TACTICAL WEAPON
GACIAC
GUIDANCE & CONTROL
INFORMATION ANALYSIS CENTER

PROCEEDINGS OF THE OPEN SESSIONS OF THE

WORKSHOP ON IMAGING TRACKERS AND AUTONOMOUS ACQUISITION APPLICATIONS FOR MISSILE GUIDANCE

19 - 20 NOVEMBER 1979

GACIAC-PR-80-01

SPONSORED BY THE

U.S. ARMY MISSILE COMMAND
REDSTONE ARSENAL, ALABAMA

AND THE

JOINT SERVICE GUIDANCE AND CONTROL COMMITTEE

NOTICES

Proceedings. This Proceedings is being published by the Tactical Weapon Guidance and Control Information Analysis Center (GACIAC) as part of its services to the guidance and control community. GACIAC is a DLA administered, DoD Information Analysis Center, operated by IIT Research Institute under Contract DSA 900-77-C-3840, sponsored by the U.S. Army Missile Command, Redstone Arsenal, Alabama. Mr. Howard C. Race, DRSMI-RN, is the project manager.

Reproduction. Permission to reproduce any material contained in this document must be requested and approved in writing by the GACIAC, IIT Research Institute, 10 West 35th Street, Chicago, Illinois 60616. This document is only available from GACIAC and is not to be distributed by DTIC.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER: GACIAC-PR-80-01

2. GOVT ACCESSION NO.: AD-A093954

3. RECIPIENT'S CATALOG NUMBER:

4. TITLE (and Subtitle): Proceedings of the Open Sessions of the Workshop on Imaging Trackers and Autonomous Acquisition Applications for Missile Guidance Held at Redstone Arsenal, Alabama on 19-20 November 1979

5. TYPE OF REPORT & PERIOD COVERED: Workshop Proceedings November 1979

6. PERFORMING ORG. REPORT NUMBER: GACIAC-PR-80-01

7. AUTHOR(s): Various, Workshop Chairman: Dr. Lewis G. Minor, USA MICOM

8. CONTRACT OR GRANT NUMBER(s): DSA900-77-C-3849

9. PERFORMING ORGANIZATION NAME AND ADDRESS:
GACIAC, IIT Research Institute
10 West 35th Street
Chicago, Illinois 60616

10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS:

11. CONTROLLING OFFICE NAME AND ADDRESS:
U.S. Army Missile Command
ATTN: DRSMI-REO
Redstone Arsenal, Alabama 35898

12. REPORT DATE: November 1979

13. NUMBER OF PAGES: 516

14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office): 519

15. SECURITY CLASS. (of this report): UNCLASSIFIED

15a. DECLASSIFICATION/DOWNGRADING SCHEDULE:

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release. Distribution unlimited.

DTIC ELECTE JAN 19 1981 S C

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

All secondary distribut ALL SECONDARY DISTRIBUTION OF THIS REPORT WILL BE MADE BY GACIAC.

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Missile Guidance	Noise Filtering	Object Recognition
Image Processing	Pattern Recognition	Linear Trainable Classifiers
Image Enhancement	Algorithms	Optical Image Data Processing
Image Enhancement Filters	Texture Analysis	
Spatial Filtering	Edge Cooccurrence	Correlation

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

This Proceedings contains papers presented in, and submitted for, the Open Sessions of the Workshop on Imaging Trackers and Autonomous Acquisition Applications for Missile Guidance that was held at the US Army Missile Command, Redstone Arsenal, Alabama, on 19-20 November 1980. This volume contains papers from sessions on: Image Processing and Pattern Recognition Algorithms for Missile Guidance, Advanced Imaging Trackers for Missile Guidance, and Image Understanding and Autonomous Acquisition for Missile Guidance.

DD FORM 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

410948

19. Key Words (Continued)

Image Trackers	Hybrid Correlation Algorithms
Image Sensors	Feature Matching
Imaging IR Seekers	Image Correlation
Video Tracking	Digital Trackers
Target Tracking Methodologies	Target Hand-Off
Lie Theoretic Methods	IR Seekers
Multi-Mode Trackers	IR Detectors
Cruise Missile Guidance	Precision Cruise Missile Guidance
Terminal Homing	Autonomous Acquisition
Target Discrimination	Object Discrimination

GACIAC-PR-80-01 NOVEMBER 1979

PROCEEDINGS OF THE OPEN SESSIONS WORKSHOP ON IMAGING TRACKERS AND AUTONOMOUS ACQUISITION APPLICATIONS FOR MISSILE GUIDANCE

Sponsored by the

U.S. ARMY MISSILE COMMAND
REDSTONE ARSENAL, ALABAMA

and the

JOINT SERVICE GUIDANCE
AND CONTROL COMMITTEE

GACIAC — A DoD Information Analysis Center
Operated by IIT Research Institute, 10 W. 35th St., Chicago, IL 60616
Sponsored by the U.S. Army Missile Command, Redstone Arsenal, AL 35898

PREFACE

The "intelligence quotient" of imaging missile systems has increased dramatically since the first imaging tracker for missile systems was built in the early 1960's at MICOM. Terms like "Smart Missile," "Intelligent Missiles" and even "Brilliant Missiles" are being used to describe future missile systems. In some cases, the terms are a mildly humorous overstatement of what really could be accomplished. However, the terms do provide a hint of what to expect in the future.

Future missile systems are expected to reach new levels of sophistication brought about by the explosion of developments in the fields of image processing, pattern recognition, signal processing, and VHSIC. Surprisingly, the cost of these new missiles will probably be dominated by mechanical and optical components rather than complexity in the electronics. As a case in point, the cost of the keyboard, display and packaging--rather than that of the electronics--is beginning to determine the price of many hand calculators.

The goal for future missile systems is to have the capability of achieving lock-on-after launch. If the goal is ever attained, warfare as we know it will be revolutionized. The obvious reason is that systems now limited by the physics of optical resolution can operate at ranges limited only by the missile propulsion system. Missile control systems and conventional trackers are also sure to be effected by the burgeoning technology highlighted by many of the fine papers presented at this conference.

If the success of a conference is measured by the quality of the papers and the number of knowledgeable attendees, then the November conference was quite successful.

A successful conference is the result of the efforts of many people. A hardy thanks goes to the Co-Chairmen, the GACIAC Committee, the MICOM Protocol Office, the presenters, and all those who attended.

LEWIS G. MINOR, Chairman

US ARMY MISSILE COMMAND

WORKSHOP ON

IMAGING TRACKERS AND AUTONOMOUS ACQUISITION
APPLICATIONS FOR MISSILE GUIDANCE

CHAIRMAN: Dr. Lewis G. Minor, Advanced Sensors Directorate, MICOM

CO-CHAIRMAN: Dr. Charles R. Christensen, Research Directorate, MICOM

CO-CHAIRMAN: Mr. S. Richard F. Sims, Advanced Sensors Directorate, MICOM

AGENDA

18 November 1979

5:00 - 10:00 P.M. Preregistration (Hilton Inn)

19 November 1979

7:30 - 8:30 A.M. Registration
Rocket Auditorium, Building 7120

8:30 - 8:35 A.M. Administrative Remarks
Dr. Lewis G. Minor,
MICOM Workshop Chairman

8:35 - 8:45 A.M. Keynote Address
Dr. Julian S. Kobler,
Director, US Army Missile Laboratory

SESSION IA 8:45 - 12:00 A.M. IMAGE PROCESSING AND PATTERN RECOGNITION ALGORITHMS FOR MISSILE GUIDANCE

Chairman: Dr. Azriel Rosenfeld, University of Maryland

IA-1 Dr. R.M. Haralick, "A Facet Model for Image Data: Regions, Edges, and Texture."

IA-2 Dr. L.S. Davis, "Texture Analysis Using Edge Cooccurrence."

IA-3 Dr. D.L. Milgram, "Range Image Processing."

IA-4 Dr. R.J. Atkinson, "Classificiation/Object Recognition Software Technique Appraisal."

IA-5 Dr. J. Sklansky, "Multiple Class Piecewise Linear Trainable Classifiers."

IA-6 Dr. B.V. Dasarathy, "An Unsupervised Learning Approach to Identification and Categorization of Edges in Images."

SESSION IB 8:45 - 12:00 A.M. IMAGE PROCESSING AND PATTERN RECOGNITION ALGORITHMS FOR MISSILE GUIDANCE

Chairman: Dr. Richard Hartman, Director, Research Directorate, MICOM

IB-1 Mr. E.H. Takken and A.F. Milton, "Phase-Slipped Time Delay and Integration."

IB-2 Mr. W.H. Haas and V. Kumar, "The Synthesis of FFT Based Image Enhancement Filters."

IB-3 Dr. T.S. Huang, "Noise Filtering in Moving Images."

IB-4 Dr. C.A. Halijak, "From Numerical Transforms to Spatial Filters."

IB-5 Dr. L.J. Pinson and J.L. Baumann, "Image Enhancement for Man-in-the-Loop Target Acquisition Systems."

IB-6 Dr. T.R. Edwards, "Two Dimensional Convolute Integers for Optical Image Data Processing."

SESSION IIA 1:30 - 4:45 P.M. ADVANCED IMAGING TRACKERS FOR MISSILE GUIDANCE

Chairman: Dr. Charles H. Church, Assistant Director for Technology, Directorate of Army Research Development and Acquisition, Department of the Army

IIA-1 Captain Benjamin Reischer, "Target Tracking Methodologies Present and Future."

IIA-2 Dr. T.G. Newman and D.A. Demus, "Lie Theoretic Methods in Video Tracking."

IIA-3 Mr. D.A. Orton and G.N. Yutzi, "The AI^{2}S Microprocessor Based Multi-Mode Tracker."

IIA-4 Dr. J.A. Ratkovic, "Hybrid Correlation Algorithms--A Bridge Between Feature Matching and Image Correlation."

IIA-5 Mr. J.R. Hamer and R.A. Satterfield, "Performance Comparison of a Laboratory Digital Tracker to the Human Eye."

IIA-6 Dr. R.B. Merrill and J.J. Hwang, "Image Processing for Selective Impact."

SESSION IIB 1:30 - 4:45 P.M. ADVANCED IMAGING TRACKERS FOR MISSILE GUIDANCE

Chairmen: Dr. Jess B. Huff, Director, Guidance and Control Directorate, MICOM
Mr. William Malcolm, Guidance and Control Directorate, MICOM

IIB-1 Dr. A.M. Savol, E. Noges, and A.J. Witsmeer, "An Iterative Feature Matching Algorithm."

IIB-2 Dr. J.K. Yoo, Dr. J.J. Hogan, P.H. McIngvale, and W.W. Malcolm, "Image Preprocessing for Automatic Target Hand-Off for Imaging Sensors."

IIB-3 Mr. D.D. Pasik, Dr. H.R. Dessau, and R.A. Walter, "A Digitized Video Slicing Technique for Correlation Processing."

IIB-4 Mr. J.A. Knecht, "Automatic Hand-Off from FLIR Acquisition Device to Imaging IR Seeker."

IIB-5 Dr. J.S. Boland III, H.S. Ranganath, and W.W. Malcolm, "Improved Method for Correlation of TV Sensor Images."

IIB-6 Mr. T.W. Iler, M. Pitruzzello, and P.H. McIngvale, "Design and Evaluation of an Automatic Hand-Off Correlator."

20 November 1979

7:30 - 8:30 A.M. Registration
Visitor Reception Center, Gate 1

8:15 - 8:45 A.M. MICOM In-House Research and Future Plans in Imaging Missile Technology
Dr. Lewis G. Minor, MICOM Workshop Chairman

SESSION IIIA 8:45 - 12:00 A.M. IMAGE UNDERSTANDING AND AUTONOMOUS ACQUISITION FOR MISSILE GUIDANCE

Chairmen: LTC. Larry Druffel, DARPA
Cmdr. Thomas F. Wiener, DARPA

IIIA-1 Dr. M. Vigg, Dr. C.C. Ormsby, and E.R. Edge, "Application of Imaging Sensor Technology to Precision Cruise Missile Guidance."

IIIA-2 Dr. M. Svedlow, "Evaluation of Scene Matching Concepts for Autonomous Acquisition Terminal Homing."

IIIA-3 Mr. D. Panton, "Reference Image Preparation for Precision Cruise Missile Guidance."

IIIA-4 Dr. T. Binford, "Stereo Correspondence, Automatic Registration and Surface Modeling: Problems and Approaches."

IIIA-5 Dr. B. Horn, "Automatic Construction of Synthetic Images."

IIIA-6 Dr. M. Tenebaum, "Automatic Registration of an Image with a Data Base."

SESSION IIIB 8:45 - 12:00 A.M. IMAGE UNDERSTANDING AND AUTONOMOUS ACQUISITION FOR MISSILE GUIDANCE

Chairmen: Mr. Dale Holter, Chief, Infrared Technology, Advanced Sensors Directorate, MICOM
Mr. Charles Hester, Advanced Sensors Directorate, MICOM

IIIB-1 Dr. D.P. Panda and Dr. R.K. Aggarwal, "An Automatic Image Recognition System for Military Image Analysis."

IIIB-2 Mr. D.N. DeFoe, "Target Discrimination With Multi-Dimensional Sensors."

IIIB-3 Dr. H.C. Schau, "Man-Made Object Detection."

IIIB-4 Dr. R. Jain, "The Classification of Regions in a Difference Picture for Motion Analysis."

IIIB-5 Dr. J.F. Lemmer, "Combined Arms for Image Understanding."

IIIB-6 Mr. D.E. Soland, D.V. Serreyn, R.C. Fitch, T.G. Kopet, G.O. Prom, R.C. Reitan, and M.O. Schroeder, "Prototype Automatic Target Screener Goals and Implementation."

SESSION IVA 1:30 - 4:45 P.M. PROPRIETARY OR CLASSIFIED IMAGE PROCESSING ALGORITHMS FOR MISSILE GUIDANCE

Chairman: Mr. William J. Lindberg, Director
Advanced Sensors Directorate, MICOM

IVA-1 Dr. A.M. Savol, E. Noges, and A.J. Witsmeer, "A Novel Target Prioritization by Pattern Recognition."

IVA-2 Dr. A.M. Baird, Dr. R.B. Goldman, Dr. W.R. Owens, and N.C. Randall, "Target Recognition by Sequential Analysis of Two-Dimensional Orthonormal Series Coefficients."

IVA-3 Dr. D.P. Panda and Dr. R. Aggarwal, "MMW Augmented IR Image Processing MAIRS Technical Paper."

IVA-4 Dr. R.L. Taylor, "A Basis for Missile Classifier Design."

IVA-5 Dr. A.S. Politopoulos, "An Algorithm for Isolation of Target-Like Areas in FLIR Images."

IVA-6 Dr. H.C. Schau, "Kolmogorow-Smirnov Test as a Non-Parametric Technique for Image Segmentation and Target Detection."

SESSION IVB 1:30 - 4:45 P.M. PROPRIETARY OR CLASSIFIED IMAGE PROCESSING ALGORITHMS FOR MISSILE GUIDANCE

Chairman: Mr. James J. Fagan
Scientific and Engineering Advisor
US Army Missile Laboratory, MICOM

IVB-1 Mr. H. Mack, "Supertracker: A Programmable Parallel Pipeline Arithmetic Processor for Auto-Cueing Target Processing."

IVB-2 Dr. D.P. Panda, "Symbolic Image Pattern Matching for Guidance and Tracking."

IVB-3 Mr. R.L. Schultz, R.J. Grommes, and W.S. Everhart, "Multiple Algorithm Tracking System (MATS)."

IVB-4 Mr. J.D. Jensen, A.C. Bouley, S. Foti, and G.M. Black, "Multi-Color Infrared Detectors."

IVB-5 Mr. E.J. Boudreaux and H.L. Casey, "Imaging Seeker Surface to Surface Missile Demonstration, A Prototyping Success."

IVB-6 Mr. R.B. Russell, "A Radar Imaging Guidance System - Pershing II."

TABLE OF CONTENTS

TABLE OF CONTENTS (Continued)

TABLE OF CONTENTS (Continued)

TABLE OF CONTENTS (Continued)

SESSION IA

IMAGE PROCESSING AND PATTERN RECOGNITION ALGORITHMS FOR MISSILE GUIDANCE

Paper No. IA-1, Presented at the Workshop on Imaging Trackers and Autonomous Acquisition Applications for Missile Guidance, 19-20 November, Redstone Arsenal, Alabama.

A FACET MODEL FOR IMAGE DATA: REGIONS, EDGES, AND TEXTURE

Robert M. Haralick
Virginia Polytechnic Institute
and State University

INTRODUCTION

The world recorded by imaging sensors has order. This order reflects itself in the regularity of the image data taken by imaging sensors. A model for image data describes how the order and regularity in the world manifests itself in the ideal image and how the real image differs from the ideal image. In this paper we propose a facet model for image data which suggests some procedures for image restoration, segmenting, and texture analysis.

The facet model for image data assumes that the spatial domain of the image can be partitioned into regions having certain gray tone and shape properties. The gray tones in a region must all lie in the same sloped plane. The shape of a region must not be too jagged.

To assure smoothness of a region, the facet model assumes that for each image there exists a $K > 1$ such that each region in the image can be expressed as the union of $K \times K$ blocks of pixels. The value of K associated with an image means that the narrowest part of each of its regions is at least as large as a $K \times K$ block of pixels. Hence images which can have large values of K have very smooth regions.

To make these ideas precise, let Z_r and Z_c be the row and column index set for the spatial domain of an image. For any $(r, c) \in Z_r \times Z_c$, let $I(r, c)$ be the gray value of resolution cell (r, c) and let $B(r, c)$ be the $K \times K$ block of resolution cells centered around resolution cell (r, c). Let $\Pi = \{\Pi_1, \ldots, \Pi_N\}$ be a partition of $Z_r \times Z_c$ into its regions.

In the sloped facet model, for every resolution cell $(r, c) \in \Pi_n$, exists a resolution cell $(i, j) \in Z_r \times Z_c$ such that

(1) $(r, c) \in B(i, j) \quad \Pi_n$ (shape region constraint)

(2) $I(r, c) = \alpha_n r + \beta_n c + \gamma_n$ (region gray tone constraint)

The actual image J differs from the ideal image I by the addition of random stationary noise having zero mean and covariance matrix proportional to a specified one.

$$J(r, c) = I(r, c) + \eta(r, c) \text{ where}$$

$$E[\eta(r, c)] = 0$$

$$E[\eta(r, c)\,\eta(r', c')] = k\sigma(r - r', c - c')$$

The flat facet model of Tsuji (1977) and Nagao (1978) differs from the sloped facet model only in that the coefficients α_n and β_n are assumed to be zero.

IMAGE RESTORATION UNDER THE FACET MODEL

Image restoration is a procedure by which a noisy image is operated on in a manner which produces an image which has less noise and is close to the ideal image. The facet model suggests the following simple non-linear filtering procedure. Each resolution cell is contained in K^2 different $K \times K$ blocks. The gray tone distribution in each of these blocks can be fit by either a flat horizontal plane or a sloped plane. One of the K^2 blocks has smallest error of fit. Set the output gray value to be that gray value fitted by the block having smallest error. For the flat facet model this amounts to computing the variance for each $K \times K$ block a pixel participates in. The output gray value is then the mean value of the block having smallest variance.

The filtering procedure for the sloped facet model is more complicated and we give a derivation here of the required equations. We assume that the block lengths are odd so that one of the block's pixels is its center. Let the block be $(2L + 1) \times (2L + 1)$ with the upper left-hand corner pixel having relative row column coordinates $(-L, L)$ the center pixel having relative row column coordinates $(0, 0)$, and the lower right-hand corner pixel having relative row column coordinates (L, L). Let $J(r, c)$ be the gray value at row r column c. According to the sloped facet model, for any block entirely contained in one of the image regions.

$$J(r, c) = \alpha r + \beta c + \gamma + \eta(r, c)$$

where $\eta(r, c)$ is the noise.

A least squares procedure may be used to determine the estimates for α, β, and γ. Let

$$f(\alpha, \beta, \gamma) = \sum_{r=-L}^{L} \sum_{c=-L}^{L} (\alpha r + \beta c + \gamma - J(r, c))^2.$$

The least squares estimate for α, β, and γ are those which minimize f. To determine these values, we take the partial derivative of f with respect to α, β, and γ, set these to zero and solve the resulting equations for α, β, and γ. Doing this we obtain

$$\alpha = \frac{3}{L(L+1)\ (2L+1)^2} \sum_{r=-L}^{L} r \sum_{c=-L}^{L} J(r, c)$$

$$\beta = \frac{3}{L(L+1)\ (2L+1)^2} \sum_{c=-L}^{L} c \sum_{r=-L}^{L} J(r, c)$$

$$\gamma = \frac{1}{(2L+1)^2} \sum_{r=-L}^{L} \sum_{c=-L}^{L} J(r,c)$$

The meaning of this result can be readily understood for the case when the block size is 3 × 3. Here L = 1 and

$$\alpha = \frac{1}{6}[J(+1, \cdot)]$$

$$\beta = \frac{1}{6}[J(\cdot, 1) - J(\cdot, -1)]$$

$$\gamma = \frac{1}{9}J(\cdot, \cdot)$$

where an argument of J taking the value dot means that J is summed from -L to L in that argument position. Hence, α is proportional to the slope down the row dimension, β is proportional to the slope across the column dimension, and γ is the simple gray value average over the block.

The fitted gray tone for any resolution cell (r, c) in the block is given by

$$\hat{J}(r, c) = \alpha r + \beta c + \gamma$$

For the case where L = 1,

$$\hat{J}(r, c) = \frac{1}{6}[J(1, \cdot) - J(-1, \cdot)]r + \frac{1}{6}[J(\cdot, 1) - J(\cdot, -1)]c + \frac{1}{9}J(\cdot, \cdot)$$

Writing this expression out in full:

$$\hat{J}(r, c) = \frac{1}{18} \{J(-1, 1)\ (-3r - 3c + 2)$$

$$+ J(-1, 0)\ (-3r + 2)$$

$$+ J(-1, 1)\ (-3r + 3c + 2)$$

$$+ J(0, -1)\ (-3c + 2)$$

$$+ J(0, 0)\ (2)$$

$$+ J(0, 1)\ (3c + 2)$$

$$+ J(1, -1)\ (3r - 3c + 2)$$

$$+ J(1, 0)\ (3r + 2)$$

$$+ J(1, 1)\ (3r + 3c + 2)\}$$

This leads to the set of linear filter masks shown in Figure 1 for fitting each pixel position in the 3 × 3 block.

The sloped facet model noise filtering would examine each of the K^2 K × K blocks a pixel (r, c) belongs to. For each block, a block error can be computed by

$$\varepsilon^2 = \sum_{r=-L}^{L} \sum_{c=-L}^{L} (J(r, c) - J(r, c))^2$$

One of the K × K blocks will have lowest error. Let (r*, c*) be the coordinates of the pixel (r, c) in terms of the coordinate system of the block having smallest error. The output gray value at pixel (r, c) is then given by $\hat{J}(r^*, c^*)$ where $\hat{J}$ is the linear estimate of gray values for the block having smallest error of fit.

Haralick and Watson (1979) prove convergence of this iteration procedure.

REGION AND EDGE ANALYSIS

The image restoration iteration procedure can produce more than just a restored gray tone. For each pixel, it also produces the α, β, and γ parameters. Using these parameters we can determine whether or not neighboring pixels lie in the same connected facet. Of course doing this determination requires that the parameters for each pixel be taken out of their relative coordinate system and be placed in some absolute coordinate system. Linking together neighboring pixels with the same α, β, γ parameters permits us to identify the facets which are characterized by the connected sets of pixels that constitute them.

$\hat{J}(-1,-1)$

8	5	2
5	2	-1
2	-1	-4

$\hat{J}(-1,0)$

5	5	5
2	2	2
-1	-1	-1

$\hat{J}(-1,1)$

2	5	8
-1	2	5
-4	-1	2

$\hat{J}(0,-1)$

5	2	-1
5	2	-1
5	2	-1

$\hat{J}(0,0)$

2	2	2
2	2	2
2	2	2

$\hat{J}(0,1)$

-1	2	5
-1	2	5
-1	2	5

$\hat{J}(1,-1)$

2	-1	-4
5	2	-1
8	5	2

$\hat{J}(1,0)$

-1	-1	-1
2	2	2
5	5	5

$\hat{J}(1,1)$

-4	-1	2
-1	2	5
2	5	8

Figure 1 shows the filtering masks to be used for least squares estimation of the gray value for any position in a 3 x 3 block. Each mask must be normalized by dividing by 18.

Edge detection and region growing are two areas of image analysis which are opposite in emphasis but identical in heart. Edges obviously occur at bordering locations of two adjacent regions which are significantly different. Regions are maximal areas having similar attributes. If we could do region analysis, then edges can be declared at the borders of all regions. If we could do edge detection, regions would be the areas surrounded by the edges. Unfortunately, we tend to have trouble doing either: edge detectors are undoubtedly noisy and region growers often grow too far.

The facet model permits an even handed treatment of both. Edges will not occur at locations of high differences. Rather, they will occur at the boundaries having high differences between the parameters of sufficiently homogeneous areas. Regions will not be declared at just areas of similar value of gray tone. They will occur at the facets, connected areas where resolution cells yield minimal differences of region parameters, where minimal means smallest among a set of resolution cell groupings. In essence, edge detection and region analysis are identical problems that can be solved with the same procedure.

Recall that the facet model iterations produce the parameters α and β. The fact that the parameters α and β determine the value of the slope in any direction is well known. For a planar surface of the form

$$g(r, c) = \alpha r + \beta c + \gamma$$

the value of the slope at an angle θ to the row axis is given by the directional derivative of g in the direction θ. Since α is the partial derivative of g with respect to r and β is the partial derivative of g with respect to c, the value of the slope at angle θ is $\alpha \cos \theta + \beta \sin \theta$. Hence, the value of the slope at any direction is an appropriate linear combination of the values for α and β. The angle θ which maximizes this value satisfies

$$\cos \theta = \frac{\alpha}{\sqrt{\alpha^2 + \beta^2}} \quad \text{and} \quad \sin \theta = \frac{\beta}{\sqrt{\alpha^2 + \beta^2}}$$

and the gradient which is the value of the slope in the steepest direction is

$$\sqrt{\alpha^2 + \beta^2} \ .$$

The sloped-facet model is an appropriate one for either the flat world or sloped world assumption. In the flat world each ideal region is constant in gray tone. Hence, all edges are step edges. The observed image taken in an ideal flat world is a defocussed version of the ideal piecewise constant image with the addition of some random noise. The defocussing changes all step edges to sloped edges. The edge detection problem is one of determining whether the observed noisy slope has a gradient significantly higher than one which could have been caused by the noise alone. Edge boundaries are declared in the middle of all significantly sloped regions.

In the sloped facet world, each ideal region has a gray tone surface which is a sloped plane. Edges are places of either discontinuity in gray tone or derivative of gray tone. The observed image is the ideal image with noise added and no defocussing. To determine if there is an edge between two pixels, we first determine the best slope fitting neighborhood for each of the pixels by the iteration procedure. Edges are declared at locations having significantly different planes on either side of them. In the sloped facet model, edges surrounding regions having significantly sloped surfaces, may be the boundary of an edge region. The determination of whether a sloped region is an edge region or not may depend on the significance and magnitude of the slope as well as the semantics of the image.

In either the case of the noisy defocussed flat world, or the noisy sloped world we are faced with the problem of estimating the parameters of a sloped surface for a given neighborhood and then calculating the significance of the difference of the estimated slope from a zero slope or calculating the significance of the difference of the estimated slopes of two adjacent neighborhoods. To do this we proceed in a classical manner. We will use a least squares procedure to estimate parameters and we will measure the strength of any difference by an appropriate F-statistic.

TEXTURE ANALYSIS

Textures can be classified as being weak textures, or strong textures. Weak textures are those which have weak spatial-interaction between the texture primitives. To distinguish between them it may be sufficient to only determine the frequency with which the variety of primitive kinds occur in some local neighborhood. Hence, weak texture measures account for many of the statistical textural features. Strong textures are those which have non-random spatial interactions. To distinguish between them it may be sufficient to only determine, for each pair of primitives, the frequency with which the primitives co-occur in a specified spatial relationship. In this section we discuss a variety of ways in which primitives from the facet model can be defined and the ways in which spatial relationships between primitives can be defined.

Primitives

A primitive is a connected set of resolution cells characterized by a list of attributes. The simplest primitive is the pixel with its gray tone attribute. Sometimes it is useful to work with primitives which are maximally connected sets of resolution cells having a particular property. An example of such a primitive is a maximally connected set of pixels all having the same gray tone or all having the same edge direction.

Gray tones and local properties are not the only attributes which primitives may have. Other attributes include measures of shape of

connected region and homogeneity of its local property. For example, a connected set of resolution cells can be associated with its length or elongation of its shape or the variance of its local property.

Attributes generated by the facet model include the α, β, and γ parameters plus the average error of fit for the facet. These attributes can be used by themselves or used to generate additional attributes such as $\sqrt{\alpha^2 + \beta^2}$ from which relative extreme primitives can be defined in the following way.

Label all pixels in each maximally connected relative maxima plateau with a unique label. Then label each pixel with the label of the relative maxima that can reach it by a monotonically decreasing path. If more than one relative maxima can reach it by a monotonically decreasing path, then label the pixel with a special label "c" for common. We call the regions so formed the descending components of the image.

Spatial Relationships

Once the primitives have been constructed, we have available a list of primitives, their center coordinates, and their attributes. We might also have available some topological information about the primitives, such as which are adjacent to which. From this data, we can select a simple spatial relationship such as adjacency of primitives or nearness of primitives and count how many primitives of each kind occur in the specified spatial relationship.

More complex spatial relationships include closest distance or closest distance within an angular window. In this case, for each kind of primitive situated in the texture, we could lay expanding circles around it and locate the shortest distance between it and every other kind of primitive. In this case our co-occurrence frequency is three-dimensional, two dimensions for primitive kind and one dimension for shortest distance. This can be dimensionally reduced to two dimensions by considering only the shortest distance between each pair of like primitives.

Co-occurrence between properties of the descending components can be based on the spatial relationship of adjacency. For example, if the property is size, the co-occurrence matrix could tell us how often a descending component of size s_1 occurs adjacent to or nearby to a descending component of size s_2 or of label "c".

To define the concept of generalized co-occurrence, it is necessary to first decompose an image into its primitives. Let Q be the set of all primitives on the image. Then we need to measure primitive properties such as mean gray tone, variance of gray tones, region, size, shape, etc. Let T be the set of primitive properties and f be a function assigning to each primitive in Q a property of T. Finally, we need to specify a spatial relation between primitives such as distance or adjacency. Let $S \subseteq Q \times Q$ be the binary relation pairing all primitives which satisfy the spatial relation. The generalized co-occurrence matrix P is defined by:

$$P(t_1, t_2) = \frac{\#\{(q_1, q_2) \in S | f(q_1) = t_1 \text{ and } f(g_2) = t_2\}}{\#S}$$

$P(t_1, t_2)$ is just the relative frequency with which two primitives occur with specified spatial relationship in the image, one primitive having property t_1 and the other primitive having property t_2.

Zucker (1974) suggests that some textures may be characterized by the frequency distribution of the number of primitives any primitive has related to it. This probability p(k) is defined by:

$$p(k) = \frac{\#\{(q \in Q \mid \#S(q) = k\}}{\#Q}$$

Although this distribution is simpler than co-occurrence, no investigator appears to have used it in texture discrimination experiments.

CONCLUSION

In this paper, we considered the gray tones of an image to represent the height of a surface above the row-column coordinates of the gray tones. The observed image is then the surface of the underlying ideal image plus random noise. The ideal image is composed of a patchwork of constrained surfaces sewed together.

We called each patch a facet and in the ideal image, the facets must satisfy the constraints of the facet model for image data: the facet model constrains the shape of each facet to be exactly composed as a union (possibly over-lapping) of a given set of neighborhood shapes and constraints the surface to be a sloped plane surface of a quadratic surface.

The goal of image restoration is to recover the ideal gray tone surface which underlies the observed noisy gray tone surface. Although the noise prevents recovering the precise underlying ideal surface, we can recover that gray tone surface which is the "closest surface" to the observed noisy surface and which also satisfies the facet model constraints.

The procedure we suggested for recovering the underlying surface is an iterative one. Associated with each given pixel is a set of all the neighborhoods of given shapes that contain it. Each one of these neighborhoods can be fit with the best fitting plane surface. One of these neighborhoods will have a best fitting surface with lowest error among all the neighborhood has a height above the given pixel. The parallel iterative procedure consists of replacing each pixel gray tone intensity with the height of the best fitting surface in its lowest error neighborhood. The procedure is guaranteed to converge and actually achieves essential convergence in a few iterations. The resulting image is an enhanced image having less noise, better contrast, and sharper boundaries.

Image restoration is not the only use of the facet model. The facet model processing provides us with additional important information. By collecting together all pixels participating in the same surface facet, we transformed the pixel as our processing and analysis unit into the surface facet as our processing and analysis unit. Now edge boundaries, for example, can be defined to occur at the shared boundary of all neighboring facets whose surface parameters are significantly different. Homogeneous regions can be defined by linking together all those neighboring surface facets whose parameters are significantly the same. Texture can be characterized by the co-occurrence statistics of neighboring primitives which are not the pixel gray tones as in the usual occurrence approach but which are the facets characterized by their boundary, shape, size, and surface parameter attributes.

Paper No. IA-2, Presented at the Workshop on Imaging Trackers and Autonomous Acquisition Applications for Missile Guidance, 19-20 November 1979, Redstone Arsenal, Alabama.

TEXTURE ANALYSIS USING EDGE COOCCURRENCE

Larry S. Davis
Computer Sciences Department
University of Texas
Austin, Texas 78712
November 1979

This research was supported in part by funds derived from the Air Force Office of Scientific Research under Contract F49620-79-C-0043.

INTRODUCTION

This paper described a tool for image texture analysis called a generalized cooccurrence matrix. Describing image texture is an important problem in the design of image understanding systems. Applications as diverse as earth resources technology and medical disease diagnosis rely, to a great extent, on the ability to automatically discriminate between different image patterns, or textures.

Most approaches to texture analysis have been based on computing various statistics of the distribution of intensities in an image. For example, the grey level cooccurrence matrix counts the frequency with which pairs of intensities are found in particular relative spatial positions. Statistics can be computed from the grey level cooccurrence matrix which reflect intuitive properties of texture such as coarseness (or size of the elements in the texture) and directionality. Haralick [1] first introduced the grey level cooccurrence matrix as a texture analysis tool. Other approaches to measuring texture features based on intensity distributions include run length statistics [2], statistics computed from histograms of differences in intensity between nearby picture points [3], and statistics derived from the autocorrellation, or power spectrum, of the image [3]. Haralick [4] contains an extensive survey.

An alternative approach to describing texture is to compute texture descriptors not based on the original pattern of intensities in the image, but rather on the results of applying an edge detector to the image texture (possibly grouping the edges detected at individual pixels into longer, extended edges). Marr [5] suggested that textures could be adequately described by computing various first-order statistics of features of the primal sketch, which is a representation of the image in terms of groups of edges which form perceptually significant contours. Marr's approach is consistent with recent psychophysical results reported by Julezs [6]

which seem to indicate that a wide range of human texture perceptions can be accounted for in terms of first-order statistics of the distributions of edges, lines and termination points (i.e., line endings) in the texture.

More recently, Davis, et al [7] suggested that useful texture descriptors could be obtained by computing statistics based on the cooccurrence of edges in textures. This paper discusses that approach, based on what we call "generalized cooccurrence matrices," and includes the results of an experimental study which compared the classification power of grey level cooccurrence matrices and generalized cooccurrence matrices on a database of natural textures.

GENERALIZED COOCCURRENCE MATRICES

A generalized cooccurrence matrix, or GCM, describes texture by describing the spatial arrangement of local features in the texture. A particular GCM is defined by specifying a triple [P,S,A] where:

1) P is an image feature prototype,

2) S is a spatial predicate, and

3) A is a prototype attribute.

The prototype, P, can be regarded as a structural definition of the local image feature of interest, and generally contains a list of attributes which defines a local feature. For example, we can define the prototype edge-pixel as

edge-pixel

location: (x,y)

orientation: theta

contrast: C

Thus, an edge pixel has a location in the image, an orientation and a contrast. This information is ordinarily computed by an edge detection procedure. As a second example, consider the prototype intensity-pixel defined as

intensity-pixel

location: (x,y)

intensity: i

An intensity-pixel is simply the intensity value associated with a pixel. GCM's based on the prototype intensity-pixel will correspond to grey level cooccurrence matrices.

A spatial predicate, S, is a mapping from pairs of image features into {TRUE, FALSE}. For example, the spatial predicate, Sk, defined over pairs of edge-pixels ((x1,y1), theta1, C1), ((x2,y2), theta2, C2) is true if and only if

$$\max\{\ |x1-x2|,\ \ |y1-y2|\ \} < k$$

Other spatial predicates will be discussed later.

Suppose that F = {F1,...,Fn} is a set of local features detected in an image by a feature detection program. Each of the Fi is structured according to a particular prototype definition. For example, if the prototype edge-pixel were being used, then each Fi would be a triple containing location, orientation and contrast. Let A be one of the attributes which appears in the definition of the prototype associated with the Fi. For example, A might be the attribute orientation. Then the GCM of the set F with respect to the spatial predicate S and attribute A, $G_{S,A}$ is defined by:

$$G_{S,A}(v_1,v_2) = \frac{\#\left\{(f_i,f_j): A_{f_i}=v_1, A_{f_j}=v_2, S(f_i,f_j)=\text{TRUE}\right\}}{\#\left\{(f_i,f_j):\ S(f_i,f_j) = \text{TRUE}\right\}}$$

where #S denotes the number of elements in the set S. An unnormalized GCM can be obtained by not performing the division by the number of pairs of local features which satisfy the spatial predicate.

Figure 1 contains a simple example of an unnormalized GCM. The prototype is edge pixel, and Figure 1a contains a picture of edge pixels, marked with their orientations. The coding is H for horizontal, V for vertical, L for left diagonal and R for right diagonal. A blank pixel indicates that no edge is associated with that picture point. The spatial predicate used to form the GCM is S1. Figure 1b contains the GCM.

GCM's are a generalization of the conventional grey level cooccurrence matrix. The prototype used is intensity-pixel, the attribute of interest is intensity, and the spatial predicate assigns TRUE to pairs of pixels in particular relative spatial positions. The relative spatial positions can be specified by a set of displacement vectors D = {(dx,dy)}. The experiments described in Section 3 will use the two sets D1 = {(0,1), (1,0), (-1,0), (0,-1), (1,1), (1,-1), (-1,1), (-1,-1)} and D2 = {(0,2), (2,0), (0,-2), (-2,0)}.

Two different spatial predicates will be used for edge-pixels. The first, Sak, examines two cone shaped areas of length k, which emanate from an edge pixel and are oriented parallel to the orientation of the edge pixel. Figure 2a illustrates the spatial predicate Sak. The second spatial predicate, SNk, orients the two cones orthogonal to the orientation of the edge-pixel. Figure 2b illustrates the spatial predicate SN3.

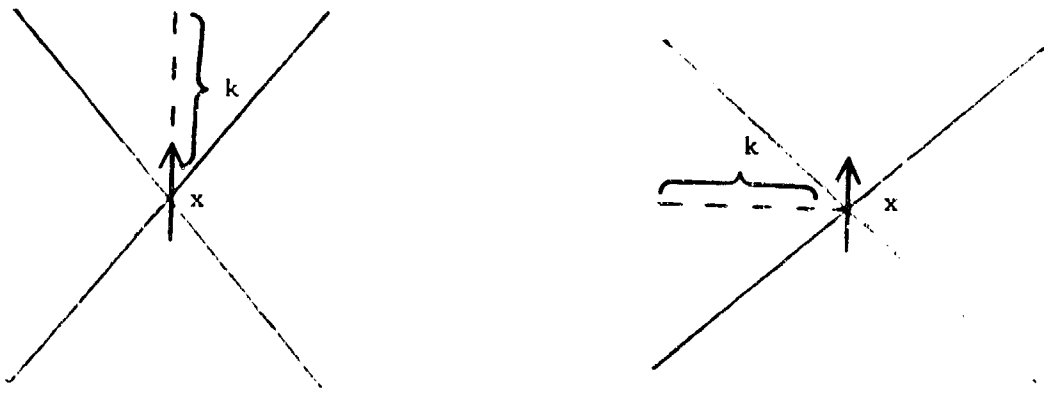
						P	H
	H	H	H		R		
L				R			
V				V			
V				V			
	H	H	H		L	H	H

a) Spatial distribution of oriented edges

12	2	2	2
2	4	2	2
2	2	0	0
2	1	0	2

b) GCM using spatial predicate S_1

Figure 1 - Example of a GCM



a) The spatial predicate S_{a_k}

b) S_{N_k}

Figure 2 - Edge-pixel spatial predicates S_{a_k} and S_{N_k}

Intuitively, Sak and SNk should be useful for determining the elongatedness and width of texture elements. For elongated texture elements, the GCM based on Sak, using attribute orientation, should have most of its power along the main diagonal, since the edges of elongated texture elements will tend to "line up." Similarly, for narrow texture elements, the GCM based on SNk should have high values along the main diagonal.

In general, the structure of GCM's can be usefully captured in a few statistics, or texture descriptors, which can be efficiently computed from the GCM. These descriptors could then serve as input to a statistical classification procedure, such as the one described in Section 3. The descriptors which we will discuss are similar to the ones proposed by Haralick [1] for grey level cooccurrence matrices. They include contrast, uniformity, entropy and correllation. See Davis, et al [7] for the definitions of these features.

EXPERIMENTAL STUDY

An experiment was performed to compare the classification power of GCM's based on edge-pixels versus GCM's based on intensity-pixels, i.e., conventional grey level cooccurrence matrices. The database for the experiment included eight classes of natural textures, including brick, striated concrete, grating, orchard, metal scrap, pebbles, shrub and tree bark. The original images were digitized to a resolution of 256x256 pixels, with each pixel quantized to six bits. A histogram normalization was applied to all of the textures so that their first-order statistics were identical. Sixteen 64x64 samples were then extracted from each of the original textures.

Edge-pixels were detected by applying an edge detector based on the Kirsch edge operator [8]. The edge detection procedure first associated a contrast and orientation at each point by applying the Kirsch operator. Points whose contrast value was below a prespecified threshold were deleted; finally, only local peaks from the remaining points were chosen as edge pixels. A more detailed discussion of the edge detection procedure can be found in [7]. Even though the edge detector does not completely outline the texture elements in the original textures, it is relatively accurate in its placement of edges (see [9] for examples of texture samples and edges).

The classifier used was a leave-one out classifier. In this method, all samples but one in the database are used as a training set. The remaining sample is then classified using the statistics derived from the training set. Thus, each sample in the database is treated once as an unknown. The results of the experiment are summarized in Table 1. For each prototype and spatial predicate, the best descriptor pair is listed along with the percentage classification. The results shown in Table 1 are consistent with those reported in [7] where edge-pixel GCM's were found to yield higher classification rates than intensity-pixel GCM's. An extended version of this experiment was described in [9]. There, a

third prototype, called an extended-edge, was included. Extended-edges correspond to connected components of constant orientation edge-pixels. Also, first order statistics of edge-pixels and extended-edges were investigated. The best classification results were obtained using GCM's based on edge-pixels.

Table 1: Classification Results

Prototype	Attribute	Spatial Predicate	Best Pair Accuracy	
Pixel-inten.	Intensity	D1	Contrast Correlation	53%
		D2	Contrast Correlation	52%
Edge-pixel	Orientation	Sa3	Correlation Uniformity	55%
		Sa7	Contrast Uniformity	59%
		SN3	Contrast Uniformity	49%
		SN7	Contrast Entropy	61%

SUMMARY

We have presented a tool for image texture analysis called a generalized cooccurrence matrix, and described its application to a texture discrimination problem. GCM's describe texture by measuring the spatial arrangement of local image features, such as edges, in the texture. To the extent that these local features characterize the size, shape and spatial arrangements of the elements which comprize the texture, the GCM's capture these important aspects of the structure of the texture. Clearly, the usefulness of GCM's is intimately related to the reliability with which we can detect local features in textures. Davis and Mitiche [10] discuss the problem of edge detection in textures, and derive an optimal edge detection procedure for cellular textures. The procedure is based on a one dimensional edge operator and a model of image texture describing cell size and placement. We are currently attempting to apply these theoretical results to the analysis of real images (the database described in this paper), in the hopes of assessing the real gain in descriptive power obtained by employing more sophisticated local feature detection algorithms.

REFERENCES

1. R. Haralick, B. Shanmugan, and I. Dinstein, "Texture features for image classification," IEEE Trans. on Systems, Man and Cybernetics,3, pp.610-622.

2. A. Rosenfeld and A. Kak, Digital Picture Processing, Academic Press, New York, 1976.

3. J. Weszka, C. Dyer and A. Rosenfeld, "A comparative study of texture measures for terrain classification," IEEE Trans. Systems, Man and Cybernetics,4, pp. 269-285.

4. R. Haralick, "Statistical and structural approaches to texture," Proc. IEEE, 67, pp. 786-805.

5. D. Marr, "Early processing of visual information," Phil. Trans. Royal Society, B, 275, pp. 483-524.

6. B. Julesz, "Differences between attentive (figure) and preattentive (ground) perception," in Proc. Image Modelling Workshop, ed. A. Rosenfeld, to appear.

7. L. Davis, S. Johns, and J. K. Aggarwal, "Texture analysis using generalized cooccurrence matrices," IEEE Trans. Pattern Analysis and Machine Intelligence, 1, pp. 251-258.

8. R. Kirsch, "Computer identification of the constituent structure of biological images," Computers and Biomedical Research, 4, pp. 315-328.

9. L. Davis, B. Clearman and J. K. Aggarwal, "A comparative texture classification study based on generalized cooccurrence matrices," to appear in IEEE Conf. on Decision and Control, Miami, Fl. December, 1979.

10. L. Davis and A. Mitiche, "Edge detection in textures," in Proc. Image Modelling Workshop, ed. A. Rosenfeld, to appear.

Paper No. IA-3, Presented at the Workshop on Imaging Trackers and Autonomous Acquisition Applications for Missile Guidance, 19-20 November 1979, Redstone Arsenal, Alabama

RANGE IMAGE PROCESSING

D. L. Milgram
C. M. Bjorklund

Lockheed Palo Alto Research Laboratory
3251 Hanover Street
Palo Alto, CA 94304

Abstract

Range images preserve the 3-D geometry of a scene as viewed from the sensor position. It then becomes possible to determine the local orientation of a surface at a particular point and to segment the image into planar surfaces. These planar surfaces can serve as primitives for matching to a model.

1. INTRODUCTION

This paper describes recent investigations into range image processing by the Lockheed Signal Processing Laboratory. The purpose of this research has been to develop an approach and a methodology for utilizing range imagery in intelligence, guidance, and recognition tasks, with particular emphasis on landmark recognition using onboard reference imagery. Our results show that range imagery processing can yield reliable, accurate descriptions of man-made scene components. These descriptions reflect the actual geometry of the scene and are independent of the sensor position. They thus constitute a robust set of primitives for scene matching or recognition, resulting in accurate vehicle position fixing. In the sections that follow we describe the problem definition, our proposed approach and the experimental results of our research.

2. PROBLEM DEFINITION

For the purposes of this discussion, we assume that the principal data gathering mechanism is a laser range finder, consisting of a laser illuminator scanning the visual field in raster fashion and a sensor which determines the distance from the sensor to each laser-designated raster point based on the return time of the laser beam. The range image is the array of range values for the raster. It is also possible to gather a reflectance image by measuring the amplitude of the return (as well as its onset); however, the reflectance image was judged to be too noisy to use as a primary source of information.

Our second basic assumption is that a model describing the planar surfaces in the scene is provided. Although the optimal structure of a scene model is by no means settled, we have assumed that it consists of a list of objects made up of planar surfaces and the descriptions of their absolute position in a standard earth-based coordinate system.

Finally, we assume that the relative orientation of the sensor is known as is an estimate of its position with respect to the model's coordinate system. This is reasonable for vehicles with onboard inertial guidance systems since they can determine orientation and can estimate position. The task addressed by this paper is to investigate algorithms for matching the sensed range image to the model.

3. APPROACH

3.1 Background

The image processing literature describes many approaches to the problem of intelligent matching to a stored model. For a review of recent research, see [1]. Nonetheless, progress has been slow. One extremely vexing source of difficulty has been the inability of the model to guide and control the segmentation and identification of image parts. This may be due to the incompleteness of the model or to its irrelevance to the sensed image. Recently, researchers have begun to focus attention on the need to model not only the contents of a scene but also its appearance to the sensor [2,3]. Increased understanding of the relationship of scene structure to scene appearance will improve our ability to solve problems in which strong scene models are available. The ranging environment provides a unique opportunity for the researcher to study the close relationship between a variety of sensed data and a model.

In contrast to sensors which measure reflectance or emission characteristics of the scene, a range sensor responds to the set of distances to nearest points along different rays from the sensor. This retains the perspective information available at the viewpoint and simplifies the reconstruction of the scene geometry [4,5]. Specifically, the range data allows us to back-compute the sensor position from the locations of recognized parts of the scene. If a sufficient number of meaningful components can be sensed, extracted and matched against the model, it then becomes possible to perform a least squares estimation of the sensor position.

In order to perform such a computation using conventional imagery, it would be necessary to find edges and corners extremely accurately and determine precisely to which points in the model they correspond. This is a much more difficult task since the 3-D nature of the scene has been lost in the imaging process. Moreover, the interiors of regions don't contribute to the position computation, thus reducing the positive effects of redundancy available to range image processing.

Another significant benefit from the use of range data is its relative insensitivity to the spectral characteristics of the scene as well as to

seasonal/diurnal variations. For example, shadows and glint do not affect the laser/sensor combination. It should also be pointed out that range geometry simplifies the construction and utilization of the target model since the model need only specify the locations and descriptions of objects in the scene. Thereafter, it is straightforward to calculate their size, shape, position, etc. in the image.

The next section presents the suggested approach to three-dimensional matching and discusses each of the steps involved. The section following it discusses the preliminary experimental results which we have obtained. The last section addresses the results and the problem areas to be investigated in future studies.

3.2 Three-Dimensional Matching Procedure

The basic objective of the matching system is to determine the location of the vehicle with relation to some fixed point, T, on the ground. The target location would be a reasonable choice for that point. Input data to the system would consist of: (1) a stored model of the scene, including a list of the plane surfaces it contains, in the form, say, of a vertex list with respect to an origin at point T; (2) a sensed range image, giving the distance from the sensor to the nearest point in the scene for an array of azimuths and elevations; and (3) the orientation and approximate vehicle location derived from the inertial guidance system and/or from the last navigation update.

The range image in angular coordinates can easily be transformed to a cartesian system whose three components represent distance from an image plane (located at the sensor and perpendicular to its line of sight) and projection along a pair of orthogonal axes in that plane. Figure 1 illustrates the transformation. In such a cartesian system, any plane can be described by a linear equation whose three parameters can be taken to be the azimuth and elevation of its normal and the perpendicular distance from the sensor to the plane (extended, if necessary). The objective, then, is to determine the vector $\vec{\Delta}$ from S, the origin of the sensor-based coordinate system, to target point T. Identification of an extracted plane in the sensed image with one in the model gives information about the component of $\vec{\Delta}$ normal to that plane. Thus, a minimum of three such corresponding pairs of planes with non-coplanar normals must be identified in order to determine $\vec{\Delta}$ completely. In practice it is expected that many more such pairs would be found. The proposed system to determine $\vec{\Delta}$ by plane matching consists of the four steps shown in Figure 2.

The aggregation procedure in the first block begins by attempting at each pixel to fit a plane to a small (say 5×5) neighborhood of the pixel (Figure 3). The results of the fitting process are three plane parameters and a residual indicating the goodness of fit. Then, the algorithm performs a systematic labeling of groups of adjacent pixels having residual smaller than some threshold and parameters within a limited range. Each labeled group is individually fitted by a least squares plane. The plane is described by its planar parameters and its location, e.g., the center of the bounding

rectangle of the extracted plane in sensed image coordinates. These six parameters for each plane*, together with the number of pixels in the plane, would be stored as an entry in a "sensed plane list".

A similar list is prepared from the reference model, using the best information about the vehicle location, in the step described in the diagram as "reference list generation". The basic techniques of model projection in perspective have been well studied in the computer graphics literature [6] and allow us to compute the set of planar surfaces visible from the estimated sensor position and to compute their descriptions - orientations, positions, visible area, etc. However, it is not necessary to create the pixel-by-pixel raster display since the features are being directly utilized for matching.

It is important to distinguish the feature primitives derived from the model or a sensed range image with those derived conventionally from an intensity image. Geometric measurements from the former relate directly to the actual scene, e.g., heights, distances, surface areas; while measurements from conventional imagery are normally expressed in terms of image-related descriptors such as pixels. Scene measures will thus remain constant in range experiments using the same scene. Image measures can be expected to vary as the imaging environment changes.

The matching step of the algorithm compares the two lists in order to identify sensed primitive planes with predicted model surfaces. The primary discriminant is the plane orientation since this measure is unaffected by translations of the sensor position. Other auxiliary discriminants are the relative positions of the primitive planes (e.g. front to rear, left to right), their areas and their adjacency relations. These are used to resolve ambiguities in the matching process. Depth first search is a simple control mechanism to guide the search. The evaluation function is the number of well-matched model surfaces. More generally, a smarter control process for "growing" the match solution would use the strong geometric cues available; this remains a topic for future research.

Fig. 4 shows a two-dimensional example of how identified planes can locate the sensor position. Point T is the assumed target position and Point S is the actual sensor position. The vector distance to the target, $\vec{\Delta}$, is the final result which is passed to the guidance package. $\vec{\Delta}$ is found by using the distances $d^{(T)}$ computed from the model, and the distances $d^{(S)}$ derived from the sensed range images. One can see that the difference between a matched pair, $d_i^{(T)} - d_i^{(S)}$, gives the component of the offset normal to the i^{th} plane. In the 3-dimensional case, $\vec{\Delta}$ could be uniquely determined by three non co-planar normals. To improve the accuracy, one uses the set of matched planes to compute a least squares solution.

An important attribute of our proposed approach is the close cooperation possible between the reference preparation task and the extraction of sensed primitives. Currently, the reference model is a list of planar surfaces to be matched with a similar list of sensed planes. However, the target scene

* Azimuth, elevation, and perpendicular distance of the plane to the sensor axes origin; the pair of location parameters; and the residuals.

may contain buildings or other structures with non-planar surfaces which cannot be incorporated directly into the model. This might happen if, for example, the target scene contained curved surfaces (e.g., silos, LNG tanks, etc.). The reference model can not simply ignore the existence of these structures, since they will appear in the sensed range image. We propose that the reference model be allowed to specify certain rectangular solid regions in the 3-D scene as containing unmodeled structure. Sensed surface primitives appearing to lie within the unmodeled regions would be ignored during the match process. If a more sophisticated model incorporating lists of curved (as well as planar) surfaces is subsequently developed, the automatic techniques used to describe curved surfaces will be applicable as well as the extraction of curved primitive surfaces from the sensed range image. We therefore propose and anticipate a parallel maturing of the modeling task and the range processing task.

Recent work at Lockheed has been directed toward demonstrating how well these ideas can be expected to work on real data, as typified by the range images and wire frame models from a test set provided by the DARPA Autonomous Terminal Homing program. These experiments, described in the next section, have concentrated on the feasibility of extracting planes from the range data. The success of these experiments makes it desirable to pursue the research to the point of developing and testing a complete software system based on this approach.

3.3 Experimental Results

A number of experiments have been performed to validate the concepts underlying the approach presented in Section 3.2. The tests were run on data base of four sensed range images and two synthetic range images displaying a variety of viewpoints of a single building site (Hughes Aircraft Company, Culver City, CA).

Several results pertain to the aggregation of pixels into well-defined primitive groups, each of which is at a single orientation. Figures 1 and 3 illustrate the geometry of the sensor and the local plane fitting. The goodness of fit at each point is measured by the residual. Pixels lying well within planar surfaces exhibit lower residuals than pixels lying near edges, since the 5×5 neighborhood fits better at surface interiors than at surface borders. Figure 5 displays the computed residuals (scaled for visibility) for both a sensed range image and a synthetic range image. The synthetic image is being used here only for ease of comparison. In practice, of course, the model is used analytically. Note that by considering pixels with low residuals it is possible to define regions corresponding to a single model surface.

Figures 6 and 7 illustrate two steps in the extraction of primitive planar regions. In the first, pixels on horizontal surfaces are identified as having unit normals projecting in the y direction. A single threshold segments the image into horizontal and non-horizontal pixels. These latter points are further separated by considering the value of $\cos x$, the component of the surface normal in the x direction. As can be seen in Figure 7, the vertical wall pixels are segregated into left and right wall points according to the value of $\cos x$ at each point. Each of the regions resulting from this segmentation defines a primitive for the match process.

The correspondence of sensed range primitives to model surfaces is illustrated in Figure 8. Identical segmentation applied to both a synthetic and a sensed range image produces primitive match regions which are strikingly similar. Moreover, the primitive match regions maintain their similarity regardless of viewpoint. This is evident in Figure 9 in which the four sensed range images are identically segmented into primitives. The primitives are well-defined and preserve the physical size and shape of the wall surfaces which they represent. These experiments support our approach by demonstrating that range images can be processed to yield regions which match model surfaces, thereby allowing an accurate position update.

4. CONCLUSIONS

This paper has explored model-based scene matching using range imagery. The advantages of using such imagery rather than more conventional reflectance (intensity) imagery are:

- Preservation of the three-dimensional nature of the scene, e.g., actual sizes, distances, etc.
- The surface primitives are directly established from the sensed data, not inferred.
- Insensitivity to shadows, specularity, time of day and viewing position.
- Contribution of all raster points (not just edges, lines, or corners) to position confirmation.
- Parallel evolution of the model building and model matching activities.

Preliminary experiments illustrate the identification of planar surfaces based on local properties of the range imagery as predicted by the model. Our results indicate that such primitives are strong cues for matching with the structural model. We anticipate that a reliable position fixing system can be based on this approach.

Acknowledgments

The authors wish to acknowledge O. Firschein for helpful discussions and L. Staley for expert typing.

5. REFERENCES

1. Shirai, Y., "Recent Advances in 3-D Scene Analysis", Proc. 4th IJCPR, Kyoto, Japan, November 1978, 86-94.

2. Horn, B. K. P. and B. L. Bachman, "Using Synthetic Images to Register Real Images With Surface Models", CACM, Vol. 21, Nov. 1978, 914-924.

3. Barrow, H. G. and J. M. Tenenbaum, "Recovering Intrinsic Scene Characteristics from Images", in Computer Vision Systems, A. Hanson and E. Riseman, eds. Academic Press, New York, NY, 1978, 3-26.

4. Nitzan, D., Brain, A. E., and R. O. Duda, "The Measurement and Use of Registered Reflectance and Range Data in Scene Analysis", Proc. of the IEEE, Vol. 64, No. 2, Feb. 1977, 206-220.

5. Duda, R. O., Nitzan, D. and P. Barrett, "Use of Range and Reflectance Data to Find Planar Surface Regions", IEEE PAMI, Vol. 1, July 1979, 259-271.

6. Newman, N. M., and R F. Sproull, Principles of Interactive Computer Graphics, McGraw-Hill, 1979.

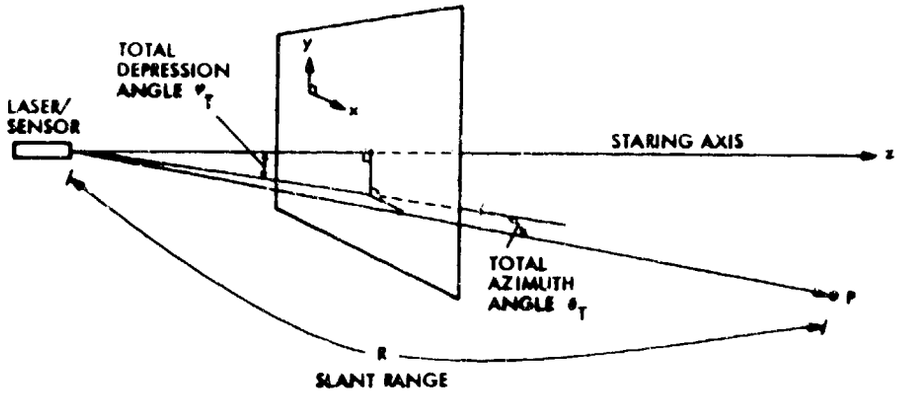


Figure 1 Raster Geometry of the Sensor

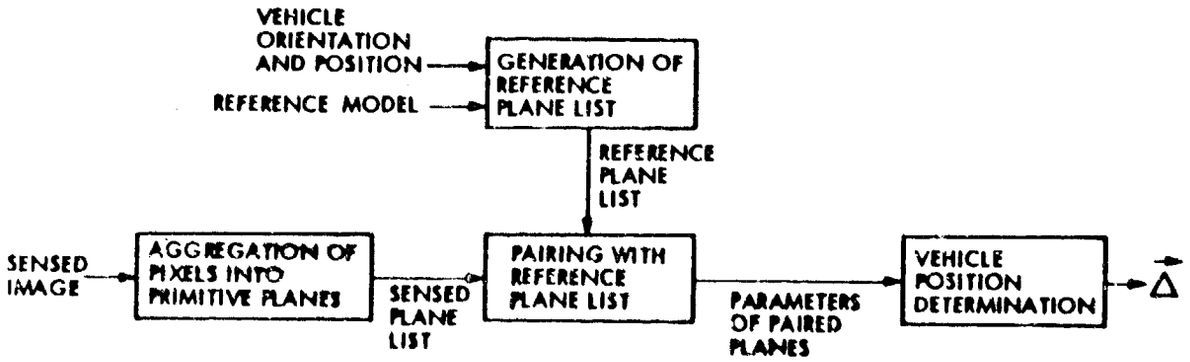


Figure 2 Block Diagram of Matching System

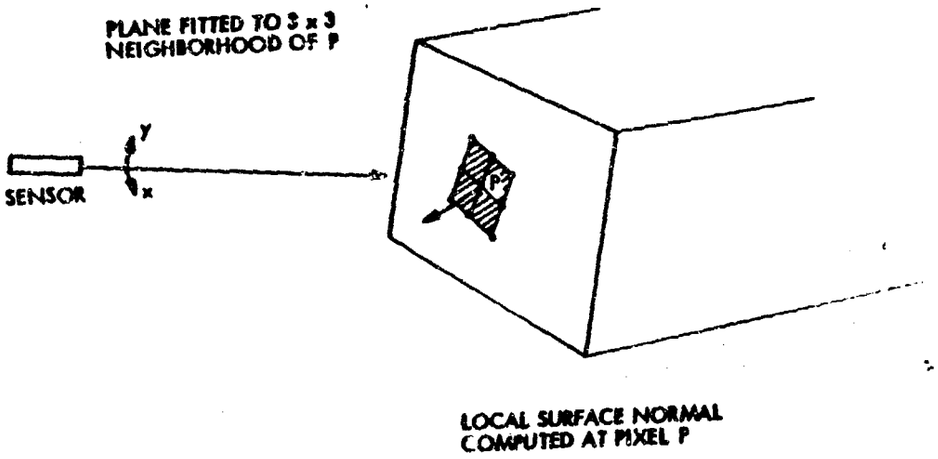


Figure 3a Computing the Local Orientation at a Pixel

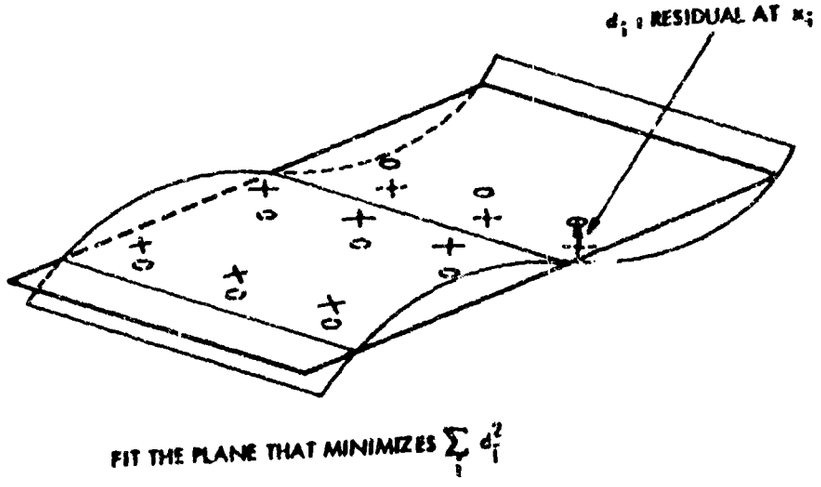


Figure 3b Planar Approximation to 3-D Data

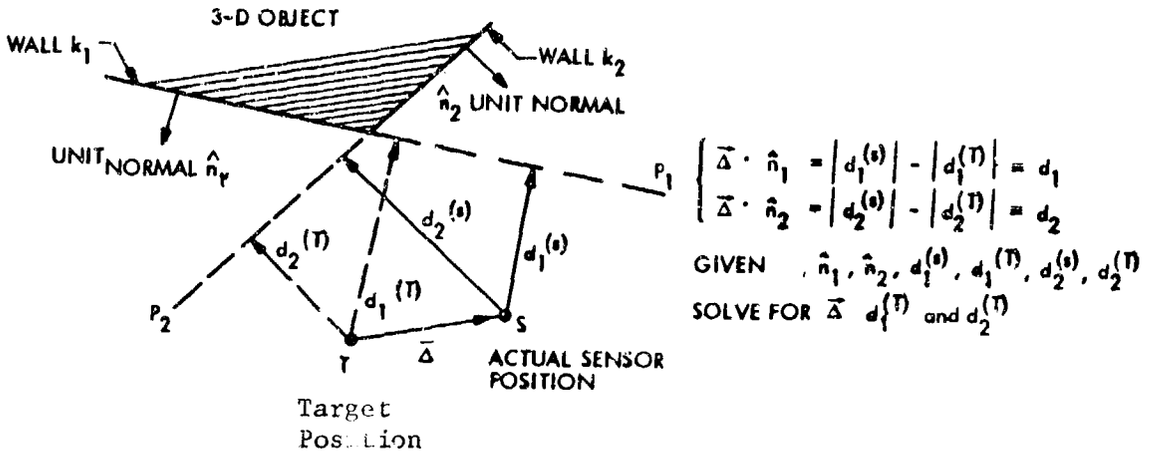


Figure 4 Two-Dimensional Example Showing Determination of Vehicle Location, $\vec{\Delta}$, From a Pair of Matched Planes

a b

c d

Figure 5 Residuals to Indicate Surface Interiors/Borders

a. Synthetic image residuals
b. Synthetic image
c. Sensed image residuals
d. Sensed image

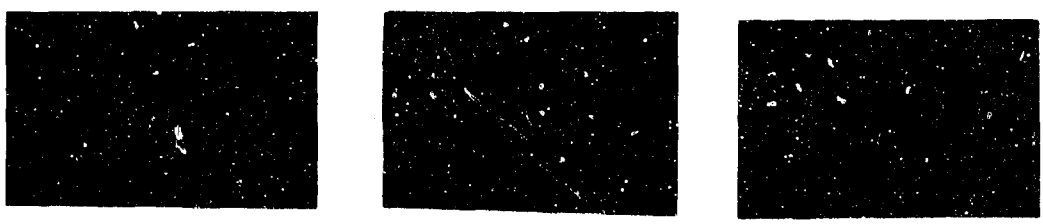
a b c

Figure 6 Horizontal Surfaces

a. Sensed range image

b. The normal component in the y direction (rescaled)

c. Thresholded y component - light points lie on horizontal surfaces



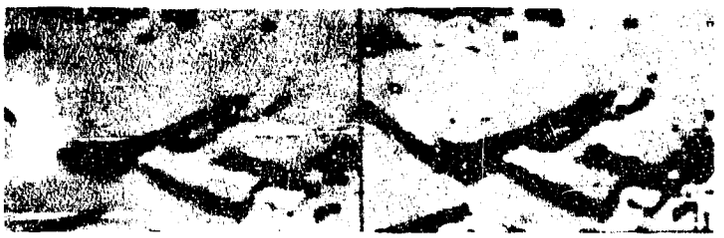
a b c

Figure 7 Vertical Surfaces

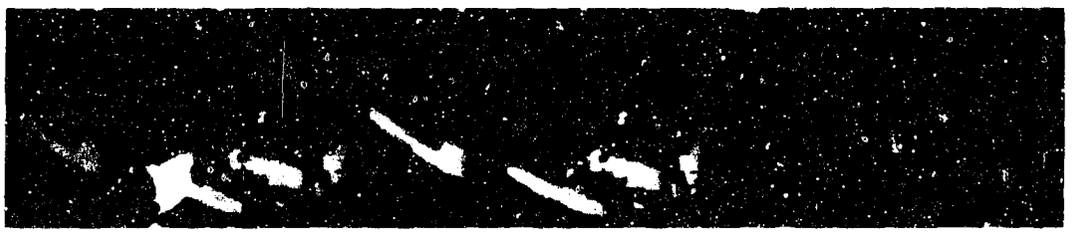
a. The normal component in the x direction (rescaled)

b. Left facing vertical surfaces

c. Right facing vertical surfaces



a b



c d e

f g

h i j

Figure 8 Correspondence of Sensed Range Image to Synthetic Range Image

a - e Sensed image; f - j Synthetic image

a,f Horizontal surfaces
b,g Horizontal surfaces (thresholded)
c,h Vertical surfaces
d,i Left facing surfaces
e,j Right facing surfaces

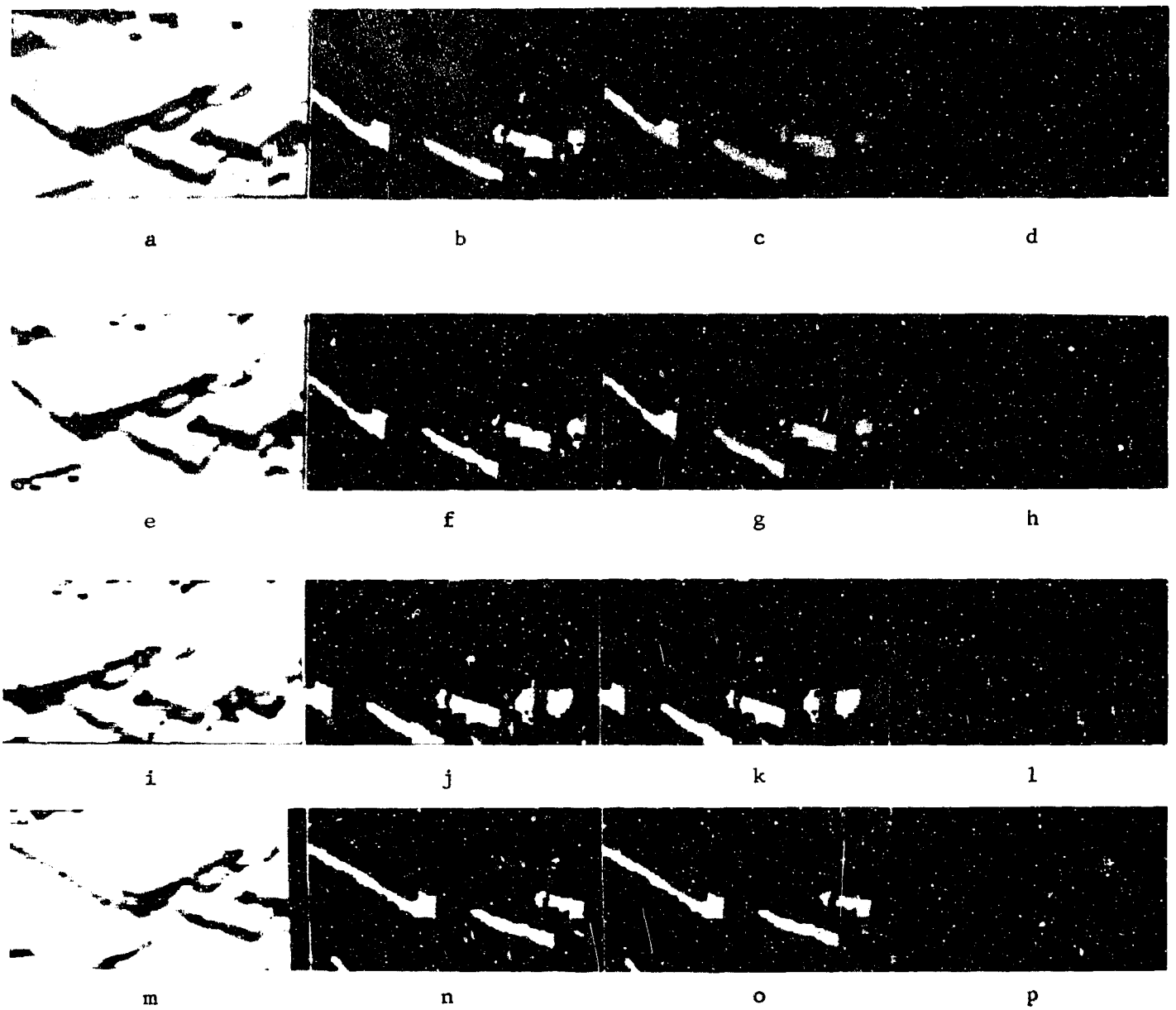


Figure 9 Stability of Primitive Regions at Differing Viewpoints

a,e,i,m Horizontal surfaces
b,f,j,n Vertical surfaces
c,g,k,o Left facing vertical surfaces
d,h,l,p Right facing vertical surfaces

Paper No. IA-4, Presented at the Workshop on Imaging Trackers and Autonomous Acquisition Applications for Missile Guidance, 19-20 November 1979, Redstone Arsenal, Alabama.

CLASSIFICATION/OBJECT RECOGNITION SOFTWARE TECHNIQUE APPRAISAL

R.J. ATKINSON

GENERAL ELECTRIC COMPANY
HUNTSVILLE, ALABAMA

ABSTRACT

An abundance of software techniques and variations of techniques exists for obtaining useful information from the large volumes of data generated by current sensors. Problems in system development are the choice of a suitable variation for a particular task, and determining the benefits of applying the processing in various sequences. As a result, the requirement exists to assess and evaluate representative techniques. This paper presents criteria and methods for evaluating information extracting techniques.

Also presented are results of applying various techniques in combination, such as edge detection and template matching, object detection and multispectral classification, geometric manipulation followed by classification.

WHY IS TECHNIQUE EVALUATION NECESSARY?

In recent years, there has been a great deal of interest in classification/pattern recognition techniques as evidenced by the hundreds of journal articles and books on the subject, with the main concentration of effort being on technique development. The large number of different approaches tends to indicate that no single approach is able to satisfy a large class of users.

Results evaluating different techniques have been published, usually by the original developer. This tends to preclude the application of other known techniques. Generally, the evaluations are performed on different computers, so that it is difficult to compare the operational characteristics of different techniques.

As a result, the evaluations that are available are difficult to piece together to obtain an overall appraisal concerning technique development. However, the appraisal is important because the utility of the systems is highly dependent upon the accuracy versus cost with which information can be obtained from imaging sensors.

The major part of the problem of obtaining a comprehensive evaluation of classification technique development is that it is a formidable task and requires an overall coordinated attempt at standardization of:

- evaluation criteria,
- unbiased evaluation procedures, and
- computer hardware and software programming practices.

Since the resources already existed within NASA, the establishment of such standardization practices became objectives of the Office of Applications Data Management Program in 1975. (Ref. 1) Inclusion of representative software for all existing techniques, evaluation by a variety of users, and dissemination of techniques and results were provided for by the formation of an Image Coding Panel representing six NASA centers.

Studies have also been undertaken by the General Electric Company in order to complement their involvement in the Landsat missions and development of image analysis systems such as IMAGE 100. (Ref. 2)

BACKGROUND: CLASSIFICATION/PATTERN RECOGNITION TECHNIQUES

The input to a pattern recognition system is a sequence of observations which are called measurement vectors or feature vectors. The user might have varying degrees of knowledge about the measurements. He might, in some cases, know the categories he is looking for and the ground truth (i.e., the class designations) at a small subset of locations from the remotely sensed image. When the ground truth is known, the method is said to be "supervised" and when there is no knowledge of ground truth the method is said to be "unsupervised." Another type of division is made depending upon the knowledge of the multivariate probability distribution for each class. When the distributions are known only in functional form with a finite set of unknown parameters to be determined on the basis of observed samples, this is called "parametric learning." Situations where even the functional form of the distributions are unknown call for "nonparametric learning."

EVALUATION CRITERIA

There are probably many ways to evaluate classification techniques, but from a user's point of view, the three most important areas of concern appear to be:

- the resources required to run the program and perform an analysis,
- a description of the analysis process, and
- the performance of the technique.

The quantitative aspects of the resource requirements are essentially concerned with the computer hardware necessary to run the program. The purpose of process descriptions is to provide a reasonable understanding of the classification analysis process and the role that a user plays in the analysis, as well as to contrast differences and highlight similarities between the various techniques. The performance characteristics are intended to indicate operational costs, cost/benefits, and maximum capabilities of the various techniques. Those quantities that can be enumerated are:

- computer time,
- relative accuracy in terms of direct pixel comparison of ground truth data and classification maps,
- maximum number of channels,
- maximum data set size,
- maximum number of clusters or classes,
- cost/benefit estimates in terms of relative accuracy and the use of conventional techniques, and
- manhours required by the user in the analysis.

The last two items tend to be subjective, since they depend on the type and quality of ground truth as well as on the human capabilities applied to the successive phases of photo-interpretation, classifier training, and iteration a number of times through the analysis process to attain satisfactory results.

PROCEDURES

In conducting a systematic assessment of classification techniques, certain procedures must be adopted to achieve consistency in results and to assure that relative comparisons have meaning. It is most important that measures of technique performance be free from biases introduced unintentionally by persons conducting the evaluation. Some of the principal factors to be considered in technique assessment include:

- choice of data sets and their preparation for analysis,
- use and treatment of Ground Truth Data to assure compatibility with the remotely sensed imagery,
- selection of samples within the imagery to be used for training supervised classifiers to recognize particular classes, and
- methods for comparing results of different classification techniques.

ACQUISITION OF DATA SETS

The data sets to be used in evaluating classification methods should be:

- sufficiently large and varied so that statistically significant numbers of data elements are present in several classes of interest,
- multivariate, since the majority of classification techniques are structured to analyze multivariate data, and
- similar as possible to data encountered in real applications.

Most of the tests were performed on a 1200 by 1200 pixel segment of Landsat data covering Mobile Bay, the City of Mobile, agriculture, forest, and wetland regions. Six scenes obtained from 1972 to 1975 were used. (Ref. 3)

ACCURACY COMPARISONS

An evaluation of the accuracy of the classification maps is a necessary part of comparing classification methods. The principal idea is to use joint histograms (contingency tables) to show the dissimilarities between maps. The joint histogram of maps 1 and 2 is defined as a Matrix A with

a_{ij} = number of simultaneous occurrences of classes i and j in maps 1 and 2, respectively.

In the case of comparisons between two maps with known labels, the map accuracy is defined as the total number of simultaneous occurrences of identical labels in the two maps expressed as a percentage of the total number of points in either of the maps. In terms of the joint histogram, this simply amounts to 100 Trace (A)/(Sum of all the elements of A).

INVENTORY COMPARISONS

In many instances, the users are simply interested in the inventories or the percentage occupancies of the various classes over a given region, rather than the point-by-point occurrences of the classes. It is reasonable to expect that the accuracy of the inventories derived from any classification method should be greater than the point-by-point accuracy of the corresponding classification map. The percent inventory accuracy is defined as

$$100\left[1 - \frac{\sum_{i=1}^{m} \left| P_{1i} - P_{2i} \right|}{2N}\right]$$

where P_{1i} and P_{2i} are the populations of the class i in maps 1 and 2, m is the number of classes, and N is the total population. The definition assures that the measure is between 0 and 100 percent, agreeing with the intuitive concept of similarity. This equation is derived by assuming 100 percent similarity, less the absolute value differences of the class populations as a percent of the total population. The division by 2 arises because samples in error are counted twice (as errors of omission and commission). An alternative derivation is obtained by counting for each class the populations in the two maps which are similar until one of the counts exceeds the population in either map. Thus, the equation has the form

$$100 \quad \frac{\sum_{i=1}^{m} \min\left(P_{1i}, P_{2i}\right)}{N}$$

CLASSIFICATION BY DENSITY SLICING

Density slicing refers to the process of identifying regions or objects in an image by choosing a range of densities (a density slice) corresponding to each. Inspection of multiband imagery reveals that significant classes of homogeneous terrain cover can be identified visually by the reflectance characteristics within single bands. The method is appealing because of its simplicity, since correlation of the reflectance values among several spectral bands is not required.

The density ranges can be chosen manually by examining the density values in each region of interest, or the spectral band and the density range for each class may be selected by a feature selection and linear classification algorithm restricted to one spectral band. The latter method was tested for inclusion in this report as the algorithms were available.

RESOURCE REQUIREMENTS

Input/Output

- number of classes and spectral bands present in the data,
- a set of training samples,
- tape of data samples to be classified with the measurements for each spectral band arranged in vector format, and
- output tape of classification results.

Program Memory

- an array to store the training samples indexed by class number, feature number, and sample number,
- input/output buffers,
- arrays to store the discriminant coefficients by class, and the order in which discriminants are tested,
- total work space dimensioned four times the number of training samples for use in training, and
- subroutine storage of 11.2 Kbytes.

ANALYSIS PROCESS

In order to verify that the best possible spectral band is chosen to discriminate any given class from all the others, a quantitative band (feature) selection method is applied first. Usually, this confirms what is visually obvious. Using a set of training data samples whose classifications are known, the signed distances of samples in different classes from the discriminant point are computed for each spectral band. The spectral band chosen is that for which the distance is a maximum.

Linear discriminant functions are then computed for each class, using the spectral band chosen by the above criterion for each class. The coefficients in the discriminant function are chosen by an iterative procedure. (Ref. 4) The two coefficients determined (constant term and data value multiplier) may be used in a linear discriminant function, as is done in this case, or may be used to calculate the density ranges occupied by each class.

PERFORMANCE CHARACTERISTICS

The algorithm can operate on large numbers of spectral bands and classes. The size of the data set to be classified is immaterial, as the classification is done on a point-by-point basis. The classification rate was 4450 pixels/second. The total storage required is 100 Kbytes using buffers for 1200 samples.

MAXIMUM LIKELIHOOD CLASSIFIER

The maximum likelihood classifier is a supervised, parametric technique and is probably the most widely known and used multichannel data classification method. A set of data samples, whose classifications are known, is required to define the parameters of the functions which are used to determine the classes of unknown data samples. The required parameters are those which define the Gaussian distributions for each class of the training data, namely the mean vectors and covariance matrices.

RESOURCE REQUIREMENTS

Input/Output

- number of classes and features (spectral bands) present in the data,
- the Gaussian parameters (mean vectors and covariance matrices),
- tape of data samples to be classified, in feature vector arrangement, and
- output tape of classification results.

Program Memory

- an array to store the training samples by class number, feature number, and sample number,
- input and output buffers,
- arrays to store the mean vectors and covariance matrices by class number, and
- subroutine storage of 8.4 Kbytes.

ANALYSIS PROCESS

It is assumed that the distribution of training data for a single class approximates the bell-shaped curve of the Gaussian or normal distribution.

The parameters--mean values and covariance matrices--completely define the Gaussian distribution functions. These parameters are easily determined for each class under consideration from the known set of training samples.

When the Gaussian parameters have been estimated, the Gaussian probability distribution for each class is completely defined. Thus, given any unknown feature vector, it is possible to compute the probability of this feature vector belonging to any one of the classes under consideration. Assignment is made to the class for which the probability is greatest; this is termed the maximum likelihood method of classification. For faster computation, the logarithm of the probability is computed and the decision function takes the form

$$G_i = \ell n\, P_i - \frac{1}{2} \ell n \quad K_i - \frac{1}{2} (X-M_i)^T K_i^{-1} (X-M_i)$$

where

P_i is the probability of class i being present, M_i is the mean vector, and K_i is the covariance matrix. The decision point between two classes occurs when the probabilities are equal, and is not midway between the means when the widths of the distributions are unequal.

PERFORMANCE CHARACTERISTICS

The performance of a maximum likelihood classifier with respect to accuracy and speed may be inferred from an examination of the method itself. If the data samples do obey the Gaussian distribution for each class, this method produces optimum results. However, the actual data samples belonging to a given class may produce a multimodal or skewed histogram. Typical causes of this effect in earth resources data are differing soil conditions, sun angle, crop health and maturity, and the widely varying reflectivity of man-made objects. In the case of such a multimodal distribution, the Gaussian parameters

which are computed do not accurately describe the actual distribution, and the classification accuracy is reduced. The maximum likelihood classifier is relatively slow because the classification of a data sample requires the evaluation of the decision function for each class being considered.

This method will operate satisfactorily on large numbers of spectal bands and classes. The size of the data set to be classified is immaterial to the process, as each data point is classified independently. Approximately one second is required to compute the distribution functions. The classification speed for six classes is approximately 650 pixels/second. The total storage required is 90 Kbytes using buffers for 1200 samples.

LINEAR CLASSIFIER

The linear classifier described here is a supervised, nonparametric technique. Thus, the initial phase of the classification process consists of the definition of a set of discriminant functions using data samples whose classifications are known.

In separating one class of objects from one or more other classes, it is desirable to de-emphasize the characteristic features that the classes may have in common, and to emphasize where possible the features that are unique to the class of interest. The Linear Classifier concept depends upon this assumption, and aims at developing a single measure of a class's composite features. This measure, the discriminant, is formed by adding the value of each feature (reflectance value or brightness in the case of multiband imagery), after each feature has been weighted according to its usefulness in separating the class of interest from the other classes.

RESOURCE REQUIREMENTS

Input/Output

- the number of classes and spectral bands in the data,
- a set of training samples,
- tape of data samples to be classified, arranged in feature vectors, and
- output tape classification results.

Program Memory

- an array to store the training samples by class number, band number, and sample number,
- input and output buffers,
- arrays to store the discriminant coefficients by class, and the order in which discriminants are tested,
- total work space dimensioned (three plus the number of bands) times (the number of training samples) for use in training, and
- subroutine storage of 11.1 Kbytes.

ANALYSIS PROCESS

Nonparametric methods are so termed because the parameters of the distribution functions of the data are not used. The training algorithm determines the values of the weighting factors "w" to be used in a discriminant function of the form

$$G = w_0 + w_1x_1 + w_2x_2 + \ldots + w_nx_n$$

A set of weights is determined for each class of data, the value of a weight reflecting the significance of its associated feature in separating the class from its companion classes. Thus, for each unknown feature vector, a value of G is obtained for each class. There are two approaches possible in the application of linear classifiers. In the first, the discriminant functions are designed such that one class may be separated from each of the other classes, pair wise.

In the second approach, the one employed at NASA-MSFC (Ref. 5), the discriminant functions are designed such that one class may be separated from all of the other classes considered collectively as one class. Unlike the first approach in which all discriminants are calculated concurrently, here the discriminants are calculated sequentially. The sequential nature of testing results in a speed advantage over the parallel procedure employed in the first approach. The class which is to be separated from the others should be widely separated from the discriminant hyperplane and from the other classes. The criterion used is the sum of the signed distances of the training data samples from the plane. Samples which are incorrectly discriminated are given negative distances. The coefficients of the discriminant function are determined by setting up a system of discriminant equations (one for each training sample). The method consists of maximizing the total distance of the training samples from the discriminant hyperplane. (Ref. 4) This process is repeated for each class until a single class remains. Samples are classified by evaluating the discriminant functions sequentially until a positive value is obtained.

PERFORMANCE CHARACTERISTICS

This method will operate satisfactorily on large numbers of spectral bands and classes. The size of the data set to be classified is immaterial.

Approximately one minute is required to compute the discriminant coefficients. Data was classified into six classes at the rate of 4590 pixels/second. The total storage required is 125 Kbytes using buffers for 1200 samples.

SPATIAL AND SPECTRAL CLUSTERING PROGRAM (SSCP)

The SSCP can be run in either an unsupervised or supervised mode and is composed of two modules which are run separately. The first module allows a user to select training areas manually or will automatically select training areas based upon the spatial and spectral characteristics of the data set and automatically merges data from training areas that are spectrally similar. The second module classifies each individual pixel according to whether or not it belongs to one of the described classes. Each class is described by a mean vector and a set of eigenvectors and eigenvalues, which are derived from module one and used in module two. The classification is thresholded, which usually results in some pixels remaining unclassified.

RESOURCE REQUIREMENTS

The resources required by a user, as far as a knowledge of the data set is concerned, can range from very little knowledge to considerable detailed information, since the program can be run in either a supervised or unsupervised mode. There are two reports that mathematically describe the program and present results on aircraft scanner and Landsat multispectral data. (Ref. 6, Ref. 7)

The program, as it is currently used, is run in two parts. The first part acquires the statistics necessary to classify the data and uses 206 Kbytes of core memory. This part of the program also utilizes four tape drives and eight sections of disc storage, each of which contains 2340 blocks (records) of 1028 bytes. One of the tapes contains previously acquired statistics, if there are any, the second tape contains the reformatted data, and the third tape contains the output statistics used in classifying the data. The fourth tape is optional and contains the cluster map.

The second part of the program classifies the individual pixels based upon the acquired statistics and utilizes 110 Kbytes of core memory. This part of the program also utilizes three tapes which contain the input statistics, the input reformatted data, and the output classification map. One section of disc is reserved that contains 2340 blocks of 3300 bytes.

ANALYSIS PROCESS

The program contains two modules which are presently run separately. The first module performs three different operations on the data, while the second module only classifies the data. Thus, the entire program consists of a boundary routine, a spatial clustering routine, a spectral merging routine, and a classification routine.

The purpose of the boundary routine is to establish boundaries when the spectral vector distance between the pixel in question and the previous adjacent scan and column pixels is large enough. The spatial clustering routine uses the boundary map as an input and searches the boundary map for homogeneous areas. The purpose of the spectral merging routine is to determine which spatial clusters are spectrally similar and which ones are spectrally distinct. The inputs to this routine are the raw data and the cluster map or training area coordinates which provide the program with information on where to fetch the raw data for each cluster. Once the data have been fetched, the following quantities are calculated for each cluster:

- pixel population,
- mean value for each channel (i.e., mean vector),
- covariance matrix,
- eigenvectors, and
- eigenvalues.

The data belonging to each cluster are then enclosed by a surface in the multispectral space whose dimension is equal to the number of channels of data. This closed surface is a hyperellipse whose center is the mean vector, whose orientation is given by the eigenvectors, and whose extent is governed by the magnitude of the eigenvalue associated with its eigenvector. The rule for spectrally merging two clusters is that the mean vector of each cluster must be contained in the other cluster's closed surface. When two or more clusters are spectrally

merged, the previously mentioned quantities are recalculated for the combined data of the merged clusters. Once the merging process has been completed, the remaining clusters are called classes.

The classification program then classifies each pixel as to whether it belongs to a particular class or none of the classes. The rule for classification is that the pixel vector first must be contained within the closed surface defining a class, and, secondly, if it is contained within more than one such surface, the pixel vector is assigned to the class whose center location (mean vector) is the closest. The inputs to the classification program are the raw data and the class statistics which are the mean vectors, eigenvalues, and eigenvectors.

PERFORMANCE CHARACTERISTICS

SSCP is dimensioned to handle a maximum of 12 channels of data, 150 unmerged clusters, or 46 input training areas, and 42 final classes. The clustering part of the program is dimensioned to handle a strip of data 256 columns wide and as many scans as desired, while the classification program is dimensioned to handle 824 columns of data and as many scans as desired.

In the unsupervised mode, the program tends to work best on large data sets where there is the opportunity to find large (typically 10 x 10 pixel arrays) homogeneous areas on the boundary map.

The output of the classification program is a tape containing the classification map and a listing of the class population and percentages. Typically, it is possible to classify at least 90 percent of the data using the program in the unsupervised mode. The urban category is usually the most difficult to classify using the unsupervised mode because urban areas tend to become boundaries.

In the clustering program, the length of time required to produce a boundary map is directly proportional to the number of pixels in the data set or about 500 pixels per second and the spatial and spectral merging rate varies but falls in the range 300 $\pm$ 60 pixels/second. The classification program time also appears to be linear with the number of pixels for a given number of classes, except for a variation due to the percent of the data classified. This variation is caused by the classification logic which checks the class assignment made to pixels spatially adjacent to the pixel in question. As a result, the classification rate ranges from 450 to 660 pixels/second.

HISTOGRAM INSPIRED NEIGHBORHOOD DISCERNING UNSUPERVISED (HINDU) SYSTEM

This technique comes under the category of unsupervised, nonparametric classification techniques and is most suited for application to environments wherein neither ground truth nor information about the distributions underlying the data are available. The program is highly automated and requires little human interaction. User's subjective influence on the process is limited to specification of the maximum and minimun limits on the number of clusters, and the approximate range in the values of measurements.

RESOURCE REQUIREMENTS

The program, as is currently implemented, is dimensioned to handle four dimensional data sets. There is no critical limitation on the number of scan lines and the number of data points/scan line, and accordingly, there is no strict limitation on the size of the data set. A scene of 1200 data points/scan line calls for a core memory requirement of 170 Kbytes. For input/output of data sets and labels, two tape drives are called for by the program.

ANALYSIS PROCESS

The major components of the HINDU system are:

- Histogram Generator,
- Cluster Formulator,
- Discriminant Designer, and
- Label Designator.

The function of the histogram generator, as the name implies, is to generate the multidimensional histogram of the input data set. This histogram analysis leads to a set of multidimensional cells occupied by the input data set. The cell widths cover several data levels, effectively smoothing the histogram. The output of this histogram generator consists of arrays containing the measurement space addresses, frequencies, and data averages for each cell.

The output of the histogram generator is processed by the cluster formulator to create the clusters (of cells) and define their boundaries. This is achieved by a sequential procedure consisting of the following steps:

- identification of the current lowest density cell,
- connection of this cell to its higher density neighbors by reassignment of the contents of this cell to these neighbors in proportion to their current density levels,
- storage of these connections in memory in the form of a connectivity matrix, and
- updating of the density and average arrays to reflect the changes due to reassignment.

This sequential processing is continued until all the originally non-empty cells are processed. As is to be expected, this processing leads to a finite number of cells whose contents remain unassigned, there being no higher density neighbors to these cells. These cells are considered as candidate cluster nuclei and those deemed significant have their updated density values higher than a threshold value. The connectivity matrix can then be processed to trace out the connections of each cell up to these significant cluster nuclei and thereby identify the clusters of cells surrounding each nucleus cell. Such cells are considered to represent the fuzzy boundary separating the corresponding clusters.

The discriminant designer determines the set of hyperplanes which discriminate between each pair of clusters. The conventional methods of learning the discriminant functions based on error-correcting procedures and solution of linear inequalities are not well suited in view of the fact that there exists a significant amount of information in terms of cells representing the fuzzy boundaries.

The methodology adopted here tackles this modified problem environment by ensuring that the hyperplane represents an optimum fit to the fuzzy boundary in addition to fulfilling its traditional role of being a discriminant between the two identified clusters. This is achieved by viewing it again as a linear inequality problem, but with certain additional minimization constraints and establishing an equivalent unconstrained linear inequality problem amenable to conventional techniques. (Ref. 8) (Here, the Ho-Kashyap algorithm (Ref. 4) is adopted to handle the equivalent unconstrained linear inequality problem.)

The label designator essentially consists of a table of class numbers corresponding to the centroids of the histogram cells. The class numbers of the individual samples are derived by looking up this table for the corresponding entries. The class numbers of the centroids are determined by the discriminant hyperplanes designed earlier.

PERFORMANCE CHARACTERISTICS

The method is designed for processing relatively large data sets of moderate dimensionality under unsupervised environments wherein computational economy is a significant factor in dictating the choice of the technique to be employed. This method does not involve intersample distance computations, a common feature of many other clustering approaches, and hence the computational load increases only linearly with increase in data size. The execution speed is somewhat dependent on the number of clusters, but is near 4000 pixels/second overall.

SUPERVISED TABLE LOOKUP METHODS

All of the previously described supervised classification techniques are applied to the multispectral vector occurring at each picture element location. Using this approach, the classification time will be proportional to the number of classes and the number of picture elements. When processing speed is a major consideration, a considerable advantage can be obtained by using a table lookup technique.

There are two methods for determining the extent of the table required to accommodate the input data. The first is to construct a table adequate to classify all vectors with components in the expected range of the data. The second is to find all of the unique measurement vectors in the image data and their frequency of occurrence and label the picture element locations with a number that identifies the vector that belongs there. Any classification approach can be used to obtain a classification inventory from the table of vectors, and a classification map can be produced by replacing each picture element vector number with the corresponding class number.

ELLTAB

The name ELLTAB stands for ELLiptical TABle, which gives a partial description of the program. ELLTAB is a version of the (supervised) Gaussian maximum likelihood method, implemented using a table lookup technique. The program is an application of the general table lookup pattern recognition method devised by Eppler. (Ref. 9) The general idea of the method is, in the training phase, to precompute the possible results of the decision rule, as a function of

position in feature (measurement) space, and store them in a table. Then, in the classification phase, each measurement vector is used to enter the table, which tells the class.

In constructing the table, each possible result only needs to be computed once, while in conventional implementations of pattern recognition techniques, the same calculation could be performed several times. The time per point required for the classification phase is approximately proportional to the number of classes. Since the classification itself is performed simply by looking up results in a table, the time required is not at all dependent on the classification rule used in preparing the table.

RESOURCE REQUIREMENTS

ELLTAB was originally written in FORTRAN V for the UNIVAC 1108 computer. Conversion of a program from one computer to another may be much less than straightforward, and unless considerable time and effort are expended, an inferior version of the program might result. For these reasons, ELLTAB was tested on the 1108.

ELLTAB consists of two executable modules, ELIPSE and ASSIGN. Each contains a main program and several subroutines. ELIPSE constructs the lookup table (training phase), which is then used by ASSIGN to classify a scene (table lookup phase). The two modules are executed separately. ELIPSE requires about 30K words of core storage. About 70 percent of this space is used for dat storage. One tape drive is required for the (output) table tape. The other executable module, ASSIGN, requires about 27K words of core storage with the array used to hold the combined lookup table for all classes dimensioned 9000. Three other buffer arrays corresponding to three tapes are used: the table tape, the (input) data tape, and the (output) classification tape.

ANALYSIS PROCESS (Ref. 10)

The table lookup method of pattern recognition is motivated by a desire to reduce the total amount of computation required for classifying large data sets, possibly using complex decision rules. After a step that partitions feature space into regions according to some decision rule and constructs tables incorporating this information, classification of multispectral data is performed simply by entering the tables, which have a form essentially independent of the decision rule.

The table-building phase could use any method of partitioning measurement space and constructing tables. ASSIGN explicity uses the Gaussian maximum likelihood method. The tables describe hyperellipsoids in four-dimensional space. Assuming first that the regions for the classes do not overlap, the statistics derived from training data are used to determine the ellipsoids. The sizes are given by the quadratic threshold values Q specified. Table size is sensitive to the value of Q. If there is no overlap between classes, nothing else is necessary. For regions of overlap, points are assigned to the class for which the likelihood discriminant function has the greatest value.

In the classification phase, the preceding class is assumed. If that hypothsis fails, the other classes are tested in order of decreasing a priori probability. The testing is done as follows: one component at a time of the point is tested to see whether it lies within the permissible range of values for that class. The tables, then, contain a description of the class boundaries, along with "pointers" to tell where in the tables to look to find the limits for the next test. The order of utilizing components of measurements is chosen for each class to minimize the size of the table for that class.

PERFORMANCE CHARACTERISTICS

Since ELLTAB is an implementation of the multivariate Gaussian maximum likelihood decision rule, its performance (e.g., with regard to the type of classification errors it may yield, etc.) should be similar to that of other implementations of the method. Because of the quadratic threshold feature, some data points will generally be assigned to the unclassified "class."

There is essentially no limit to the number of data points ELLTAB can process in a single run, since it classifies one scan line at a time.

The value of Q corresponding to excluding an average of 100 points from each of six classes (an exclusion probability of 0.01154) is Q = 12.96. Using this value for each class, the run of ELIPSE to make a table tape took 0.8 minute (CPU time). The time for classification was 250-300 microseconds per pixel or up to 4000 pixels/second. Probably, large homogeneous areas could be classified faster than regions where there are frequent changes between classes. Classification time should increase with the number of classes (as in the case with other classification programs).

VECTOR CLASSIFICATION

Classification time can be significantly reduced if the unique vectors and the number of times that they occur are extracted from the image for an inventory. A classification map can also be constructed by replacing the multispectral vector at each picture element with one number that identifies the vector that belongs there, and then by replacing the vector number with the class number to which that vector was assigned using a table lookup procedure. Thus, each unique vector is only processed once, and the answer may be applied many times. The classification time will then depend more on the number of unique vectors in an image which is typically less than five percent of the number of picture elements.

RESOURCE REQUIREMENTS

The program which calculates the histogram of the four-dimensional Landsat vectors requires arrays for storing the vector components (four components per word) and the frequency of occurrence of each vector. These should be of length approximately 50 percent greater than the expected number of vectors. The subroutine requires 3770 bytes of storage. This step is followed by the implementation of a classification routine which classifies the table of vectors. The modifications required are slight for those classifiers which accept vector input.

ANALYSIS PROCESS

A straightforward table of occurrences cannot be used because the maximum possible number of vectors from Landsat data is 128 x 128 x 128 x 64 = 134,217,728. Consequently, a divisor, base, and multiplier are applied to a vector to compute a location in a shorter table. Each component of the vector is divided by the divisor to obtain the remainder for each component. Using the specified base, the remainders are used to obtain a four digit number. Since this number is not unique with respect to input vectors, the number and hence the available table locations are multiplied by the multiplier. This final number is the table location at which the search for new vectors begins. Additional details and results are given in Reference (11).

The table location of each data sample is written in a file, the table of vectors is classified, and the numbers in the file are replaced with class numbers.

PERFORMANCE CHARACTERISTICS

Because of the vector tables, the storage requirements are high, typically 500 Kbytes for a large image.

The histogram generation rate varies greatly with the distribution of vectors, but is approximately 9000 input vectors per second. The execution time increases if the length of the frequency table is not somewhat greater than the number of vectors found (due to greater searching to bypass the full regions of the table).

The overall processing rate to extract the unique vectors, determine how many times they occur, and to label the picture elements with the corresponding vector numbers; to classify the unique vectors; and, to convert the vector numbers to a class number for each picture element is 6500 pixels/second. The vast majority of the time is spent extracting the vectors and labeling the picture elements, but the total process is still one and one-half times faster than using the linear classifier to classify a vector at every picture element.

VECTOR REDUCTION PLUS CLASSIFICATION

The most obvious way to further reduce processing and storage costs is to approximate the multispectral imagery and hence reduce the number of unique vectors contained in the table (Ref. 12). The effects of such a reduction on processing costs and classification results were examined by combining vectors with their neighbors. The reduction was accomplished by superimposing a grid, of spacing greater than unity, on the measurement space and changing the value of each vector component to that of its nearest grid point. Thus, the vectors contained within cubes the size of the grid separation are merged to a centroid. The grid separations can be increased until the spectral structure of the data is smoothed to the extent that multispectral classification is hindered. The limit of vector merging can be established by requiring that a sufficient number of natural clusters remain in the data. This may be accomplished by a clustering technique such as the HINDU system described earlier. In this study, three classification techniques using the centroids of the occupied grid cells were employed. These additional techniques are supervised, in that a set of training samples, whose classification is known, is input to the system.

RESOURCE REQUIREMENTS

The computer resources employed by these classifiers are similar to those given previously for the HINDU system, viz. a core memory requirement of 170 Kbytes, two tape drives for input and output, and external storage for the histogram cell addresses. An additional input requirement is a set of labeled training samples.

ANALYSIS PROCESS

The three classification techniques used in this system were:

- nearest neighbor
- maximum likelihood
- piece wise linear

The training samples are used to evaluate the required parameters of the classification algorithm, such as the Gaussian parameters of the distributions or the coefficients of linear discriminant functions.

The table lookup is applied to a smaller set of vectors only instead of all possible feature measurements. The reduced set is defined here as the centroids of the contents of all the occupied cells resulting from a multidimensional histogram analysis. The centroids of all the occupied histogram cells are classified on the basis of nearness to one or the other of the training samples, by using the classical maximum likelihood classifier with the estimated parameter values, or by a set of discriminant hyperplanes whose parameters are determined by the set of training samples The classification, or table lookup, phase requires the use of the incoming easurement vector to locate the proper element of the table, which contains the class.

PERFORMANCE ASSESSMENTS

The CPU time required is divided between the histogram analysis time and the table creation and lookup time. The overall processing rate is approximately 3600 pixels/second.

ERRORS DUE TO VECTOR REDUCTION

Since the reduction in the number of vectors was accomplished by changing the component values, it is necessary to examine the error introduced in the data and consequently in the classification. If the error is deemed too large, it may be decided to merge only those vectors with a low frequency of occurrence. This will be a large number of vectors, with a correspondingly large reduction in the length of the table of vectors required to describe the image. However, this large number of vectors represents only a small part of the image area, due to their low occurrences. For this study, all of the vectors from a 1.44 million pixel scene were extracted and some or all of the vectors were reduced in a nearest neighbor fashion, i.e., merged in groups of three so that the maximum change in a component was ± 1.

If the distribution of unique vectors and the variances per spectral band are known, it is possible to predict quite accurately the average mean square error per band with respect to the original data. For nearest neighbor merging, two-thirds of the numbers in the spectral distributions will change by an absolute value magnitude of one.

The following table (Ref. 13) shows the predicted and actual average mean square error per band, the number of vectors left after the reduction, the inventory accuracy, and classification map accuracy for the cases of not merging any vectors, for merging vectors that occur 15, 30, and 45 times or less, and for merging all of the vectors. Sequential linear classification was used. It can be seen that NN merging of all components has a small effect on the imagery and the classification results while reducing the vector table length by a factor of 11.4.

FREQUENCIES OF MERGED VECTORS	NUMBER OF VECTORS	AVERAGE MEAN SQUARE ERROR PER BAND PREDICTED/ACTUAL	INVENTORY ACCURACY PERCENT	CLASSIFICATION MAP ACCURACY PERCENT
NONE MERGED	27696		98.24	72.46
1-15	8525	0.0314/0.0315	98.18	72.43
1-30	6694	0.0502/0.0501	98.11	72.32
1-45	5856	0.0649/0.0649	98.10	72.26
ALL MERGED	2420	0.667 /0.671	95.49	70.35

GE IMAGE 100 CLASSIFICATION TECHNIQUES

Although direct comparisons cannot be made on the basis of IBM 360 classification speed, it is important to consider the impact of an interactive system on the appraisal of techniques. Consequently, the performance of three classifiers implemented on a PDP 11/45 based IMAGE 100 were evaluated by General Electric. (Ref. 2) A standard maximum likelihood method was included and need not be discussed.

PARALLELEPIPED CLASSIFIER

In brief, the IMAGE 100 operates simultaneously, under human supervision, on two to four bands (generally) of Landsat data. The operator, interacting with a display of classification results and/or histogram displays, selects upper and lower data bounds relative to specific training sites representing a known class of material. If desired, the selected upper and lower bounds can be applied directly to the displayed scene segment and interactively modified to establish a final classification in virtually real time. The upper and lower bounds established by the operator for each class are the only data points determining the resulting classification, an advantage because, in spite of histogram distortion, the upper and lower bounds of a class are relatively stable in the presence of compression/decompression and calibration, whereas the distribution of counts between limits is distorted.

ANALYSIS PROCESS

In training, multispectral brightness data (gray levels) within the training area are automatically measured, and their upper and lower spectral limits are used to define a single spectral cell. This spectral cell is the first-cut signature of the class within the training set. All screen pixels that lie between the bounds of this signature are then identified on the color monitor image display.

Further refinement of these signatures was possible through a manual interactive refinement technique. The objective of this signature refinement was to obtain spectral signatures with characteristically low omission and commission errors This interactive procedure, called histogram trimming, allows the machine operator to adjust the range (large cell gray level limits) of any one or all of the Landsat spectral bands that comprise the four channel signature.

PERFORMANCE CHARACTERISTICS

The parallelepiped approach applied in an interactive mode appears to be an effective classification tool. An advantage of this approach is that the operator can see the results of his classification immediately on a class-by-class basis. The operator can evaluate his classification in terms of both visual classification maps and histogram graphic displays.

FEATURE SPACE CLASSIFIER

The feature space classifier, as currently implemented, is a two-axis classifier. These axes can be defined as two selected Landsat bands, ratio of bands, principal components or virtually any combination of data space finally reduced to a two-axis projection. In essence, it is a parallelepiped classifier; however, the graphic presentation of two-axis feature space coupled with the highly interactive mode of operation eliminates the need for designating training sites.

ANALYSIS PROCESS

The classifier utilizes a non-parametric approach that depends on the interactive definition of multispectral classes in two dimensional feature space. The approach is normally applied by partitioning a feature plot of MSS 5 versus 7 spectral space of a Landsat subscene. The feature axes, however, can be defined as any other bands or as a variety of band combinations. Although training sites are not required, improved separation in two bands may be obtained by rotating the measurement space axes to be along the eigenvectors of the covariance matrix.

PERFORMANCE CHARACTERISTICS

Feature space partitioning is a highly interactive technique that can be performed quickly and yields very accurate results. Use of two dimensional data allows display of all channels and a high level of interaction in selecting upper and lower bounds in two-space and observing the result on the color display of the scene. This highly interactive approach efficiently couples spatial pattern recognition and context perceptions of the operator with the number crunching capabilities of the machine. This approach has been applied successfully in many other classification exercises. The results indicate that the non-parametric approaches tested (parallelepiped and feature space) have advantages over the tested parametric approach (maximum likelihood) when compared in terms of classification accuracy, processing time requirements and operational considerations.

All three approaches yielded approximately equivalent accuracy results. The time required to perform the complete analysis of the study area (from data input to numerical results extraction) varied considerably with the classification approaches. First, parallelepiped and feature space approaches require far fewer digital operations per pixel than the maximum likelihood classifier to assign a pixel to a class. This number of processing steps becomes an important consideration as the demand on a processor increases, especially if interactive rates are desired. Secondly, the most time-consuming function in both the parallelepiped and maximum likelihood approaches is definition of training sets.

Ease of operation comparisons among the three approaches is difficult because the operation depends on how the approach has been implemented on a processing system. Of the three approaches tested as implemented on the IMAGE 100 systems used, feature space partitioning was the most efficient, followed by parallelepiped and finally maximum likelihood.

CLASSIFICATION SUMMARY

The performance characteristics are summarized in the following table and in Figures 1 and 2.

CLASSIFICATION METHOD	STORAGE REQUIRED KBYTES	CLASSIFICATION SPEED PIXELS/SECOND	INVENTORY ACCURACY PERCENT	CLASSIFICATION MAP ACCURACY PERCENT
DENSITY SLICING	100	4450	77.5	66.5
MAXIMUM LIKELIHOOD	90	650	93.8	70.4
SEQUENTIAL LINEAR	125	4590	92.6	69.4
SPATIAL-SPECTRAL CLUSTERING	206	660	79.3	65.5
HINDU CLUSTERING	170	3900	92.7	79.4
ELLTAB (1108)	120	4000	87.9	71.9
VECTOR CLASSIFICATION	500	6500		
REDUCED VECTOR	170	3600		
1. NEAREST NEIGHBOR			81.7	61.2
2. LINEAR			90.0	68.8
3. MAXIMUM LIKELIHOOD			91.3	69.7
IMAGE 100				
1. MAXIMUM LIKELIHOOD			95.9	
2. PARALLELEPIPED			92.3	
3. FEATURE SPACE			97.2	

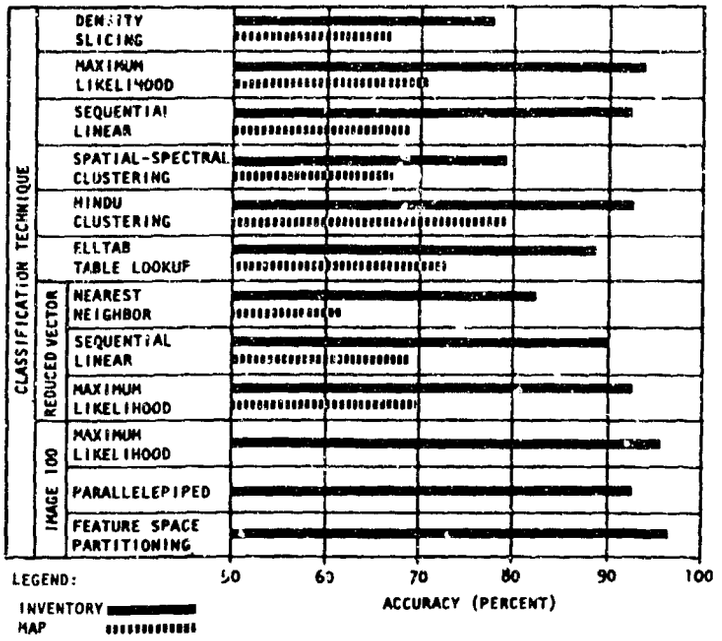


Figure 1. Performance Characteristics (Accuracy)

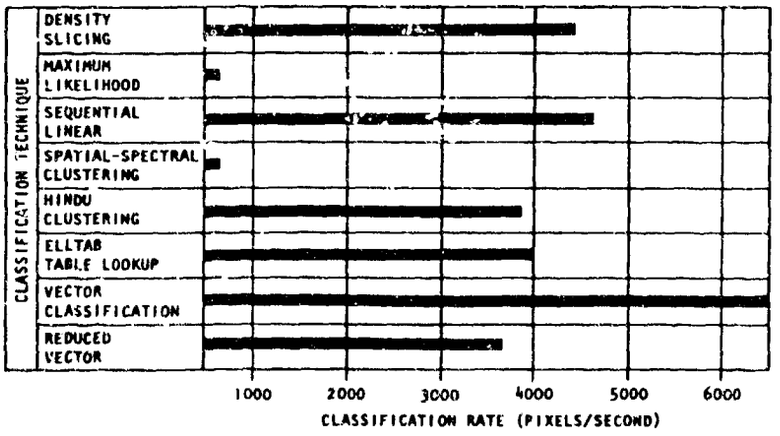
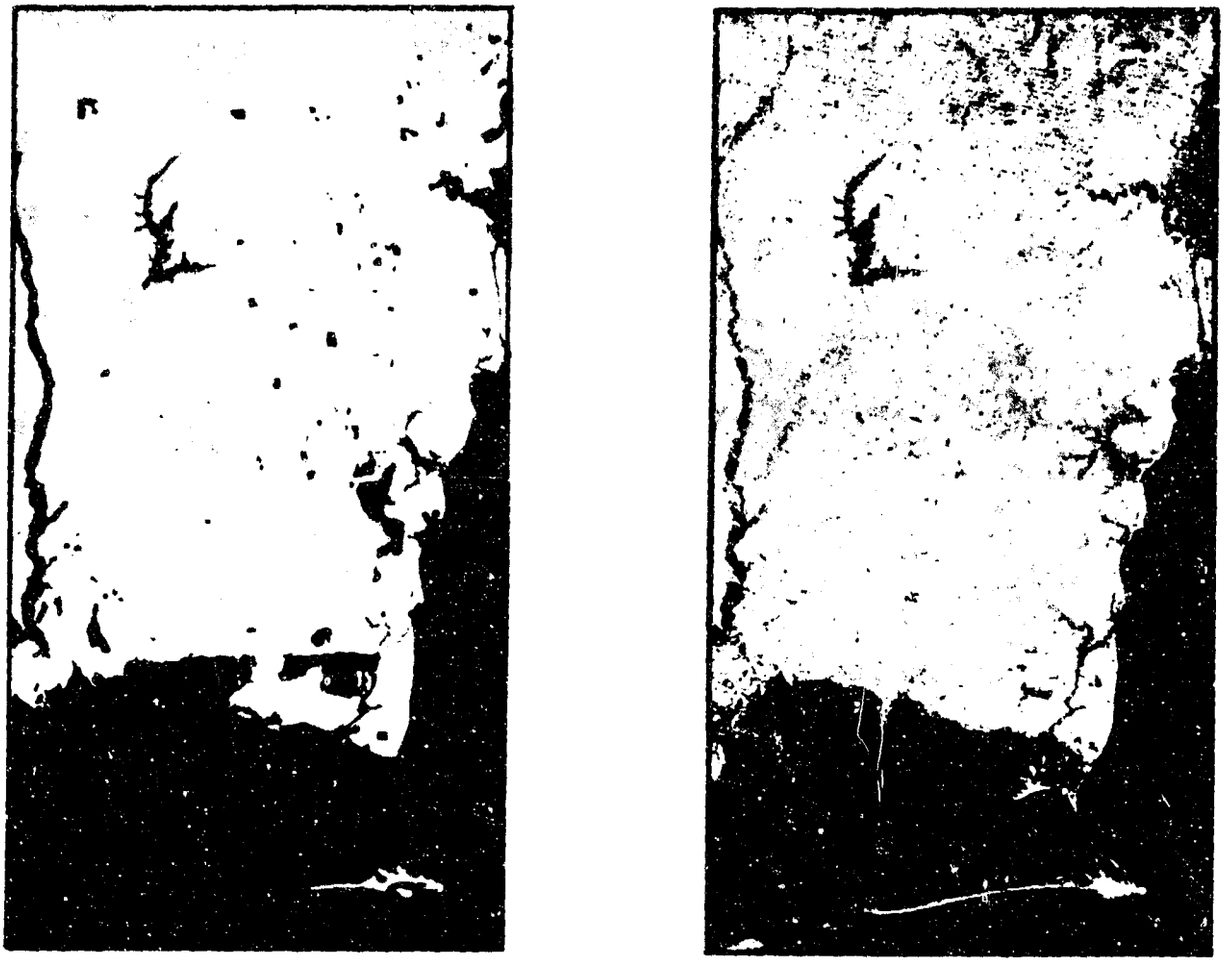


Figure 2. Performance Characteristics (IBM 360/75 Rate)

A pictorial example of a classification map compared to a ground truth map is shown in Figure 3.



(a) Ground Truth Map

(b) Sequential Linear Classification of December 1973 Landsat Data

Figure 3. Maps of a Classification Test Site

Based on classification accuracy, there is no single outstanding technique. This is because the sensor data levels are continuous from class to class and not separated into distributions which would match the assumptions of, for example, Gaussian or linear separability. The pixel by pixel accuracies are much lower than the inventory accuracies because classification is done on an individual pixel basis resulting in misclassification of isolated pixels and boundary pixels. This effect can be reduced by classifying whole objects on the basis of the majority class of the pixels in the object.

OBJECT DETECTION AND CLASSIFICATION

A great amount of information is also carried in the spatial characteristics of sensor imagery. This information is extracted by techniques such as edge

detection and template matching, which may allow recognition of shapes. Template matching using five sizes of circular templates applied to aerial photography of a peach orchard is shown in Figure 4. However, many groups of objects are not distinguishable by shape or outline alone. The spectral information which is available should also be used. If multispectral classification is applied to the pixels comprising the objects, they may be identified as different objects while possessing identical shapes. Classification as healthy or declining trees is shown in Figure 5. (Ref. 14)

Figure 4. Template Matching Example

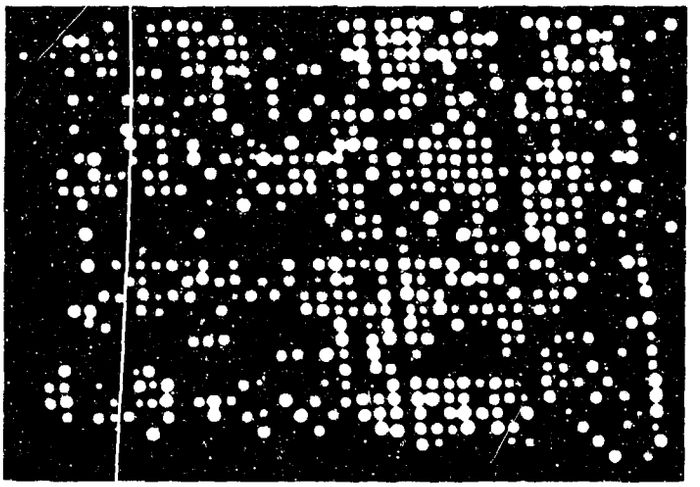


Figure 5. Detected Object Classification

GEOMETRIC MANIPULATION AND CLASSIFICATION

It may be necessary to change the geometry of the imagery for purposes such as merging data from different sensors, removing sensor distortions or overlaying maps to select ground truth areas. The data values in the manipulated image must be interpolated from those in the original data, and this is usually accomplished by one of three methods: nearest neighbor, bilinear, or bicubic. For a 1.44 million pixel data set, the results are given in the following table.

It may be seen that the effects of the geometric transformation on the classification performance are minimal, changing the accuracies by amounts on the order of one percent. Bilinear interpolation results in the highest map accuracies, apparently due to the slight smoothing of the data by this method.

CLASSIFICATION METHOD	GEOMETRIC MANIPULATION METHOD	INVENTORY ACCURACY PERCENT	CLASSIFICATION MAP ACCURACY PERCENT
SEQUENTIAL LINEAR	NONE	98.23	72.93
	NEAREST NEIGHBOR	98.87	72.57
	BILINEAR	99.14	74.16
	BICUBIC	98.58	72.43
MAXIMUM LIKELIHOOD	NONE	97.79	73.83
	NEAREST NEIGHBOR	97.97	73.52
	BILINEAR	97.76	74.87
	BICUBIC	98.60	73.24

CONCLUSIONS

The accuracies obtained by various types of classification techniques do not vary greatly. The processing speeds do, but this can be overcome by the use of table lookup. The fastest method is by classification of only the set of unique measurement vectors in the data set. Classification errors occur at isolated points and boundaries, which can be overcome in some cases by object detection. Some data modification, such as interpolation for geometric manipulation, is not highly detrimental to classification, due to the inherent data overlap among classes.

REFERENCES

1. Robert R. Jayroe, Robert Atkinson, B.V. Dasarathy, Matthew Lybanon and H.K. Ramapriyan, "Classification Software Technique Assessment," NASA TD D-8240, May 1976.

2. G.F. Chafaris, "Image Processing Investigations," General Electric Technical Information Series No. 77SDB002, December 1977.

3. R.J. Atkinson, B.V. Dasarathy, M. Lybanon, and H.K. Ramapriyan, "A Study and Evaluation Of Image Analysis Techniques Applied to Remotely Sensed Data," Final Report, Contract NAS8-32107, October 1976.

4. Y.C. Ho and R.L. Kashyap, "A Class of Interactive Procedures for Linear Inequalities," J. Siam on Control, Vol. 4, 1966.

5. A.D. Bond and R.J. Atkinson, "An Integrated Feature Selection and Supervised Learning Scheme for Fast Computer Classification of Multi-Spectral Data," Remote Sensing of Earth Resources, Vol. I, F Shahrokhi, Ed., U. of TN Space Institute, March 1972.

6. R.R. Jayroe, "Unsupervised Spatial Clustering with Spectral Discrimination," NASA TN D-7312, May 1973.

7. R.R. Jayroe, P.A. Larsen, and C.W. Campbell, "Computer and Photogrammetric General Land Use Study of Central North Alabama," NASA TR R-431, October 1974.

8. B.V. Dasarathy, "Discriminant Hyperplane Abstracting Residuals Minimization Algorithm for Separating Clusters with Fuzzy Boundaries," Proc. IEEE, Vol. 64, April 1976.

9. W.G. Eppler et al, "Table Look-Up Approach to Pattern Recognition," Proc. 7th International Symposium on Remote Sensing of Environment, U. of Michigan, Ann Arbor, May 1971.

10. W.G. Eppler, "An Improved Version of the Table Look-Up Algorithm for Pattern Recognition," Proc. 9th International Symposium on Remote Sensing of Environment, U. of Michigan, Ann Arbor, April 1974.

11. Robert R. Jayroe, "A Fast Routine for Computing Multidimensional Histograms," NASA TM 78133, October 1977.

12. Robert R. Jayroe and Debrah Underwood, "Vector Statistics of Landsat Imagery," NASA TM 78149, December 1977.

13. R. Jayroe, R. Atkinson, L. Callas, J. Hodges, B. Gaggini, and J. Peterson, "Evaluation of Registration, Compression, and Classification Algorithms," NASA TM 78227, February 1979.

14. Robert J. Atkinson, "Digital Computer Processing of Peach Orchard Multispectral Aerial Photography," NASA CR 149998, October 1976.

Paper No. IA-5, Presented at the Workshop on Imaging Trackers and Autonomous Acquisition Applications for Missile Guidance, 19-20 November 1979, Redstone Arsenal, Alabama.

MULTIPLE-CLASS PIECEWISE LINEAR TRAINABLE CLASSIFIERS

Jack Sklansky
School of Engineering
University of California
Irvine, California 92717

ABSTRACT

When applying computers to the analysis of signals or images, one often must classify parts of the signals or images into several classes. Examples of such classes are tumor, calcification and blood vessel in chest radiographs; and tank, jeep, and building in scenes analyzed by guided missiles. The previous theory of automatic classifiers was mostly devoted to two-class classifiers. We describe a new technique for the design of multiple-class classifiers.

Out technique combines our earlier theory of trainable linear classifiers with the available methods for the design of multiple-output logic networks.

Our technique is based on the assumption that the optimal decision surfaces can be approximated by piecewise linear surfaces $\{L_i\}$ with little effect on the classification errors; and that the optimal decision surfaces depend mostly on subsets of feature vectors from distinct classes that are close to one another in feature space.

Visualizations of the relationships of the linear segments of $\{L_i\}$ to one another in multidimensional feature space are provided by adjacency graphs and incidence graphs relating various polyhedral regions in feature space. These graphs facilitate interactive design of the classifier.

Each linear segment of the piecewise linear decision surfaces is designed by a training procedure that yields near-minimal classification errors for that segment. Thus the effectiveness of each segment of the decision surfaces reflects the design data in the part of feature space associated with that segment.

The use of switching theory and mathematical methods for the design of logic networks leads to efficient sequential decisions for the multiple-class classifier. These sequential designs tend to minimize the number of computations required for the assignment of a previously unclassified feature vector to a class.

Piecewise linear surfaces offer the further advantage of relatively simple implementation by special-purpose digital electronic hardware.

INTRODUCTION

In the application of computers to the analysis of signals or images, one often must partition these signals or images into several categories or classes. For example, the analysis of a medical radiograph often requires outlining and labeling regions such as heart, calcification, tumor, and blood vessel. Effective missile guidance often requires the segmentation of a scene into classes such as tank, jeep, and building.

Earlier classification techniques are mainly suited to just two or three classes, and to cases where the optimum decision surfaces are either approximately linear or approximately quadratic. In practice there are many forms of distributions of labeled data which cannot be adequately separated by linear or quadratic decision surfaces. In these cases the Bayes-optimum surfaces [7] are highly nonlinear.

Our technique is based on the following property of Bayes-optimum decision surfaces: the Bayes surface often passes through regions of feature space where the hulls of subsets of feature vectors from different classes overlap or where the data from these classes are very close to one another. We refer to such regions as encounter zones [3]. Figure 1 illustrates three of these encounter zones. In most practical situations the decision boundary depends principally on the data within these zones.

In our technique each linear segment of the decision surface is positioned by training a hyperplane only on a subset of data lying within an encounter zone. We use two forms of decision graphs -- adjacency graphs and incidence graphs -- to visualize the relationships among the linear segments and the polyhedral decision regions. These graphs help us to reduce the number of hyperplanes. After choosing the piecewise linear decision surface, we exploit switching theory to minimize the decision logic and the average computation time.

The localized training and the decision graphs give our method great versatility, and yield both a near-minimum number of hyperplanes and error rates near the Bayes optimum.

Our technique consists of three major parts:

1. Find "closed opposed" pairs of data prototypes. (These pairs represent the encounter zones.)

2. Find a set of hyperplanes separating clusters of data represented by close opposed pairs of prototypes. For this purpose use our theory of trainable linear classifiers [1,2,3]. These hyperplanes produce piecewise linear decision surfaces separating polyhedral decision regions in feature space. These decision regions represent classes to which unknown feature vectors are assigned by the classifier.

Display the relations among the segments and the polyhedral decision regions by decision graphs. These graphs facilitate interactive simplification of the decision surface, often leading to a reduction in the number of hyperplanes and the number of linear segments. This approach to interactive design obviates the need for mapping of the data into two-dimensional space for human visualization.

3. Use multiple-output switching theory to minimize the number of hyperplanes and the computation time in the multiple-class decision logic [4,5,6].

CLOSE OPPOSED PAIRS OF PROTOTYPES

We assume that the design data consists of a finite set of d-dimensional feature vectors $\chi = \{\underline{x}_i\}$ in R^d. Each $\underline{x}_i$ is labeled by one of c classes $\{\omega_1,\ldots,\omega_c\}$. If $\underline{x}_i$ is labeled by ω_j, we say that $\underline{x}_i \in \omega_j$. We refer to χ as a training set or design set. We assume that the dissimilarity between any pair of feature vectors $(\underline{x}_i, \underline{x}_j)$ is measured by the Euclidean distance between them:

$$D(\underline{x}_i,\underline{x}_j) \equiv ||\underline{x}_i - \underline{x}_j|| \equiv \left[\frac{1}{d}\sum_{k=1}^{d}(x_{ik} - x_{jk})^2\right]^{\frac{1}{2}}$$

In order to give approximately equal significance to all of the coordinates of feature space, we assume that each feature has been normalized. The choice of the form of normalization depends on the shapes of the distributions of these data in feature space [1]. Often one may effectively normalize the data by subtracting the sample mean within each class and dividing by the sample standard deviation -- yielding, for each class, a training set whose projection on each feature axis has a mean of zero and a variance of unity.

Using an iterative clustering procedure [7,8] (the choice of the procedure does not seem to be critical), we segment the data into clusters. Each cluster is represented by its centroid or prototype. Then we find close pairs of prototypes in opposite classes by a procedure to be described below. We refer to such a pair as a closed opposed pair or link. We refer to the cluster of data represented by a prototype as a protocluster. Each close opposed pair represents a subset of an encounter zone.

The number of prototypes is specified by the designer of the classifier. The designer usually will have to experiment with this number until he finds the configuration that gives best results. This number should be as small as possible while large enough so that the set of prototypes serves adequately as a "skeleton" for the data.

Encounter zones are identified with close opposed pairs of protoclusters, defined in the following way. Let M_k denote the set of prototypes formed from data in class ω_k. We say that a pair of prototypes $(\underline{\mu}_i, \underline{\nu}_j)$ is opposed

iff $\underline{\mu}_i \in M_r$ and $\underline{\nu}_j \in M_s$, $r \neq s$. An opposed pair of prototypes $(\underline{\mu}_i, \underline{\nu}_j)$ is now defined to be close opposed if and only if

$$D(\underline{\mu}_i, \underline{\nu}_j) = \min_{\underline{\nu}_k \in M_s} D(\underline{\mu}_i, \underline{\nu}_k) = \min_{\underline{\mu}_k \in M_r} D(\underline{\mu}_k, \underline{\nu}_j),$$

where $D(\underline{a}, \underline{b})$ denotes the Euclidean distance between $\underline{a}$ and $\underline{b}$. That is, $\underline{\mu}_i \in M_r$ and $\underline{\nu}_j \in M_s$ are close opposed iff $\underline{\mu}_i$ is closer to $\underline{\nu}_j$ than to any other prototype in M_s, and vice versa.

Let Π_{rs} denote the set of close opposed pairs for classes ω_r and ω_s. It must contain at least one member, namely the opposed pair for which $D(\underline{\mu}_i, \underline{\nu}_j)$ is minimum. A procedure for constructing Π_{rs} is obtained from its definition:

Step 1: For each $\underline{\mu}_i \in M_r$, find the closest $\underline{\nu}_j(\underline{\mu}_i) \in M_s$, $r \neq s$.
The link set

$$L_{rs} \equiv \{(\underline{\mu}_i, \underline{\nu}_j(\underline{\mu}_i)) \mid \underline{\mu}_i \in M_r\}$$

is saved.

Step 2: For each $\underline{\nu}_j \in M_s$, find the closest $\underline{\mu}_i(\underline{\nu}_j) \in M_r$, $r \neq s$.
The link set

$$L_{sr} \equiv \{(\underline{\mu}_i(\underline{\nu}_j), \underline{\nu}_j) \mid \underline{\nu}_j \in M_s\}$$

is saved.

Step 3: The set of close opposed pairs for the pair of classes (ω_r, ω_s) is:

$$\Pi_{rs} = \{L_{rs} \cap L_{sr}\}.$$

The sets $\{\Pi_{rs}\}$ are our realizations of encounter zones.

The set Π_{rs} is an approximate representation of the gap between the classes ω_r and ω_s and thus leads to an initial decision boundary for this gap. At times it is useful to enlarge this set by extending the concept of close opposed pairs to that of k-close-opposed pairs. The algorithm for finding $\Pi_{rs}^{(k)}$, a set of k-close-opposed pairs, is like that for finding Π_{rs}, except that the instruction "find the closest prototype" in Steps 1 and 2 is changed to "find the k closest prototypes." This means, for example, that $(\underline{\mu}_i, \underline{\nu}_j) \in L_{rs}$ if and only if no more than k-1 prototypes in M_s are closer to $\underline{\mu}_i$ than is $\underline{\nu}_j$. L_{sr} is similarly redefined. Clearly,

$$\Pi_{rs}^{(k)} \subseteq \Pi_{rs}^{(k+1)}, \text{ and } \Pi_{rs}^{(1)} = \bigcap_k \Pi_{rs}^{(k)}.$$

A set of 3-close-opposed pairs for three classes is illustrated in Figure 2. In this figure the three classes are labeled A,B, and C; the protoclusters are represented by circles; the prototypes are represented by dots at the centers of the circles; the links $\{L_{rs}\}$ are represented by straight line segments joining the prototypes.

THE DECISION SURFACE

In the next stage of the design process, a near-minimal set of decision hyperplanes that separates subsets of the k-close-opposed pairs is found. Training procedures are used to find near-Bayes-optimum positions of these hyperplanes in feature space.

We find these hyperplanes sequentially. First we find a hyperplane that separates the closest among the k-close-opposed pairs, because placing a hyperplane in a constricting or neck-shaped part of the interclass gap seems likely to separate more close opposed pairs than a hyperplane in other parts of the gap. Let $(\underline{\mu}_I, \underline{\nu}_J)$ denote this pair. For simplicity, we choose the hyperplane that is the perpendicular bisector of $(\underline{\mu}_I, \underline{\nu}_J)$. The equation of this hyperplane is

$$[\underline{x} - \frac{1}{2}(\underline{\mu}_I + \underline{\nu}_J)]^T (\underline{\mu}_I - \underline{\nu}_J) = 0.$$

Call this hyperplane $\hat{H}_I$.

Next we find those pairs of prototypes that are correctly classified by $\hat{H}_1$. Denote these pairs by $\{\underline{\mu}_i(\hat{H}_1), \underline{\nu}_j(\hat{H}_1))\}$. Let $P_i(\hat{H}_1)$ denote the region of feature space associated with $\underline{\mu}_i(\hat{H}_1)$. We refer to $P_i(\hat{H}_1)$ as a prototype region.

Next we use $\hat{H}_1$ as the initial hyperplane, and the data in the prototype regions $\{P_i(\hat{H}_1)\}$, $\{P_j(\hat{H}_1)\}$ as the training set in a training procedure that finds a near-Bayes optimum separation of the training set in these regions. The training procedure should be nonparametric, because the data in the prototype regions are likely to be nongaussian. For this purpose we recommend the window training procedure [1]. One form of this training procedure is given by the following recursive equation:

$$\underline{v}(n+1) = \begin{cases} \underline{v}(n) + \dfrac{(-1)^k \|\underline{w}\|}{(1+n)^{1/4}} \left[y - \dfrac{\underline{v}^T \underline{y}}{\|\underline{w}\|^2} \begin{pmatrix} 0 \\ \underline{w} \end{pmatrix} \right] & \text{for } |\underline{v}^T \underline{y}| \, (1+n)^{1/2} \leq \|\underline{w}\| \\ \underline{v}(n) \text{ otherwise} \end{cases}$$

where

$$\underline{v}(n) = [v_o(n), w_1(n),\ldots,w_d(n)] \equiv [v_o(n), \underline{w}(n)]$$

= augmented weight vector at iteration n,

$$\underline{y}(n) = [1, \underline{x}(n)]$$

= augmented feature vector at iternation n.

Let H_1^* denote the hyperplane obtained by the training process using $\hat{H}_1$ as the initial hyperplane. We say that H_1^* is the hyperplane obtained by "training on $\hat{H}_1$." H_1^* may or may not separate the same set of pairs of k-close opposed prototypes as $\hat{H}_1$. If H_1^* does not separate the same set as that of $\hat{H}_1$, the training process may be repeated, treating H_1^* as the initial hyperplane of the repeated training process. We suggest that the training process be repeated until two successive repetitions separates the same pairs of prototypes. Call the final hyperplane H_1.

Next the prototypes separated by H_1 are removed from the set of k-close opposed hyperplanes, the closest among the remaining k-close opposed pairs computed, and another near-Bayes-optimum hyperplane H_2 computed in a manner similar to that for H_1.

In this way, a set of near-Bayes-optimum decision hyperplanes is computed, each hyperplane separating a subset of the data that forms the set of k-close opposed pairs of prototypes.

DECISION GRAPHS

Two types of graphs facilitate interactive simplification of the decision surface: adjacency graphs and incidence graphs. Both graphs are derived from a set of minterms representing the polyhedral volumes enclosed by the set of decision hyperplanes.

We explain the decision graphs by the piecewise linear decision curves shown in Figure 3. In this figure the feature space is two-dimensional. Thus the decision segments here are straight line segments, and the decision surfaces are polygonal curves. These decision surfaces partition the 2-space into four decision regions: R_1, R_2, R_3, R_4.

Let $z_i(x)$ denote a binary variable associated with linear segment S_i and feature vector $\underline{x}$. The value of $z_i(\underline{x})$ is 0 or 1 depending on whether $\underline{w}_i^T\underline{x} - |\underline{w}_i|p_i$ is negative or positive, where $\underline{w}_i$ is a weight vector from the origin, normal to segment S_i, and p_i is the distance of the origin from the hyperplane containing S_i. Let H_k denote that hyperplane. Let $\underline{z}(\underline{x})$ denote the vector formed by all the $z_i(\underline{x})$'s. The H_k's partition the feature space into a nonoverlapping set of convex polyhedra $\{r_j(\underline{x})\}$. For all $\underline{x} \in r_j(\underline{x})$, $\underline{z}(\underline{x})$ has a fixed value, which we denote by $\underline{z}_j$. We refer to $\underline{z}_j$ as a polyhedral minterm. If we choose $\underline{z}$ arbitrarily, it may or may not

be a polyhedral minterm. We refer to such a $\underline{z}$ as a minterm. Let Z denote the set of polyhedral minterms $\{\underline{z}_j\}$.

In Figure 3, each linear decision segment is identified by an encircled number. Thus ③ denotes segment S_3. A number enclosed by a square is a label for a hyperplane. Thus [4] denotes hyperplane H_4. The weight vector $\underline{w}_i$ normal to S_i has the direction shown by the arrow emanating from S_i. The polyhedral minterms are denoted by $\{\underline{z}_r^{(i)}\}$, where r denotes the r^{th} polyhedron and i denotes decision region R_i. Each $\underline{z}_r^{(i)}$ is shown inside its corresponding polyhedron. Note that segment S_5 has no arrow, because its hyperplane coincides with that of segment S_2.

The first step toward constructing the adjacency graph is to find Z. Each polyhedral minterm--i.e., each member of Z--must yield a consistent set of inequalities of the form

$$\underline{w}_j^T \underline{x} - |\underline{w}_j| p_j \begin{cases} > 0 \text{ for } z_{ji} = 1 \\ < 0 \text{ for } z_{ji} = 0 \end{cases}$$

$$\text{for } i = 1, \ldots, m,$$

where z_{ji} is the i^{th} component of $\underline{z}_j$. I.e., there must be at least one real vector $\underline{x}$ that satisfies the above set of inequalities. To find Z, we check the consistency of the above set of inequalities for each of the 2^m possible m-vectors $\underline{z}_j = (z_{j1}, \ldots, z_{jm})$. If the inequalities are consistent, $\underline{z}_j \in Z$; otherwise $\underline{z}_j \notin Z$. The consistency of these inequalities may be checked by the method of finding "feasible solutions" in linear programming [9].

In Figure 3, the consistency check yields the following members of Z. (Here the components of each minterm represent hyperplanes H_4, H_6, H_3, H_2, H_5, H_1, in that order.)

Region R_1:	$\underline{z}_1^{(1)}$ =	100001	$\underline{z}_5^{(1)}$ =	111001
	$\underline{z}_2^{(1)}$ =	100011	$\underline{z}_6^{(1)}$ =	111101
	$\underline{z}_3^{(1)}$ =	100101	$\underline{z}_7^{(1)}$ =	111100
	$\underline{z}_4^{(1)}$ =	101101	$\underline{z}_8^{(1)}$ =	101100
Region R_2:	$\underline{z}_1^{(2)}$ =	000001	$\underline{z}_3^{(2)}$ =	010011
	$\underline{z}_2^{(2)}$ =	000011	$\underline{z}_4^{(2)}$ =	110011
Region R_3:	$\underline{z}_1^{(3)}$ =	010010	$\underline{z}_4^{(3)}$ =	111011
	$\underline{z}_2^{(3)}$ =	011011	$\underline{z}_5^{(3)}$ =	101011
	$\underline{z}_3^{(3)}$ =	011010	$\underline{z}_6^{(3)}$ =	101001

Region R_4: $\underline{z}_1^{(4)} = 111010$ $\underline{z}_3^{(4)} = 011110$

$\underline{z}_2^{(4)} = 111111$ $\underline{z}_4^{(4)} = 111110$

For every pair of decision regions (R_i, R_j), $i \neq j$, find all pairs $\{(\underline{z}_r^{(i)}, \underline{z}_s^{(j)})\}$ such that $\underline{z}_r^{(i)}$ and $\underline{z}_s^{(j)}$ are adjacent--i.e., such that $|\underline{z}_r^{(i)} - \underline{z}_s^{(j)}| = 1$, where $|\underline{x}|$ = "city-block" or "Hamming" distance. Note that if $\{\underline{z}_r^{(i)}, \underline{z}_s^{(j)}\}$ are adjacent, then

$$|\underline{z}_{rk}^{(i)} - \underline{z}_{sk}^{(j)}| = \delta_{km} \equiv \begin{cases} 1 \text{ for } k = m \\ 0 \text{ for } k \neq m \end{cases}$$

for some positive integer m. We say that an adjacent pair $(\underline{z}_r^{(i)}, \underline{z}_s^{(j)})$ is a segment element of hyperplane m, and denote it by e_{ij}^m.

Next find the set of segment elements $\{e_{ij}^m\}$ for a given linear segment S_k of hyperplane m. We call this the segment set for segment S_k. For convenience we let S_k denote this segment set as well as the segment. For every pair (i,j) we can find a set of segments (or segment sets) that separate R_i from R_j.

Next find every polyhedral minterm region $\underline{z}_k$ that shares a decision hyperplane with one of the segment sets, and which is a unit Hamming distance from one of the polyhedral minterm regions in the segment set. Then find all pairs of segment sets (or segments) that share one of the $\underline{z}_k$'s. Call such a pair (S_i, S_j).

Define

$$d(S_i, S_k) = \text{distance between } S_i \text{ and } S_k$$

$$= \begin{cases} 0 \text{ if } i = k \\ \text{twice the sine of half of the external angle between } S_i \text{ and } S_k \text{ if } S_i \text{ and } S_k \text{ are neighbors} \\ \infty \text{ otherwise} \end{cases}$$

In the adjacency graph each node denotes a segment S_i, and every arc joins a pair (S_i, S_k) for which $d(S_i, S_k) < \infty$. The arc for (S_i, S_k) is labeled by $d(S_i, S_k)$.

In the example illustrated in Figure 3, the segment sets of the decision surface separating R_1 from R_5 are

$$S_3 = \{(\underline{z}_2^{(1)}, \underline{z}_5^{(3)}), (\underline{z}_1^{(1)}, \underline{z}_6^{(3)})\}$$

$$S_4 = \{(\underline{z}_4^{(1)}, \underline{z}_6^{(3)})\}$$

$$S_5 = \{(\underline{z}_5^{(1)}, \underline{z}_6^{(3)})\}$$

$$S_6 = \{(\underline{z}_5^{(1)}, \underline{z}_4^{(3)})\}$$

The adjacency graph for this decision surface is shown in Figure 4. The nodes in this graph are labeled by the indices of $\{S_i\}$; the arcs are labeled by $d(S_i,S_j)$.

The incidence graph for Figure 3 is obtained by representing every polyhedral region by a node, and joining by an arc every pair of nodes that represent adjacent polyhedral regions. Every arc is labeled by its associated hyperplane. Every node is labeled by its associated minterm $\underline{z}_r^{(i)}$ and decision region R_i. In addition, an arc may be labeled by a decision segment, if applicable.

Figure 5 shows the incidence graph for Figure 3. To simplify the drawing each node in this graph is labeled only by the index of its decision region, and each arc by the index of its hyperplane. To distinguish the four classes, we represent the nodes by squares, triangles, circles, and inverted domes for R_1, R_2, R_3, and R_4, respectively.

MINIMAL DECISION LOGIC

Our multiple-class classifier generates the hyperplanes $\{H_i\}$ in a prescribed sequence. For each H_i, the classifier determines whether $\underline{x}$ lies on the "positive" or "negative" side of H_i. In particular, if the equation of H_i is

$$\underline{v}_i^T \underline{y} = 0$$

where $\underline{y}$ is the augmented vector $\begin{pmatrix} 1 \\ \underline{x} \end{pmatrix}$, then the classifier determines whether $\underline{v}_i^T \underline{y}$ is positive or negative. If $\underline{v}_i^T \underline{y}$ is negative, a C is generated. If $\underline{v}_i^T \underline{y}$ is positive, a 1 is generated. The 0's and 1's of successive $\underline{v}_i$'s are combined in a logical network (or "switching function") to produce one of $c + 1$ assignments of $\underline{x}$: $R_1,\ldots,R_{c+1}$.

The first c of these assignments correspond to the c classes $\{\omega_i \mid i=1,\ldots,c\}$. R_{c+1} corresponds to "undecided." (In some applications, R_{c+1} is omitted.) As soon as one of the R_i's is produced by the logical network, the sequence of hyperplanes is terminated, and the current assignment of $\underline{x}$ is accepted.

The switching function may be designed so as to use a near-minimal number of hyperplanes in each decision, thereby yielding a near-minimal computation time for a given configuration of computer hardware. To see how this may be achieved, let $z_i(\underline{x})$ denote the binary-valued function of $\underline{x}$ such that

$$z_i(\underline{x}) = 0 \text{ if and only if } \underline{v}_i^T \underline{y} < 0,$$

$$z_i(\underline{x}) = 1 \text{ if and only if } \underline{v}_i^T \underline{y} \geqslant 0.$$

Let

$$\underline{z}(\underline{x}) = [z_1(\underline{x}), z_2(\underline{x}), \ldots, z_m(\underline{x})]^T$$

$$= \text{a column vector formed by the } z_i(\underline{x})\text{'s.}$$

Let $\{S_i\}$ denote the set of decision surfaces derived from our training processes and our analysis of the decision graphs. Let $\underline{\Omega}(\underline{z})$ denote a (c+1) - component binary-valued vector function of $\underline{z}$ such that the ith component corresponds to the decision region R_i. When $\underline{z}$ is feasible, only one component of $\underline{\Omega}(\underline{z})$ is 1. When $\underline{z}$ is not feasible, all components of $\underline{\Omega}(\underline{z})$ are "don't care's", denoted by δ. (If we wish we may reduce the number of components of $\underline{\Omega}(\underline{z})$ to the smallest integer greater than $\ell n(c+1)$, and apply a decoder to $\underline{\Omega}(\underline{z})$ to obtain the desired output. But it is not clear whether the cost of the decoder is less than the savings in implementing $\underline{\Omega}(\underline{z})$.)

To find $\underline{\Omega}(\underline{z})$, construct a "population table" for the $\underline{x}$'s in the training set: for each $\underline{x}$ in the training set, find $\underline{z}(\underline{x})$. For each possible $\underline{z}$, count the number of $\underline{x}$'s in ω_i for which $\underline{z}(\underline{x}) = \underline{z}$. Call this the "population" $N_i(\underline{z})$. Do this for $i = 1, \ldots, c+1$. Let

$$Q = \sum_{i=1}^{c+1} N_i(\underline{x}).$$

If Q is small, let $\underline{\Omega}(\underline{z}) = \delta$. (The threshold separating "small" from "not small" must be determined empirically.) If Q is not small, let

$$\Omega_i(\underline{z}) = 0 \text{ or } 1$$

depending respectively on whether or not

$$N_i(z) \geqslant N_j(\underline{x})$$

for all j.

We explain our procedure for minimizing the decision logic by the following example. Suppose the population table for the $\underline{z}$'s yields the function $\underline{\Omega}(\underline{z})$ specified by Table 1.

$\underline{z}$			$\underline{\Omega}(\underline{z})$		
z_1	z_2	z_3	Ω_1	Ω_2	Ω_3
0	0	0	1	0	0
0	0	1	0	1	0
0	1	0	1	0	0
0	1	1	1	0	0
1	0	0	0	0	1
1	0	1	0	0	1
1	1	0	0	1	0
1	1	1	δ	δ	δ

TABLE 1

Using Karnaugh maps or other 2-level logic minimization techniques to minimize the logic relating each Ω_i to $\underline{z}$, we obtain

$$\Omega_1 = \bar{z}_1(z_2 \vee \bar{z}_3)$$

$$\Omega_2 = z_1 z_2 \vee \bar{z}_1 \bar{z}_2 z_3$$

$$\Omega_3 = z_1 z_2$$

From an examination of these equations we obtain the following decision sequence. Compute the unknown $\underline{x}$ on hyperplane H_1. If $z_1 = 1$, then we need only examine H_2 in order to arrive at a decision. If $z_1 = 0$, compute the unknown $\underline{x}$ on hyperplane H_3. If $z_3 = 0$, then $\underline{x}$ is assigned to R_1. If $z_3 = 1$, compute the unknown $\underline{x}$ on hyperplane H_2. This tends to minimize the average number of computations for each assignment of $\underline{x}$ to a decision region.

When the number of minterms and the number of decision regions are large, then one may use computer programs for the minimization of multiple-output logic networks. An excellent example of such a program is described by Svoboda and White [10]. When the number of minterms and the number of decision regions are intermediate in size, one may use manual procedures for minimizing the covering set of multiple-output prime implicants [5].

CONCLUDING REMARKS

Supporting experience for the techniques described here has been obtained for two-class classifiers for relatively complex distributions of data in two-dimensional and three-dimensional feature space [3]. This experience encourages us to believe that our technique has great versatility, and yields both minimum decision logic as well as near-Bayes optimality.

Another advantage of our technique is provided by the linear algebraic equations representing the decision hyperplanes. We believe these linear equations can be economically and compactly implemented in special purpose digital electronic hardware.

These techniques have not yet been tested on multiple-class data. We hope to carry out such tests and report on their outcomes at a later date.

ACKNOWLEDGMENT

This paper is an outgrowth of earlier work on two-class piecewise linear classifiers carried out at Spectra Research Systems, Irvine, California, in collaboration principally with Dr. Leo Michelotti. That work was reported in [3]. The work on multiple-class classifiers was subsequently carried out at the Pattern Recognition Project of the School of Engineering, University of California, Irvine, California, and was supported by the National Institute of General Medical Sciences under U.S. Public Health Service Grant No. GM-17632.

REFERENCES

1. J. Sklansky, G. Wassel, Pattern Classifiers and Trainable Machines, Springer Verlag, New York 1980.

2. G. Wassel, J. Sklansky, "An Adaptive Nonparametric Linear Classifier," Proceedings of the IEEE, Vol. 64, No. 8, pp. 1162-1171, August 1976.

3. J. Sklansky, L. Michelotti, "Locally Trained Piecewise Linear Classifiers," IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 1, No. 2, pp. 101-111, March 1980.

4. R. E. Prather, Introduction to Switching Theory, Allyn and Bacon, Boston, Mass., 1967.

5. A. D. Friedman, P. R. Menan, Theory and Design of Switching Circuits, Computer Science Press, Woodland Hills, California 1975.

6. T. C. Bartee, I. L. Lebow, I. S. Reed, Theory and Design of Digital Machines, McGraw-Hill Book Co., 1962.

7. R. O. Duda, P. E. Hart, Pattern Classification and Scene Analysis, John Wiley and Sons, New York, 1973.

8. M. R. Anderberg, Cluster Analysis for Applications, Academic Press, New York, 1973.

9. R. D. Luce, H. Raiffa, Games and Decisions, J. Wiley and Sons, New York, 1967.

10. A. Svoboda, D. E. White, Advanced Logical Circuit Design Techniques, Garland Publishing Co., New York, 1978.

LIST OF FIGURES

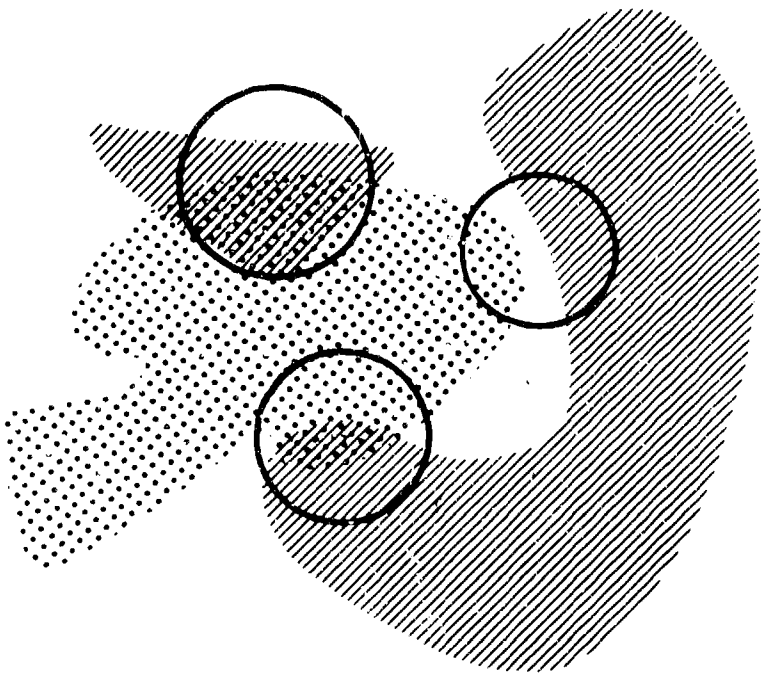


Figure 1. Encounter zones between two classes of data in feature space.

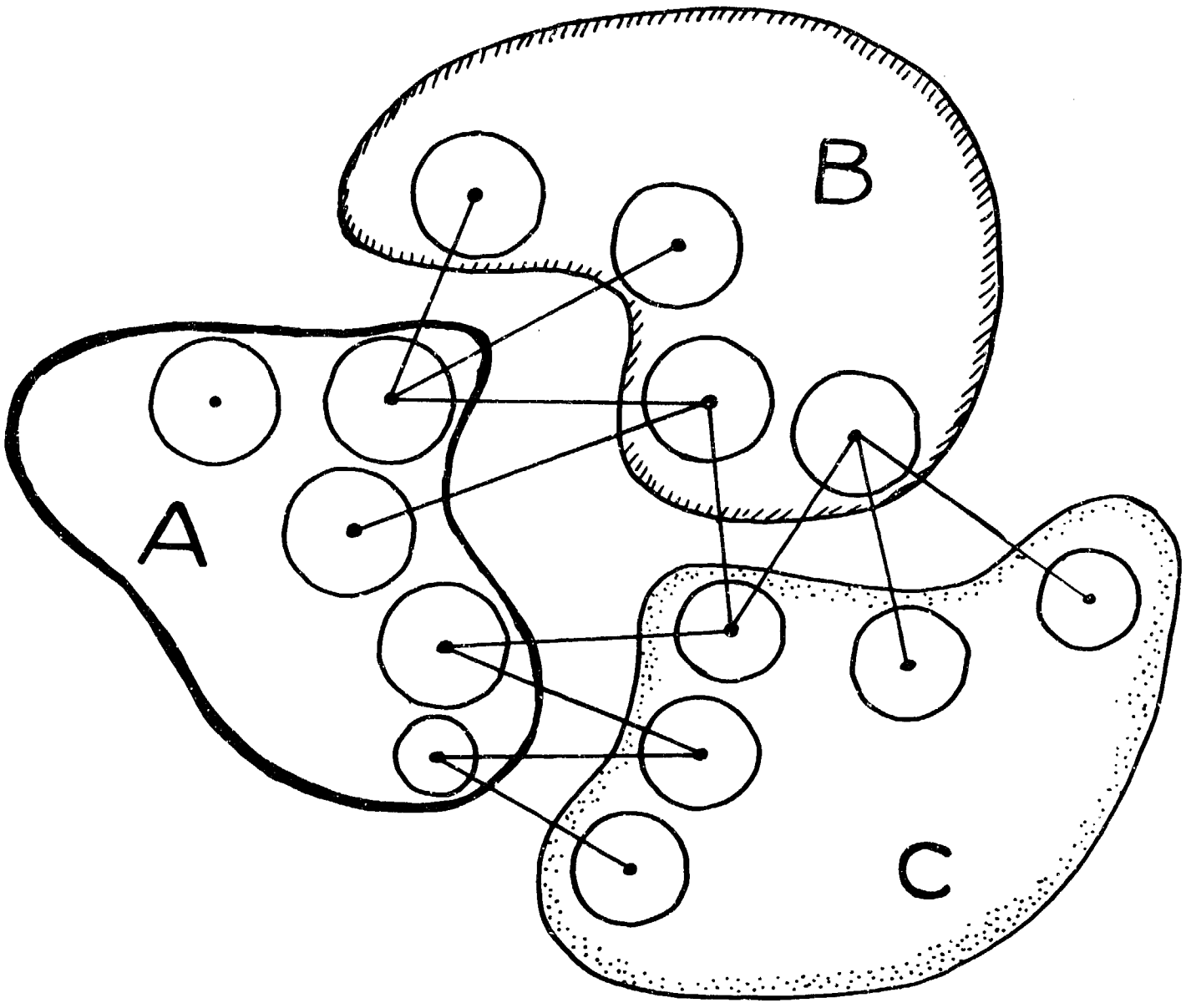


Figure 2. A set of 3-close-opposed pairs for three class regions.

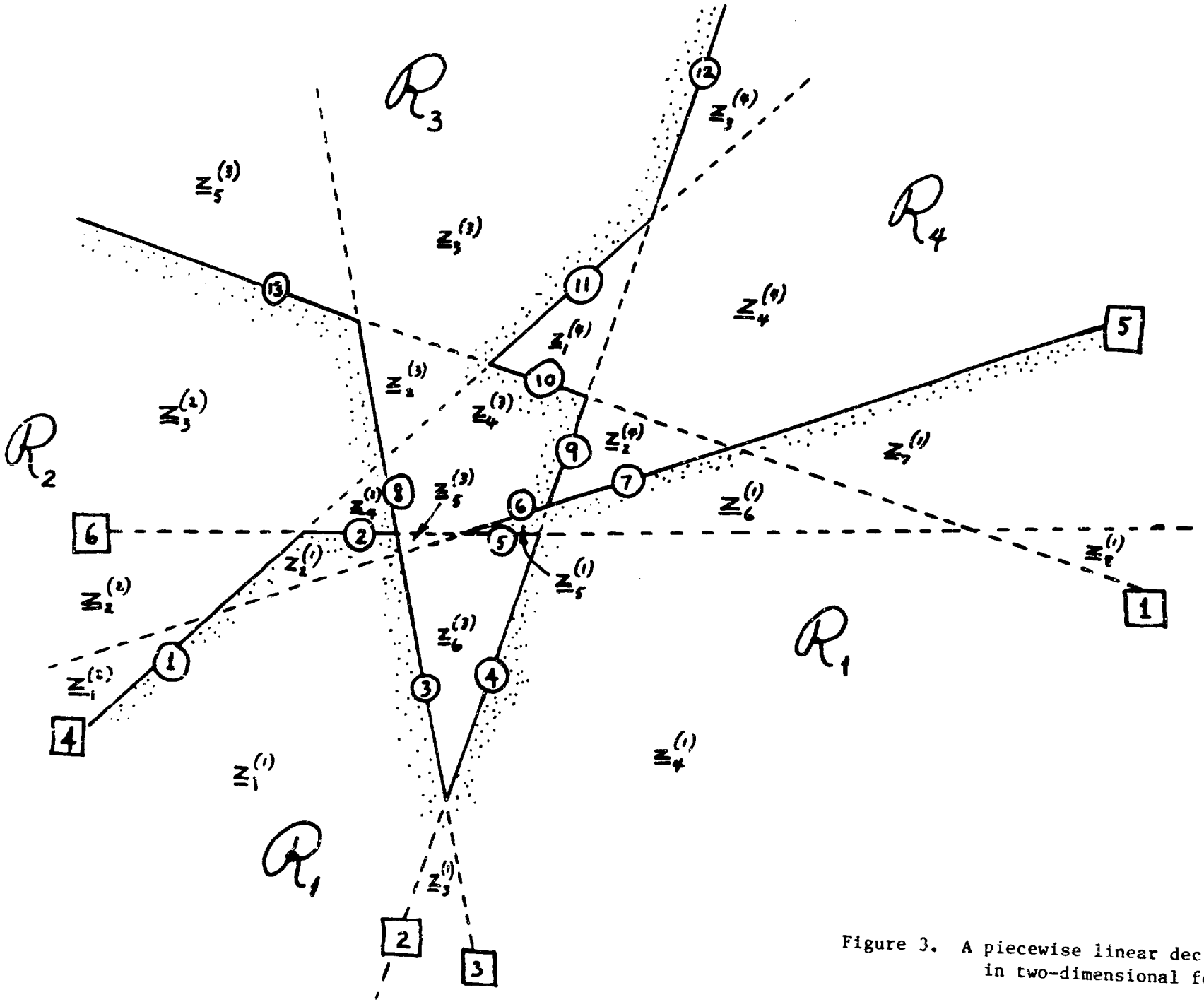


Figure 3. A piecewise linear decision curve in two-dimensional feature space

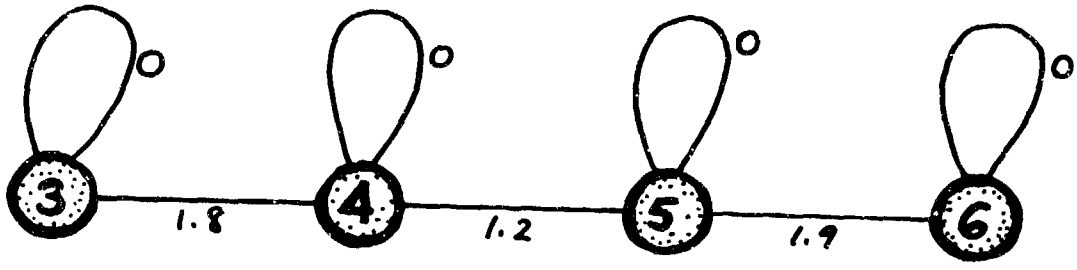


Figure 4. Adjacency graph for decision surface separating R_1 from R_3 in Figure 3.

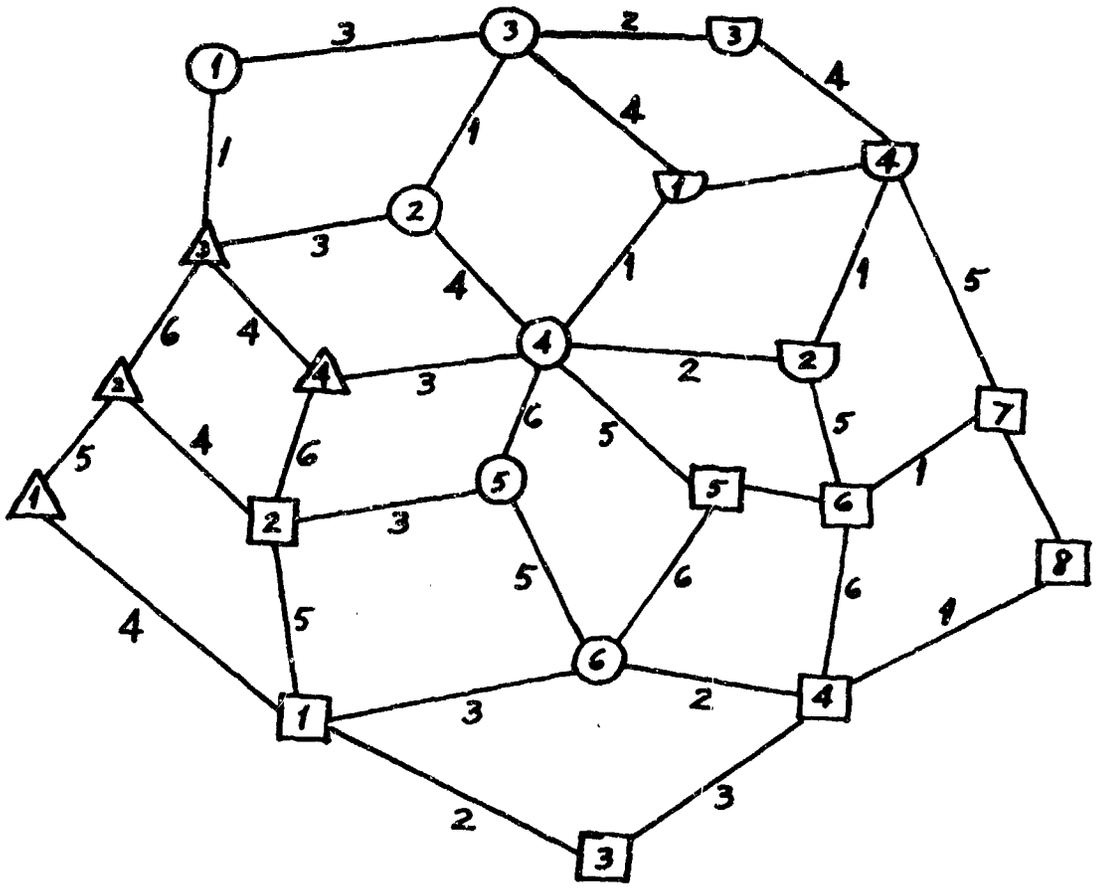


Figure 5. The incidence graph for the decision regions in Figure 3.

Paper No. IA-6, Presented at the Workshop on Imaging Trackers and Autonomous Acquisition Applications for Missile Guidance, 19-20 November 1979, Redstone Arsenal, Alabama.

AN UNSUPERVISED LEARNING APPROACH TO IDENTIFICATION AND CATEGORIZATION OF EDGES IN IMAGES

Belur V. Dasarathy
M&S Computing, Inc.
P. O. Box 5183
Huntsville, AL 35805

ABSTRACT

The objective of this paper is to explore the scope for deploying the powerful tools in the domain of pattern recognition in the nontraditional role of a preprocessor for image segmentation through edge detection/categorization. The edge detection problem is viewed as a problem of learning in unsupervised environments, and the available information in the form of the input image is restructured into a multidimensional data base for such learning. Details of this approach and preliminary experience of its implementation are discussed.

1. INTRODUCTION

Autonomous acquisition of targets by imaging trackers requires a capability to locate targets within the field of view by a process of segmentation of the image into regions of interest. This image segmentation task is accomplished by using a variety of tools generally clubbed together under the common term: Image Processing [1]. Some typical tools for image segmentation include thresholding [2], edge detection [3], region growing [4], etc.. However, most of the approaches generally restrict themselves to the physical two-dimensional image plane and each pixel therein is described by a single scalar descriptor at any given instant in the processing stream. Once the target areas are isolated, features based on its shape, size, texture, etc., are extracted, and this information is processed through classical pattern recognition tools to derive the identification labels of these targets: i.e., to perform target recognition/classification.

The objective of this study is to explore the feasibility for expanding the role of these pattern recognition oriented tools to cover the domain of target acquisition in addition to its traditional role of target classification. This problem of extraction of features which are descriptive of targets can itself be looked upon as a problem in unsupervised learning, and the relevant experience in such learning can be brought to bear on this task. Viewing the image segmentation task as one of edge detection, one could visualize this as the problem of unsupervised learning (and categorization o.) edges in images. Under the classical edge detection approach, this would be limited to determining the gradients(s) at each pixel point in the image, and viewing the resultant gradient image by appropriate thresholding to locate the significant edges.

However, this procedure has its limitations in that it is incapable of distinguishing among equal gradient valued edges. The drawback arising out of this limitation becomes apparant when one considers the fact that it is entirely possible that the gradient at the target-background boundary can be numerically equal to gradients elsewhere in the image at some natural boundaries inherent in the scene. Instead, if one could simultaneously examine the pixel intensity value and the gradient value at each of these pixels, it is easy to see that discrimination among the different but equal gradient valued edges can be attempted. This can, in principle, be further extended to include a variety of other possible derived information at each pixel position. In effect, one could conceive of a multidimensional attribute space in which such unsupervised learning of edges is to be attempted.

The succeeding sections explore the scope of this concept, present a feasible methodology, and report on some preliminary results which confirm the feasibility and effectiveness of the approach.

2. UNSUPERVISED LEARNING APPROACH

As stated earlier, most studies with very few exceptions (typical of which is the work of Panda and Rosenfeld [5]) base their decision process on a single scalar at each pixel position. Although Panda and Rosenfeld [5] do consider both the pixel intensity and edge value together, their study is essentially restricted to manual assessment of the joint usefulness of these attributes rather than development of an automated unsupervised learning methodology capable of considering a general multiattribute set. Here our emphasis is on the latter aspect, and, accordingly, a specific set of attributes was chosen purely for illustrative purposes.

At the first instance, the learning environment is viewed as completely unsupervised and nonparametric. This naturally leads to clustering techniques as the most viable approach for accomplishing this learning. In view of the large data size involved in the context of most images encountered in practice, clustering methods based on intersample distance measures (in the selected attribute space) are deemed impractical. Accordingly, the most suitable approach would be the one based on assessment of density of samples in this multidimensional attribute space. On the basis of prior experience in this problem area, a multidimensional histogram based approach was chosen [6]. However, this approach which was designed to identify major clusters, required significant modifications, since in this application the emphasis is more on identification of smaller clusters. The major steps of the resultant approach, as shown in Table 1, are now considered in detail.

Table 1. MAJOR STEPS OF THE PROPOSED APPROACH

- Select a Set of Candidate Attributes.
- Obtain the Multidimensional Histogram of the Image Data Corresponding to the Selected Attributes.
- Formulate Clusters by Traversing trough the Hills and Valleys of the Histogram Space.
- Develop Intercluster Boundaries and Identify Cluster Class Labels of all Pixels Individually.
- Review/Threshold Labeled Image to Collect Desired Segments of the Image.
- Refine the Image Segments of Interest.

2.1 Selection of Candidate Attributes

This is a key step in the process. The effectiveness of the total processing is dependent to a large extent on the choice of the attributes, as this defines the attribute space in which the learning is carried out. This effort cannot be purely analytical or computational in its scope, as the initial choice is tied to an understanding of the physics of the problem, and will be based mainly on prior experience in the area. This leads to a possibly subjective list of potential attributes from which a subset (or a linear/nonlinear combination thereof, including possible transforms) is to be chosen. This subset selection can, however, be automated by defining appropriate figures of merits for individual and/or subset of attributes. For example, variance can be a measure of merit in that we are seeking attributes along which there is sufficient spread to permit some kind of discrimination. Also, univariate histogram analysis [7] along these attributes may be revealing as multimodal distributions denote presence of separable clusters. Thus, an effective figure of merit, which takes into account these factors, can be defined. This is currently under development and will be reported in due course.

2.2 Multidimensional Histogram Generation

Basic to the generation of the multidimensional histogram is a definition of the geometry of the histogram cells in the N-dimensional space. Conceptually, the most satisfactory shape of these cells would be one that would ensure that each cell is equidistant from all the neighboring cells surrounding it. The simplest shape that has this property in a two-dimensional space is the hexagon. Extension to three and higher dimensional space can be visualized by construing the hexagonal cell as being composed of six equilateral triangles each joined to two others along its sides. The three-dimensional equivalent of this hexagon cell can then be thought of as a set of 16 simplicies (the three-dimensional equivalent of the two-dimensional equilateral triangle) put together such that three of its faces are shared by three similar simplicies. The three-dimensional space is then viewed as a set of these 16 faced polytopes, each with 16 such neighboring polytopes. Conceptually, this could be extended to higher dimensions also as the space derived by putting together appropriately dimensioned simplicies. However, a histogram implementation, with such complex shaped histogram cells, in terms of identifying the location of samples in these cells, is computationally complex. Accordingly, recourse is taken to an implementation based on the much simpler definition of histogram cells as hyper rectangular objects (or cubes) with N pairs of parallel hyperplanes, each pair being perpendicular to every other pair. Of course, here each cell will have a total of 3^N-1 neighbors, consisting of N sets of 2^iNC_i neighbors, each set being at a uniquely different distance (from the central cell) depending on the number of coordinates in which they differ from the cell under consideration. Of these,

2N neighbors are fundamental or first-order neighbors differing in only one coordinate from the central cell, and unless specified otherwise, these 2N neighbors are deemed to be the neighbors of the cell.

Let

$$X = \{X^j = \{x_i^j : i = 1, \ldots N\} : j = 1, \ldots P\}$$

be given set of 'P' pixels, each described by a 'N' dimensional attribute vector, which are to be processed to derive a set of 'M' inherent clusters where M is to be self-learned by the clustering scheme.

Let k_i be the number of cell divisions prescribed externally as the parameter of the histogram along the attribute i. Then, the total number of attribute subspaces or regions defined in the N dimensional attribute space for generating the histogram is

$$K = \prod_{k=1}^{N} k_i$$

(Note: $k_i \geq 2$ V i = 1, ...N for effective use of all the attributes). Let P_k be the number of pixels assigned to the k^{th} subspace as determined by the multidimensional histogram analysis

$$\text{i.e } \sum_{i-1}^{K} P_k = P$$

This process of generating the multidimensional histogram of the given data set, although conceptually straightforward, could lead to complexities in implementation. For example, even with large grid sizes, i.e., with small number of cell divisions being prescribed for the histogram analysis, a large dimensional data environment could lead to rather astronomical values of K which represent a corresponding core memory demand on the computational facility. Virtual memory is not a satisfactory solution as this increases input/output operations to impractical levels. However, in practice, it is observed that very many of these K subspaces remain empty even after all the P pixels have been assigned to their corresponding cells. This is because of the inherent distributions of the pixels of the different edge classes and regions of sparse density separating the clusters of pixels. One can take advantage of this fact by requisitioning storage corresponding to the nonempty cells only. Thus, in reality, one needs only a fraction: $K_f (K_f << K)$ memory locations to store the set of samples spread superficially over K cells in the multidimensional space. This Optimal Kerneiling, i.e., indenting for only as many storage locations as are essential, calls for an implementation similar to the standard techniques for storage of sparse arrays. Under

such a mode of operation, as each pixel is input to the system, a check has to be made as to whether an appropriate storage already exists or a fresh storage is to be requisitioned. This of course necessitates maintaining a register of indented bins containing their addresses which correspond to the location of the cell in the multidimensional attribute space. Thus, a net savings in core memory requirements can be visualized if less than half of the K cells are populated. However, in practice, far less than K/2 cells are populated, and the histogram analysis package can be easily implemented in this mode. Furthermore, the histogram analyser, as designed here, in addition to storing the density of the nonempty cells, keeps track of the averages of the pixel data values (in each attribute) in each of these nonempty cells in order to define the centroids of the cells at a later stage. This calls for an additional set of 'N' arrays each of a length equal to the density array. Thus, the total memory requirements, if not optimized, would be $(N+1)\cdot K$ locations. However, optimizing the storage requirements by storing information pertaining to nonempty cells only, one would need only $(N+2)\cdot K_f$ locations including the additional array needed to store the addresses of the nonempty cells. Thus, the percentage savings achieved by an optimal implementation will be all the more significant. As is to be expected, this implementation increases CPU time, as whenever a sample is input, a check has to be made against the array of addresses of storage bins indented at that stage to determine the need for indenting a new store. This traditional tradeoff between memory and CPU time has to be assessed considering several factors such as pixel set size P, its dimensionality N, number of grids K, and weighing them against available computational facilities in terms of core size and time. Experience has shown that, in general, unless one has access to an exceptionally large core and the spread in the data is such as to permit choice of unusually low k_i values, it is far more practical to go in for the optimum core implementation as described above. This permits relatively more freedom of choice in grid sizes for the histogram generation process, because time limitations can be viewed as relatively open-ended as compared to memory limitations which are necessarily very finite. Any resultant increase in CPU time can perhaps be tolerated far more easily than large increase in memory. In most cases, this increase in CPU time is almost insignificant even for relatively large data sets.

Here, the choice of appropriate values for k_i is obviously in the hands of the analyst and the process of clustering is indeed sensitive to these values in terms of the level of the resultant categorization. The larger the grid sizes (lower the k_i values), the coarser the grid and the smoother is the histogram which then necessarily leads to fewer clusters and vice versa. This is in fact a desirable latitude or freedom, as one can either look for only major clusters or classes, or go in for more minute classification depending on one's needs and limitations in terms of computational costs. This is certain to be different in different applications; hence, an option in terms of externally choosing the

level or fineness of the learning system may indeed be desirable. Here, one could visualize unequal grid sizes along the different attribute directions depending on the spread of the data, and perhaps variable grid sizes even along one attribute direction. Of course, such grid positioning is difficult to conceive of unless one has a priori knowledge about the variations in the density along the different attribute dimensions. Thus, the choice of the grid positions along the different attribute directions is dictated by whether or not a preprocessing (as discussed earlier) in terms of unidimensional histogram analysis has been carried out. Such a preprocessing, while essential for the purpose of overcoming the "curse of dimensionality" through dimensionality reduction, is not so necessary merely from the point of view of locating the appropriate grid positions. In the event such a preprocessing has been undertaken for the purpose of attribute ordering and selection, the information derived therein may be utilized to position the grids at the significant valleys of the histograms of the corresponding attributes, and thereby enhance to some extent the reliability of the ensuing multidimensional histogram analysis. If, however, no preprocessing is contemplated for dimensionality reduction, it is not advisable to go in for it merely to locate the grid positions, especially when relatively small grid sizes are employed in the histogram process. In such cases, equal grid sizes (leading to possibly unequal k_i values depending on the spread of the data in the different directions) may be employed in the multidimensional histogram evolution. Thus, the outputs of this histogram analysis package are: the address array storing the addresses of all nonempty cells, the density array storing the number of pixels assigned to these cells and 'N' average arrays storing the N attribute values averaged over all the samples assigned to the corresponding cells. This represents the most significant part of the computational expense of the proposed learning scheme. The computational effort, involved in checking the address array each time a new pixel is input to determine whether an appropriate set of bins already exist, in proportional to K_f, and represents the major part of this expense. In this context, one could visualize having the address array ordered and instituting a binary search through a recursive array segmentation procedure, which can conceivably reduce the computational effort of the search. But, this process of having the address array in order at all times calls for reordering each time a new cell (and a corresponding new entry in the address array) is encountered. Thus, in most cases, the expense of keeping the array organized wipes out the advantages to be gained by an organized search procedure.

2.3 Cluster Formulation

The output of the histogram analyzer, consisting of the density, sample averages, and addresses of all the nonempty histogram cells, is input into another processor designed to develop the inherent clusters through merging of the cells with their higher density neighbors. This merging technique essentially consists of connecting each cell in the multidimensional histogram

space with its higher density neighbors and processing the resulting merger monitor matrix to develop the boundaries of each of the clusters inherent in this histogram space. These boundaries are necessarily fuzzy in that many of the cells along the common joint boundaries cannot always be uniquely identified with only one of the clusters, but are likely to be identified as belonging to the set of clusters sharing the particular boundary.

Furthermore, this merging of cells with their higher density neighbors, can be carried out with or without updating of the density and average values in the cells. If this merger or connectivity is carried out without any changes in the density and average values, the process leads to the identification of the hills and valleys as they exist in the histogram space defined by the input. If, for example, some of these hills are the results of overlapping distributions, then identifying the centroids and boundaries of such hills would lead to clusters which may in reality correspond to a mixture of more than one inherent category. This error is especially likely whenever coarse grid sizes are employed in generating the histogram. On the other hand, if the pixels contained in each cell were to be reassigned into its higher density neighboring cells during the process of connecting them, with the density and average values of these cells being updated accordingly, a certain transformation or distortion is induced on the histogram terrain leading to creations of new peaks wherever the gradients are relatively small. This choice brings up the question of whether such artificial distortion is desirable or indeed even tolerable. There is, of course, no mathematically justifiable answer to this query, as desirability of an externally induced distortion depends on the user's subjective needs. Following the adage that the end justifies the means, one has to decide on the basis of the nature of the resulting clusters. In general, it is to be expected that this distortion will lead to relatively larger number of clusters as compared to the undistorted version of the connectivity procedure. This has been experimentally confirmed.

Therefore, the choice between the two alternatives is clearly dependent upon the user's tolerance to the two types of errors, each of which are likely to be caused by one of the two approaches: The undistorted histogram could lead to more than one of the inherent classes being lumped into a single cluster, and the distortion process can lead to breaking up a single cluster into subclusters. It is therefore necessary to view the likelihood or relative probability of these errors occuring in a given environment. But such probabilities are never known a priori. However, we do know that for coarse grid sizes the chances of occurrence of the former type of error is relatively higher and vice versa. This can be kept mind in deciding on the approach to be employed for the given problem. Another aspect to be considered is that in most cases, the latter error of breaking up a single class into subclusters is the lesser of two evils. This is because one could always combine or merge them together, if need be, at a later stage without much difficulty. But overcoming the former type of error is hardly ever feasible at a later stage. The computational view point also supports such a choice in that operating even at a

relatively course grid level, i.e., at relatively less computational expense, (when the likelihood of the latter type of error is small) one can derive a correspondingly larger number of clusters, i.e., attain a finer level of discrimination. This is especially true in this application as we are particular of detecting all cluster classes, however small in population.

This dictates, in most cases, the choice to be that of updating the density and average values at each stage of the connectivity process through proportionate reassignment of the contents of the cell under processing to all of its neighboring higher density cells. (At very coarse grid sizes, the reassignment causes no effective changes in the relative ordering of the cells in view of the comparatively large gradients in the histograms, and both approaches would result in essentially the same cluster set.)

The connectivity process leads to categorizing the originally nonempty cells, depending on whether there were any inward and/or outward connectivity (or sample reassignment in the case of distortion inducing connectivity process) to or from these cells, into one of the following six sets:

- The set of cluster nuclei cells, which, having only lower density neighbors, had inward connectivity but no outward connections;

- The set of cluster interior cells, which, having both lower and higher density cells, had both inward and outward connections, the latter being limited to cells belonging to the same single cluster;

- The set of saddle point cells, which again had both inward and outward connections, the latter leading onto the cluster nuclei of more than one cluster;

- The set of valley point cells, which, having only higher density neighbors, had no inward connections from other cells but the outward connections leading onto more than one cluster as in the case of saddle point cells;

- The set of exterior boundary cells, which, again, had no inward connectivity, but had its outward connections limited to cells belonging to a single cluster as was the case of cluster interior cells.

- The set of isolated singularity cells, which, having no nonempty neighbors, are completely unconnected with the rest of the cells, and are viewed as independently single cell clusters. (Depending on their density levels and/or their relative distances from the outer clusters, they could be small but significant regions of interest or noise.)

This categorization is derived by processing the connectivity matrix developed during the process of connecting the cells and identifying the terminal cluster nuclei cells for each cell by tracing the connectivity through this matrix. This, along with appropriate flag arrays to denote existance of inward and outward connectivities, completes the categorization of the cells.

2.4 Discriminant Design and Pixel Labeling

The processing carried out thus far has, in effect, defined completely the set of all the clusters, their nuclei and their boundaries, albeit in an implicit sense. A more explicit definition of the cluster boundaries can best be achieved by developing the discriminant hyperplanes (assuming of course, that the clusters are linearly separable; otherwise appropriate nonlinear discriminant surfaces may be defined in a similar fashion), separating the clusters. The problem of determining the discriminant hyperplanes in this case is more complex than is the case in the classical discrimination problem given supervised training data sets. Of course, the centroid of each cell can be viewed as a pseudo training sample of the class corresponding to the cluster of this cell. But, the complexity arises from the fact that many of these cells, such as valley and saddle point cells, are associated with more than one cluster, and they form a fuzzy boundary between the corresponding clusters. Thus, the problem is one of defining discriminent functions to separate clusters with fuzzy boundaries. This is tackled here by using a previously reported algorithm (DHARMA[8]).

Once this discriminant design is completed, each histogram cell represented by its centroid can be labeled as to its cluster allocation. All the pixels alloted to each of these cells are accordingly labeled by a table look-up approach using the previously stored histogram information.

The net result of this processing is now a clustered image with each pixel designated by a cluster number representing the knowledge accumulated by the unsupervised learning effort.

2.5 Labeled Image Assessment

The labeled image in effect represents a segmented image with each set of pixels with identical cluster values representing a specific segment of interest. Depending on the attributes used, these segments have different physical significance. For example, in using pixel intensity and gradients as attributes, the cluster segments are likely to represent targets, target-background boundaries, background subregions, background subregion boundaries, etc. The extraction of the segments of interest is now a routine task in view of the available pixel by pixel labeling. Once the segments are developed, the next step is, as before, obtaining features such as slope/size measures for target identification purposes. However, the extracted segments can be further refined if desired prior to feature measurements, so as to improve the reliability of these measurements. As is likely, the extracted

segments represent a far smaller data set compared to the total image. This makes it feasible to bring into action more sophisticated pattern recognition tools [9] which otherwise are computationally expensive to be applied to the total image.

If multiple targets are likely in the scene, one could visualize this refinement activity as a preamble to target classification. If identity of some of the targets can be established by external means (for example on the basis of prior information, processing of images of the same scene at an earlier date, etc.) then it is possible to utilize more sophisticated pattern recognition tools of learning, such as learning under an imperfect teacher in developing reliable target identification capabilities. While these possibilities are being explored, the present study is being reported to demonstrate the feasibility of utilizing pattern recognition tools in the nontraditional role of feature extraction.

3. IMPLEMENTATION EXPERIENCE

The processing methodology presented thus far was implemented on a PDP-11/70 using a Night Vision Lab (NVL) data set. After some preliminary assessment, four attributes were selected to demonstrate the unsupervised learning approach developed here. These were: the pixel intensity value, averaged vertical and averaged horizontal gradients (each averaged over the 3 x 3 neighborhood of the pixel under consideration), and the Laplacian. The photographs, which show the raw and segmented images, clearly bring out the effectiveness of the methodology in delineating the edges of targets of interest. (Color slides being shown at the oral presentation bring out distinctly the different categories of edges.) The processing resulted in 12 clusters. Of these, Clusters 1 through 8 are seen to essentially correspond to background areas and edges between the subregions of the background. Clusters 9 through 12 represent the target and target/background edges. Segmentation in these terms effectively delineates the target from the background. The grouping of pixels in this region can, if needed, be further refined to make the subsequent feature measurements (size/shape descriptors) more reliable for target classification. While further work of a more detailed nature in terms of attributes assessment and development of more effective attributes are on the anvil, this presentation is being made mainly to portray the viability of extending pattern recognition concepts and methodology to the traditional domain of image segmentation and edge detection.

REFERENCES:

1. A. Rosenfeld, "Picture Processing: 1978," Computer Graphics and Image Processing, Vol. 8, pp. 354-393, 1979.

2. J. S. Weszka, "A Survey of Threshold Selection Techniques," Computer Graphics and Image Processing, Vol. 7, pp. 259-265, 1978.

3. S. A. Dudani and A. L. Luk, "Locating Straight Line Edge Segments on Outdoor Scenes," Pattern Recognition, Vol. 10, pp. 145-157, 1978.

4. J. I. Toriwaki and T. Fukumura, "Extraction of Structural Information from Grey Pictures," Computer Graphics and Image Processing, Vol. 7, pp. 30-51, 1978.

5. D. P. Panda and A, Rosenfeld, "Image Segmentation by Pixel Classification in Gray Level-Edge Value Space," IEEE Transactions on Computers, Vol. C-27, pp. 875-878, 1978.

6. B. V. Dasarathy, "HINDU: Histogram Inspired Neighborhood Discerning Unsupervised Pattern Recognition System," Proceedings of the Purdue Symposium on Machine Processing in Remotely Sensed Data, 1976.

7. B. V. Dasarathy, "AHISMA: Ad Hoc Histogram Information Measure Sensing Algorithm for Feature Selection in the Context of Histogram Inspired Clustering Techniques," Proceedings of the IEEE, Vol. 64, pp. 1446-1447, 1976.

8. B. V. Dasarathy, "DHARMA: Discriminant Hyperplane Abstracting Residuals Minimization Algorithm for Separating Clusters with Fuzzy Boundaries," Proceedings of the IEEE, Vol. 64, pp. 823-824, 1976.

9. B. V. Sheela and B. V. Dasarathy, "A New Algorithm for Optimal Partitioning and Learning in Non-parametric Unsupervised Environments," Int. Jl. of Computer and Information Sciences, Vol. 8, pp.239-253, 1979.

SESSION IB

IMAGE PROCESSING AND PATTERN RECOGNITION

ALGORITHMS FOR MISSILE GUIDANCE

Paper No. IB-1, Presented at the Workshop on Imaging Trackers and Autonomous Acquisition Applications for Missile Guidance, 19-20 November 1979, Redstone Arsenal, Alabama.

Phase-Slipped Time Delay and Integration

E. H. TAKKEN
and
A. F. MILTON
Naval Research Laboratory
Washington, D. C. 20375

Abstract

A new form of time delay and integration with serially scanned detector arrays is proposed in order to facilitate reliable automatic detection of point source targets with scanning infrared search systems which have limited sampling frequency.

Introduction

Automatic target detection requires the use of a threshold exceedence sensing device. A standard arrangement for scanned systems is shown in Fig. 1. An "optimized" filter is placed after the detector in order to maximize the peak signal-to-noise ratio passed on to the threshold exceedence sensor. To be truly optimum this detection filter must be specified using the power spectral density of both the anticipated target signal and of the background clutter.[1]

Unfortunately, neither the target signal nor the background clutter can be well characterized a priori in many infrared systems. A prescription for circumventing some of the problems caused by imprecise scene clutter information has been addressed previously.[2,3] The problem discussed here is the one arising from variability in target signal caused by limited sampling frequency in the direction of scan. This variability arises through aliasing of frequency components in the signal which are higher than the Nyquist frequency and is due to a randomness of phase between the position of point source targets and the timing clock of a scanning discrete-time sampled infrared sensor. The problem occurs both for scanning CCD and CID arrays and for systems passing the output of a conventional detector through CCD delay-line electronics thereby creating a sampled analog signal.

The problem with such discrete-time sampled systems is that they split up space into discrete cells in the scan direction. Just as with staring mosaics, the sensor's response to a point source will depend on whether the blur circle image of the source happens to fall on or

between the discrete cells. Phase-slipped time delay and integration (PSTDI) is a method for introducing an extra MTF before sampling to reduce the components of the signal above the Nyquist limit thereby reducing discrete-time cell boundary effects in the scan direction. An analogous high speed mechanical dither could accomplish the same result in the cross-channel direction but is not analyzed here.

The Model

Figure 2 depicts the instantaneous signal output from an infrared detector being scanned by the blurr circle of a point source. For the purposes of modeling this bell-shaped curve is represented here by the raised-cosine signal $\frac{1}{2} + \frac{1}{2} \cos(\pi\, t/t_d)$, for $|t| \leq t_d$ but zero for $|t| \geq t_d$. As indicated in the figure the half-width for this model signal is t_d, with the blurr circle first reaching the detector at $t = -t_d$ and then finally leaving the detector at $t = +t_d$. The first zero in the power spectral density of this signal occurs at $f = \pm 1/t_d$.

When this continuous signal is integrated and sampled by a CCD-type sensor the result is a series of charge packets. The case of sampling time t_s equal to the dwell time t_d is indicated in Fig. 2b. The cross-hatched areas show the successive parts of the continuous signal that are integrated in the CCD. For what is called Phase (a) the first integration sampling period happens to commence simultaneously with initial contact of the model blurr circle with the detector. The initial charge packet then results from an integration over the left half of the continous signal.

For what is called Phase (b) the sampling clock is displaced by half a sample period. In this case the initial charge packet results from an integration over only the first quarter of the continuous signal. The numerical packet sizes given in Fig. 1b, $S_a = (0, .5, .5)$ and $S_b = (.091, .818, .091)$, are derived from integration of the model raised-cosine signal over the regions indicated.

Of course, Phase (a) and Phase (b) are not the only situations that could occur, because in fact the timing relationship between the target signal and the sampling clock is totally arbitrary. It is this multiplicity of possibilities that makes it impossible to define a matched or optimized filter in the usual way. For the case shown in Fig. 2 the Nyquist frequency is $1/(2\tau d)$ which results in considerable aliasing.

What was done in references 2 and 3 was to specify the filter relative to a signal defined as the average overall possible signal phases. This average signal for $t_s = t_d$ was shown to consist of three charge packets proportional to $S_{Avg} = (.149, .703, .149)$. A tapped-delay-line filter matched in white noise will then simply have the weights $w_{wn} = S_{Avg}$, while a filter optimized to detect this average signal in a low frequency $1/f^2$ clutter background was shown to have the weights $w_{1fc} = (-.184, .370, -.184)$.

Figure 3 shows the filter's peak amplitude output $S^T \cdot w_{1fc}$ as a function of sampling rate $r = t_d/t_s$ for two different filters. Considerable variability is indicated at low sample rates depending on the phase of the input. More variability is evidenced for the high-pass filter (w_{1fc}) since aliasing has more effect on higher frequencies. In practical systems sample rates and signal-to-clutter ratios are often limited so this variability can make detection less reliable.

Time Delay and Integration (TDI)

With serial scan systems a single detector is replaced by a linear array of detectors oriented in the direction of scan and the outputs from this array are coherently added together with time delay and integration to increase signal-to-noise ratio. Standard TDI does not help however with signal-to-clutter ratios since clutter is relatively stationary with time. Figure 4 shows an example of how time delay and integration can be accomplished with an off-chip CCD delay line. For standard TDI, timing of the shift register is designed to ensure that the charge packets moving down the register end up being a coherent sum of the sampled outputs from each of the successive detectors.

A general relationship for standard TDI is that the number of packet intervals between delay line inputs

$$x_2/vt_s = K \qquad (1)$$

must be an integer. Here x_2 is the interdetector spacing, v is the scan velocity at the focal plane and t_s is the time interval between samples.

The desired value for this parameter K depends both on the ratio x_1/x_2 of detector width to interdetector spacing and on the desired degree of over sampling, since the number of samples taken per dwell time is just

$$r \equiv t_d/t_s = x_1/vt_s = K(x_1/x_2). \qquad (2)$$

The larger the sampling rate r, the larger the delay line length must be in order to accommodate larger values for the parameter K.

Phase-Slipped TDI

The intent of phase-slipped TDI is to perturb the timing relationships of standard TDI in such a way so as to introduce a slight phase shift between the signals sampled by successive detectors in the TDI array. If these delays are introduced evenly with a large number of detectors, an extra MTF filter before sampling of sinc ($\pi t_s f$) will be introduced. This has a value of 0.65 at the Nyquist frequency $f = 1/2t_s$ and falls rapidly thereafter. Thus, the high-frequency content of the signal will be attenuated before sampling, and aliasing will be reduced. The final phase-slipped TDI sum will still be variable but

much less than before. To achieve the desired result one wants the first and last detectors in the array to initiate signal sampling at times differing by nearly a whole sampling time t_s.

Phase-slipped TDI is accomplished by changing the relationship in Eq. (1) to

$$x_2/vt_s = K(1 \pm 1/N) \tag{3}$$

where N is the number of detectors in the TDI linear array. As a practical matter phase-slipped TDI can be most easily implemented by dropping the CCD clock frequency $f_c = 1/t_s$ to

$$f_c' = f_c/(1 + 1/N) \tag{4}$$

This approach will of course also have an influence on the number of samples per dwell.

Figure 5 shows an example of phase-slipped TDI signal summing for the cases of a two-detector array with sampling being made at the rate of once per dwell time. New Phase (a) and new Phase (b) refer to the extreme phase relationships for the case of phase-slipped TDI. Clearly with phase-slipped TDI the output is less variable and in all cases is more like the new S_{Avg} appropriate after PSTDI. For $t_s = t_d$ and $N \to \infty$ the new S_{Avg} becomes (.001, .186, .627, .186, .001).

PSTDI Performance

The final effectiveness of phase-slipped TDI must be judged by the consistency of the peak signal output from a filter optimized to detect the appropriate S_{Avg} in the existing clutter environment. If the signal interference consists of low-frequency $1/f^2$ clutter the filter weights are w_{1fc}. The PSTDI signals most similar and dissimilar to S_{Avg} are S_b and S_a which have to be evaluated for various values of r and N. The raised cosine instantaneous signal model of Figs. 2 and 5 is assumed, and resultant filter outputs $S_a^T \cdot w_{1fc}$ and $S_b^T \cdot w_{1fc}$ are first normalized to $S_{Avg}^T \cdot w_{1fc}$ and then plotted in Fig. 6.

The figure shows that phase-slipped TDI is effective even for small values of N. With just a two-stage PSTDI the filtered peak target signal is already highly reproducable even for low values of the sampling rate. This satisfies the need for reliable target detection with an automatic threshold exceedence sensor. Some signal-to-clutter penalty is of course paid on the average by reducing the MTF and spreading the signal with PSTDI. However, since the peak of S_{Avg} is only slightly attenuated this loss in signal-to-clutter should both be small and less than the variability that occurs without phase-slipped TDI.

Aliasing could also be reduced by defocussing the optics and increasing the size of the blurr circle. However, since this approach would simply degrade an existing MTF rather than introduce a new, multiplicative one, the unwanted high frequencies could not be significantly attenuated without also inflicting loss of the desired signal at frequencies below the Nyquist limit. Phase-slipped TDI attenuates troublesome aliasing frequencies before sampling with less effect on the desired signal. Since a more consistent signal is derived from point-source targets, the use of PSTDI will help both with automatic threshold detection and with any post detection clutter rejection algorithms that depend upon accurate measures of peak amplitude.

REFERENCES

1. M. J. Skolnick, Introduction to Radar Systems (McGraw-Hill Book Company, 1962, NY). Expressions for optimal signal detection filters are given in Eqs. (9.27) and (9.28).

2. R. Nitzburg, E. H. Takken, D. Friedman and A. F. Milton, "Spatial Filtering Techniques for Infrared (IR) Sensors," Society for Photo-Optical Instrumentation Engineers, Vol. 178-Smart Sensors.

3. E. H. Takken, D. Friedman, A. F. Milton and R. Nitzburg, "A Least-Mean-Square Spatial Filter for IR Sensors," Applied Optics, December 1979 (Note that Fig. 7 and Table I of this reference need correction. The S_b values for k = 1 of the LNS filter should both be dropped by 0.1 in Fig. 7. In Table I the last two rows should read 0.269, 0.141, 0.0717 and 0.136, 0.188, 0.299.

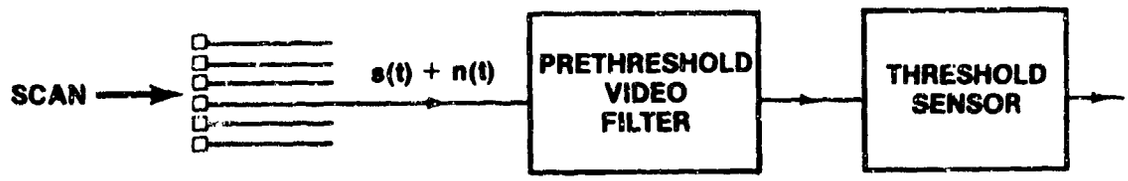


Fig. 1 - Pictoral representation of an automatic signal detection system using a parallel scan linear array.

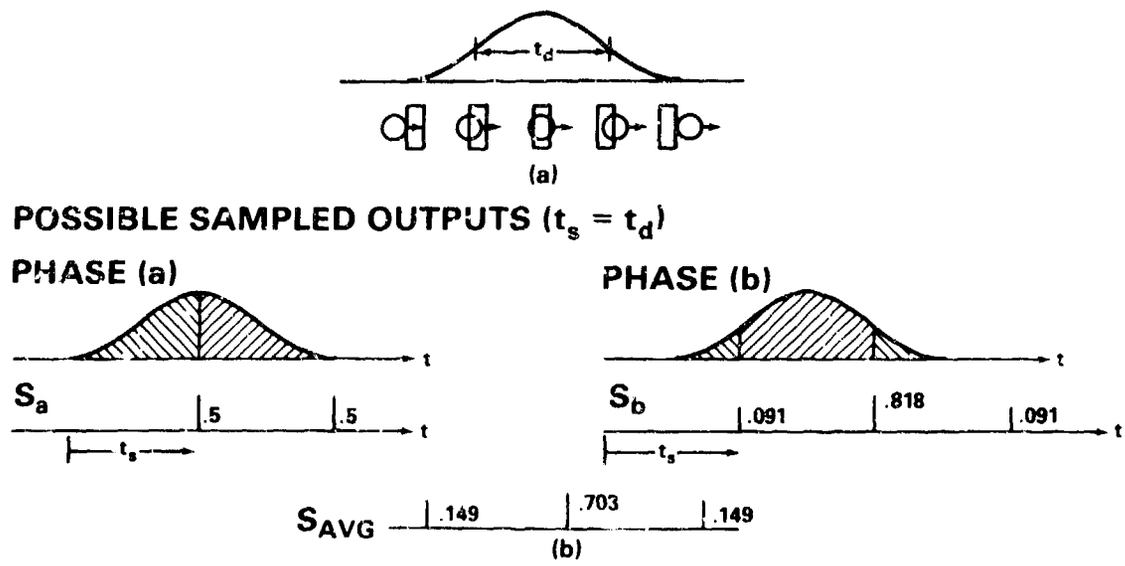


Fig. 2 - The (a) instantaneous versus the (b) time-sampled signals of a point-source blurr circle in a scanning infrared sensor. Phase (a) and Phase (b) differ by a shift of half a sample time of the discrete readouts relative to the input instantaneous signal.

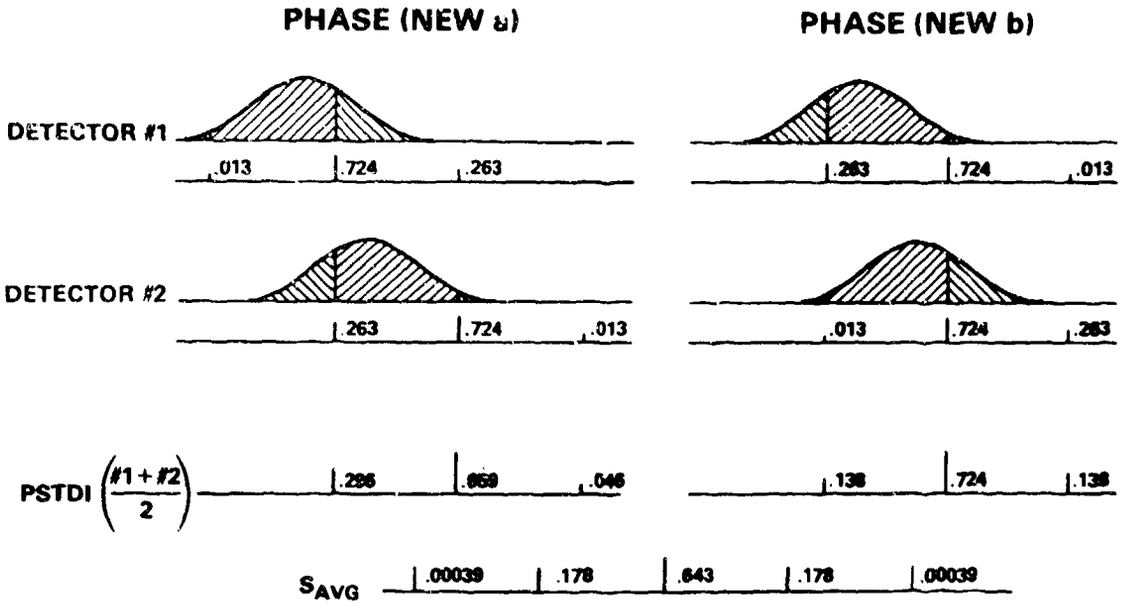


Fig. 5 - Two-detector phase-slipped TDI for $t_s = t_d$. The numerical charge-packet weights shown are derived from integrations over the indicated cross-hatched areas of the raised-cosine signal. No individual signal contains more than four packets although S_{Avg} calculated in the same way as for Fig. 3 contains five.

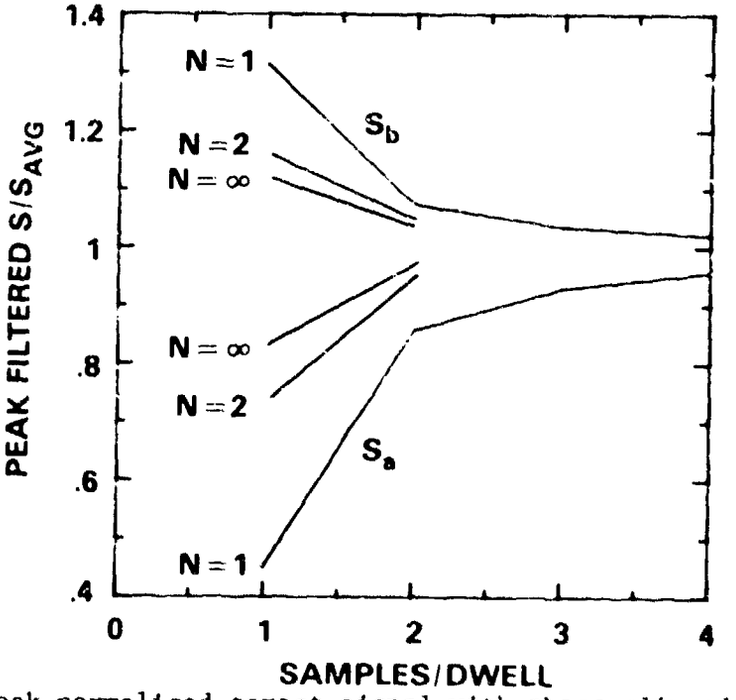


Fig. 6 - Peak normalized target signal with phase-slipped TDI after passing through detection filters optimized for the appropriate post-PSTDI S_{Avg} (N, r) in $1/f^2$ low-frequency clutter. N represents the number of detectors in the TDI row and r the number of samples per dwell time.

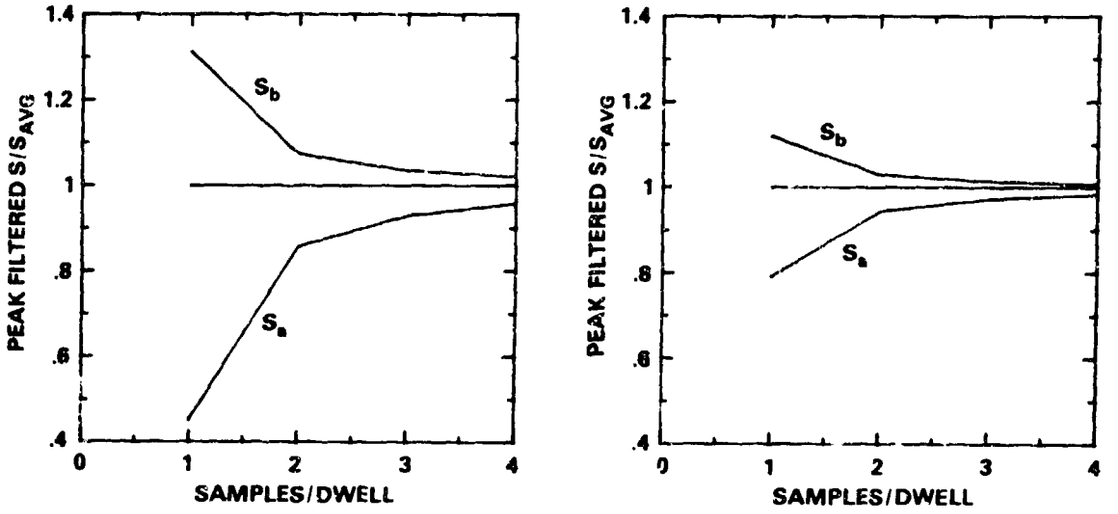


Fig. 3 - a) The normalized peak output of discrete-time detection filters optimized for S_{Avg} in $1/f^2$ clutter. b) The normalized peak output of a discrete-time detection filters optimized for S_{Avg} in white noise. The outputs are normalized to that for an input of S_{Avg} for each sampling rate.

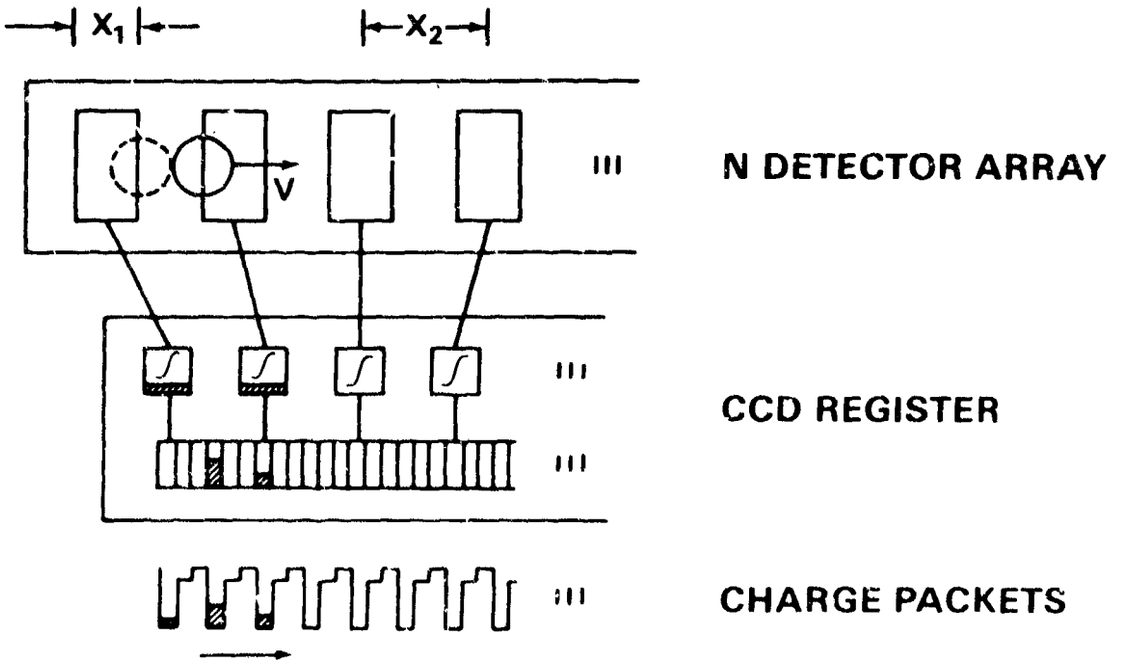


Fig. 4 - Time delay and integration using an off-chip CCD delay line. The case shown is for a three-phase shift register and a sampling rate of once per dwell. x_1 is the detector width and x_2 is the interdetector spacing.

Paper No. IB-2, Presented at the Workshop on Imaging Trackers and Autonomous Acquisition Applications for Missile Guidance, 19-20 November 1979, Redstone Arsenal, Alabama.

THE SYNTHESIS OF FFT BASED IMAGE ENHANCEMENT FILTERS

William H. Haas and V. Kumar

Rockwell International, Satellite Systems Division
12214 Lakewood Blvd., Downey, California 90241

ABSTRACT

A generalized approach to the synthesis of FFT-based image enhancement filters is considered here. An example is presented to demonstrate its application. Specific filter structures are introduced that utilize a fast Fourier transform (FFT) algorithm to perform both filter synthesis and the filtering operation itself. A frequency domain synthesis technique based on ideal filters is briefly introduced and applied to the problem of enhancing edge detail in two-dimensional areal images. These techniques are applicable to imaging target trackers and adaptive acquisition systems.

1. INTRODUCTION

Image processing applications are increasing at a rapid rate. Enhancement of images is a special part of image processing which is vital to all sorts of two-dimensional processing techniques. In particular, imaging target trackers and adaptive acquisition systems for missile guidance dictate enhancement techniques that are not only simple to implement but also exhibit high performance. Increasing the computational burden without a significant increase in performance would become costly and at the same time impractical.

A frequency domain synthesis technique using FFT-based image enhancement filters is presented. This technique can be applied to the synthesis of filters that are implemented directly in the frequency domain. These filters also can be approximated with space domain transversal or recursive structures.

New filter structures for real-time digital image processing are becoming possible due to the development of efficient integral transform algorithms and recent developments in solid-state electronic devices that permit economical use of high-speed parallel and pipelined processors in relatively small systems. Two such adaptive filter structures are presented in Figure 1. With these structures, receiver filter inputs, $r(x)$, are used with a model for the signal, $S(j\omega)$, to be enhanced or detected to adaptively synthesized filters rather than just adjust them. The filters are not gradually adjusted in a control loop so as to seek the optimum but are periodically synthesized to be optimum for the measured inputs. In this paper, an algorithm is presented that would be implemented in the FFT filter synthesis block.

2. THEORY

The filter synthesis presented in this paper is based on a general method that can be called "the method of ideal filters".[1]

By this method one can obtain an optimum realizable filter in two steps:

1. Obtain an ideal* filter transfer function in the frequency domain.
2. Approximate this ideal transfer function with the transfer function of some constrained realizable filter.

*To clarify our terminology, an ideal filter is not constrained by stability or reliazability while an optimum filter is.

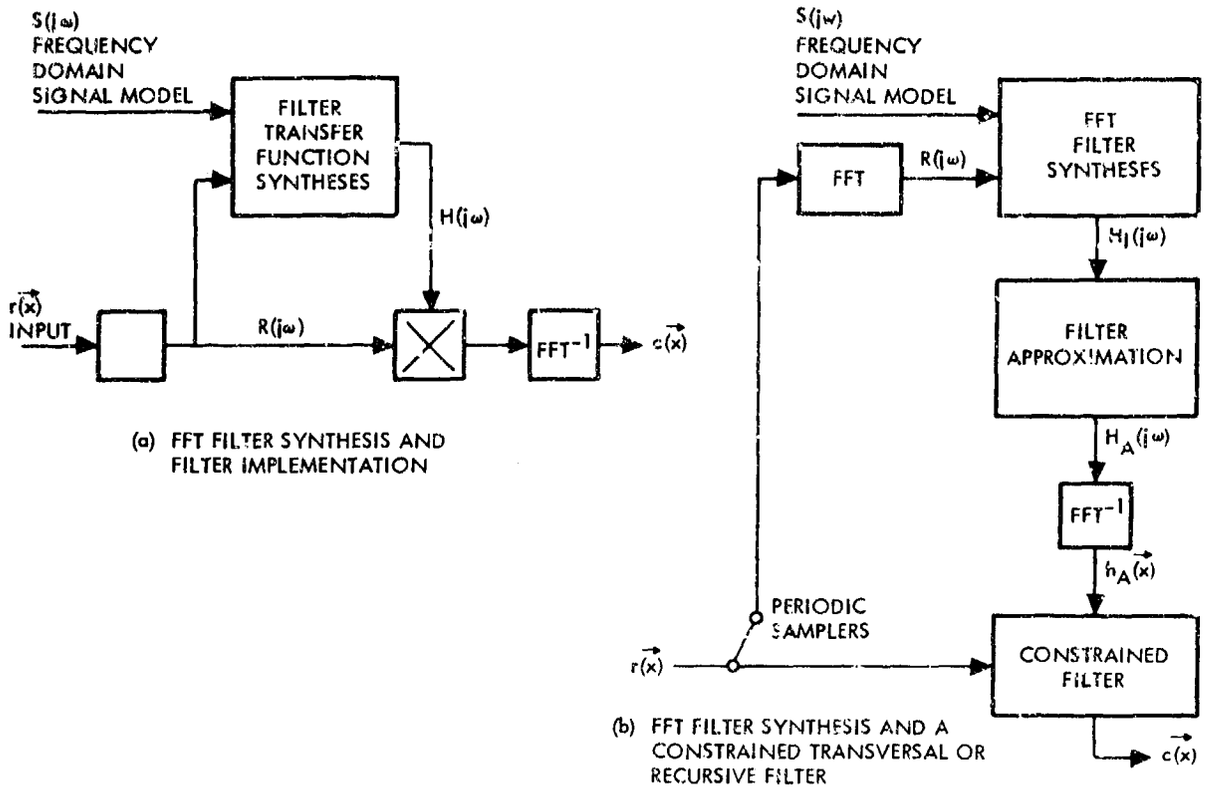
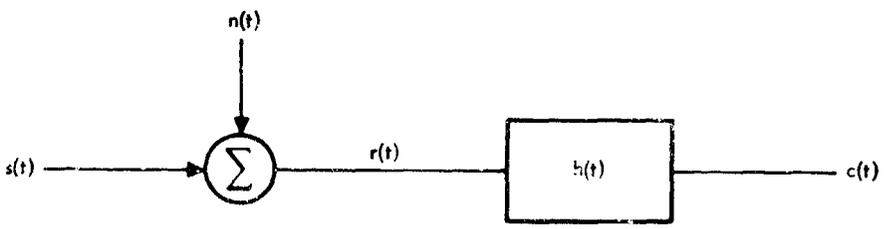


Figure 1. Basic FFT-Based Filter Structures

A block diagram of a simple signal-in-additive-noise is presented in Figure 2 to define notation. Table 1 shows a list of such ideal filter transfer functions. The filter approximation $H_A(j\omega)$ will be optimum if the distortion index D, defined by

$$D = \int_{-\infty}^{\infty} \left| R(j\omega) \right|^2 \left| H_A(j\omega) - H_I(j\omega) \right| \, d\omega \qquad (1)$$

has been minimized.



s(t) = THE INPUT SIGNAL

n(t) = THE INPUT NOISE

r(t) = THE TOTAL FILTER INPUT

h(t) = THE FILTER IMPULSE RESPONSE

c(t) = THE FILTER OUTPUT

Figure 2. Block Diagram of Basic Linear Filtering System Assuming Additive Noise

Table 1. Ideal Linear Filter Transfer Functions

ITEM	TRANSFER FUNCTION $H(j\omega)$
1. GENERAL ESTIMATION FILTER FORMULA	$\frac{R(-j\omega)D(j\omega)}{R(j\omega)R(-j\omega)} = \frac{\text{DESIRED INPUT-OUTPUT CROSS-CORRELATION}}{\text{INPUT AUTOCORRELATION SPECTRUM}}$ $= H_E(j\omega)$
2. GENERAL DETECTION FILTER FORMULA	$\frac{S(-j\omega)}{R(j\omega)R(-j\omega)} = \frac{\text{CONJUGATE OF SPECTRUM TO BE DETECTED}}{\text{AUTOCORRELATION OF SIGNAL TO BE REJECTED}}$ $= H_D(j\omega)$
3. UNCORRELATED ESTIMATION DETECTION	$H_E(j\omega)H_D(j\omega)$
4. UNCORRELATED[1] ESTIMATION (WIENER)	$\frac{\lvert S(j\omega)\rvert^2}{\lvert S(j\omega)\rvert^2 + \lvert N(j\omega)\rvert^2}$
5. CORRELATED ESTIMATION (GENERALIZED WIENER)	$\frac{S(j\omega)\lvert S(-j\omega) + N(-j\omega)\rvert}{\lvert S(j\omega) + N(j\omega)\rvert^2}$
6. AUTOCORRELATION[1] ESTIMATION	$\frac{S(-j\omega)\lvert S(j\omega)\rvert^2}{\lvert S(j\omega)\rvert^2 + \lvert N(j\omega)\rvert^2}$
7. CLASSICAL DETECTION (PRE-WHITENED MATCHED)	$\frac{S(-j\omega)}{\lvert N(j\omega)\rvert^2}$
8. HIGH RESOLUTION DETECTION	$\frac{S(-j\omega)}{\lvert S(j\omega)\rvert^2 + \lvert N(j\omega)\rvert^2}$
9. PULSE SHAPING DETECTION	$\frac{S(-j\omega)F(j\omega)}{\lvert S(j\omega)\rvert^2 + \lvert N(j\omega)\rvert^2}$, $F(j\omega)$ = DESIRED PULSE SHAPE

A block diagram of a linear systems model for an edge-enhancement filtering problem is presented in Figure 3. For this problem, it is assumed that the system blur transfer function $G(j\omega)$ is known, that the total filter input $R(j\omega)$ is known, an edge model $E(j\omega)$ is given, but that neither the non-edge spatial structure, $N(j\omega)$, nor the specific image edge structure, $S(j\omega)$, are known *a priori*. The problem then is to synthesize an optimum filter, $H_I(j\omega)$, that will tend to transform edge features modeled by $E(j\omega)$ into features with the arbitrarily chosen model, $F(j\omega)$, and to attenuate other features of the selected ideal filter.

Filter No. 9 in Table 1—the pulse-shaping detection filter—seems to be the most appropriate for our application. It has been shown[2] that this filter can be viewed as a least square estimator as well as a detection filter. With reference to Figure 3, if we knew $N(j\omega)$, the noise expectation spectrum, and $S(j\omega)$, the signal expectation spectrum, then our ideal filter could be written as

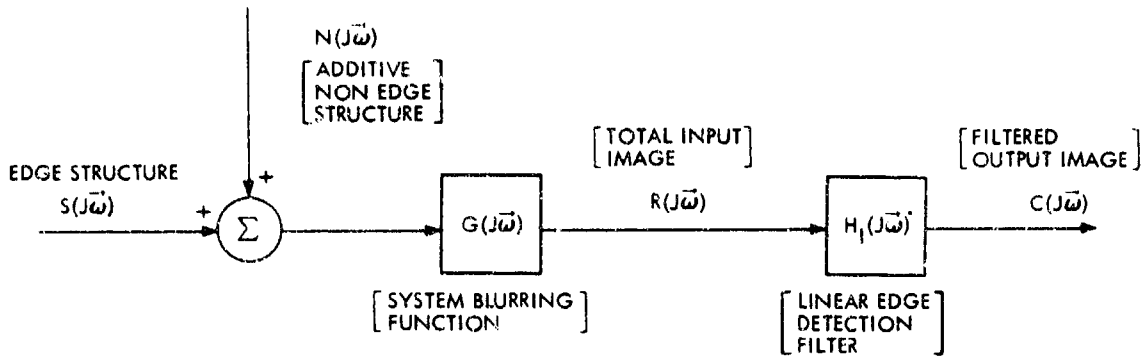
$$H_I(j\omega) = \frac{S(-j\omega) \cdot F(j\omega) \cdot G(j\omega)}{\left[\lvert S(j\omega)\rvert^2 + \lvert N(j\omega)\rvert^2\right]} \tag{2}$$

However, we do not know specifically $S(j\omega)$ or $N(j\omega)$, though we know the blurred image $G(j\omega)$ $R(j\omega)$ and a model for the signal, namely $E(j\omega)$. This leads us to consider a filter of the form

$$H_I(j\omega) = \frac{E(j\omega)\, F(j\omega)\, G(j\omega)}{\lvert R(j\omega)\rvert^2} \tag{3}$$

$$= \frac{E(-j\omega)\, F(j\omega)\, G(j\omega)}{\lvert S(j\omega)\rvert^2 + \lvert N(j\omega)\rvert^2 + 2S(j\omega)\, N(-j\omega)^*} \tag{4}$$

*This is really $S(j\omega)\, N(-j\omega) + S(-j\omega)\, N(j\omega)$ but it does not matter since we must consider both positive and negative frequencies together.



$\vec{\omega}$ = SPATIAL FREQUENCY VECTOR

THE PURPOSE OF $H_I(j\vec{\omega})$ IS TO MAXIMIZE THE PEAK ABSOLUTE INTENSITY OF THE EDGE STRUCTURE OUTPUT RESPONSE TO THE RMS INTENSITY OF THE NON EDGE STRUCTURE OR NOISE OUTPUT RESPONSE

Figure 3. Model of Linear Edge Enhancement Filtering Problem

which is similar to the pulse-shaping detection filter except for the $2\,S(j\omega)\,N(-j\omega)$ cross-spectral density term in the denominator.

For the typical arbitrary scene that will be input to an autonomous imaging tracking or guidance system, $S(j\omega_i)$ and $N(j\omega_i)$ ordinarily will not be correlated with $S(j\omega_k)$ and $N(j\omega_k)$, $\omega_i \neq \omega_k$. The $2S(j\omega)\,N(-j\omega)$ cross-spectral density term can hence be made arbitrarily small by forming a smoothed or windowed approximation to $|R(j\omega)|^2$ which we shall write as $\widehat{|R(j\omega)|^2}$. A general expression for a proposed ideal filter to perform edge enhancement can then be written

$$H_I(j\omega) = \frac{E(-j\omega)\,G(j\omega)\,F(j\omega)}{\widehat{|R(j\omega)|^2}} \tag{5}$$

Of particular note concerning this proposed ideal filter is that it can be synthesized from (1) *a priori* definitions of the features to be enhanced, $E(j\omega)$, (2) what the features should look like after they are enhanced, $F(j\omega)$, (3) the assumed optical blurring function $G(j\omega)$, and (4) a smoothing of the observed input $R(j\omega)$ to yield $\widehat{|R(j\omega)|^2}$. It is because the noise is buried in the $R(j\omega)$ term that the proposed filter will be adaptive to the noise that is encountered.

3. ILLUSTRATIVE EXAMPLE

To illustrate the performance of our proposed filter, we used a Landsat image* of Columbus, Ohio, as an input, R(x,y), and ran it through the algorithm. This image is presented in Figure 4. The smoothed power spectrum estimate of the input image, $\widehat{|R(j\omega)|^2}$, shown in Figure 5, was obtained by taking FFT power spectral estimates of each X-direction row of image pixels and averaging them. An edge model, e(x), was generated and is presented in Figure 6. Its magnitude spectrum $|E(j\omega)|$ is presented in Figure 7. $G(j\omega)$ and $F(j\omega)$ were assumed to be equal to unity. The X direction magnitude response of the resulting ideal filter is presented in Figure 8.

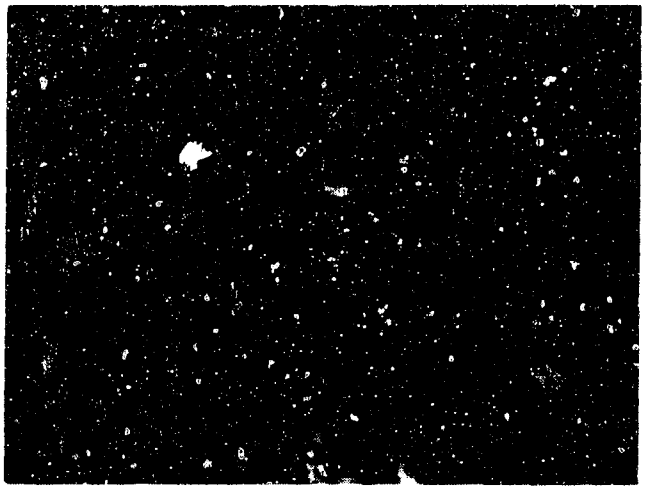


Figure 4. Landsat Image Over Columbus, Ohio, r(x,y)

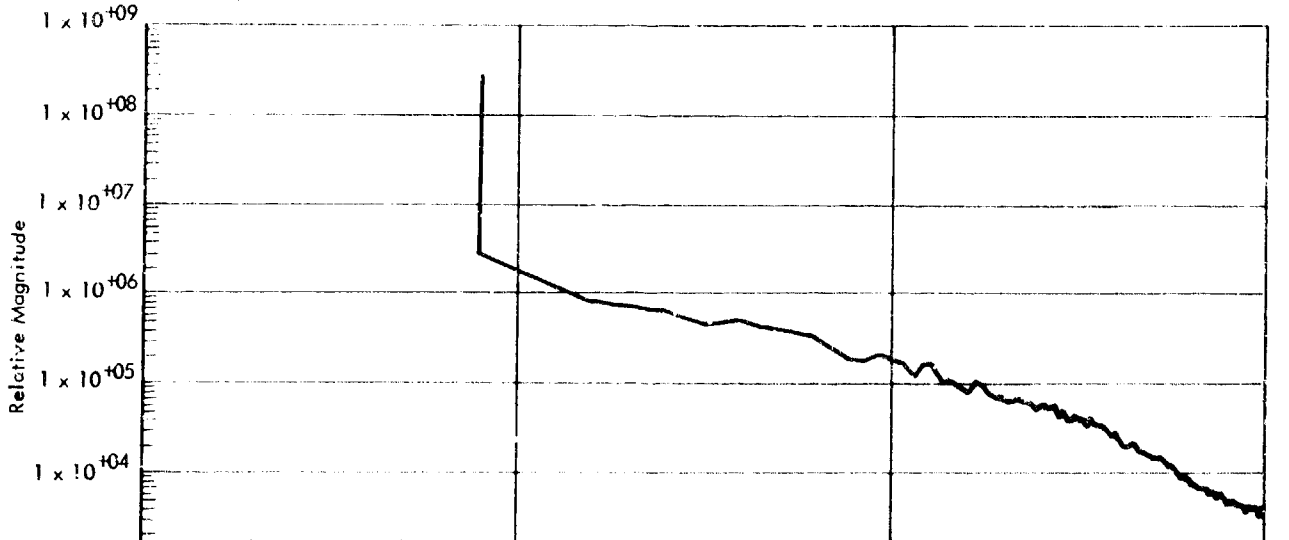


Figure 5. Averaged x Direction Power Spectrum, $\overline{|R(j\omega)|^2}$

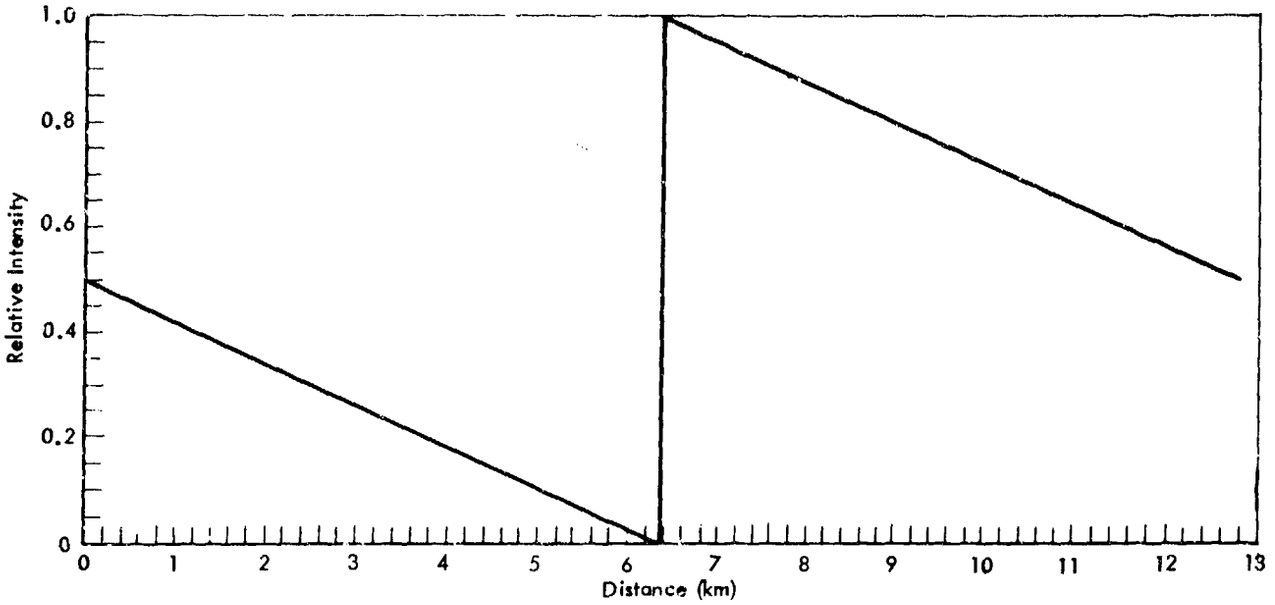


Figure 6. Model of an Edge, e(x)

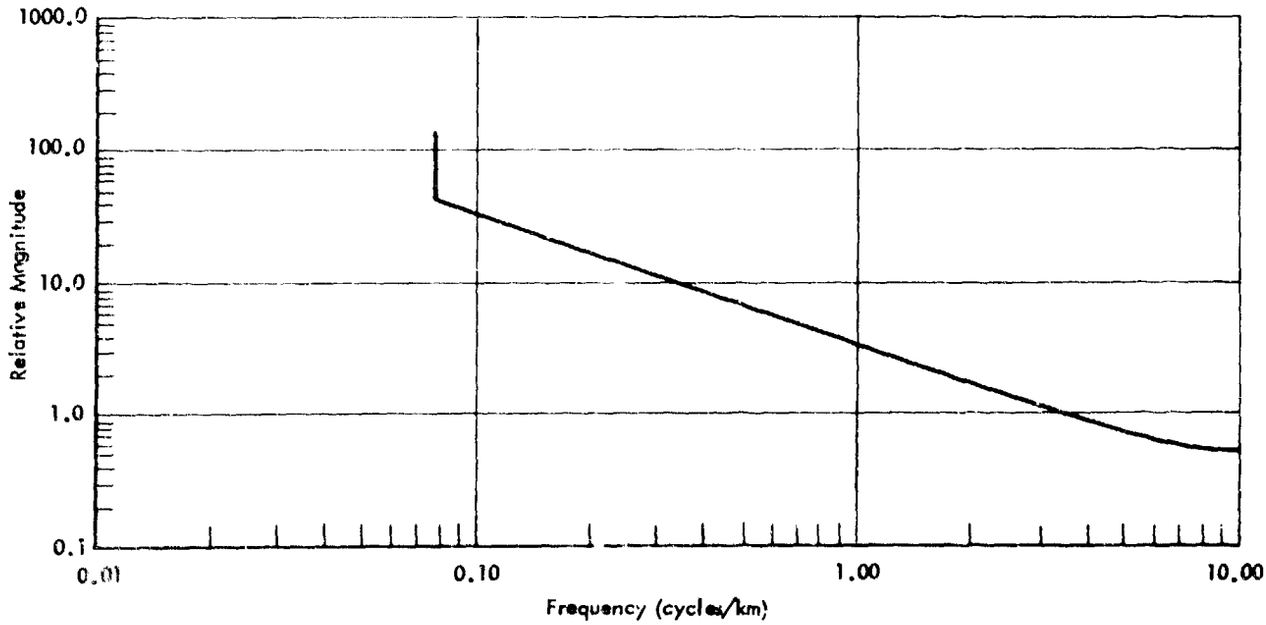


Figure 7. Magnitude Spectrum of an Edge, $E(j\omega)$

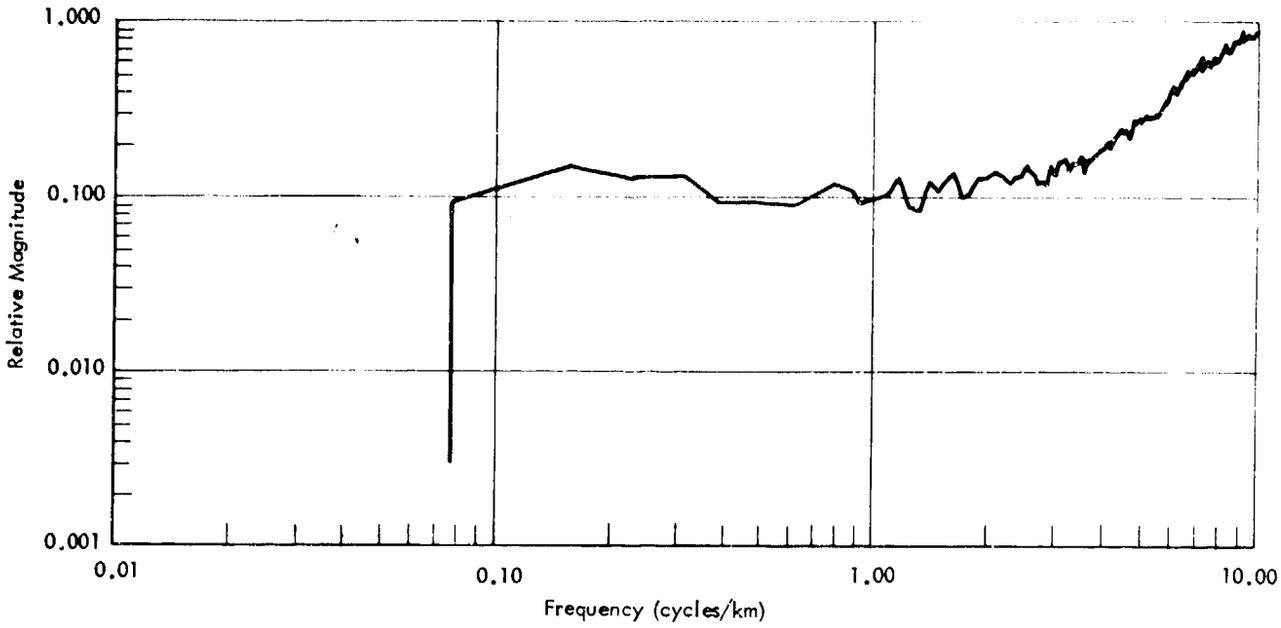


Figure 8. Edge Enhancement Filter Magnitude Response, $H(j\omega)$

A two-dimensional filter was generated from the one-dimensional filter by rotation in the frequency domain such that

$$H_I(j\omega_r) = H_I(j\omega_X), \quad \omega_r = \sqrt{\omega_X^2 + \omega_Y^2} \tag{6}$$

Convolving the input image with the two-dimensional spatial filter yielded the filter output, c(x,y), as shown in Figure 9. In this image the dark gray shades indicate negative edges and the lighter gray shades indicate positive edges. $|\, c(x,y)\,|$ and $LOG_N[c(x,y)^2]$, which illustrate the edges more graphically, are presented in Figures 10 and 11, respectively.

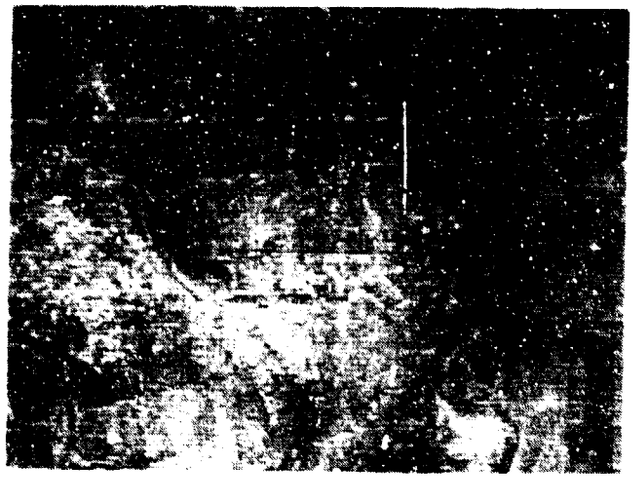


Figure 9. Edge Enhanced Lansat Image, c(x,y)

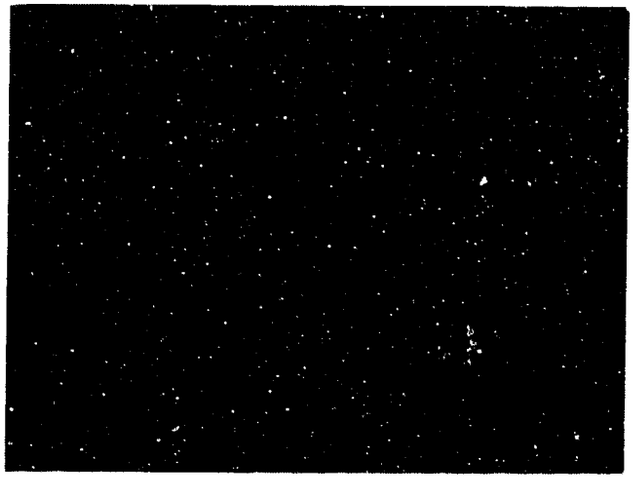


Figure 10. Landsat Image Edges c(x,y)

To reduce the degradation of non-edge features, two additional filters, $H_1(j\omega)$ and $H_2(j\omega)$, whose magnitude responses appear in Figures 12 and 13, were used. These filters are defined as:

$$H_1(j\omega) = \sqrt{\frac{E(-j\omega_x)}{\widehat{|R(j\omega_x)|^2}}} \tag{7}$$

$$H_2(j\omega) = \sqrt{H_1(j\omega_x)} \tag{8}$$

Output images obtained with $H_1(j\omega)$, $c(x,y)$ and $|c(x,y)|$ are presented in Figures 14 and 15, respectively, and output images obtained with the $H_2(j\omega)$ filter, $c(x,y)$ and $|c(x,y)|$ are presented in Figures 16 and 17, respectively. These images all show different degrees of enhancement.

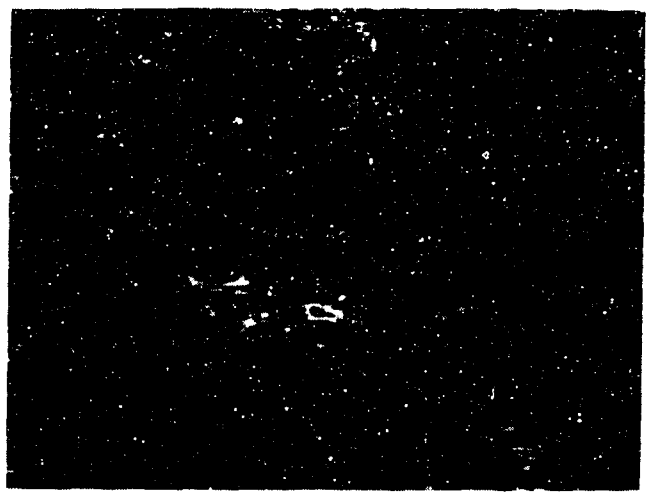


Figure 11. Landsat Image Edges, L_N $c(x,y)^2$

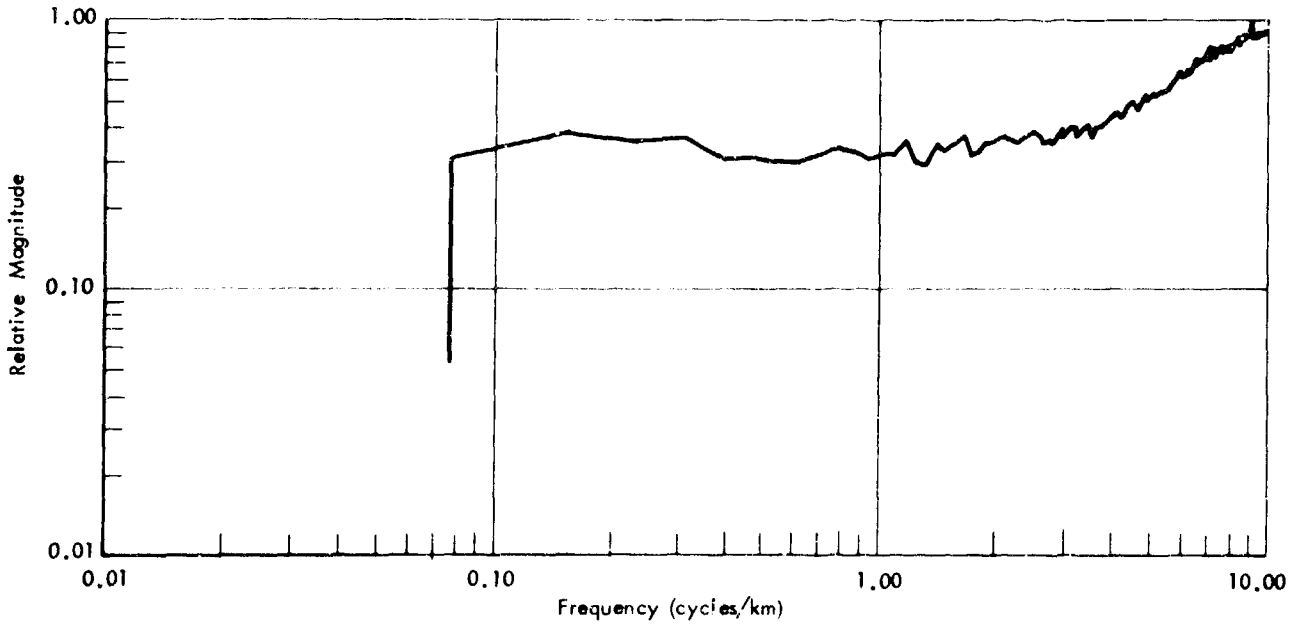


Figure 12. Edge Enhancement Filter Magnitude Response $H_1(j\omega)$

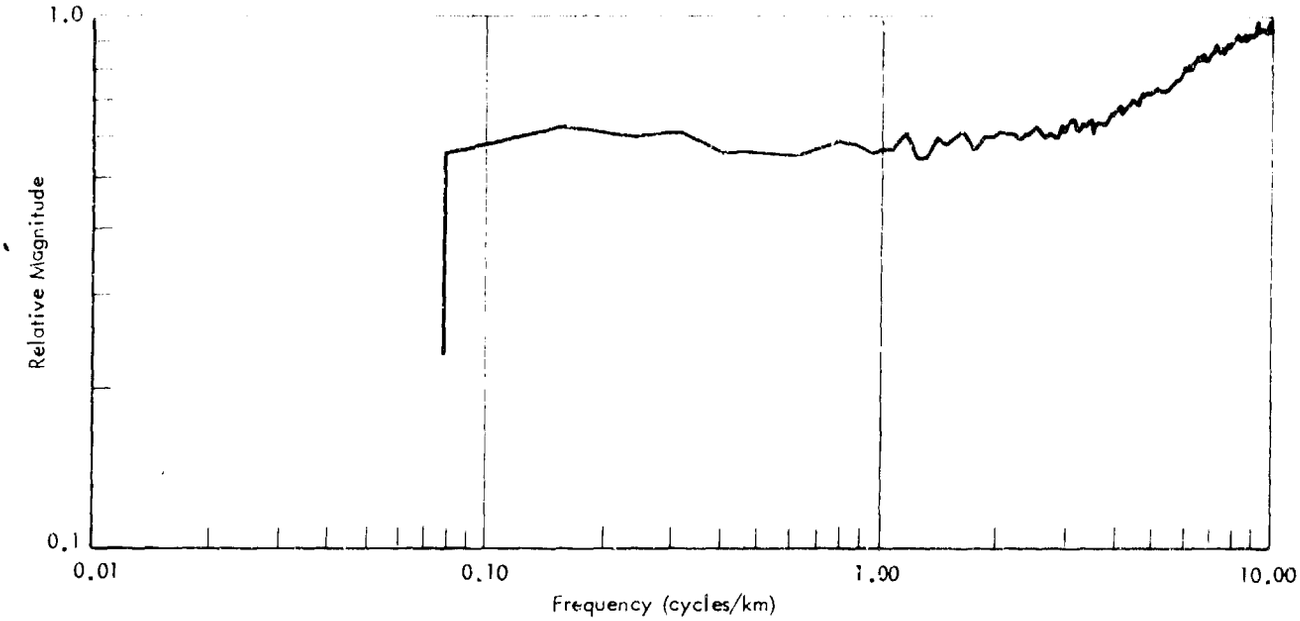


Figure 13. Edge Enhancement Filter Magnitude Response $H_2(j\omega)$

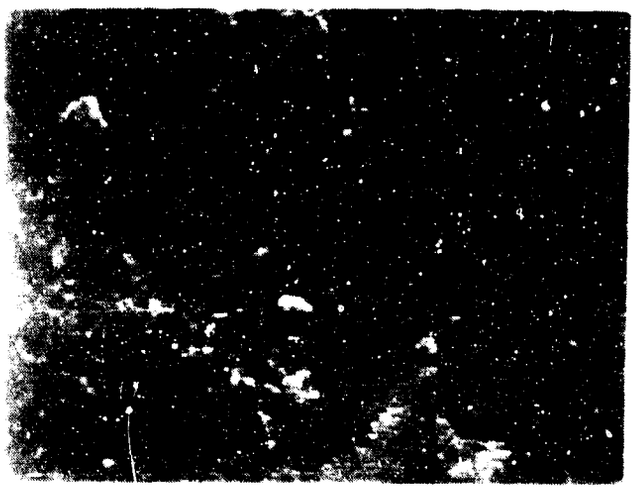


Figure 14. Areal Image Enhanced by Filter $H_1(j\omega)$

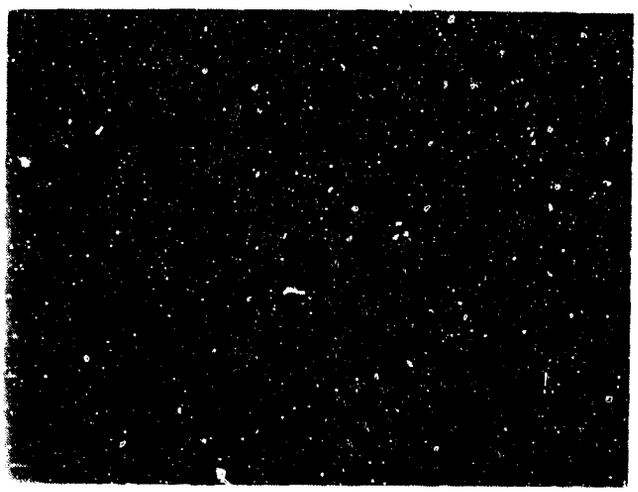


Figure 15. Absolute Value of Image Enhanced by $H_1(j\omega)$

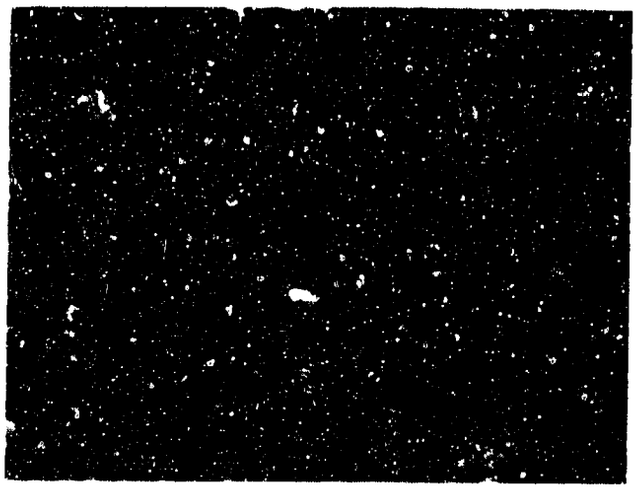


Figure 16. Same Image Enhanced by Filter $H_2(j\omega)$

Figure 17. Absolute Value of Image Enhanced by $H_2(j\omega)$

4. CONCLUSIONS

Synthesis of FFT-based image enhancement filters is presented here and its application is demonstrated for an areal image. The synthesis technique presented here involves a general frequency domain approach and yields edge environment filters that are much simpler for their mathematical representation than others recently presented.[3] While the theory leads to an ideal filter, realizable filters can be obtained through approximations. For the specific example illustrated here, a mild edge enhancement technique identifies areas of intensity variation while maintaining much of the image characteristics. On the other hand, a strong enhancement filter shows the steep variations only, and thus other non-edge details are hidden.

REFERENCES

1. Haas, W.H., G.S. Dixon, and N.Y. Soon, "Performance of Mosaic Sensor Dim Target Detection Algorithms," Proc. SPIE, Vol. 178 (April 1979), pp. 25-32.

2. Haas, W.H. and C.S. Lindquest, "Linear Detection Filtering in the Context of a Least Squares Estimator for Signal Processing Applications," Proc IEEE Intl. Conf. on Acoustics, Speech, and Signal Processing (April 1977), pp. 651-654.

3. Shanmugam, K.S., F.M. Dickey, and J.A. Green, "An Optimal Frequency Domain Filter for Edge Detection in Digital Pictures," IEEE Trans or Pattern Analysis and Machine Intelligence, Vol. PAMI-1, No. 1 (January 1979), pp. 31-49.

*This image was used simply because it was readily available.

Paper No. 1B-3, Presented at the Workshop on Imaging Trackers and Autonomous Acquisition Applications for Missile Guidance, 19-20 November 1979, Redstone Arsenal, Alabama.

NOISE FILTERING IN MOVING IMAGES

T. S. Huang
School of Electrical Engineering
Purdue University
West Lafayette, Indiana 47907

ABSTRACT

A number of linear and nonlinear temporal filters for noise reduction in image sequences have been simulated on computer. The results will be presented on a TV monitor from a video tape. Among the filters studied, temporal median filter along estimated direction of motion appears to give the best results.

INTRODUCTION

Many moving images collected by visible and infrared scanners are corrupted by random and burst noise (including line drop out). Reducing the noise will facilitate target detection, recognition, and tracking. In this paper we discuss temporal filtering techniques and present computer-simulation results of the application of a number of linear and nonlinear temporal filters to several noisy image sequences.

STRAIGHT TEMPORAL FILTERING

Let $f_k(i,j)$ denote the gray level of the ijth picture element (ith row, jth column) of the kth frame of the image sequence, and $g_k(i,j)$ that of the corresponding picture element in the filtered image sequence. A nonrecursive straight temporal filter over (2K+1) frames is defined by

$$g_k(i,j) = F\{ f_{k-K}(i,j), f_{k-K+1}(i,j), \ldots, f_k(i,j), \ldots, f_{k+K}(i,j) \} \quad (1)$$

Two examples are:

(i) Linear time-variant filtering

$$F\{ x_{k-K}, x_{k-K+1}, \cdots, x_k, x_{k+1}, \cdots, x_{k+K} \}$$

$$= \sum_{m=-K}^{K} a_m x_{k+m} \tag{2}$$

where a_m are constants.

(ii) Median filtering

$$F\{ x_{k-K}, \cdots, x_k, \cdots, x_{k+K} \}$$

$$= \text{Median}\,(x_{k-K}, \cdots, x_k, \cdots, x_{k+K}) \tag{3}$$

A recursive straight temporal filter is defined by

$$g_k(i,j) = F\{ f_{k-K}(i,j), f_{k-K+1}(i,j), \cdots, f_k(i,j); g_{k-M}(i,j), g_{k-M+1}(i,j), \cdots, g_{k-1}(i,j) \} \tag{4}$$

where K and M are positive integers.

Two examples are:

(iii) Linear time-invariant filtering

$$F\{ x_{k-K}, \cdots, x_k; y_{k-M}, \cdots, y_{k-1} \}$$

$$= \sum_{n=-K}^{0} a_n x_{k+n} + \sum_{m=-M}^{-1} b_m y_{k+m} \tag{5}$$

where a_n and b_m are constants.

(iv) Median filtering

$$F\{ x_{k-K}, \cdots, x_k; y_{k-M}, \cdots, y_{k-1} \}$$

$$= \text{Median}\,(y_{k-M}, y_{k-M+1}, \cdots, y_{k-1}, x_k) \tag{6}$$

In the experiments reported in this paper, filters (i) and (ii) are included with $(2K+1) = 3$ and $a_m = 1/3$. Note that for white Gaussian random noise, averaging in the temporal direction of N frames will reduce the noise variance by a factor of N. Median filtering will reduce the noise variance by a factor of only $2N/\pi$. However, for reducing salt-and-pepper noise and burst noise (including line dropout), median filtering is much more effective [1].

MOTION-COMPENSATED TEMPORAL FILTERING

Both averaging and median filtering (in the temporal direction) will degrade (blur) moving objects. To reduce this degrading effect, we propose to estimate the direction of motion at each picture element and then do the filtering along that direction.

We shall consider the nonrecursive filtering case, the recursive case being entirely similar. To obtain the filtered point $g_k(i,j)$, we track the object point located at the ijth element of the kth frame over the $(2K+1)$ frames to be used in the filter expression, Eq. (1). Let the coordinates of this object point in the $(k+m)$th frame be $\vec{u}_{k+m}$, $m = -K, -K+1, \ldots, -1, 0, 1, \ldots, K-1, K$. Thus $\vec{u}_k = (i,j)$. The filtering is defined by

$$g_k(i,j) = F \{ f_{k-K}(\vec{u}_{k-K}), \ldots, f_k(i,j), \ldots, f_{k+K}(\vec{u}_{k+K}) \} \tag{7}$$

Two examples are:

(v) Linear time-invariant filtering, motion compensated.

(vi) Median filtering, motion compensated.

In the experiments reported in this paper, filters (v) and (vi) are included with $(2K+1) = 3$, and $a_m = 1/3$. The motion was estimated in the following way. The sample variances are calculated for the 9 triplets $\{ f_{k-1}(i-m, j-n), f_k(i,j), f_{k+1}(i+m, j+n) \}$ for $\{ n=0, m=0, \pm 1, \pm 2, \pm 3, \}$ and $\{ m=0; n= \pm 1 \}$.

The triplet with the smallest variance is taken as the direction of motion. For example, if the variance is smallest for n=0; m=2, then

$$\vec{u}_{k-1} = (i-2,j)$$

$$\vec{u}_k = (i,j)$$

$$\vec{u}_{k+1} = (i+2,j)$$

and the linear filter output will be

$$g_k(i,j) = \frac{1}{3} f_{k-1}(i-2,j) + \frac{1}{3} f_k(i,j) + \frac{1}{3} f_{k+1}(i+2,j).$$

EXPERIMENTAL RESULTS

The experimental results were obtained by computer simulation using the Digital Video Store System [2] at INRS-Telecommunication. Three input sequences were used: (a) panning, (b) zooming, (c) conductor. Each sequence contains 36 frames (at 30 frames per second). Each frame contains approximately 256x256 samples with 8 bits per sample.

Four temporal filters were applied to each of these 3 sequences. These are filters (i), (ii), (v), and (vi) as described earlier in this paper.

The filtered results were recorded on a video tape. Part of this tape will be played at the workshop.

CONCLUDING REMARKS

The performance of the temporal filters can be compared only by reviewing the filtered sequences on a TV monitor. However, the following general conclusions can be stated.

(1) Overall, the motion-compensated median filter performs the best (in terms of reducing noise and preserving motion).

(2) Edges of slow-moving large objects are preserved remarkably well by median filtering even without motion compensation.

To improve the performance of these filters, one direction is to use more accurate motion estimation techniques, e.g., those proposed by researchers in the inter-frame coding area [3,4]. More generally, more sophisticated modeling of the image sequence is required, since in many cases because of change in illumination and obstruction, etc., an object point simply cannot be tracked.

ACKNOWLEDGEMENT

This work was carried out at INRS-Telecommunications, University of Quebec, where the author was a Visiting Professor June 1-August 31, 1979. The encouragement of Prof. M. L. Blostein, Director of INRS, and Prof. B. Prasada, Head of the Video Communication Group is greatly appreciated. The author also wishes to thank Profs. E. Dubois and S. Sabri for their invaluable help.

REFERENCES

1. B. Justuson, Median filters: Statistical properties, in "Two-Dimensional Filters and Transforms," ed. by T. S. Huang, Springer-Verlag, to be published 1980.

2. R. Johnson, J. Mastronardi, and G. Mony, A digital television sequence store, IEEE Trans. on Comm., vol. 26, no. 5, pp. 594-600; May 1978.

3. J. Limb and J. Murphy, Computer Graphics and Image Processing, 1975, pp. 311-327.

4. A. Netravali and Robbins, BSTJ, March 1979, pp. 631-670.

Paper No. IB-4, Presented at the Workshop on Imaging Trackers and Autonomous Acquisition Applications for Missile Guidance, 19-20 November 1979, Redstone Arsenal, Alabama.

FROM NUMERICAL TRANSFORMS TO SPATIAL FILTERS

Charles A. Halijak
University of Alabama in Huntsville
Huntsville, Alabama 35807

ABSTRACT

Matrix representation of a derivative depends on the numerical transform whereas the matrix representation for a high pass filter depends on the optimal Rader-Gold transform in digital filters, a corollary to the numerical transform. Matrix representation reveals the distinctness of the gradient and the high pass spatial filter. Applications of the gradient are given to divergence, Laplacian of images. Applications of the high pass filters are to edge and wedge detectors. The spatial DC notch filter is a hybrid of the gradient and the high pass filter. Vector space aspects of derivative matrices and physical realizations are presented.

NUMERICAL TRANSFORM

The numerical transform's [1] main purpose is to simulate linear dynamical systems on the digital computer. The subject begins with integration formulas such as

$$Z\left(\frac{1}{s}\bar{f}\right) \doteq \frac{Tz}{1-z}\, Z\bar{f} \tag{1}$$

$$Z\left(\frac{1}{s}\bar{f}\right) \doteq \frac{T}{1-z}\left[Z\bar{f}-f_o\right] \tag{2}$$

Z = Uniform sampler,

T = Sampling interval in seconds,

z = Exp $(-Ts)$, the delayor,

$\bar{f} = \bar{f}(s) = Lf(t)$.

Approximation goodness requires that $0<T<1$.

There exists no need for differentiation formulas in digital simulation of dynamical systems but they are important in image processing.

Digital filters are a corollary to digital simulation and the high pass filter and derivative are often equated. One purpose of this paper is to show that they are distinct indeed.

DIFFERENTIATION

Suppose attention is given only to Eqn. (2). Let $f(t) = dg/dt$. Then $f=s\bar{g}-g_0$ and some calculations lead to the differentiation formula

$$Z\bar{f} = \frac{1-z}{T} Z\bar{g} + (\dot{g}_0 - \frac{g_0}{T}). \tag{3}$$

If $Z\bar{f}+Z\bar{g}$ are replaced by the n-vectors $\tilde{f}$ and $\tilde{g}$, then a matrix representation for the derivative is at hand after one approximates $\dot{g}_0$ by $(g_1-g_0)/T$.

If $(1-z)$ is replaced by $(I-H)$ where

I = diag (1,1,1,...1), (number of 1's=n)

H = subdiag (1,1,...1), (number of 1's=n-1)

then one must account for the additive term

$$\dot{g}_o - \frac{g_o}{T} \doteq \frac{g_1 - 2g_o}{T} \tag{4}$$

which only occurs in the first component of the vector output One can then state that $(g_1 - 2g_0)$ is replaced by a matrix-vector product where the first row of the matrix is (-2, 1, 0, 0, 0) and the remaining rows are all zero and the column vector is $(g_0, g_1, g_2, g_3, g_4)^1$, where a 5-vector exemplifies the general n-vector case. The matrix representation of the derivative is the matrix in

$$\begin{pmatrix} f_0 \\ f_1 \\ f_2 \\ f_3 \\ f_4 \end{pmatrix} = \frac{1}{T} \begin{pmatrix} -1 & 1 & 0 & 0 & 0 \\ -1 & 1 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 \\ 0 & 0 & -1 & 1 & 0 \\ 0 & 0 & 0 & -1 & 1 \end{pmatrix} \begin{pmatrix} g_0 \\ g_1 \\ g_2 \\ g_3 \\ g_4 \end{pmatrix} \tag{5}$$

In n-vector symbolism, the above formula becomes

$$\underset{\sim}{f} = \frac{1}{T} D \underset{\sim}{g} \tag{6}$$

and D is definitely different from (I-H). Furthermore, because the first two rows of D are identical, D is a singular matrix. The rank of D is (n-1). Therefore, there exists a vector u such that Du=0.

Some vector notation needs to be formed before further study of D and (I-H) can proceed. Let u denote the n-dimensional column vector with all 1's and r denote the n-vector such that r' = (0,1,2,3,...,n-1). It is verbally convenient to call r the 'ramp vector', and to call u the 'all-unit vector'.

There is a need for vector counterparts of t^m where m = 0,1,2,3,... . The cases t^0 and t^1 are analogous to u and r respectively. For the general case t^m one can define r^m such that $(r^m)' = (0^m, 1^m, 2^m, 3^m, \ldots, (n-1)^m)$.

The D matrix accurately calculates the derivative of u and r; namely Du=0, Dr=u. However, (I-H) calculates the derivative of r only; namely (I-H) u=e_1, (I-H) r=u. Recall that e_i is the Euclidean n-vector with 1 in the i-th position and 0's elsewhere. Furthermore, $Dr^n \neq nr^{n-1}$ for $n \geq 2$.

The product rule does not apply in the matrix algebra with scalar elements; that is

$$D(fg') \neq (Df)g' + f(Dg)'. \tag{7}$$

LEMMA 1: If D is any n x n matrix, f and g are n-vectors, all have scalar elements, then

$$(D(fg')) = ((Df)g') \tag{8}$$

$$((fg')D') = (f(Dg)') \tag{9}$$

and the rank of the resulting tensor products can be 0 or 1 depending on whether D is singular. The negation of the product rule is an immediate corollary.

On the positive side, one obtains

$$D(f(Dg)') = (Df)\ (Dg)' \tag{10}$$

which is akin to the Ragazzini-Zadeh identity,

$$Z(\bar{f}Z\bar{g}) = (Z\bar{f})\ (Z\bar{g}), \tag{11}$$

in Z-transforms [3].

D-MATRIX AND VECTOR CALCULUS

Interesting associations are noted when n-vectors are replaced by n x n matrices. Using the vector calculus as a source of $g(x,y)$, $\partial/\partial x$ and $\partial/\partial x$ and $\partial/\partial y$, $x,y\in[0, (n-1)T]$, one can discretize

(a) $g(x,y)$ into G, an n x n matrix.

(b) $\frac{\partial g}{\partial x}$ into $\frac{1}{T}$ GD' where prime denotes matrix transpose,

(c) $\frac{\partial g}{\partial y}$ into $\frac{1}{T}$ DG,

(d) $\frac{\partial^2 g}{\partial x \partial y}$ into $\frac{1_2}{T}$ DGD'

(e) Dg into $\frac{1_2}{T}$ (GD'D'+DDG).

In effect, the derivative matrix D can operate on row vectors or column vectors of G, or both.

(f) $\nabla.\vec{g}$ into $\frac{1}{T}$ (DF+HD')

(g) $(\nabla x\vec{g}).\ \hat{k}$ into $\frac{1}{T}$ (FD'-DG)

For the sake of completeness, one should include the cross product in

(h) $\vec{a}x\vec{b}$ into $\begin{pmatrix} 0 & b_3 & -b_2 \\ -b_3 & 0 & b_1 \\ b_2 & -b_1 & 0 \end{pmatrix} \quad \begin{pmatrix} a_1 \\ a_2 \\ a_3 \end{pmatrix}$

These discretization from the vector calculus will be shortly modified into edge detectors. However, the D matrix and filters need to be studied first.

DIGITAL FILTERS

The numerical transform source of (I-H) remains to be found.

- The starting point is trapezoidal quadrature

$$Z(\frac{1}{s}\bar{f}) = \frac{T}{2}\ \frac{1+z}{1-z}\ [Z\bar{f} - \frac{f_o}{1+z}] \qquad (12)$$

which is obtained by averaging Eqn. (1,2).

- The second step is the Complete Tustin Program:

  If $\bar{\bar{y}} = \frac{N(s)}{D(s)} \bar{\bar{x}}$ and $w = \frac{2}{T} \frac{1-z}{1+z}$ then

  $$[Z\bar{y} - \frac{y_o}{1+z}] = \frac{N(w)}{D(w)} [Z\bar{x} - \frac{x_o}{1+z}] \qquad (13)$$

  is the discrete simulation of the analog response $\bar{y}$.
  This program introduces the replacement of s by w.

- The third step is to demand spectral equivalence! Success strongly depends upon a fuzzy definition of a low pass filter in the context of Butterworth filters with cut-off frequency ω_0. The conclusion is:

  If $\bar{y} = \frac{N(s)}{D(s)} \bar{x}$ is the response of an analog low pass filter with cut-off frequency ω_0 and if $s \to \omega_0 \tanh(\pi s/4\omega_0) \stackrel{\Delta}{=} \omega_0 w$
  then $Z\bar{y} = \frac{N(\omega_0 w)}{D(\omega_0 w)} Z\bar{x}$ is the spectrally equivalent digital response.

The sampling interval T of the sampler Z is constrained by

$$\omega_0 T = \pi/2. \qquad (14)$$

The pay-off is optimality [2]. The digital Butterworth filter is flatter than the analog filter in the in-band; the slope of the digital filter is greater than the slope of the analog filter at the cut-off frequency.

It is convenient to normalize the cut-off frequency by setting $\omega_0 = 1$. A list of normalized analog, digital, and spatial low pass Butterworth filters is

$\frac{1}{1+s}$	$\frac{1+z}{2}$	$\frac{1}{2}(I+H)$
$\frac{1}{1+\sqrt{2}s+s^2}$	$\frac{(1+z)^2}{(2+\sqrt{2})+(2-\sqrt{2})z^2}$	$((2+\sqrt{2})I+(2-\sqrt{2})H^2)^{-1}(I+H)^2$
$\frac{1}{1+2s+2s^2+s^3}$	$\frac{(1+z)^3}{6+2z^2}$	$(6I+2H^2)^{-1}(I+H)^3$.

An unexpected conclusion is that $1/1+s$ is spectrally equivalent to $(1+z)/2$ but <u>not</u> to $1/(1-ze^{-T})$.

A list of normalized high pass Butterworth filters is

$\frac{s}{1+s}$	$\frac{1-z}{2}$	$\frac{1}{2}(I-H)$
$\frac{s^2}{1+\sqrt{2}s+s^2}$	$\frac{(1-z)^2}{(2+\sqrt{2}) - (2-\sqrt{2})z^2}$	$((2+\sqrt{2})I+(2-\sqrt{2})H^2)^{-1}(I-H)^2$
$\frac{s^3}{1+2s+2s^2+s^3}$	$\frac{(1-z)^3}{6+2z^2}$	$(6I+2H^2)^{-1}(I-H)^3$.

The uncommon factors of each order filter in these two lists are (I+H), (I-H) and their powers which are also filters but require less computational effort than the complete Butterworth filter. For instance,

$$\frac{(I-H)^3}{2} = 1/8\ (I-3H+3H^2+H^3)$$

requires less effort than

$$(6I+2H^2)^{-1}\ (I-H)^3;$$

the former requires three subdiagonals whereas the latter requires (n-1) subdiagonals. One can verbalize these notions by calling $1/8(I-H)^3$ a <u>local</u> spatial filter and $(6I+2H^2)^{-1}\ (I-H)^3$ a <u>global</u> spatial filter. In the sequel, only local spatial filters will be considered.

EDGE DETECTORS

A normalized high pass filter, I-H, can be employed as an edge detector of distinct VH- squares with vertical and horizontal edges.

LEMMA 2: If g is a binary vector, then (I-H)g is a vector that displays either the start, or the start-and-end, or end of a burst of 1's. The start is indicated by +1 and the end is indicated by -1. The original vector g is recoverable with

$$(I-H)^{-1} = \sum_{k=o}^{n-1} H^k$$

where $H^o\underline{\Delta} = 1$. Moreover, $|(I-H)g|$ is binary and invertible by $(I+H)^{-1}$ in the modulo 2 arithmetic only in the case of a start-and-end and end result.

LEMMA 3: The number of binary n-vectors containing bursts of 1's is the modified Fibonacci number ϕ_n where $\{\phi_n\} = \{1,2,3,5,8,13,\ldots,\}$ and $n = 1,2,3,\ldots$

LEMMA 4: The probability that a binary n-vector contains bursts of 1's is $\phi_n/2^n$. Moreover

$$\frac{\phi_{n+1}}{2^{n+1}} < \frac{\phi_n}{2^n} \quad \text{and} \quad \lim_{n\to\infty} \frac{\phi_n}{2^n} = 0.$$

Specializations of the divergence and curl motivate the next result.

LEMMA 5: If C is a binary matrix that contains well-separated closed VH-square contours, and if

$$\nabla_+ C \underline{\underline{\Delta}} C(I-H)' + (I-H)C$$

$$\nabla_- C \underline{\underline{\Delta}} C(I+H)' - (I+H)C \tag{15}$$

then

$$|\nabla_+(\nabla_- C)| = G, \; \mathcal{D}(G) = 0 \tag{16}$$

where G is a binary matrix with VH- squares subject to the condition that $\mathcal{D}(G) = 0$. Here, $\mathcal{D}(G)$ is a matrix whose diagonal is the diagonal of G and whose off-diagonal terms are all zero. Under the specialized condition of Lemma 6, $|\nabla_+(\nabla_- C)|$ fills in the contours of C except for zero diagonal elements of the matrix.

The commuted situation requires binary thresholding associated with the absolute value which is symbolized and defined in the scalar case by

$$|f|_{(2)} \underline{\underline{\Delta}} \begin{cases} 0 \text{ if } f = 0 \\ 1 \text{ if } f \neq 0 \end{cases} . \tag{17}$$

The scalar case is directly extendable to each element of a matrix.

LEMMA 6: If G is a binary matrix that contains well-separated VH- squares, and ε is a sparse matrix with 1's located only at the bottom right corner of every square contour in G then

i) $\nabla_+ G$ detects almost-closed

ii) contours of the VH- squares

iii) $|\nabla_-(|\nabla_+ G|_{(2)} + \varepsilon)| = G - \mathcal{D}(G)$

iv) ∇_- is the reconstructor of the contour.

Almost-closed VH-square contours are closed contours with the bottom right corner element equal to zero.

A superinvariance property immediately follows; namely

$$G - \mathcal{D}(G) - \delta \leq |\nabla_+ |\nabla_- G|_{(2)}| \tag{18}$$

where δ is a sparse binary matrix such that

i) $\delta \underline{\Delta} |\nabla_- \varepsilon|$,

ii) The number of 1's in δ is almost twice the number of 1's in ε.

The next Lemma is motivated by the second partial derivative.

LEMMA 6: If G is a binary matrix with well-separated VH-squares, then

$$|(I-H)G(I-H')| = C$$

is a binary corner detector. The reconstructor is right transpose and left multiplications by $(I+H)^{-1}$ in the mod 2 arithmetic.

WEDGE DETECTOR

The previous section developed detection and reconstructions of images with jumps. It is natural to extend this study to images with slope jumps. The test image is a truncated pyramid whose middle horizontal, middle and vertical and two diagonal crossections have the form (0,1,2,4,5,5,5,5,5,4,3, 2,1,0). The sites of the slope jumps are wedges and three distinct wedges contours are:

i) A closed square contour due to the pyramid base;

ii) An inner closed square contour due to the pyramid top;

iii) The four sloping wedges which connect corners of the inner and outer closed square contours.

The combined appearance on an image is that of a closed contour formed from four open trapezoidal contours. Thus, contact is made with the closed contour detector, ∇_+ of the previous section.

Interaction of ∇_+ on a truncated pyramid image G yields

$$\nabla_+(\nabla_+ G) = \nabla_+^2 G + 2(I-H)^2 G(I-H')^2$$
$$\nabla_+^2 G \underline{\Delta} G(I-H')^2 + (I-H)^2 G \qquad (19)$$

and the latter is the Laplacian. Iteration of ∇_- on the closed contour of open trapezoidal contours, C, yields

$$\nabla_-(\nabla_- C) = \nabla_-^2 C - 2(I+H)^2 C(I+H')^2$$
$$\nabla_-^2 C \underline{\Delta} C(I+H')^2 - (I+H)^2 C \qquad (20)$$

and the latter is the complementary Laplacian.

Much complexity causes one to refrain from detailed statements about both the Laplacian and its complement. However, by itself, $|\nabla_+^2 G|_2$ is a

wedge detector wherein the sloping wedges are represented by sub- and super-diagonal block matrices. Of course, one can conjecture that the complementary Laplacian is a candidate reconstructor according to Lemma 6.

THE DC AND RAMP NOTCH FILTERS

The DC notch filter is a very low cut-off frequency high pass filter and is realized by $(1-z/2)^m$ where m is an integer, $z = \exp(-s\pi/2)$. The new cut-off frequency and an inequality are

$$\omega_{om} = \frac{2}{\pi} \cos^{-1}(-1+2^{1-(1/m)}), \tag{21}$$

$$0 < \omega_{om} \leq 1 \tag{22}$$

where

i) equality holds when $m = 1$

ii) $\lim_{m\to\infty} \omega_{om} = 0.$

The corresponding local spatial filter is the nxn matrix.

$$\frac{(I-H))^m}{2}, \quad m << n,$$

with the same cut-off frequency formula but with a different inequality,

$$\omega_{on} < \omega_{om} < 1. \tag{23}$$

The DC notch filter eliminates u, the signal's constant component. This task cannot be performed by $(I-H)^m$ because $(I-H)^m u$ is a vector with binomial coefficients followed by zeros. Small computational effort requires the form $(I-H)^{m-k}D^{(k)}$ where $k << (m-k) << n$. Least computational effort occurs for $k = 1$ and perhaps $k = 2$.

Operators D, DD and $D^{(2)}$ are doing the annihilation in Lemma 6 and 7. Therefore, gradient, divergence and second gradient, Laplacian, in the vector calculus can be given additional meaning as DC notch and ramp notch filters. This filter meaning induces two residue classes whose simplest elements are D, the gradient, and $D^{(2)}$, and the second gradient.

The spatial notch filters of Lemmas 7 and 8 are better performers than their z-transform precursor, $(1-z^m/2)$. This is because analog and digital filters are defined modulo their nonzero transient responses.

NOTCH FILTERS

This section develops the second derivative matrix and then proceeds to develop notch filters from D and $D^{(2)}$.

Test computations yield desirable and undesirable results such as:

i) $D^0 = 0$, $Du = 0$, $Dr = u$, $D^2r - D(Dr) = 0$;

ii) $D^2u - D(Du) = D^2u - Do = -2(\hat{e}_1 + \hat{e}_2) \neq 0$.

DD works but D^2 does not work on <u>both</u> u and r. However, repair is easily contrived by redefining D^2 as

$$D^{(2)} \triangleq \begin{pmatrix} 1 & -2 & 1 & 0 & 0 & 0 \\ 1 & -2 & 1 & 0 & 0 & 0 \\ 1 & -2 & 1 & 0 & 0 & 0 \\ 0 & 1 & -2 & 1 & 0 & 0 \\ 0 & 0 & 1 & -2 & 1 & 0 \\ 0 & 0 & 0 & 1 & -2 & 1 \end{pmatrix} \tag{24}$$

In general, $D^{(2)}$ has rank (n-2) and it annihilates the n-vectors u and r. This idea generalizes to $D^{(k)}$ $1 \leq k < n$ with higher order binomial coefficients and k + 1 top rows similar.

One can now construct four primitive u-notch and r-notch filters which eliminate u and r components from a signal vector.

LEMMA 7: The form $(I-H)^{m-1} D$, $z \leq m < n$, is a DC notch filter but not a ramp notch filter.

LEMMA 8: The forms $(I-H)^{m-2}D(I-H)$, $(I-H)^{m-2}DD$ and $(I-H)^{m-2}D^{(2)}$, $3 \leq m < n$, are ramp notch filters; moreover, the latter two forms are also DC notch filters.

LEMMA 9: All cut-off frequencies of the four notch filters are ω_{om}.

VECTOR SPACE ASPECTS OF NOTCH FILTERS

In this section, we consider only $D^{(k)}$ the nxn k-th derivative matrix, $1 \leq k < n$, rank $\ell \geq 1$, and typify it by D, all in a vector space [4] setting. Also needed is the observation that $D\hat{e}_k$ is the k-th column of D.

LEMMA 10: For any nxn singular matrix of type D and integer rank ($\ell = n-k$), there exists a <u>canonical elementary column transformation</u> (CECT) such that

$$\begin{bmatrix} I \\ \hline D \end{bmatrix} \xrightarrow{\text{CECT}} \left[\begin{array}{c|c} (K|\bar{K}) \\ \hline (O|Q) \end{array}\right], \quad (DK|D\bar{K}) = (O|Q) \tag{25}$$

where

$(O|Q)$ is an nxn upper triangular matrix,

$(K|\bar{K})$ consists of the null column vector set, K, and its complementary vector set $\bar{K}$.

Moreover, arbitrary non-zero linear combinations of the column of the null set, K, produce the null space of D.

The canonical elementary column transformation proceeds as follows:

i) Start with $D\hat{e}_n$, the last column of D;

ii) Perform k leftward ECT's on $\left(\frac{I}{D}\right)$ to zero all off-diagonal terms on the bottom row;

iii) Go to the column (n-1) diagonal element and use k leftward ECT's to zero all off-diagonal elements on the (n-1)st row.

iv) Continue while outside the first n-p columns;

v) Inside the region of the fist n-p columns use ECT's to zero all n-p columns.

This algorithm is effective because of the sparse upper triangular region of all matrices $D^{(k)}$ and the k subdiagonals in the lower triangular region.

The column vectors of the non-singular matrix $(K|\bar{K})$ can be used as a basis for any vector x in $V_n(I)$ and Dx belongs to the subspace $V_{n-p}(Q)$ and $Q = D\bar{K}$.

If one uses $D^{(2)}$ for example, then K = (u-r,r) issues forth. One finds u in the null space by means of the linear combination (u-r) + (r). Moreover, $\bar{K}$ can be found easily from, r; namely

$$\bar{K} = (Hr, H^2r, H^3r, \ldots, H^{n-p}r). \tag{26}$$

The general case requires the matrix B which contains the first k n-vectors of the Right Pascal Triangle for the binomial coefficients and b the last column vector of B. Each column of B represents truncated inverse binomial coefficients.

LEMMA 11: The null-space and complementary space of $D^{(k)}$ are given by the lower triangular form

$$(B|\bar{B}) = (B|Hb, H^2b, \ldots, H^{n-k}b). \tag{27}$$

This specific form of $(K|\bar{K})$ is extremely easy to compute after the canonical elementary column transformations decoded its form.

Because of the lower triangular form of $(B|\bar{B})$, one can observe that $\hat{e}_1, \hat{e}_2, \ldots, \hat{e}_k$ are included in the null space and are excluded from the complementary space. In addition, $(B|\bar{B})$ forms a natural basis for any x in $V_n(I)$ that leads to the next conclusion.

LEMMA 12: If x belongs to $V_n(I)$, then $D^{(k)}x$ is independent of the first k columns of $D^{(k)}$. Moreover, the fixed column vector of $D^{(k)}$ is $\hat{e}_n$; that is $D^{(k)}\hat{e}_n = \hat{e}_n$ for all k and n such that $1 \leq k \leq n-2$. This Lemma is a

play on the fact that $D^{(k)}x$ is a vector that belongs to the column space of $D^{(k)}$; but if $D^{(k)}$ is singular, then "$D^{(k)}x$ belongs to the complementary space" is brought back to $D^{(k)}$ by the independence conclusion.

If $k = n-2$ and x is any vector in $V_n(I)$ then $D^{(n-1)}x = a\hat{e}_n$. A drastic mutilation has occurred and one must insist on $k << n$. This demand is equivalent to insisting on relatively large rank short of full rank for $D^{(k)}$.

Our u-notch and r-notch filters depend on D and $D^{(2)}$ respectively. One way of lessening the independence phenomenon is to limit the context to low order derivative matrices. Everything has its limitations!

Lastly, one can loop back to Lemma 11 and obtain something reminiscent of real variable multiple derivatives.

LEMMA 13: There exist nonzero linear combinations on the null space B of $D^{(k)}$ such that

$$(K|\bar{K}) = (u,r,r^2,\ldots,r^{k-1}|\bar{B}). \tag{28}$$

It seems that the notion of power vector belongs to the boundary of vector space concepts.

PHYSICAL REALIZATIONS

Spatial low pass, high pass, DC notch and ramp notch filters, D and $D^{(2)}$, can be realized directly by software on a matrix-vector oriented digital computer.

Dedicated hardware realizations will be faster and proceed as follows. The low pass filter $(1+z)^m$ and high pass filter $(1-z)^m$ are realized with parallel full adders and parallel shift registers <u>without</u> initializations.

These devices are connected together in the feed-through manner to form a non-recursive filter and linked to each other through carry lines.

The derivative realization is almost similar to the previous low pass and high pass filter realizations except that

i) A single delay is needed for each parallel realization,

ii) Each delay flip-flop is initialized with bits from

$$[(g_1 - 2g_o)/T]_2.$$

Here, it is useful to denote $[\alpha]_2 + [\beta]_2 z$ by $[\alpha + \beta z]_2$.

There remains the problem of transcribing the matrix $D^{(2)}$ into the Z-transform.

If: $f(t) = \ddot{g}(t)$

then:

$$Z\bar{f} = \left(\frac{1-z}{T}\right)^2 Z\bar{g} + \left(\frac{g_2-2g_1}{T^2}\right) + \left(\frac{g_2-3g_1-3g_0}{T^2}\right) z. \qquad (29)$$

The two additive terms indicate the need for two initializations in the parallel shift register realizations of the second derivative, $D^{(2)}$.

If g is a single image column vector then the simultaneous DC notch and ramp notch filter operation yields

$$[I-H]^{m-2} D^{(2)} g.$$

This is realized with the aid of nexted round parenthesis convention by

$$([I-H]^{m-2} (D^{(2)} (g))) \rightarrow ([1-z]_2^{m-2} (Z\bar{f})) \qquad (30)$$

where

$$Z\bar{f} = \left[\frac{g_2-2g_1}{T^2}\right]_2 z^0 + \left[\frac{g_2-3g_1-3g_0}{T^2}\right]_2 z + \left[\left[\frac{1-z}{T}\right]_2^2 (Z\bar{g})_2\right] \qquad (31)$$

A straight parenthesis sequence follows the left-to-right order convention. The first two additive terms display the two initializations and their placements.

The same techniques apply to simpler DC notch filter $[I-H]^{m-1}D$ except that its Z-transform precursor is known; indeed the precursor derived D, the first derivative matrix.

CONCLUSIONS

The origin of the derivative matrix in the numerical transform and the origin of the spatial filter in the optimal Rader-Gold transform has been displayed. Applications to edge detector, reconstructors and DC notch filters are interesting consequences.

A vector space analysis of multiple derivative matrices displays their null spaces and their independence sets. Physical realization heavily depends on the originating numerical transform.

REFERENCES

1. C. A. Halijak, "A Numerical Transform with ε-quadrature." Computers and Electrical Engineering, Vol. 4, pp. 257-260; 1971.

2. C. A. Halijak, "From Switched Capacitors to Digital Filters." Proceedings of the First Rocket City Seminar, University of Alabama in Huntsville; pp. 1-9; March 1979.

3. J. R. Ragazzini, L. A. Zadeh, "Analysis of Sampled Data Systems." Trans. AIEE, Vol. 71, pp. 225-234; November 1952.

4. S. Perlis, "Theory of Matrices." Addison-Wesley Publishing Co., Cambridge, Mass.; 1951.

Paper No. IB-5, Presented at the Workshop on Imaging Trackers and Autonomous Acquisition Applications for Missile Guidance, 19-20 November 1979, Redstone Arsenal, Alabama.

IMAGE ENHANCEMENT FOR MAN-IN-THE-LOOP TARGET ACQUISITION SYSTEMS

Lewis J. Pinson
Associate Professor
University of Tennessee
Space Institute
Tullahoma, Tennessee 37388

James L. Baumann
Research Engineer
U.S. Army Missile Command
Redstone Arsenal, Alabama 35809

INTRODUCTION

A major operational need facing the next generation of Scout (ASH) and Attack (AAH) helicopters is to detect targets from nap-of-earth altitudes, on a realistic battlefield in a complex and cluttered scene. The current technology base supports the development of a complement of sensors spanning a region of the spectrum from visible to far infrared. Precision pointing and stabilization has been demonstrated to insure that target detection and recognition requirements can be met. However, the problem for the operator to detect low contrast targets in a complex and cluttered scene at long ranges and minimum exposure time to insure adequate survivability still exists. The targets of interest here are primarily the single, high threat target which will not be contained with the main body of target tanks and will not present many detection cues. This work is oriented primarily for TADS type acquisition systems for airborne missile fire control, although it has general application to any video format imaging system. It holds high potential for improving target signature for seeker lockon and tracking and can possibly simplify correlation techniques for missile seeker handoff by preprocessing the seeker and sensor imagery.

Image processing techniques offer potential for improved performance through target acquisition (autonomous or man-in-loop) at greater standoff ranges or in less time and through automatic handoff of identified targets from a precision pointing and tracking system to the missile seeker. Other facets of the fire-and-forget concept, tracking and homing, guidance, also have potential for improved performance through image processing techniques, are beyond the scope of this paper. Emphasis here is on man-in-loop target acquisition.

TARGET ACQUISITION

Performance of an observer in acquiring a target is usually given in terms of probabilities of specific subjective decisions by the observer. Target acquisition is taken to mean the detection and/or recognition of potential target classes by a human observer viewing a display of the imaged scene. Performance in terms of detection probability and recognition probability can be related to quantifiable system performance measures for the sensor system.

$$P_{D,R} = P_1 \cdot P_2 \cdot P_3 \cdot P_4$$

where

P_1 is a search-term probability

P_2 depends on contrast

P_3 depends on resolution

P_4 depends on noise.

Based on this model, image processing methods which improve contrast, resolution or signal-to-noise are candidates for application to the target acquisition problem.

Other parameters which are implicit in the model are listed below. All elements from the target/background through the optical path, sensor, and display to the observer will influence performance.

a) Target coordinates on the display.

b) Target angular sub-tense at the observer's eye.

c) Target-background contrast.

d) Displayed target and background radiance levels.

e) Resolution of the system.

f) Target dimensions on the display.

g) Two-dimensional noise at the display.

h) Photon noise at the display.

i) Video electronic noise.

j) Background structure.

k) Display dimensions.

l) Eye integration period (0.1-0.2 sec).

m) Eye fixation period ($\sim$0.3 sec).

n) Search time available to observer.

IMPROVEMENT METHODS

For viewing conditions which produce marginal contrast, resolution and/or signal-to-noise ratio, target acquisition performance is improved by enhancing these quantitative measures of image quality. Improvement in these measures can be accomplished to varying degrees in a number of ways including sensor/display system design optimization and video signal processing. Based on a study by Southern Research Institute, a number of options were identified, as shown in Figure 1, for potential image enhancement. Some of the methods in Figure 1 (e.g., items 1 through 6 and 10) are more accurately described as elements of good sensor design. Other methods such as 9, 11, 13 and 14 are more accurately described as video signal processing methods. The division between these two general methods is not fixed; however, some specific methods may be characterized either way.

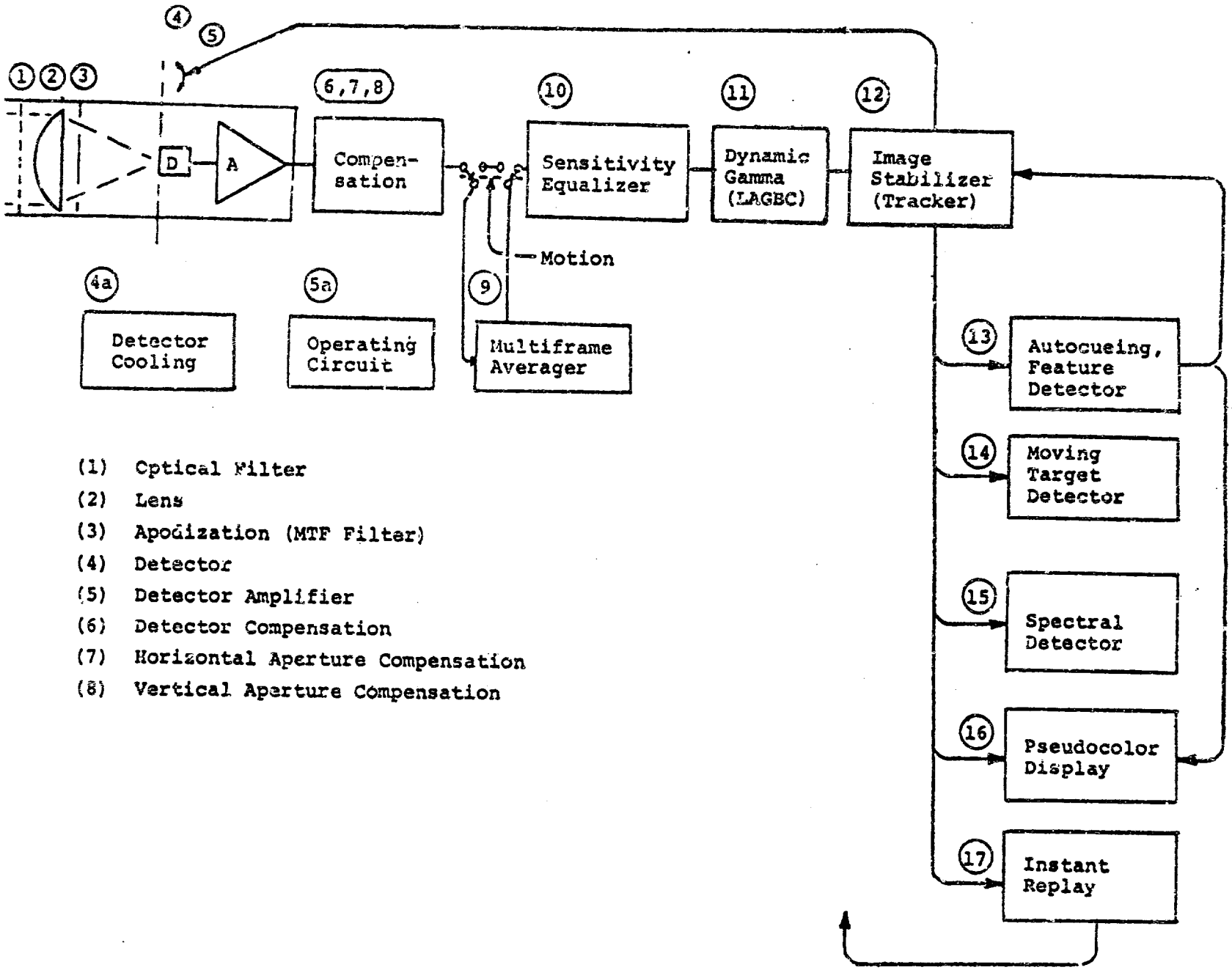
The primary conclusion of the Southern Research effort was that attention should first be paid to optimizing the operating characteristics of the imaging system components. Then specific video image processing methods should be investigated for further improvement in target acquisition capability.

IMAGE PROCESSING

Image processing consists of a number of inter-related disciplines and may be described in terms of four general categories as shown in Figure 2. The four categories, enhancement, restoration, registration, and pattern recognition, may be combined in an interactive fashion to best achieve the objectives for a given imaging problem. For example, image registration performance may be improved if the imagery is first restored, enhanced or represented by descriptive features which are part of a pattern recognition scheme.

The process of acquiring a target (detection and recognition) is considered to be the outcome of a pattern recognition process. The recognition may be subjective (by an operator viewing the image) or autonomous (wherein features of the image are compared with a library of features representing different target classes). In either case, the accuracy of the decision (target or non-target) can be improved by selected image processing methods.

It is desirable to have methods for quantitatively evaluating the expected improvement in performance effected by a given processing method. This quantitative comparison of various processing methods, in conjunction with practical constraints such as processing speed and complexity, form the basis for choosing algorithms to be implemented in hardware. This technique for processor selection is outlined in Figure 3. Methods for evaluation are given in Figure 4 and example evaluation measures are listed in Figure 5.



(1) Optical Filter
(2) Lens
(3) Apodization (MTF Filter)
(4) Detector
(5) Detector Amplifier
(6) Detector Compensation
(7) Horizontal Aperture Compensation
(8) Vertical Aperture Compensation

Figure 1. TV System with Image Enhancement Functions

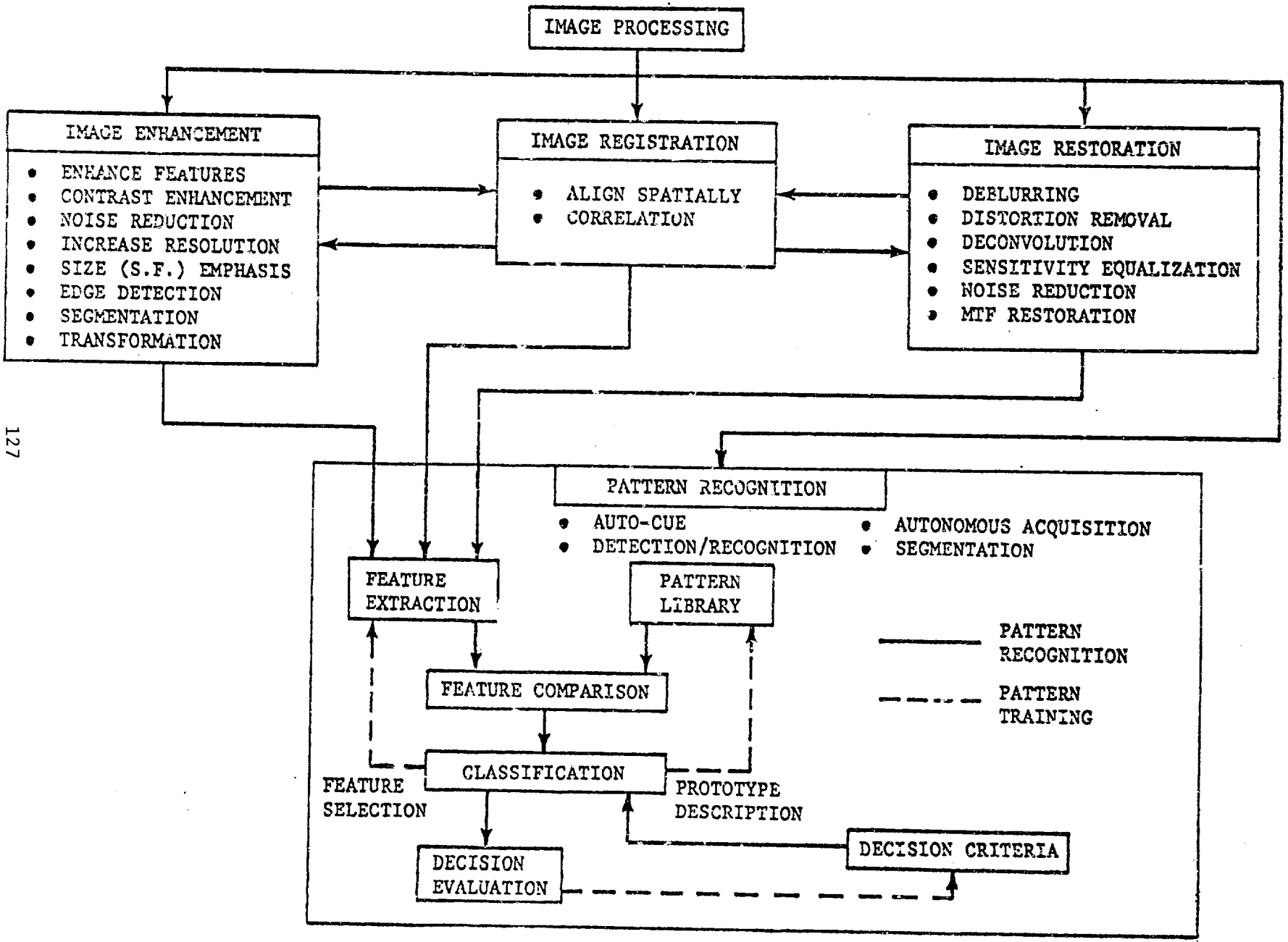
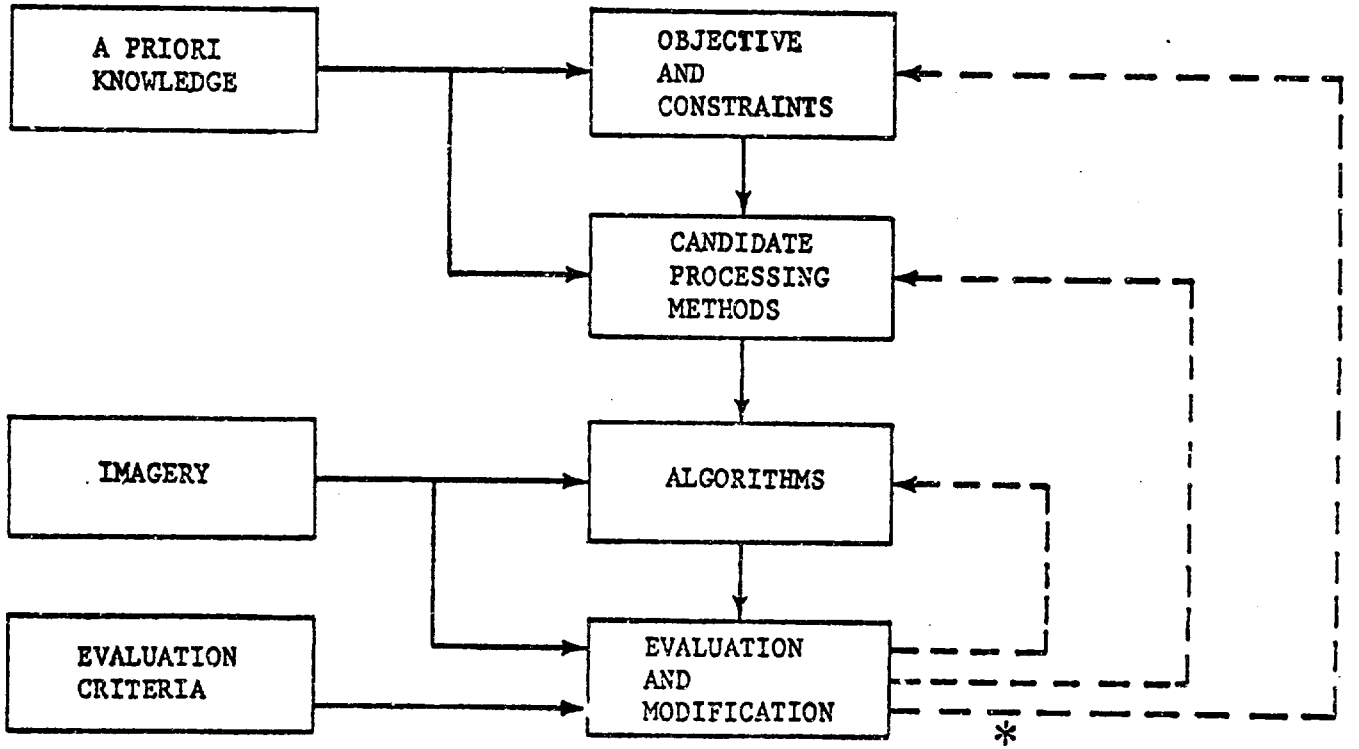


FIGURE 2. IMAGE PROCESSING INTERACTIONS



*MODIFICATION LOOP - QUESTION ON WHETHER IT CAN OR SHOULD BE REAL-TIME ADAPTIVE

FIGURE 3. TECHNIQUE FOR PROCESSOR SELECTION

PURPOSE - TO DETERMINE IF A PROCESSING METHOD IMPROVES THE ABILITY TO ACHIEVE STATED OBJECTIVES

IMPROVED ABILITY CAN BE BASED ON EVALUATION MEASURES

SUBJECTIVE: e.g. LESS FATIGUE IN TASK PERFORMANCE; A "BETTER" IMAGE

e.g. P_D, P_F, HIGHER CONTRAST, BETTER ACCURACY, ETC.

METHODS FOR EVALUATING PROCESSOR AND PARAMETER VARIATION

1. ANALYSIS - GENERAL STUDY OF PROPERTIES; MATHEMATICAL DEVELOPMENT; PARAMETER AND FUNCTION DEFINITIONS
2. SIMULATION WITH SIMPLE KNOWN IMAGES - ANALYTICAL OR COMPUTER ANALYSIS OF SPECIFIC, SIMPLE IMAGES.
3. EMPIRICAL APPLICATION TO REALISTIC IMAGES - COMPUTER ANALYSIS BASED ON APPLICATION TO TYPICAL IMAGERY.

FIGURE 4. PROCESSOR EVALUATION

SUBJECTIVE

- LESS FATIGUE
- BETTER IMAGE
- FASTER RESPONSE

PSYCHOVISUAL

- STATISTICAL TASK PERFORMANCE
- P_D
- P_F
- RESPONSE TIME

STATISTICAL/QUANTITATIVE

- SNR
- RESOLUTION
- CORRELATION ACCURACY
- P_D, P_F (AUTONOMOUS ACQ AND AUTO-CUE)
- SHAPE FEATURE STATISTICS
  - PERIMETER/$\sqrt{\text{AREA}}$
  - NUMBER OF EDGES
  - NORMALIZED EDGE LENGTH HISTOGRAM
  - SUCCESSIVE EDGE SLOPE DIFFERENTIAL HISTOGRAM
- INTENSITY AND CONTRAST MEASURES
  - INTENSITY HISTOGRAM (TARGET/BACKGROUND)
  - TARGET/BACKGROUND AVG. CONTRAST
  - TARGET/BACKGROUND PEAK CONTRAST
  - SOBEL GRADIENT EDGE HISTOGRAM
  - GRADIENT HISTOGRAM ACROSS EDGES
- TEXTURE FEATURES
  - DIRECTIONAL GRAY LEVEL DIFFERENCE HISTOGRAMS (VARIOUS SPACINGS)
  - THRESHOLDED-INTENSITY AREA HISTOGRAMS

NOTE: HISTOGRAMS MAY BE CHARACTERIZED BY MOMENTS (MEAN, STANDARD DEVIATION, SKEW, EXCESS)

FIGURE 5. EVALUATION MEASURES

The processing methods to be applied to the imagery can be global (applied equally across the image) or local-area-adaptive. They can be linear or non-linear and can be applied in either the spatial domain or spatial frequency domain. Specific processing methods which offer potential for improved performance include the following:

1. 3 x 3 Moving Window Average (or n x n)

   - Contrast enhancement
   - Edge enhancement
   - Easily implemented in hardware
   - Digital or analog implementation
   - Inexpensive add-on

2. Edge Detection/Enhancement

   - Potential for reduced scene information
   - Options on degree of edge emphasis
   - Easily implemented in hardware
   - Digital or analog implementation
   - Inexpensive add-on

3. Local Area Gain and Brightness Control

   - Locally dependent contrast enhancement
   - Simultaneous contrast enhancement and dynamic range suppression
   - Adaptive to scene statistics
   - Digital recursive and non-recursive implementations
   - Real time operation

4. Histogram Equalization

   - Non-linear gray scale transformation
   - Maximizes entropy of total image
   - May obscure target details for some scenes
   - Not recommended for FLIR images, but untested on TV images
   - 1 frame processing lag

5. Histogram Specification

   - Allows gray scale optimization for human vision (hyperbolization)
   - May obscure target details for some scenes
   - Must develop criteria for specifying a given histogram shape
   - 1 frame processing lag

6. Pixel Sensitivity Equalization

- More of a problem for discrete detector arrays than for e-scan photosurfaces (such as Vidicon)
- Requires uniform reference across array
- Should be accomplished internal to the sensor (sensor design instead of signal processing)

7. Multi-Frame Averaging

- Improves S/N (by $\sqrt{N}$, N frames)
- Requires time proportional to number of frames to be added
- Requires that frames be registered (generally are not registered due to sensor motion)
- Requires full frame storage

8. Median Filtering (non-linear)

- Improves S/N (n x n window)
- Single frame processing
- Does not degrade edges
- Requires sorting of n^2 pixels
- Implementable as separate 1-D, n-element filters with slightly degraded performance

9. Hysteresis Filter

- Single frame image smoothing
- Adjustable amplitude dead band applied to pixel values
- Potential loss of target detail

10. Scene Adaptive Low Pass Filter

- Linear, adaptive processor
- Filter pass-band adaptive to gradient or curvature within a 7 x 7 (or n x n) window
- Non-recursive implementation can require up to 10 x 10 filter size
- Recursive implementation possible in real-time

11. Inverse Filtering (Wiener)

- Restores resolution
- Requires model for MTF degradation
- Typically aimed at restoring e-beam and atmospheric MTF degradations

12. Super-Resolution

- Requires sampling at greater than Nyquist rate
- Increases computational burden because of more pixels
- Usual purpose is to enlarge a selected sub-area of the image (can thus keep number of pixels within limits)

PRESELECTION AND FURTHER EVALUATION

Based on 1) a thorough search of the literature for descriptions and analyses of candidate processing methods, 2) compatibility with objectives and constraints of the intended application, and 3) potential for modification, adaptive flexibility, and new device availability several methods were identified for further analysis and detailed evaluation.

For the objective of improving contrast and/or resolution and/or signal-to-noise ratio, the following six processing methods were identified for further analysis. Based on known characteristics, these processing methods offer varying degrees of simplicity, flexibility and expected improvement.

1. Convolutional Window (enhancement, restoration)
2. Edge Detection (enhancement)
3. LAGBC (adaptive enhancement, restoration)
4. Scene Adaptive LPF (noise reduction, smoothing)
5. Median Filter (noise reduction)
6. Histogram Modification (enhancement)

For the above processing methods, there are options on parameter values and methods for implementation. A thorough evaluation is required in terms of the target acquisition objective to form the basis for selection of specific methods for hardware implementation. This required evaluation has the following elements:

1. Parametric Analyses - Relate parameter values to specific quantitative and subjective evaluation measures.

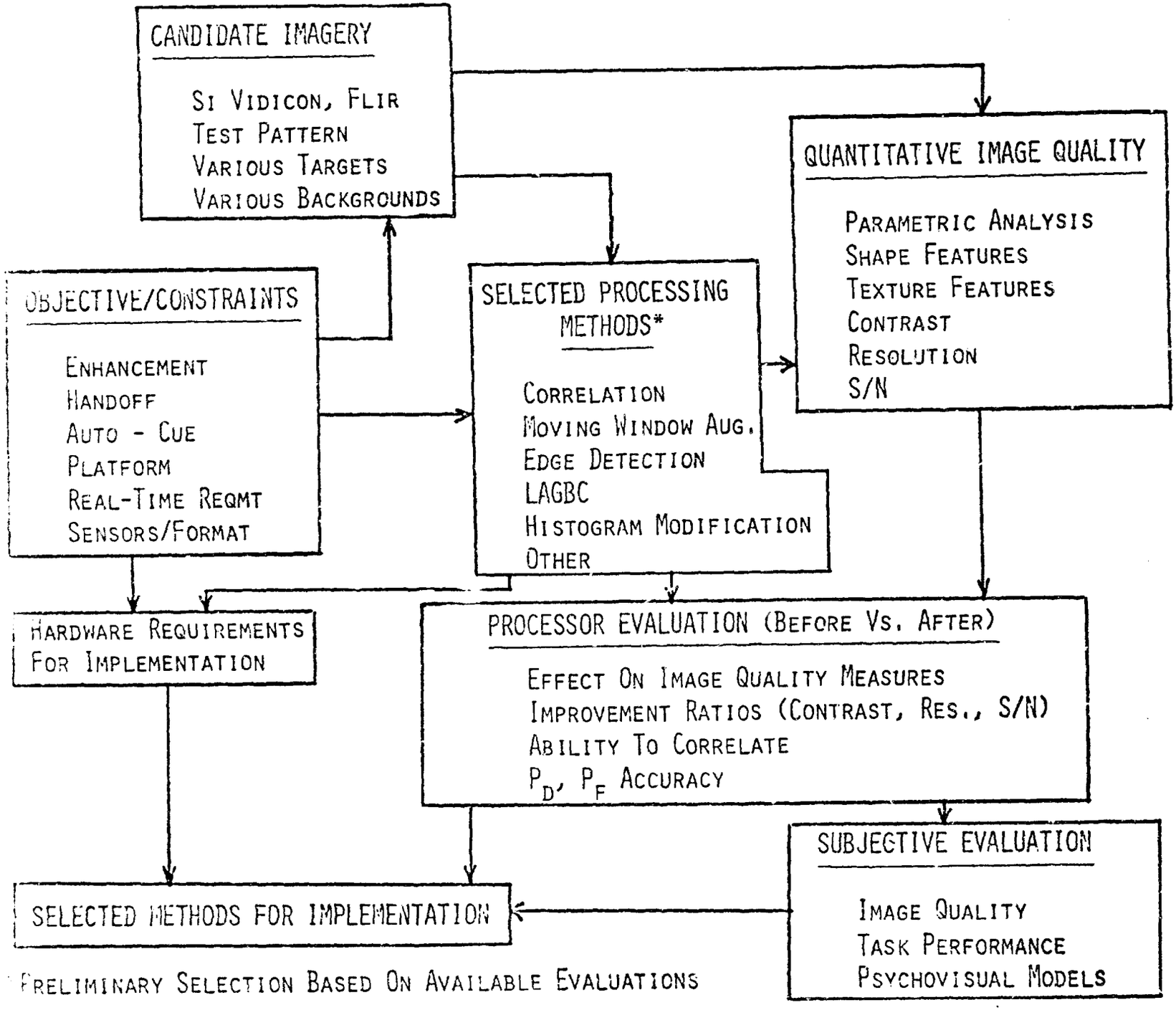
2. Multiple Processor Interactions - Algorithms which improve one performance measure (e.g., contrast, resolution, S/N) often do so at the expense of the others. Relative performance of serial and parallel implementations of multiple processors is required.

3. Adaptive Methods - Look for ways to make processors adaptive to scene content to optimize performance.

4. Verify/Validate - Candidate processing algorithms must be applied to typical TADS type imagery and evaluated by computer simulation.

Further detail in the processor selection/validation effort is given in Figure 6.

FIGURE 6. PROCESSOR SELECTION/VALIDATION STEPS



*PRELIMINARY SELECTION BASED ON AVAILABLE EVALUATIONS

CONCLUSION

There is evidence to support the contention that relatively simple image processing methods offer improved performance for TADS type target acquisition systems. This improved performance is related through psychovisual experimentation to quantifiable image quality measures. Thus, evaluation of a given image processing method can be in terms of quantifiable image improvement and in terms of subjective image quality.

From a list of potential processing methods, six have been identified as offering high potential for this application. Methods for further evaluation are given.

Paper No. IB-6, Presented at the Workshop on Imaging Trackers and Autonomous Acquisition Applications for Missile Guidance, 19-20 November 1979, Redstone Arsenal, Alabama.

Two-Dimensional Convolute Integers for Optical Image Data Processing

By

Thomas R. Edwards
Marshall Space Flight Center
Huntsville, AL 35812

ABSTRACT

Regression-generated Two-Dimensional Convolute Integers for optical image digital data processing present truly two-dimensional low pass, high pass, and band pass filtering with zero phase shifting and false magnification. As image enhancement this results in noise suppression, background subtraction, contour or edge sharpening, with minimal loss of resolution over the physical optics. Topographical directionality is available through generation of a normal image, i.e., an orthogonal surface. Physical optics resolution can be enhanced by false magnification. The logic, applied in a weighted, nearest-neighbor, nonrecursive, moving, smoothing, averaging type algorithm is fast and readily implemented in hardware. The entire package can reside immediately behind the physical optics and function as an image logic preprocessor.

INTRODUCTION

Regression-generated convolute integers for non-phase shifting, nearest-neighbor, weighted, moving smoothing, averaging type digital filters are well- stablished techniques found ubiquitously in one dimensional applied spectroscopy[1-5]. Two earlier references to these techniques can be found for the two-dimensional case[6,7]. But in reviewing the cornucopia of existing two dimensional filtering techniques[8], these rather powerful procedures, so readily hardware implementable, are conspiciously missing. Two Dimensional Convolute Integers can perform the following functions when convoluted with the data in an image:

1. Low pass filtering
2. High pass filtering
3. Band pass filtereing
4. Normal surface generating
5. False magnification or re-registration
6. Nonlinear magnification
7. Edge or contour enhancement
8. Noise filtering
9. One-pass multiple convolution

All these tasks can be accomplished in video time frame real-time hardware[10].

THEORY

Regression theory is at least a century old; therefore, there is nothing new about the calculations required to generate the Two Dimensional Convolute Integers. The only theoretical requirement of the data is equal interval spacing; the displacements between pixel elements in the x-direction must all be equal and either equal to or a multiple factor of the pixel displacements in the y-direction.

Four equivalent concepts must be simultaneously considered when developing these coefficients, Figure 1. Nonrecursive, nearest-neighbor, weighted, moving, smoothing average is equivalent to the convolution or folding together of a local region in an image with a weighting function and then moving on to the adjacent region. But these two ideas are equivalent to surface fitting of a local region, replacing a pixel element with one calculated by fitting the local region with a surface of some order, and again repeating the operation on the adjacent region. All these operations result in filtering and, in fact, possess all the normally desired filter characteristics in the fastest type of software algorithm or rather inexpensive, easy-to-build hardware.

In sampled data theory, convolution coefficients are equivalent to the weighting coefficients used to obtain a nearest-neighbor average, Figure 2. Merely describing the convoluting function in digital sampled data form leads to the statement that the weighting coefficients in a nearest-neighbor average are convolution coefficients. That these convoluting weighting coefficients in a nearest-neighbor average are regression coefficients also may be a bit more difficult to see but nonetheless is a very straightforward result.

Regression calculations or least squares analysis is in no way affected by the fact that a curvilinear polynomial is to be fitted to a one-dimensional data stream or an arbitrary surface is to be fitted to a matrix of data points in two dimensions, Figure 3.

Viewing the steps in Figure 3 leads to nothing unusual up to the normal equations associated with regression calculations. At this point, most investigators have failed to make the association between the weighting coefficients of a nearest-neighbor average and the regression coefficients resulting from surface fitting, Figure 4. In matrix representation, this association becomes clearer. Recognize that at the center of the data mask, position 0,0, each regression coefficient represents not only the value of the partial derivative but also an intensity value calculated from the data. View the matrix expression for a nearest-neighbor weighted average along with the matrix expression for the regression coefficients. Consider the individual scalar regression coefficients evaluated at the center of the data mask and equate them to a scalar intensity value. This new scalar intensity value can in turn be represented by a set of weighting coefficients and a normalizer, as in nearest-neighbor weighted averaging. But now, these newly defined weighting coefficients and normalizer are seen to be universal sets of numbers, independent of the data, dependent only on the surface order and the data mask size. These new weighting coefficients and their associated normalizer are convolution coefficients derived from two-dimensional regression calculations and can be appropriately described as regression-generated Two-Dimensional Convolute Integers. The integer aspect of their description arises from the fact that only integer values are used in their calculation.

TYPICAL FILTERS

A typical filter mask is seen in Figure 5. Note the great deal of symmetry associated with the coefficients in the filter mask. Only one quadrant of coefficients is needed to uniquely specify a complete set of coefficients. This lends speed to the weighted moving smoothing algorithm need to address the image data. Data mask locations having Two Dimensional Convolute Integers of equal value need only be added or subtracted prior to multiplication. Since addition of two integers is significantly faster than multiplication,

considerable processing time is saved by utilizing all the symmetry properties available. Viewing only the upper left-hand quadrant, allows the Two-Dimensional Convolute Integers to be expressed in a compact form, Figure 6.

FILTERS

The concept of filtering, hitherto addressed but not fully expressed, is rather simply stated for weighted, nearest-neighbor type averaging. A low pass filter should pass a constant intensity value. A high pass or band pass filter should not pass the constant value, Figure 7. A low pass filter is a noise suppressor or smoothing filter; whereas a high pass is a roughing filter. By applying Cramer's rule for the calculation of the individual regression coefficients, these filtering properties are readily satisfied.

TEST CASE

A very simple test case helps to clarify these concepts, Figure 8. Consider an arbitrary surface as represented in the figure. Calculate the intensity at each point in a 5 x 5 pixel data mask. If the data are noise-free, then fitting a surface to the data and calculating the intensity value at the center of the data mask by least squares should yield an intensity value of 10, the center point value. Now apply a two-dimensional regression to these intensity values by the nearest-neighbor weighted averaging using the regression-generated, Two-Dimensional Convolute Integer coefficients for a 5 x 5 filter mask, second or third order surface, smoothing filter. The coefficients are those seen in Figure 5. The products of the filtering coefficients and the intensity values when summed and divided by the normalizer are indeed just 10, the intensity at the center point of the data mask.

Smoothing these noise-free data merely regenerates the data but indicates that applying these filter coefficients is equivalent to a two-dimensional regression calculation.

RESULTS

An important aspect of meteorology is the ability to track clouds. Whereas cloud images in the computer are difficult to track, Figure 9, cloud contours are less difficult. Generating cloud contours by these filters is relatively straightforward . A normal surface of the original image is generated. A normal surface is by definition a surface, every point of which is the magnitude of the gradient evaluated at that point. For Two-Dimensional Convolute Integers this represents fitting a surface to a local region, calculating the partial derivative in both the x and y directions evaluated at the center of the region, and then obtaining the magnitude of the gradient. Now the gradient represents the greatest rate of change within a region and is therefore a very high pass filter. The gradient of the cloud data, enhancing contours, is seen in Figure 10. As a contour or

edge enhancement technique, generating the normal surface in a truly two-dimensional sense allows for excellent feature selection. But as almost all image analysis investigators recognize, generating a derivative also generates excessive noise and tends to degrade an image. However, Two-Dimensional Convolute Integers allow for multiple convolution in a single pass of the algorithm, i.e., two filter functions applied simultaneously. Thus, combine the gradient filter with a smoothing filter to suppress noise. This is band pass filtering via regression. A breast X-ray is seen in Figure 11, and the ability to view a cancer tumor, denoted by arrow, is enhanced by the band passed, gradient plus a smooth image seen in Figure 12. The ability to detect the tumor is definitely enhanced by this technique.

HARDWARE

A patent disclosure has been filed which represents a hardware design for a general purpose Two-Dimensional Convoler, Figure 13. The design is straightforward and can clock a filtered data point every 70 nanoseconds using existing IC chips. This rate approximates video rates of 60 frames per second with a raster 512 points square. All that's involved are shift registers in a delay chain scheme, adders and multipliers, as appropriate to a moving, smoothing, nearest-neighbor weighted averaging scheme. The Two-Dimensional Convolute Integer coefficients are loaded according to what output is desired, i.e., a noise-filtered contour-enhanced target, or an enhanced weld failure displaying the fault. This type hardware box can represent a video preprocessor residing immediately behind the imaging optics, performing a whole host of functions. Needless to say, the cost for parts in such a video preprocessor is rather inexpensive and the design rather straightforward.

CONCLUSION

In conclusion, let me address yet another aspect of these ideas which also resides within the domain of Two-Dimensional Convolute Integers and was mentioned previously.

The data masks considered so far have all been odd numbered in size, i.e., 3 x 3 or 7 x 7. The filter point, the new calculated value, has been a replacement value for the point at the center of the data mask.

Now consider an even numbered data mask, 4 x 4 or a 6 x 6. The calculated value, the new intensity, is again located at position 0,0; but this position being the center of the data mask, is interstitial--no point initially resides there. When the filter is moved along in its moving smoothing fashion, an interstitial line is generated. The intensity values on this line are excellent in that they are good fitted, weighted, nearest-neighbor averages. When the filter is passed over the image, every other line is an interstitial line hitherto not present; the data set has doubled. The number of line pairs

per millimeter has doubled. Thus, the resolution of the physical optics has been enhanced, or via this method of false magnification an image may be enlarged or magnified by a factor of two without a significant loss of information and in real-time hardware on a video screen.[11] A patent disclosure has been filed which represents a hardware design for a Two-Dimensional Convolute Integer Interstitial Point Generator, Figure 14.

ACKNOWLEDGMENT

The author wishes to express grateful appreciation to Dr. Robert Jayroe, Data Systems Laboratory MSFC, for generating the results shown in Figures 9, 10, 11 and 12.

REFERENCES

1. Savitzky, A. and Golay, M.J.E; Anal Chem 3b, 1964, p. 1627.

2. Steiner, J et al.; Anal Chem 44, 1972, p. 1906.

3. Enke, C. G. and Nieman, T. A.; Anal Chem 48, 1976, p. 705A.

4. Wood, L.D. and Hockens, S.N.; Geophysics 6, 1970, p. 1005.

5. Edwards, T. R. and Knight, R. D.; Instrument and Control Systems, September 1974, p. 73.

6. Jansson, P. A.; Journal Opt. Soc. Am., February 1972, p. 195.

7. Edwards, T. R. and Knight, R. D.; NASA TMX 64949, May 1975.

8. Rosenfeld, A. and Kak, C. A.; Digital Picture Processing: Academic Press, 1976, Chapters 6 and 7.

9. Termonia, Y. and Deltour, J; private communication.

10. Edwards, T. R.; Invention Disclosure MFS # 25294, Two-Dimensional Convolute Integer Filter - A General Purpose Hardware Convoler; August 1979.

11. Edwards, T.R.; Invention Disclosure MFS # 25295, An Interstitial Point Generator - Two-Dimensional; August 1979.

LIST OF FIGURES

TWO DIMENSIONAL CONVOLUTE INTEGERS
FOUR EQUIVALENT CONCEPTS

NON–RECURSIVE, NEAREST – NEIGHBOR, WEIGHTED, MOVING SMOOTHING AVERAGE

$$z'_{rc}(x,y) = \sum_{i=-m}^{m} \sum_{j=-n}^{n} C_{ij}\ z_{ij}\ (x,y) / \text{Norm} \quad ,$$

CONVOLUTION – FOLDING TOGETHER A REGION

$$z'_{rc}(x,y) = \int \Omega(\alpha,\beta)\, z(x-\alpha, y-\beta)\, d\alpha\, d\beta \quad .$$

SURFACE FITTING – REGRESSION CALCULATIONS.

$$z'_{rc} = \sum_{i=0}^{m} \sum_{j=0}^{n} A_{ij}\ x_r^i\ y_c^j \quad .$$

FILTERING IN TWO DIMENSIONS

FIGURE 1

THE WEIGHTING COEFFICIENTS C_{ij}'s ARE CONVOLUTION COEFFICIENTS

$$z'_{r,c}\ (x,y) = \int \Omega\ (\alpha,\beta)\, z\,(x-\alpha, y-\beta)\, d\alpha\, d\beta\ .$$

A FILTER CAN BE CONSIDERED AN OPERATOR WHICH FORMS THE FILTERED DATA BY INTEGRATING THE RAW DATA OVER A WEIGHTING FUNCTION $\Omega(\alpha,\beta)$:

THIS INTEGRAL IS DEFINED AS THE CONVOLUTION OF $z\,(x-\alpha, y-\beta)$ WITH $\Omega(\alpha,\beta)$. IN A DIGITAL FILTER, THE WEIGHTING FUNCTION IS OF THE FORM

$$\Omega(\alpha,\beta) = \sum_{i=-m}^{m} \sum_{j=-n}^{n} C_{ij}\, \delta\ (\alpha+i, \beta+j) / \text{NORM}\ ,$$

WHERE δ IS THE DIRAC DELTA FUNCTION REPRESENTING THE DISCRETE SAMPLING OF THE DATA

NOW SUBSTITUTING THIS DISCRETE FUNCTION AND INTEGRATING OVER THE DELTA FUNCTION YEILDS A NEAREST NEIGHBOR WEIGHTED AVERAGE

$$z'_{rc}\ (x,y) = \sum_{i=-m}^{m} \sum_{j=-n}^{n} C_{ij}\, z\,(x+i, y+j) / \text{NORM}\ .$$

FIGURE 2

THE WEIGHTING COEFFICIENTS C_{ij}'s ARE REGRESSION COEFFICIENTS

$$Z'_{r,c}(X,Y) = \sum_{i=0}^{m} \sum_{j=0}^{n} A_{ij}\ x_r^i\ y_c^j .$$

THE INTENSITY $Z'_{r,c}(X,Y)$, AT ROW – r, COLUMN – c ON THE SURFACE OF ORDER m + n IS DESCRIBED BY THE REGRESSION COEFFICIENTS A_{ij}.

LET z_k REPRESENT THE ACTUAL INTENSITY AT THE kTH POSITION ON AN IMAGE, AND FIT THE SURFACE TO THE DATA SET $z_k(x,y)$ BY THE RESIDUAL SUM OF SQUARES,

$$\delta 2 = \sum_{k=1}^{np} \left(z_k - \sum_{i=0}^{m} \sum_{j=0}^{n} A_{ij}\ x_k^i\ y_k^j \right)^2 .$$

SELECT THE BEST FIT REGRESSION COEFFICIENTS BY SETTING

$$\partial \delta 2 / \partial A_{uv} = 0$$

AND REPRESENT THE RESULTING NORMAL EQUATIONS IN MATRIX FORM

$$\underline{\overline{XY}}^T \cdot \underline{\overline{XY}} \cdot A = \underline{\overline{XY}}^T \cdot Z ,$$

NEAREST NEIGHBOR WEIGHTED AVERAGING IN MATRIX FASHION IS

$$Z'_{r,c}(X,Y) = C \cdot Z/NORM$$ – A SCALAR EXPRESSION

AND THE REGRESSION COEFFICIENTS ARE

$$A = \underline{\overline{XY}}^T \cdot Z \ \left[\underline{\overline{XY}}^T \cdot \underline{\overline{XY}} \right]^{-1}$$ – A VECTOR EXPRESSION

NOTE THE SIMILARITY AND RECOGNIZE THAT AT THE CENTER OF THE DATA MASK, LOCATION 0,0, THE REGRESSION COEFFICIENTS REPRESENT AN INTENSITY VALVE AND ALL PARTIAL DERIVATIVES ASSOCIATED WITH THE SURFACE ORDER.

THUS EQUATE

$$A_{ij} \mid_{0,0} = Z'_{0,0}(X,Y)$$ – THE MAIN THEORETICAL CONSIDERATION

FIGURE 3

TWO-DIMENSIONAL REGRESSION

MATRIX REPRESENTATION OF THE NORMAL EQUATIONS

$$\overline{\underline{XY}}^{T} \cdot \overline{\underline{XY}} \cdot A = \overline{\underline{XY}}^{T} \cdot Z \ ,$$

MATRIX REPRESENTATION OF THE NEAREST NEIGHBOR WEIGHTED AVERAGE

$$Z'_{rc} = C \cdot Z/Norm$$

MATRIX REPRESENTATION OF THE REGRESSION COEFFICIENTS

$$A = \left(\overline{\underline{XY}}^{T} \cdot \overline{\underline{XY}} \right)^{-1} \cdot \overline{\underline{XY}}^{T} \cdot Z \ ,$$

AT THE CENTER OF THE DATA MASK, LOCATION 0,0; EACH REGRESSION COEFFICIENT REPRESENTS A FILTERED INTENSITY VALUE

$$A_{ij} \mid_{o,o} = Z'_{0,0} \ (X,Y)$$

$$A_{ij} \mid_{o,o} = C \cdot Z/Norm$$

EACH REGRESSION COEFFICIENT, EVALUATED AT THE CENTER OF THE DATA MASK, CAN BE REPRESENTED BY A SET OF COEFFICIENTS, C AND NORM, INDEPENDENT OF THE DATA SET DEPENDENT ONLY ON THE SURFACE ORDER AND MASK SIZE (NUMBER OF DATA POINTS)

MAKING THE APPROPRIATE SUBSTITUTIONS LEADS TO

$$C \cdot Z/Norm = \left[\overline{\underline{XY}}^{T} \cdot Z \ [\overline{\underline{XY}}^{T} \cdot \overline{\underline{XY}}]^{-1} \right]_{ij}$$

WHERE

$$C = \left[\overline{\underline{XY}} \right]_{ij}$$

$$1/Norm = \left[\overline{\underline{XY}}^{T} \cdot \overline{\underline{XY}} \right]^{-1}$$

(CARE MUST BE EXERCISED TO AVOID MIXING SCALAR AND VECTOR EXPRESSIONS INCORRECTLY)

FIGURE 4

TWO DIMENSIONAL CONVOLUTE INTEGERS

5 X 5 FILTER MASK
2ND & 3RD ORDER SURFACE
A(00)
SMOOTHING FILTER

r \ c	−2	−1	0	1	2
−2	−13	2	7	2	−13
−1	2	17	22	17	2
0	7	22	27	22	7
1	2	17	22	17	2
2	−13	2	7	2	−13

NORMALIZER 175

FIGURE 5

TWO DIMENSIONAL CONVOLUTE INTEGERS
7 X 7 FILTER MASK

SURFACE ORDER	4 & 5	5 & 6	4 & 5	5 & 6
	SMOOTH	PARTIAL	PARTIAL	PARTIAL
r / c	A (00)	A (01) A (10)*	A (02) A (20)*	A (03) A (30)*
−3 −3	206	−1484	−823	29
−3 −2	−174	3176	1407	−92
−3 −1	−24	−2350	− 69	97
−3 0	89	0	−1030	0
−2 −3	−174	1116	−273	9
−2 −2	−279	1701	1407	−72
−2 −1	36	−4050	−389	117
−2 0	450	0	−1407	0
−1 −3	651	246	57	−3
−1 −2	89	−804	1407	−60
−1 −1	204	−5880	−597	129
−1 0	651	0	−1734	0
0 −3	89	−449	167	−7
0 −2	204	1909	1407	−56
0 −1	651	−6625	−663	133
0 0	863	0	−1822	0
NORM	4851	41580	38808	3024

*INDICATES TRANSPOSE – INTERCHANGE r/c

FIGURE 6

FILTERING PROPERTIES

LOW PASS FILTER

$$\sum_{K=1}^{N_p} C_K / \text{NORM} = 1$$

ALL C_K'S AND NORM ASSOCIATED WITH THE REGRESSION COEFFICIENT $A_{O,O}$ SATISFY THIS

HIGH PASS FILTER

$$\sum_{K=1}^{N_p} C_K / \text{NORM} = 0$$

ALL C_K'S AND NORM ASSOCIATED WITH THE REGRESSION COEFFICIENT $A_{I,J}$ $I, J \neq O$ SATISFY THIS.

FIGURE 7

TWO DIMENSIONAL CONVOLUTE INTEGERS
3RD ORDER SURFACE TEST CASE

$$Z(X,Y) = 10 + 5X + 6Y + 12XY + 2X^2Y + 21XY^2 + 54X^3 + Y^3$$

$$Z(0,0) = 10$$

A SMOOTHING FILTER APPLIED TO NOISE FREE DATA SHOULD GENERATE AN INTENSITY VALVE AT THE CENTER OF THE DATA MASK.

SURFACE INTENSITY

r \ c	−2	−1	0	1	2
−2	−588	−133	−10	105	536
−1	−465	−67	3	69	455
0	−432	−49	10	69	452
1	−483	−73	17	111	533
2	−612	−133	30	201	704

$C_{ij} \cdot Z_{ij}$ PRODUCTS

r \ c	−2	−1	0	1	2
−2	7644	−266	70	210	−6968
−1	−930	−1139	66	1173	910
0	−3024	−1078	270	1518	3164
1	−966	−1241	374	1887	1066
2	7956	−266	210	402	−9152

SMOOTHING FILTER A (00)
2ND AND 3RD ORDER SURFACE

COLUMN SUMS	10680	−3990	850	5190	−10980

TOTAL SUM ÷ NORMALIZER = 1750/175 = 10

FIGURE 8

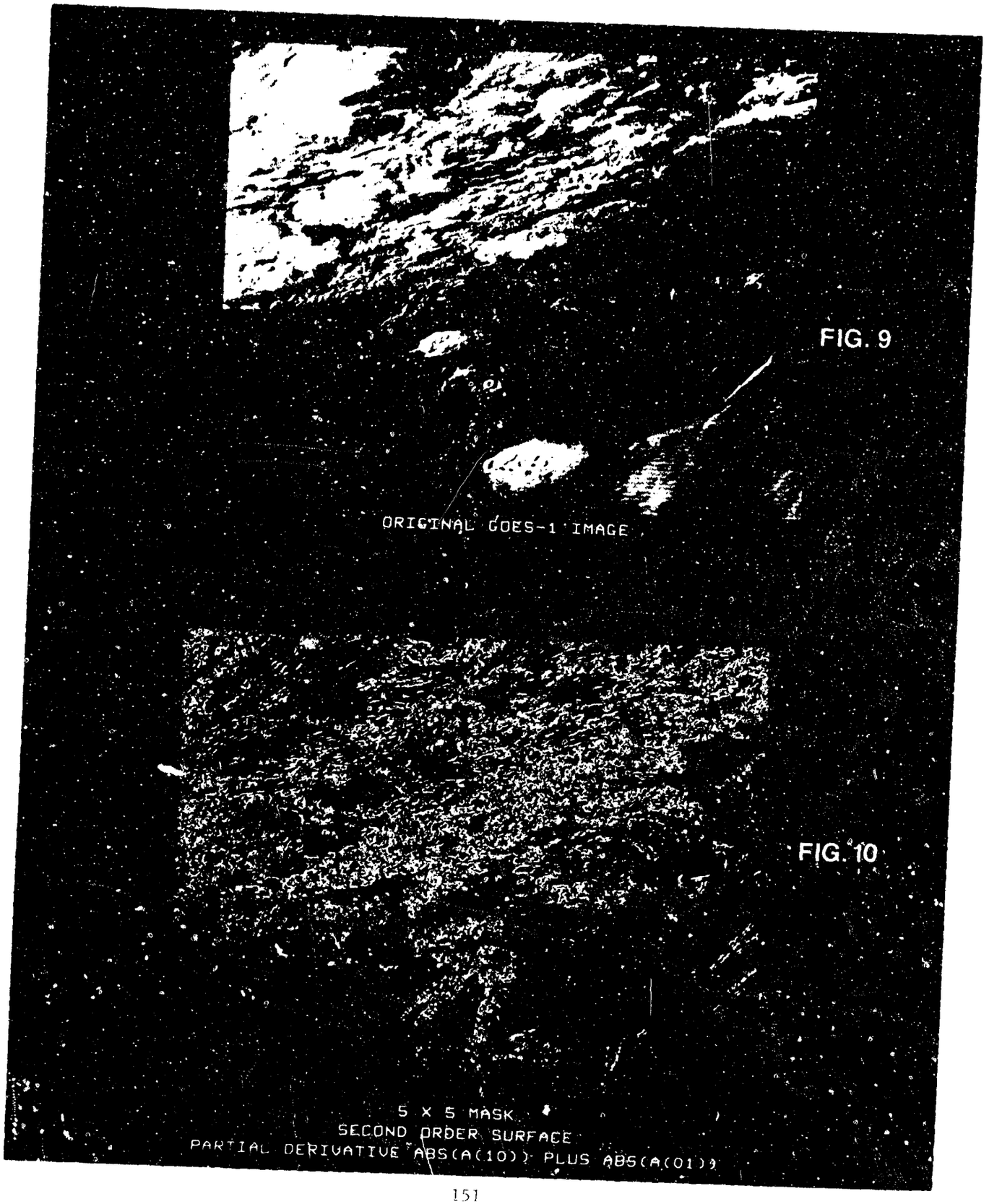

ORIGINAL GOES-1 IMAGE
5 X 5 MASK
SECOND ORDER SURFACE
PARTIAL DERIVATIVE ABS(A(10)) PLUS ABS(A(01))

FIG. 9

FIG. 10

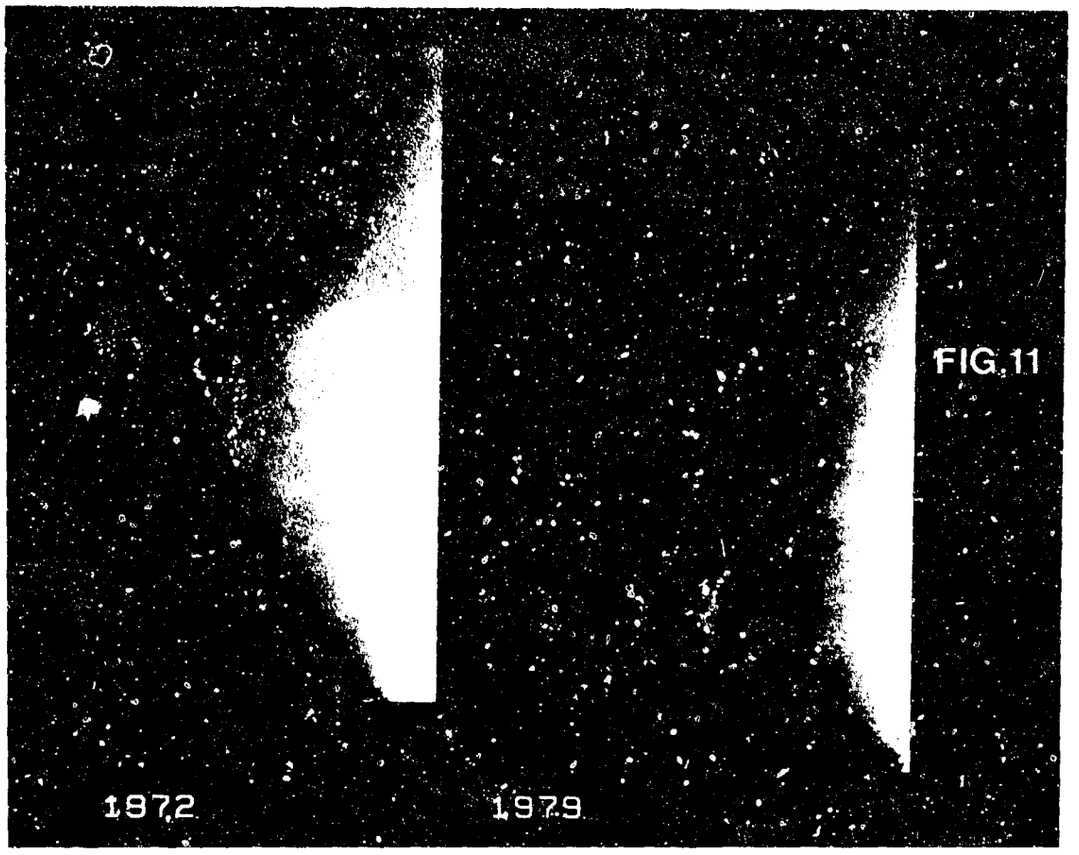


FIG. 11

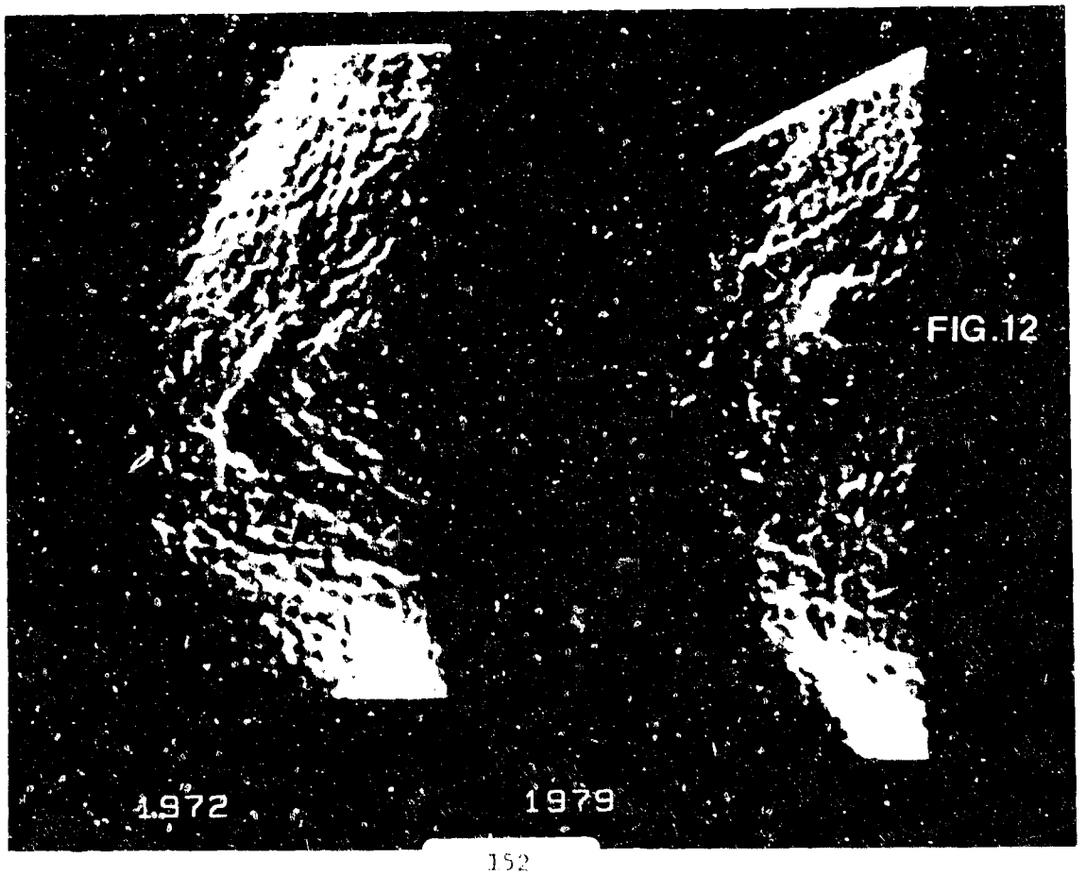


FIG. 12

TWO DIMENSIONAL CONVOLUTE INTEGER FILTER
HARDWARE DESIGN

5 X 5 FILTER MASK
DELAY CHAIN SHIFT REGISTERS, ADDERS AND MULTIPLIERS

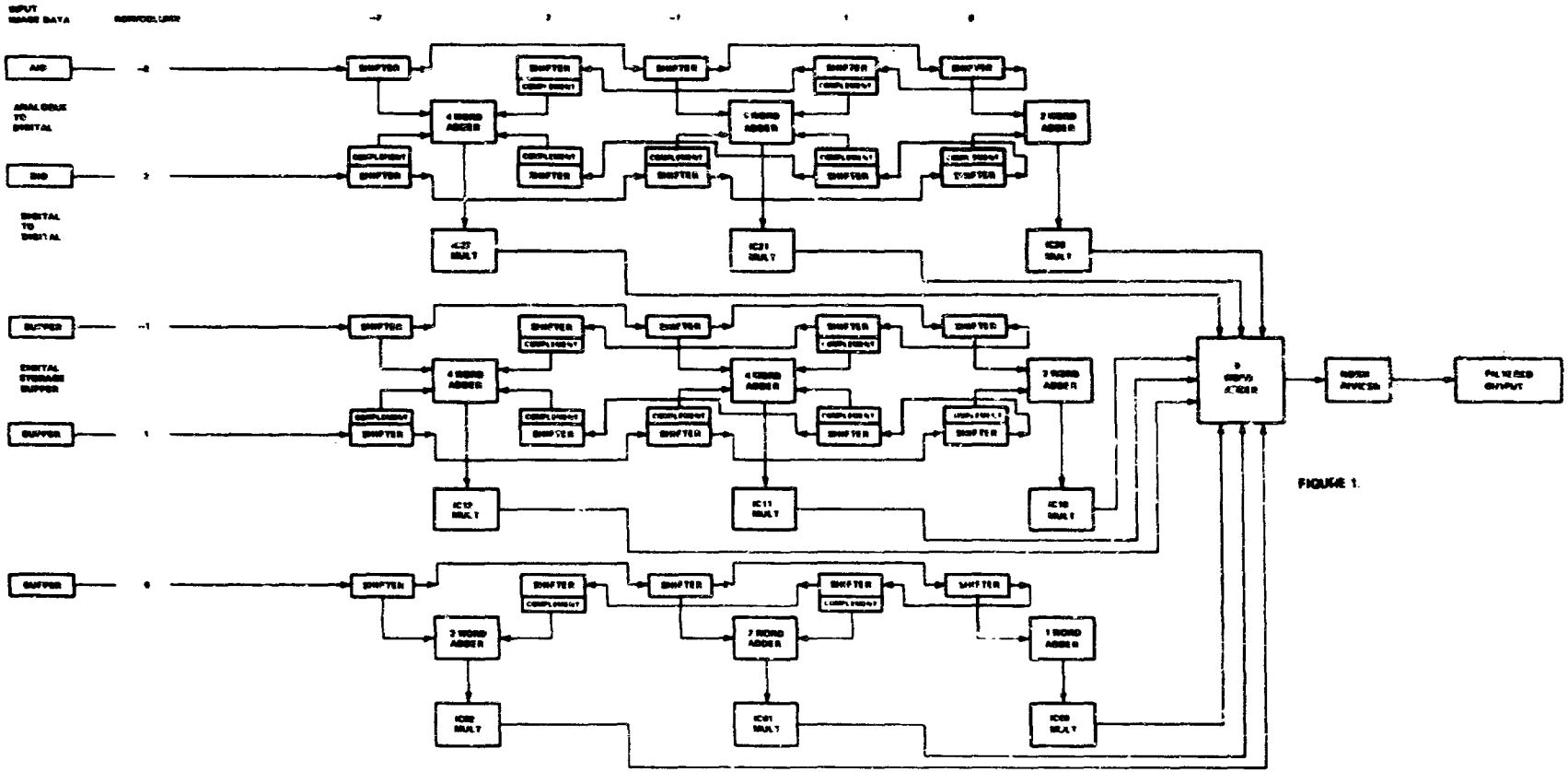


FIGURE 1.

FIG.13

TWO DIMENSIONAL CONVOLUTE INTERSTITAL POINT GENERATOR.

4 X 4 MASK
2ND ORDER SURFACE
SMOOTHING INTERSTITIAL

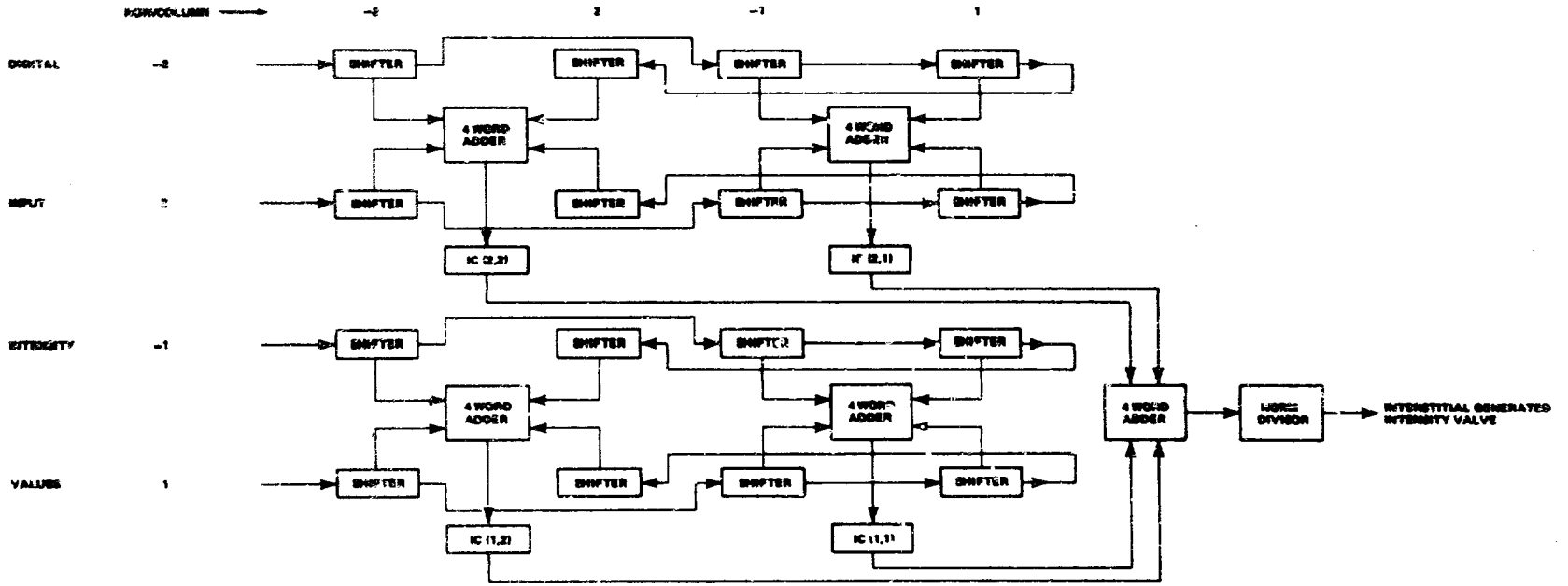


FIG 1

FIG.14

SESSION IIA

ADVANCED IMAGING TRACKERS FOR MISSILE GUIDANCE

Paper No. IIA-1, Presented at the Workshop on Imaging Trackers and Autonomous Acquisition Applications for Missile Guidance, 19-20 November 1979, Redstone Arsenal, Alabama.

TARGET TRACKING METHODOLOGIES PRESENT AND FUTURE

CPT BENJAMIN REISCHER
U.S. Army Night Vision and
Electro-Optics Laboratory
Advanced Concepts Division
Fort Belvoir, Virginia 22060

Abstract

This paper presents an overview of target tracking methodologies. The evolution of the tracker is traced from its original basic design capabilities and limitations through today's state-of-the-art (SOA) multimode trackers. Discussion is made concerning limitations of SOA trackers and consequently the necessity for development of an "intelligent" target tracker. Required capabilities of the intelligent tracker are discussed. Details concerning basic research and development work and progress made to date in the area of intelligent target tracking are discussed.

Introduction

The Night Vision & Electro-Optics Laboratory has been actively performing smart sensor research and development for the purpose of supplementing or supplanting the human observer in the target acquisition role. The target acquisition scenario requires the detection of target-like objects, tracking of the objects until recognition is possible, identification as a target for engagement, munitions launch and launch transient target reacquisition, and finally, tracking till munitions impact. Automatic tracking systems are utilized for supplementing the human observer in several of these roles. The intelligent target tracker discussed in this paper is an attempt to provide a technology base suitable for many different tracking roles. The intelligent tracker is specifically suited for systems lacking continuous man-in-the-loop interaction and is ultimately required for fully autonomous weaponry.

This paper presents an overview of the various target tracking methodologies and their application into fielded or future systems. The intelligent target tracker is presented as concepts to be explored and scenarios to be investigated.

Presently Fielded Military Trackers

Presently fielded military trackers typically are of the single mode, non-adaptive type - most commonly simple correlation or contrast (centroid of brightness). Target lock-on is achieved solely by operator command and the object being tracked is described merely as a grouping of illuminated pixels within a fixed or operator defined track window. Failure of the tracker algorithms to delineate the boundaries of the tracked object coupled with poor window sizing (a process known as "gate discipline") permits the introduction of non-target pixels (ie clutter objects) within the track window. Frequent loss of track in clutter regions is typical for these non-adaptive tracking methodologies and a man-in-the-loop scenario is required for target reacquisition after track loss.

The correlation tracker attempts window image registration on a frame-to-frame basis with repeated update of the reference image. As the tracked target moves, both target aspect changes and clutter combine to severely influence the tracker confidence. Hence for a target moving behind a large clutter object (e.g. bush) the required reference updating often causes the tracker to remain locked onto the bush and not the re-emerging target.

The contrast tracker is affected in much the same way by target obscurations. A moving target approaching another object causes both signatures to enter the track window and hence forces the centroid of brightness to be driven to a point between the two objects often causing the loss of the originally tracked target. Furthermore, since the target is known only by its intensity profile, the centroid (hence tracker aimpoint) will vary with changing target aspect. This "aimpoint wander" is one of the most significant errors in standoff tracking systems.[1] The operator's inability to maintain effective gate discipline greatly increases the probability of breaklock and further burdens him with continually performing manual target reacquisition.

Another shortcoming of presently fielded trackers is caused by the use of less than optimal algorithms (e.g. binary correlation) which are required to allow implementation in reasonable realtime packages. Despite this, the fabricated hardware is often bulky, special purpose (mission dependent) and has a significant power consumption. This often requires a redesign of the hardware to compensate for the pecularities of a new mission scenario.

Perhaps the most significant limitation is the implicit assumption that one and only one "target" exists to be tracked in the image. It is this assumption which lies at the heart of the "gate discipline" and correlation "reference update" problems. Without the duplication of tracker hardware, the technology is limited to tracking one target at a time, although the military necessity for delivering high rates of fire power effectively upon the enemy depends on simultaneous tracking of multiple targets.

State-of-the-Art Trackers

The newly emerging state-of-the-art target trackers typically are of a multimode, adaptive microprocessor-based technology. The SOA tracker typically incorporates a variety of tracking methodologies in a multimode format. These include various combinations of algorithms such as correlation, contrast, edge detection, motion detection, coast mode clutter compensation, etc. Each mode tracks well over a wide range of conditions, and each tends to fail (to maintain track) under a specific set of conditions (i.e. low contrast, high clutter, etc.). However, it is highly unusual for all modes to fail simultaneously for a given condition. Multimode trackers maintain track better than single mode trackers since we often find that when the confidence level of a given tracking mode is low, we can continue to track using another mode with a high confidence level.

This beneficial state of interaction (synergism) is implemented by one of two methods. In the earlier method, the tracker interacts with the operator by warning him of impending breaklock, and the operator then selects a different track mode. In the most recent trackers, a "controller" algorithm evaluates the various tracking methodologies, assigns a measure of track confidence to each mode, automatically selects the mode for tracking which yields the highest confidence level, and provides updated track information to the other algorithms. This creates much greater synergism than is possible with a human "controller." It is anticipated that the use of multimode algorithms will dramatically decrease the frequency of breaklock.

The use of Large Scale Integration (LSI) hardware techniques in SOA trackers permits a more optimal selection of tracker algorithms for realtime applications and permits the incorporation of advanced multimode trackers in terminal munitions and man-in-the-loop RPV scenarios. The use of microprocessor-based technology permits fine tuning of algorithms for specific applications. This is a tremendous improvement over fielded (hard-wired) trackers which are not adaptable to improved algorithms without hardware modification.

SOA trackers have an adaptive gate which attempts to close about the target and exclude the background. While this works fairly well with high contrast targets, the adaptive gate experiences difficulties with low contrast targets in high clutter areas. Another feature of SOA trackers is the coast mode clutter compensation algorithm. This algorithm uses a priori velocity information to coast the tracker through temporary occlusions (which usually causes break lock) and to reacquire the target as it emerges. This method works well if the target's velocity and direction of travel do not change while it is observed. Unfortunately SOA trackers still encounter severe problems when tracking in high clutter environments. As an example, assume the tracked target moves into a high clutter region. Perturbations in tracker confidence force the system into a reacquisition mode. Coast mode clutter compensation takes effect, and the tracking gate widens in an attempt to reacquire the target. However, clutter objects now enter the track window and significently influence the tracker algorithms, often causing the system to lock onto a clutter object rather than the true target. Additional factors which create problems for the algorithms

include low target-to-background contrast, sun-to-horizon angle (shadows, glint), target-sun aspect, background texture, etc.[1]

Some of the latest trackers being introduced have features not seen previously. These include automatic acquisition of targets by use of algorithms such as brightness or motion detection; tracking of two targets at a time, and a limited aimpoint analysis capability due to delineration of target edges. Even though they provide the best tracking schemes to date, multi-mode trackers still lack the sophistication for application into fully autonomous terminal munitions.

Military applications for SOA trackers include the Remotely Piloted Vehicle (RPV)[2], The Advanced Attack Helicopter (AAH)[3], and Lock-On-Before - Launch munitions such as the Hellfire imaging tracker (THASSID)[4].

Advanced "Intelligent" Target Trackers

The Night Vision & Electro-Optics Laboratory is currently sponsoring research for the development of an intelligent target tracker which will combine target cueing and target tracking methodologies for near zero breaklock performance.[5][6] A synergistic cuer/tracker combination is expected to lead to the development of a fully autonomous tracker. This will allow, through the use of VLSI/VHSI techniques, the intelligent target tracker (with inherent target cuer) to be applied to the fully autonomous munition. The following concepts and capabilities of the intelligent tracker are being explored:

Multiple Target Tracking

The intelligent tracker must be able to track many targets in the sensor field of view simultaneously. Tracker/cuer synergism will allow the cuer to continually inform the tracker of the location of all cued objects. The tracker will then update its memory to acknowledge the existence of a new target or reconfirm the location of known targets. Preliminary investigation in this area indicated the need for trade-off studies between the extremes of a super-fast cuer cueing a relatively simple slow tracker; or a relatively slow cuer cueing either a very fast, sophisticated tracker or alternately cueing many simpler trackers, each limited to tracking a single target. One approach currently under investigation by Westinghouse Corporation involves an auto cuer cueing every fifth frame and a single band pass correlation tracker tracking multiple targets simultaneously.[7][8] Preliminary findings are very promising.

Realization of the multiple target tracking capability will permit multiple target engagements in the ground-to-ground scenario and automatic Ripple Fire and simultaneous multiple weapon fire engagements from the AAH/HELLFIRE and RPV air-to-ground platforms.

Target Prioritization

Since the intelligent tracker works in conjunction with a cuer, target classification information for all cued objects is made available to the tracker as feature information (size, shape, range, etc.) is extracted from the sensed

scene. Classification permits prioritization of all the tracked multiple targets based on a priori knowledge about target type and threat. This capability will allow the tracker to always point to (and engage) the highest threat target first in a multiple target scenario. It is important to realize that target prioritization must be considered in conjunction with threat assessment. For example, if the tracker is located in a tank, then an enemy tank would be of higher priority than a SA-9 missile. The reverse would be true if the tracker were in an RPV.

Critical Aimpoint Selection

Since the target has been classified and track information is available, the intelligent tracker can point to the location of the most vulnerable point of the target. Munitions deployment would be directed to that point since a hit there would yield the highest probability of kill.

Autonomous Target Tracking

The intelligent tracker should track autonomously, automatically reacquiring the target as need be, from the time of acquisition until time of completed munitions deployment without any human intervention. A situation illustrating the need for this capability is when targets enter, leave, and re-enter the field of view (FOV) as happens in the RPV. In such a situation, it is very important for the tracker to "remember" the characteristics of targets outside the FOV, namely their priority, direction of travel, velocity and time since leaving the FOV. Thus, after the highest priority target has been engaged, the tracker would know approximately where to slew the sensor so as to acquire the next highest priority target, even if it is now outside the FOV of the current image.

One of the most promising methods under investigation for realizing the autonomous tracking capability is the concept of "Signature Prediction". Current tracker technologies consider only the background immediately surrounding the target being tracked. This approach leads to frequent track loss as the target moves into new background regions. The signature prediction algorithms of the intelligent tracker will monitor a broad area around the target and thus be able to predict the expected variation of the target's signature before it enters the new background region. This advanced approach will allow the tracker to track the target as it crosses the boundary between background regions.

Figure 1 depicts three scenarios which often lead to track loss. Case A depicts a target leaving a region of uniform background and crossing into a new background. In this case, the signature predictor must look ahead, place the target in the new region, and predict the degraded image. Knowledge of the background is essential since the predictor must be careful about perspective (i.e. it must not place the tank on top of a tree). In cases B1 and B2 portions of the target are occluded by terrain features. The signature predictor must anticipate that in the following frames it should expect to see only the front of the tank, or the turret, etc. In case C, the target is almost completely obscured and only isolated pixels of brightness can be seen. The predictor

must be able to anticipate that it will be tracking groups of isolated pixels and continue to track the "target" as it moves through the trees. NV&EOL has collected a target tracking/homing data base which depicts all three scenarios. It is being used for intelligent tracker development studies and is available to interested parties.

Figure 2 is an illustration of how the signature predictor works. It shows a typical scene in which the tracked target is leaving a scene of uniform background (road) and moving into a high clutter area. The signature predictor, looking ahead, notes the future background, "places" the target into the future background, determines the predicted signature, and updates the tracker of the new "reference" to be expected in the next image. Hence, when the target does indeed reach the new background, the tracker already knows the new (degraded) reference signature, and adapts to prevent track loss.

It is important to realize that significant problem areas are encountered in the signature prediction process. Chief among these are perspective (or height) estimation (namely, the a priori knowledge which enables the tracker to superimpose the target properly into its future background, i.e., under trees and on top of bushes); proper compensation for targets undertaking evasive maneuvers (rapid aspect changes); and, of course, the requirement for background segmentation and scene modeling.

Westinghouse Corporation has been investigating the Signature Prediction Concept and has made some very interesting findings.[7][8] Among these, a phenomenon known as "bridging" occurs frequently for TV imagery when a bright object approaches a dark region. The pixels located between the bright object and the dark boundary tend to change in intensity and "average" so that the forward edges of the target merge into the background and the segmentation process fails. It becomes essential for the signature predictor to "look ahead" a distance 2-3 target widths in front of the targets in order to establish a valid future reference image. Westinghouse found that tracking the "rear edge" of a target in such a situation, coupled with a "change detection" algorithm which establishes the original boundary as a reference image and looks for changes in future frames at that boundary, enables the tracker to maintain track. Before the rear edge is lost, the tracker slides forward to track on the emerging forward edge.

It was also learned that even if the signature predictor fails to predict an obscuration (i.e. due to its inability to classify background regions) and break lock occurs, it is often possible for the tracker to reacquire the target rapidly by the use of change detection. This is done by looking for changes in the signature predictors "reference" windows. Thus reacquisition becomes possible even if only a portion of the target reappears in the clutter region. Reacquisition occurs despite the fact that the target cannot be segmented and cued.

Summary

The metamorphosis of the tracker from its present state to its intelligent counterpart can best be illustrated by the following example - A Tracker for the Army's Remotely Piloted Vehicle (RPV).

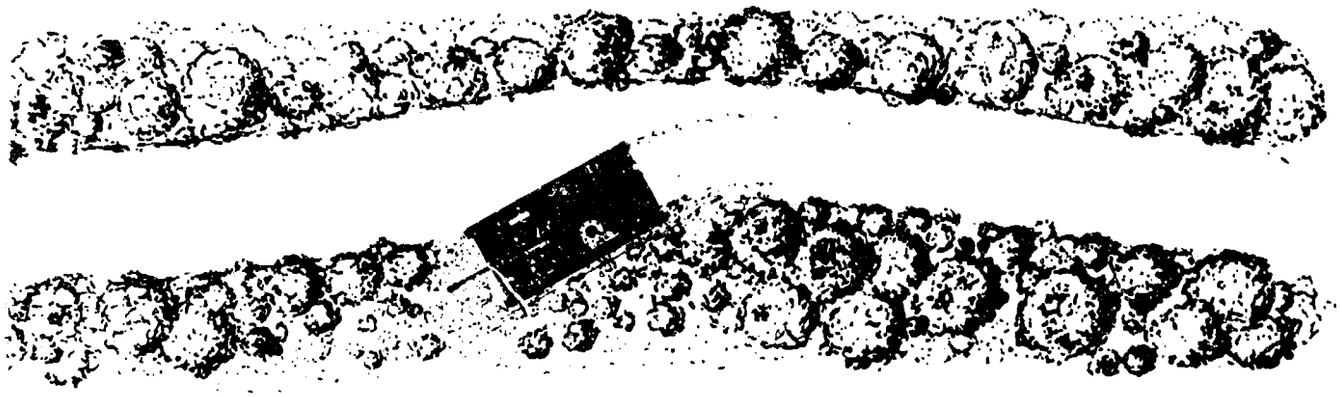
Presently Fielded Tracker - The present RPV tracker uses a contrast only technique which suffers from frequent breaklock conditions. Breaklock occurs not only due to the reasons mentioned earlier, but also due to wing occlusion. This occurs when the RPV is turning and the wing obscures a portion of the FOV. There is no reacquisition capability in the tracker itself. The operator has great difficulty in reacquiring the target. He must slew the sensor back to the target. The data link contributes to the problem since the update rate is slow, and the image is compressed and fuzzy. The data link is also subject to enemy jamming which further interferes with reacquisition. Aimpoint wander is a significant problem as the laser must be held steady on the target until the precision guided munition (PGM) impacts. Current PGM's are extremely sensitive to beam wander (aimpoint shift) on the target. Excessive aimpoint wander greatly reduces the probability of kill.

State-of-the-Art Tracker - A SOA tracker for the RPV would substantially improve the system's performance by using several tracking modes in a synergistic manner. The problems associated with wing occlusion and subsequent breaklock mentioned earlier will diminish due to the background (scene) correlation algorithm. The SOA tracker has an offset track capability and thus there exists a limited aimpoint selection capability. Since track loss should not occur as frequently as in the present tracker, and since the operator will be made aware of an impending breaklock situation, his ability to reacquire the target as well as his ability to minimize aimpoint wander should improve dramatically. A coast made clutter compensation algorithm would also be expected to reduce the frequency of breaklock.

Intelligent Tracker - An intelligent tracker working in conjunction with a target cuer should reduce the operator's function to a monitoring operation in most cases. As required, he might override the tracker's selection of highest priority target with one of his choosing. In a case of extreme jamming (when no video information can be received), it should be possible for the tracker to send a short code to the operator: "I have a target of type T, location X, Y." The operator knowing the RPV's location, could then make a decision concerning the likelihood of it being an enemy target. If the operator decided it was a valid target, he could have a PGM fired towards the vicinity of the target, and activate the laser. Meanwhile, the intelligent tracker would be tracking the target, reacquiring autonomously as need be, and maintain the laser at the target's optimal aimpoint. This process could be continued indefinitely because cuer/tracker synergism would allow autonomous acquisition and tracking of objects, classifying them, prioritizing them and selecting an aimpoint for each target.

TRACKER SCENARIOS

CASE A: TARGET LEAVING A REGION OF UNIFORM BACKGROUND INTO A NEW REGION:



CASE B1: TARGET PARTLY OCCLUDED BY AN OBSTRUCTION:

CASE B2: TARGET PARTLY OCCLUDED BY TERRAIN FEATURES:

CASE C: TARGET ALMOST COMPLETELY OCCLUDED (ISOLATED AREAS VISIBLE):

Figure 1 - Tracker Scenarios

SIGNATURE PREDICTION

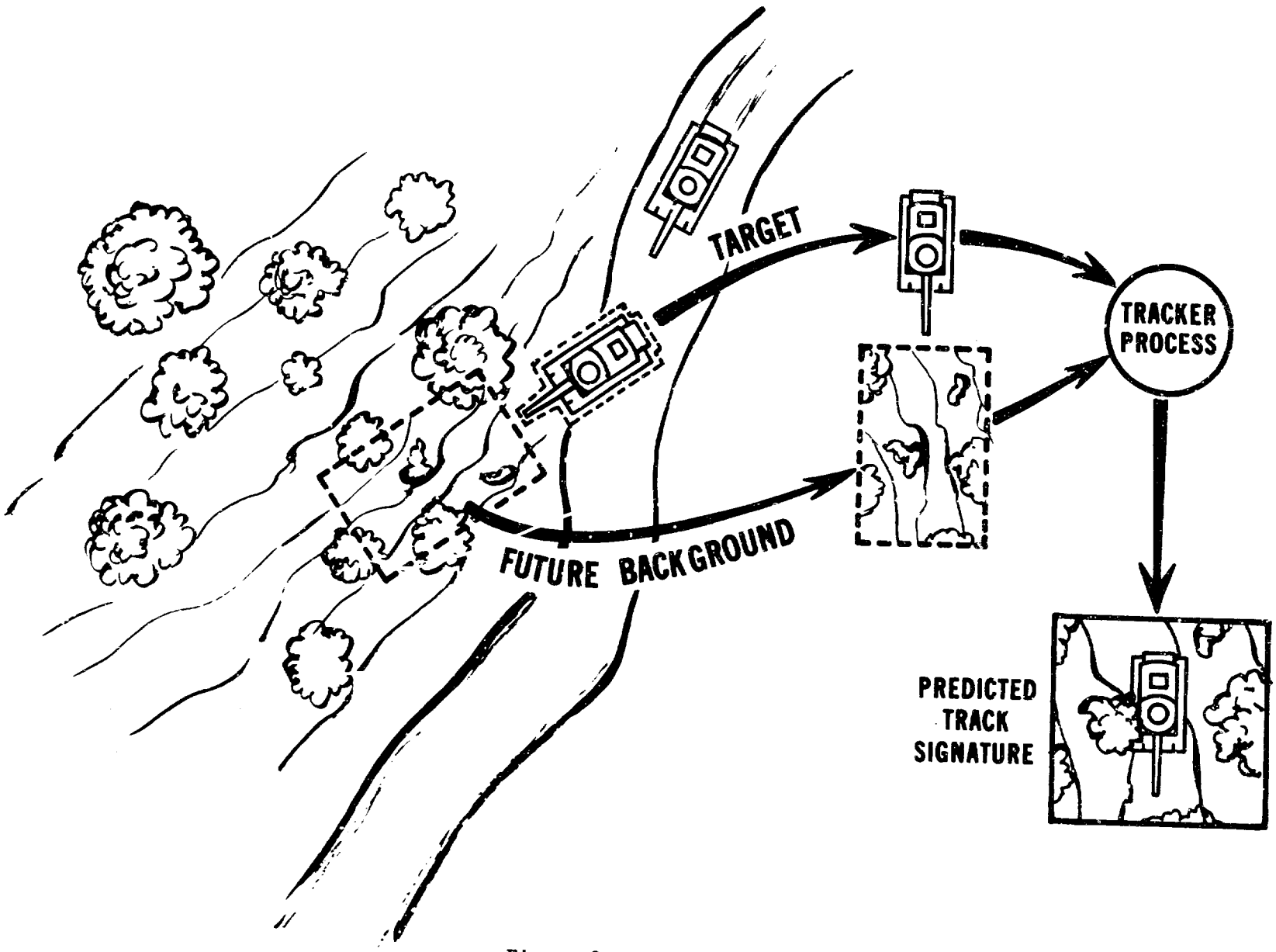


Figure 2 - Signature Prediction

Acknowledgments

The author wishes to thank Mr. John Dehne of the Night Vision and Electro-Optics Laboratory for his invaluable help and encouragement. He also wishes to thank Mrs. Mary Vernon for her skill with this manuscript.

References

1. Mini-Remotely Piloted Vehicle Precision Tracking Evaluation - Interim Report for the period Jan 78 - Dec 78, Contract DAAB07-78-C-2415.

2. Prime Item Development Specification for U.S. Army Remotely Piloted Vehicle System (U), RFP # DP-RPV-SEN-56646, 22 Nov 78.

3. Prime Item Development Specification for Target Acquisition Designation System (U), RFP # AMC-DP-AAH-H4020, 12 Apr 77.

4. Terminal Homing Applications of Solid State Image Devices, Final Report, Contract DAAK40-76-C-1083, Dec 78.

5. NV&EOL Contract DAAK70-78-C-0167, Intelligent Target Tracking Techniques, Westinghouse Electric Corporation.

6. NV&EOL Contract DAAK70-79-C-0150, Advanced Target Tracker Concepts, Honeywell, Inc.

7. Intelligent Tracking Techniques, Second Quarterly Report for the period Jan-Mar 78, Contract DAAK70-78-C-0167.

8. Intelligent Tracking Techniques, Third Quarterly Report for the period Apr-Jun 78, Contract DAAK70-88-C-0167.

Paper No. IIA-2, Presented at the Workshop on Imaging Trackers and Autonomous Acquisition Applications for Missile Guidance, 19-20 November 1979, Redstone Arsenal, Alabama.

LIE THEORETIC METHODS IN VIDEO TRACKING

Thomas G. Newman and David A. Demus
Department of Mathematics
Texas Tech University
Lubbock, Texas 79409

ABSTRACT

Consider a 2-dimensional image in which objects are in motion through trajectories describable by translation (both horizontal and vertical), rotation, and magnification. The trajectory of such an object can be completely described by a 4-vector of parameters $\lambda(t)=(\lambda_1,\lambda_2,\lambda_3,\lambda_4)$ which determine the velocities with respect to the four possible motions. If the data at time t and position x in the view plane is written as F(t,x), then we can show that

$$\frac{\partial F}{\partial t} = \sum_{i=1}^{4} \lambda_i(t) X_i F,$$

where X_1, X_2, X_3 and X_4 are certain (known) differential operators associated with the group of motions.

The derivatives appearing above may be evaluated numerically at various points in a given time slice to produce a system of linear equations which may be solved for the motion parameters. Evaluation at points within a moving rigid body leads to a vector of motion parameters unique to that particular body. In principle, at least, this technique permits application to tracking as well as segmentation of images based on relative motion of various objects.

The paper concludes by presenting the results of having implemented the above method on digitized video images.

INTRODUCTION

A complex three dimensional scene may contain an arbitrary number of objects, each of which is in motion relative to a stationary background. The trajectories of the various objects may or may not be the same. When such a scene is projected on a viewing plane (for example, through the use of a television camera), the various objects appear as moving regions which vary in time in a complex fashion as a result of their actual trajectories

in space. Variations due to certain trajectories, such as rotation about a line parallel to the image plane, are not readily predictable. Previously unseen patches of the surface of an object may be brought into view for the first time, while others may disappear. In addition, a near object may pass between the camera and a distant object, occluding all or part of the latter.

The situation is further complicated in case mobility is provided at the camera. Motion of the camera results in an opposing change in the apparent motion of all of the objects in the scene, including background. In many applications camera mobility is desirable or even necessary. For instance, in tracking applications the motion of the camera is required to stabilize a particular portion of the scene within the viewing field. Although this may in general be impossible, as with the rotating objects mentioned above, a fair degree of stabilization with respect to position, size, and orientation can be achieved.

In the following sections we present a model for describing motion in images which is valid in a large number of practical applications and which is a reasonable approximation in many others. A novel feature is that camera motion and relative motion of objects within a scene are both described within the model.

THEORETICAL MODEL

Let G be a Lie group of transformations on an analytic manifold M. Suppose G has dimension n while M has dimension m. Let x and y denote the coordinates of elements f and g in G, respectively, in a patch containing the identity element e of G. Also, let p denote coordinates of an element u of M in some patch in M. We may then express the coordinates z of the product h = fg and the coordinates q of the element v = gu, relative to suitable patches, by means of analytic functions

$$z = J(x,y) \tag{1}$$

$$q = K(y,p) \tag{2}$$

K and J are vector-valued, having values in n-dimensional space R^n or C^n and m-dimensional space R^m or C^m. Hereafter we shall assume that these underlying spaces are real. We denote the ith component of J by J_i and the jth component of K by K_j.

In order to define the Lie algebra of G we first introduce real-valued maps on G by

$$P_{ij}(x) = \frac{\partial J_i}{\partial y_j}(x,y)\Big|_{y=e}, \tag{3}$$

where i and j each range from 1 to n. The cross-section P_{*j}, which consists of the P_{ij} as i ranges from 1 to n, and j is fixed, may be thought of as a vector field in R^n. Such a vector field attaches to a point x the vector $P_{*j}(x)$. As such, $P_{*1}, P_{*2}, \ldots, P_{*n}$ form a basis for the tangent space at the point x [1,2]. In view of the correspondence between elements f in G and the coordinates in R^n, the tangent vectors are implicitly attached to the elements of G.

In terms of the above vector fields we may express the infinitesimal transformations of G by defining, for each $j = 1,2,\ldots,n$,

$$X_j = \sum_{i=1}^{n} P_{ij}(x) \frac{\partial}{\partial x_i} . \tag{4}$$

The differential operators so defined are to be considered as linear operators on the space of analytic functions on G, or, more generally, on the space of differentiable functions on G. The Lie algebra of G is simply the n-dimensional vector space consisting of all linear combinations of these operators, and will be denoted by L(G) [2].

Now it is a surprising and useful fact that the Lie algebra of G may be defined in terms of its actions on the manifold M. Analogous to (3) we define

$$Q_{\alpha j}(p) = \frac{\partial K_\alpha}{\partial y_j}(y,p)\Big|_{y=e} \tag{5}$$

for $\alpha = 1,2,\ldots,m$ and $j = 1,2,\ldots,n$. Finally, as in (4) above we set

$$X'_j = \sum_{\alpha=1}^{m} Q_{\alpha j} \frac{\partial}{\partial p_\alpha} . \tag{6}$$

The operators $X'_1,\ldots,X'_n$ span a Lie algebra L'(G) which is also of dimension n. Note that these operators act on functions defined on the manifold M.

Many interesting relationships may be shown to hold between the two representations of the Lie algebra of G as given above. However, the following property is of immediate interest to our application:

Theorem 1: Let $f: M \to R$ be differentiable and define $F: G \times M \to R$, in terms of coordinates by

$$F(x,p) = f(K(x,p)). \tag{7}$$

Then for each $j = 1,2,\ldots,n$ we have

$$X_j F = X'_j F. \tag{8}$$

Proof: First we shall show that for each $j = 1,2,\ldots,n$ we have

$$X_j K = X_j' K. \tag{9}$$

We note that from the action of G on M we obtain

$$K(J(x,y),p) = K(x,K(y,p)) \tag{10}$$

for all x,y and p in suitable coordinate patches. Application of the operator

$$\frac{\partial}{\partial y_i}\Big|_{y=e}$$

to both sides of (10) gives

$$\frac{\partial K_\alpha(J(x,y),p)}{\partial y_i}\Big|_{y=e} = \sum_{k=1}^{n} \frac{\partial J_k(x,y)}{\partial y_i}\Big|_{y=e} \cdot \frac{\partial K_\alpha(x,p)}{\partial x_k} =$$

$$\sum_{k=1}^{n} P_{kj}(x) \frac{\partial K_\alpha(x,p)}{\partial x_k} = X_j K_\alpha(x,p)$$

for the left hand side and

$$\frac{\partial K_\alpha(x,K(y,p))}{\partial y_j}\Big|_{y=e} = \sum_{\beta=1}^{m} \frac{\partial K_\beta(y,p)}{\partial y_i}\Big|_{y=e} \frac{\partial K_\alpha(x,p)}{\partial p_\beta} =$$

$$\sum_{\beta=1}^{m} Q_{\beta j}(p) \frac{\partial K_\alpha(x,p)}{\partial p_\beta} = X_j' K_\alpha(x,p)$$

on the right hand side. From this it follows that $X_j K = X_j' K$ as desired. Now setting $q = K(x,p)$ and performing a computation similar to that above, we find that

$$X_j F(x,p) = \sum_{\alpha=1}^{m} X_j K_\alpha(x,p) \cdot \frac{\partial f(q)}{\partial q_\alpha}$$

and that

$$X_j' F(x,p) = \sum_{\alpha=1}^{m} X_j' K_\alpha(x,p) \cdot \frac{\partial f(q)}{\partial q_\alpha} .$$

The result of the theorem follows immediately from this and our preliminary result.

Now let us consider a curve $t \to g(t)$ in G satisfying $g(0) = e$. In terms of a coordinate patch at e, $g(t)$ may be described by a curve $x(t)$ in R^n satisfying $x(0) = 0$. We shall consider the case in which $x(t)$ is given as the solution of an evolution equation of the form

$$\dot{x}(t) = \sum_{i=1}^{n} \lambda_i(t) P_{*i}(x(t)), \; x(0) = 0, \tag{11}$$

where $P_{*1},\ldots,P_{*n}$ are cross-sections of the array of functions given by (3), and the control functions $\lambda_1(t),\ldots,\lambda_n(t)$ are suitable continuous functions. The latter are the parameters of motion, and have the characteristics associated with velocity, thereby providing a basis for the continuity assumption.

Now let p denote the coordinates of a point u in some coordinate patch. For a differentiable map $f: M \to R$ we may define $H: R \times M \to R$ by setting

$$H(t,p) = f(g(t)u). \tag{12}$$

We recognize that $H(t,p) = F(x(t),p)$ where F is the extension of f to $G \times M$ as in Theorem 1 above. From the point of view of application, if we regard $f: M \to R$ as an image, then $H(t,p)$ represents the moving image obtained by translation due to the curve $g(t)$. We may now present our main result.

Theorem 2: In the context described above we have

$$\frac{\partial H}{\partial t} = \sum_{i=1}^{n} \lambda_i(t) X_i' H. \tag{13}$$

Proof: We have

$$\frac{\partial H}{\partial t}(t,p) = \frac{\partial F(x(t),p)}{\partial t} = \sum_{j=1}^{n} \dot{x}_j(t) \frac{\partial F}{\partial x_j}(x(t),p) =$$

$$\sum_{j=1}^{n} \Big(\sum_{i=1}^{n} \lambda_i(t) P_{ji}(x(t)) \Big) \frac{\partial F}{\partial x_j}(x(t),p) =$$

$$\sum_{i=1}^{n} \lambda_i(t) \Big(\sum_{j=1}^{n} P_{ji}(x(t)) \frac{\partial F}{\partial x_j}(x(t),p) \Big) =$$

$$\sum_{i=1}^{n} \lambda_i(t) X_i F(x(t),p).$$

By Theorem 1 we have $X_i F = X_i' F$. But clearly $X_i' F(x(t),p) = X_i' H(t,p)$, so that

$$\frac{\partial H}{\partial t}(t,p) = \sum_{i=1}^{n} \lambda_i(t) X_i' H(t,p),$$

as desired.

We should observe that the results above are presented as local properties which hold in suitable neighborhoods and appear to be highly coordinate dependent. As a matter of fact, though we shall not attempt to prove it here, the underlying vector fields continue globally throughout both G and M to give corresponding global analogues of these theorems.

The primary importance of Equation (13) lies in the fact that it gives a linear equation in the control parameters $\lambda_1,\ldots,\lambda_n$ with coefficients that are in principle observable, since the values H(t,p) constitute the data.

In the next section this result will be applied to the problem of tracking spatial objects through the use of two-dimensional projections.

APPLICATIONS TO VIDEO TRACKING

The control system for the Real-Time Videotheodolite (RTV) permits four basic motions of the camera [3]. These are azimuth, elevation, electronic rotation of the view plane, and lens zoom. When the effects of these motions on the viewing plane are scrutinized, we see that they correspond, respectively, to horizontal translation, vertical translation, rotation, and magnification - at least to a satisfactory degree of approximation. Moreover, inspection of a number of real images reveals that a surprisingly large number (but not all) motions of spatial objects, when projected on the viewing plane, are likewise well approximated by these four motions in the plane.

Thus with only a mild apology we restrict our attention in what follows to the group G generated by horizontal and vertical translations, rotation, and magnification. The corresponding generators for the Lie algebra of G are as follows:

$$X_1 = \frac{\partial}{\partial x} \tag{14a}$$

$$X_2 = \frac{\partial}{\partial y} \tag{14b}$$

$$X_3 = x\frac{\partial}{\partial y} - y\frac{\partial}{\partial x} \tag{14c}$$

$$X_4 = x\frac{\partial}{\partial x} + y\frac{\partial}{\partial y} \tag{14d}$$

In these equations we are using x and y as coordinates in the view plane $M = R \times R$ and have represented the infinitesimal transformations as they act on M.

Let us note that in the theorems of the previous section it was assumed that the trajectories of all of the points of M were derived from the same evolution equations. However, for complex scenes we find that various objects may be present which have different trajectories. A little reflection reveals, nevertheless, that the conclusions of Theorem 2 remains valid as long as we avoid the boundaries between objects or regions having different trajectories. In the present context, we may paraphrase the results of Theorem 2 as follows:

Theorem 3: Let $H(t,x,y)$ be a time varying two dimensional image. Within the interior of each object in the image which is moving along a G-trajectory, we have

$$\frac{\partial H}{\partial t} = \sum_{i=1}^{4} \lambda_i(t) X_i H, \tag{15}$$

where $\lambda_1,\ldots,\lambda_4$ are continuous functions and $X_1,\ldots,X_4$ are given in (14).

Upon evaluation of the various derivatives appearing in (15) at each point of a suitable grid, within a given time slice, we obtain a system of linear equations which may be solved for the parameters of motion, $\lambda_1,\ldots,\lambda_4$. In the example to be presented, a 3×3 grid was used.

A sequence of digitized video images showing the launch of a Hawk missile were obtained from the U.S. Army White Sands Missile Range. The images were trimmed to 128×128 pixels from full frame interlaced video in which each raster line was sampled 512 times.

One of the frames is shown in the upper left of the illustration below. Of noteworthy interest, we mention the "cold plume" region (lower left) which can be seen billowing out behind the missile. Although hardly discernible, the foreground contains several buildings and other ground clutter.

By evaluation of Equation (15) at each point of a 3×3 neighborhood of each pixel, nine equations in the four parameters $\lambda_1,\ldots,\lambda_4$ were obtained. In the upper right frame of the illustration, we see the results of scaling the horizontal translation component, λ_1 for display. The effect of image noise and truncation error is apparent from the rapid transition from white to black in this view. This component of the velocity profile was passed through a median filter to obtain the image shown in the lower left of the illustration. Finally, in the lower right we see the results of thresholding, about $\lambda_1 = 0$. In this image the dark region indicates points which are at rest relative to the camera (which was apparently successfully tracking the missile), while the white regions appear to be moving with respect to the camera.

It is interesting to note that the cold region of the plume has been correctly classified with the background, while the hot region of the plume appears to be moving with the missile.

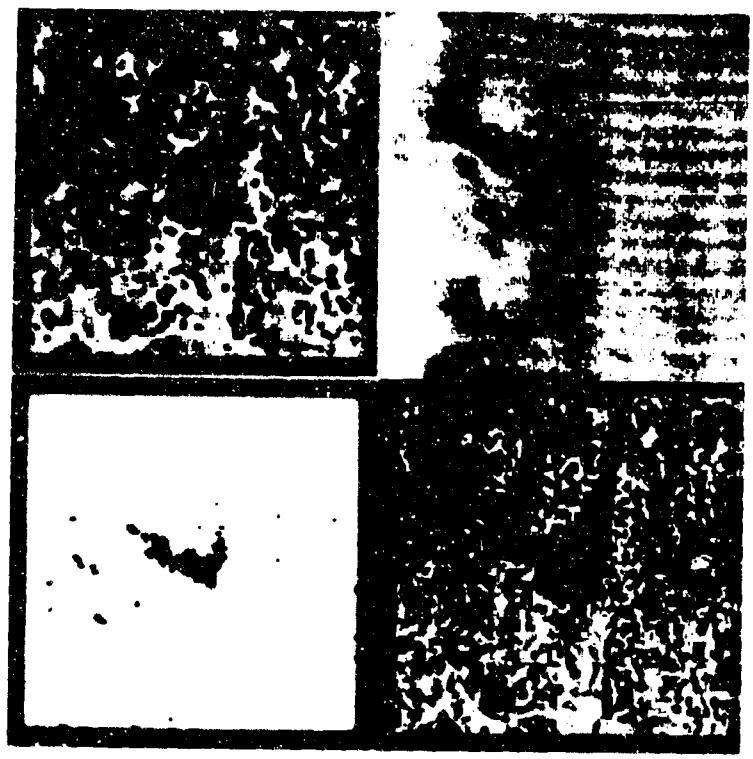


Figure 1. Processing the launch of a Hawk missile.

Similar results were obtained with other parameters and with other images. These results are encouraging, although the numerical methods employed are clearly too susceptible to noise and truncation. Better computational procedures are being explored, including one technique which is based on integration rather than differentiation.

SUMMARY AND CONCLUSIONS

We have developed a fundamental equation satisfied by moving images which uses Lie theory to determine the trajectories of various objects within an image. The theory has been implemented on real data with some success. While the implementation suffers from the effects of random noise and truncation errors, the results obtained have shown sufficient success as to be encouraging. We feel that the computations can be greatly improved by the incorporation of better numerical methods.

ACKNOWLEDGEMENT

This research was conducted under the auspices of the Joint Services Electronics Program at Texas Tech University and in collaboration with the Advanced Technology Office, Instrumentation Directorate, U.S. Army White Sands Missile Range. Support was provided by the Office of Naval Research under contract N00014-76-C-1136.

BIBLIOGRAPHY

[1] Auslander, L., Differential Geometry, Harper and Row, New York, 1967.

[2] Cohn, P. M., Lie Groups, Cambridge University Press, London, 1957.

[3] Flachs, G. M., P. I. Perez, et al, A Real-Time Video Tracking System, Final Report for Contract DADD07-D-0046, New Mexico State University, 1979.

Paper No. IIA-3, Presented at the Workshop on Imaging Trackers and Autonomous Acquisition Applications for Missile Guidance, 19-20 November 1979, Redstone Arsenal, Alabama.

THE AI^{2}S MICROPROCESSOR BASED MULTIMODE TRACKER

David A. Orton and Gerald N. Yutzi
Rockwell International
Electronics Research Center
Anaheim, California

ABSTRACT

The Advanced Infrared Imaging Seeker (AI^{2}S) Multimode Tracker was developed to meet fire-and-forget missile guidance requirements of the U. S. Army. Tracking algorithms were initially developed and simulated on an image processing general purpose computer facility. A multimode tracker organization was selected to combine correlation, contrast and moving target algorithms weighted for optimum guidance correction. A multi-microprocessor architecture was developed to implement the tracker algorithms. A Z80 Executive processor controls tracker operation, directing higher-speed AMD 2900 input and algorithm processors. Firmware was developed and integrated with the microprocessor hardware using two laboratory development systems and a Nova minicomputer for interface simulation. A flyable brassboard implementation is currently undergoing evaluation tests and will later be repackaged to meet missile constraints.

1. Introduction

The AI^{2}S multimode tracker was developed in Rockwell's Electronics Research Center for the Advanced Infrared Imaging Seeker program. This is a sophisticated new seeker system incorporating advances in IR detectors, CCD processors, tracking algorithms and microprocessor technology to meet fire-and-forget missile guidance requirements of the U. S. Army.

The focal plane array is made up of a 32x32 IR sensor whose detectors are directly connected to cells of a CCD integrating and multiplexer chip. A serial readout of detector samples is compensated and provided to the tracker at a 60-frame per second rate. The tracker function is to process this data and identify the current position of a target acquired before launch to generate in-flight missile guidance corrections.

A multimode tracker approach was selected to meet tracking requirements. Complementary tracking modes each contribute to a best estimate decision determined by a controlling Executive. The tracking modes include correlation,

a high-performance contrast algorithm (t-statistic), moving target, and momentum. Each mode independently generates a best estimate of target position and confidence from which the Executive computes a weighted mean. The Executive will monitor the performance level of each mode and reset any mode whose performance indicates a loss of track.

The tracker has been implemented in a multi-microprocessor architecture designed for high-speed complex operations. This is a flexible and modularly expandable architecture employing multiple algorithm processors which execute computational tasks scheduled and assigned by a controlling Executive processor. The Executive processor is a conventional 8-bit fixed format microprocessor (Z80) capable of performing general control and decision-making functions. Higher speed bit-slice microprocessors (AMD 2900) are used for tracker algorithm computations and image input handling.

The AI^2S system has been fabricated as a flyable brassboard which is currently undergoing field test and evaluation. Repackaging studies have established the practicality of reducing this design to meet missile space and power requirements. The following sections will describe the tracker algorithm development, multimicroprocessor architecture, hardware design, and firmware implementation of the algorithms.

2. Algorithm Development and Software Simulation

The multimode tracker function is to provide gimbal pointing and flight control for a lock-on before launch missile system. The initial inputs to the tracker are the approximate left, right, top and bottom coordinates of the target as seen on the operator's viewing screen. These data points are transformed into the approximate target position and size coordinates within the focal plane of the IR sensor which is providing images to the tracker.

The primary sensor characteristics which have affected the tracker algorithm design are outlined in Table 1. The rather small 32x32 image and overall ½°

TABLE 1

IR Sensor Characteristics

Array Size	32x32 pixels
Field of View	8 mradians (~½°)
Spectral Region	3-4 μm
Lock-on Range	7000 meters
Minimum Tracking Range	100 meters

field of view eased the low-level algorithm computational requirements somewhat, but also introduced some high level tracking complications. The tracker was required to track from a range of greater than seven kilometers down to a minimum tracking distance of 100 meters. At 100 meters the target angular displacement will greatly exceed the ½° sensor field of view while at greater range the target minimum dimensions may be as small as two pixels. The tracking algorithms must adapt to these widely varying flight phases. In addition, the tracker must handle times of partial or full target occlusion, close proximity of stationary target-like objects near the tracked target, and loss of contrast due to dust, fog or smoke at different times during the missile's flight. These prerequisites steered the algorithm design toward an executive controlled multimode approach which was capable of dynamically modifying tracker functions.

The three modes which resulted from the tracker development are: correlation, t-statistic (contrast), and moving target. A functional block diagram of the multimode configuration is shown in Figure 1.

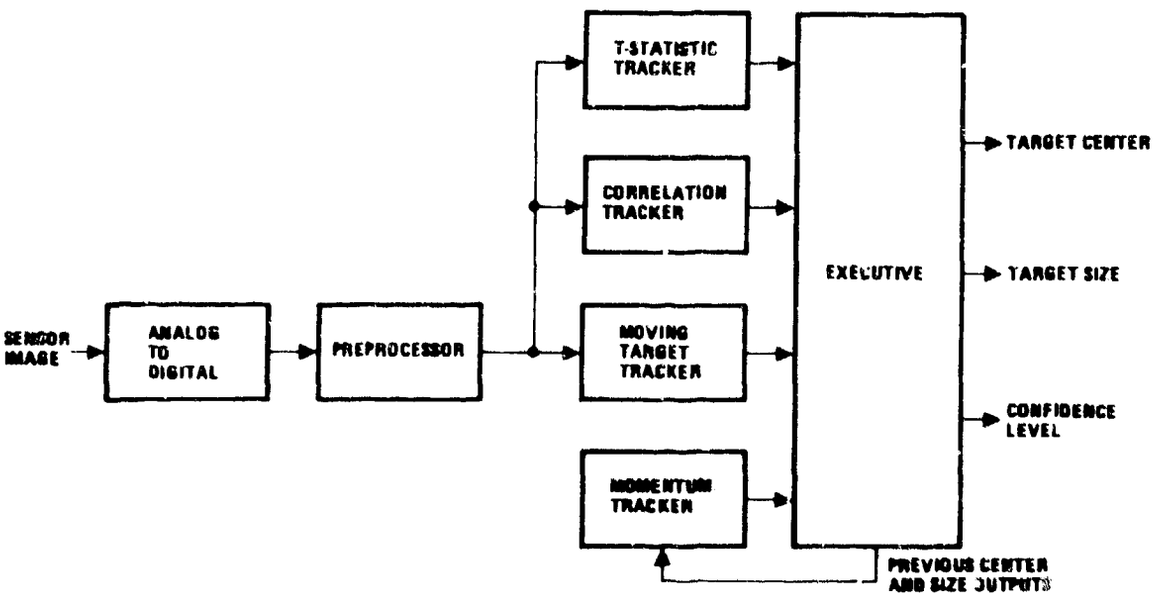


Figure 1. Multi-Mode Tracker Functional Block Diagram

The correlation subtracker utilizes a reference image and the current input image, and performs a modified correlation between the displaced input video and the reference pattern. The correlation or matching is done throughout a window surrounding the last known target position and size obtained from the Executive. The match metric is the mean difference squared between corresponding pixels and is limited to minimize the effects of high impulse type, non-Gaussian noise. An additional function of the correlation subtracker is to evaluate its own performance and to provide an estimate of its confidence level to the Executive by comparing the score at the position of the best match with the scores for the surrounding positions.

The t-statistic subtracker finds an object within an extended window around the previous target location whose pixels have the highest probability of coming from a different distribution than that of the neighboring background pixels. A measure of the intensity difference between the potential target and background areas is provided by the t-statistic.

$$t = \frac{M_t - M_b}{\sqrt{\frac{N_t S_b^2 + N_b S_t^2}{N_t + N_b - 2}\left(\frac{1}{N_t} + \frac{1}{N_b}\right)}}$$

where

M_t = mean of target

M_b = mean of background

S_t = standard deviation of target

S_b = standard deviation of background

N_t = number of pixels in target

N_b = number of pixels in background

A t-statistic is calculated for each potential target position. In contrast with the correlation subtracker, the t-statistic subtracker also determines the target size during each frame time, allowing for natural target growth as well as possible decreases in size due to aspect change. Although theoretically suited for Gaussian distribution of equal variance, the t-statistic is robust in the statistical sense and has proven very effective in its practical application.

The third tracker mode is the moving target subtracker which utilizes delayed video to generate relative motion signals between the background and the target. This is accomplished by means of a background correlator which performs the best match function between a delayed frame and the current frame over areas known not to contain the target. The previous frame is displaced the indicated amount and

the magnitude of the difference between the delayed and current image is computed. This difference image is time integrated with previous such images resulting in an image in which fixed objects appear dark and moving objects have a brightness relative to the shape of their spatial autocorrelation function. A thresholding contrast algorithm is applied to the final image to detect the position and size of moving objects. The degree of contrast, as measured by the chosen threshold's percentile level in the local intensity distribution, serves as the moving target tracker's confidence level returned to the Executive.

The Executive assimilates the outputs from each of the three tracking modes to provide a best estimate of the actual target position and size. In addition, the Executive maintains its own estimate of target position and size based on filtered past position and velocity data. The Executive monitors performance of the three modes using its own estimates and the returned confidence levels to determine if an individual subtracker should be reset. Also, in the terminal mode of missile flight the target size will exceed the sensor field of view. The Executive must detect this condition and switch to a correlation only mode.

General purpose computer simulations, in floating point software, used IR image sequences provided by Night Vision Laboratories for the majority of algorithm development. Extensive testing of the tracking algorithms was also conducted with various degrees of noise added to the sequences to aid in algorithm evaluation. A typical simulation sequence is shown in Figure 2. All floating point computations were later converted to integer arithmetic to verify performance in a fixed word size microprocessor implementation.

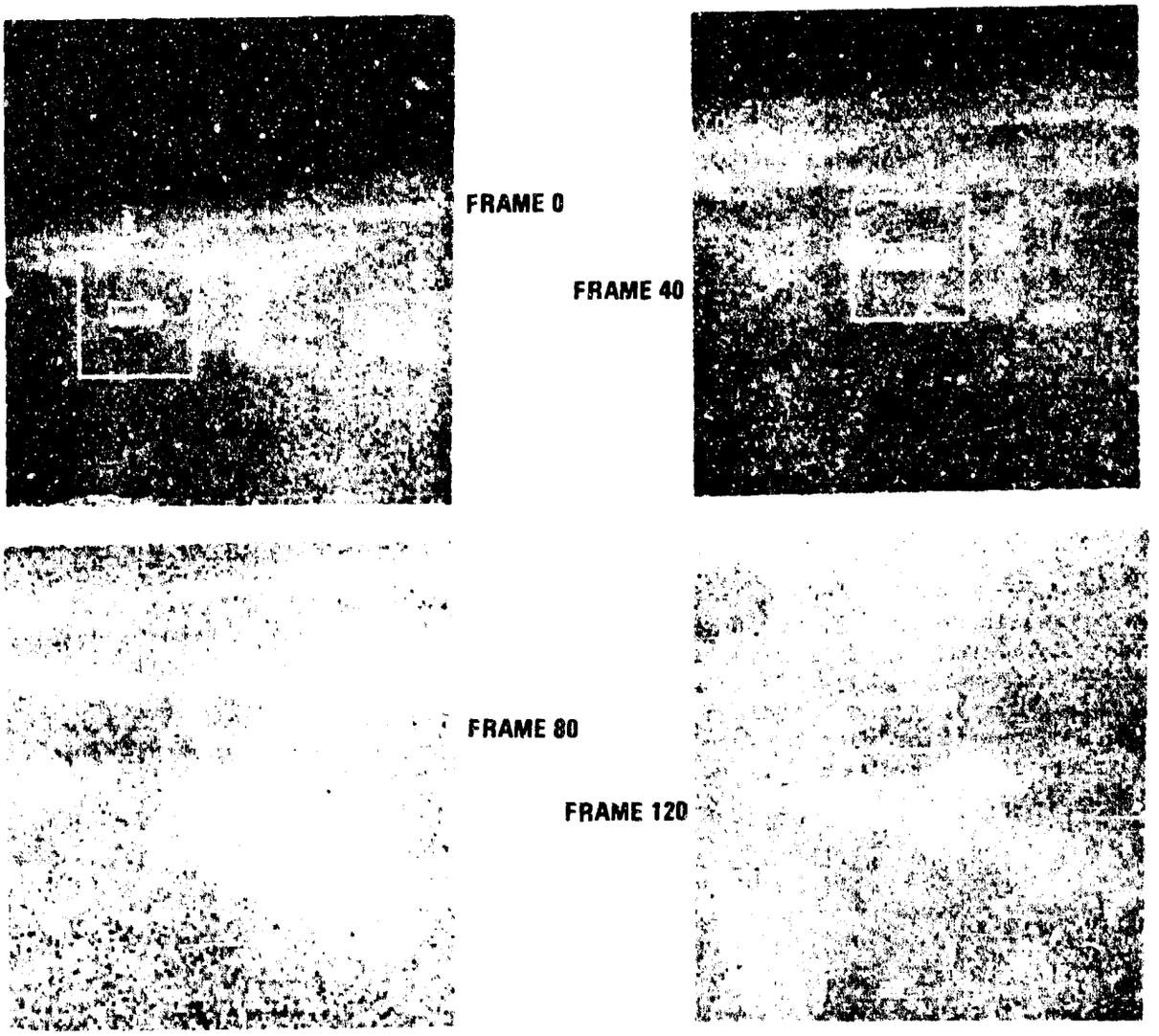


Figure 2. Sample Tracking Sequence (Noise Added)

3. Multi-Microprocessor Architecture and Hardware Design

The multimode tracker algorithms developed to meet AI^2S mission requirements are quite sophisticated and require considerable processing power for real-time implementation. A programmable processor approach was selected for the obvious reasons of flexibility in algorithm refinement and modification, later addition of new functions, and the long term reduction of hardware development and maintenance costs. An implementation study was initiated early in the AI^2S program to review all microprocessor technologies (NMOS, CMOS, I^2L, STTL, and ECL) and formats (8-bit, 16-bit, and bit-slice).

From this review, it became quite apparent that the tracker would have to be implemented in a multiprocessor configuration to execute all algorithms in real time. The widely accepted 8-bit fixed format microprocessor families could be effectively applied to executive and control functions, but could not perform tracker algorithm computations at the required speeds. Newer 16-bit fixed format microprocessors offer some performance gain over the 8-bit versions, but still not enough to implement tracking algorithms. Special support circuits to assist a fixed format microprocessor in frequently executed operations could be used, but would be very special purpose and quite complex to be really effective.

The bit-slice format microprocessor devices offer considerably more signal processing potential. These devices operate at higher clock rates and can be cascaded for required data precision. The instruction format of a slice processor has many more bits directly controlling processor and sequencer logic, in contrast to the limited number of bits in a fixed format processor instruction. This horizontal expansion of the slice processor instruction format provides for direct control of multiple functions enabling concurrent operations which would require multiple steps in a fixed format processor. Each clock cycle in a slice processor is a microinstruction cycle, while the fixed format processor will require several clocks to execute a single instruction. Real-time signal processing capability of the slice devices is about an order of magnitude above the fixed format devices.

The multi-microprocessor architecture developed for tracker implementation, shown in Figure 3, is divided into an Executive control processor and high-speed algorithm and input processors. The Executive is a conventional 8-bit fixed format Z80 microprocessor capable of performing general control and decision making functions, while higher-speed AMD 2900 bit-slice microprocessors are used for actual tracker algorithm computations and input image handling. All control interfaces are through the Executive while the input and algorithm processors interface directly with image data. A common data memory holds current and past image frames, and is shared by the two algorithm processors. A two-phase clocking scheme divides memory cycles between the two processors providing direct access for each.

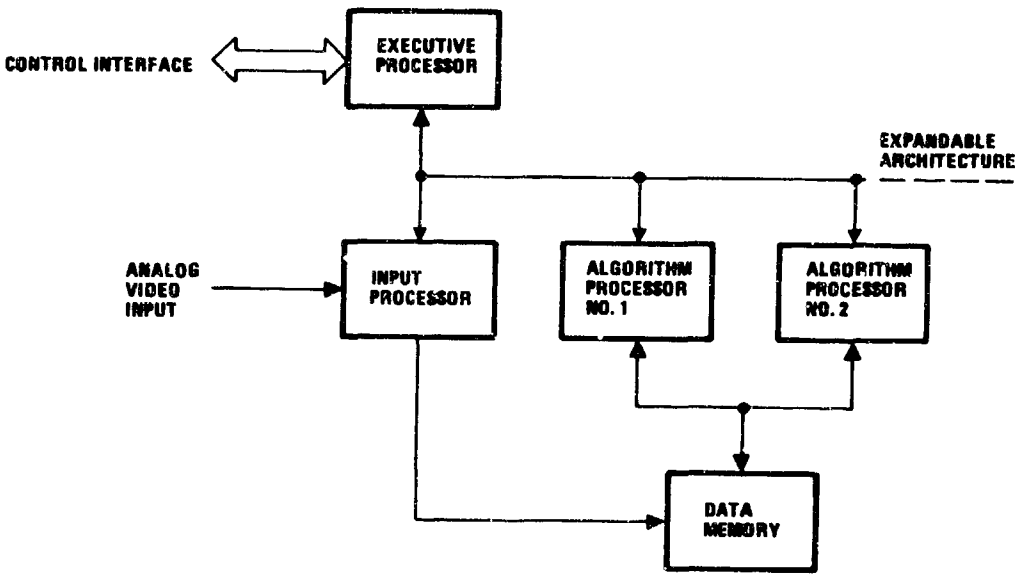


Figure 3. Tracker Multi-Microprocessor Architecture

The Executive processor is interfaced to the three-slice processors through high-speed bidirectional FIFO buffers. This organization enables a higher degree of overlap between the asynchronous processors. The Executive can queue up task assignments for the slice processors in the FIFO interfaces while the slice processors are still executing their previous assignments. A non-buffered interface would slow the high-speed slice processors down to the Executive rate during interface transfers resulting in inefficient utilization. All interfaces to the Executive are through standard Z80 parallel I/O Controllers (PIO's) using port A for bidirectional data transfers and port B for control.

The algorithm processor organization, shown in Figure 4, consists of a program sequence controller, processing logic, data memory and interface control circuits. The input processor is quite similar except for data word size, data memory interfaces, and special processing functions. The program sequence controller consists of an AMD 2910 controller device, a test condition input multiplexer, a PROM memory for program storage, and a pipeline register for instruction buffering. The AMD 2910 is a sophisticated LSI device which is programmed to generate a 12-bit program memory next-instruction-address. The device contains a microprogram counter, five-word-deep LIFO stack, loop counter or address register,

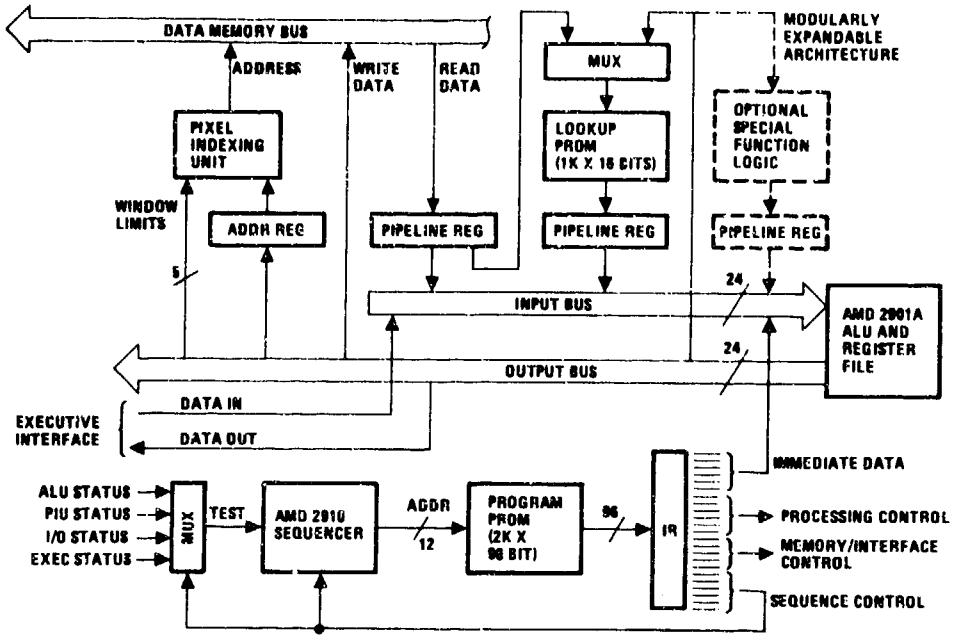


Figure 4. Algorithm Processor Organization

and a multiplexer for next-instruction-address selection. The AMD 2910 can be programmed to execute 16 sequence-control instructions providing sequential access, conditional branching to any location within a 4096 microword range, and subroutine return linkage and looping capability. The last-in first-out (LIFO) stack will accommodate up to five levels of subroutine nesting.

The test multiplexer can be programmed to select one of 16 possible test inputs (various status and interface control signals) for conditional branch instructions. The test polarity can be either true or false, and the address will control the program memory next-microinstruction access. The micro-instruction is stored in a pipeline (instruction) register to enable overlap between instruction execution and access cycles. This pipeline technique reduces maximum path propagation delays, allowing a much faster instruction cycle. The algorithm and input processor instruction formats, shown in Figure 5 are 96-bits wide including 8-bits for expansion. Storage is provided for up to 2k instructions.

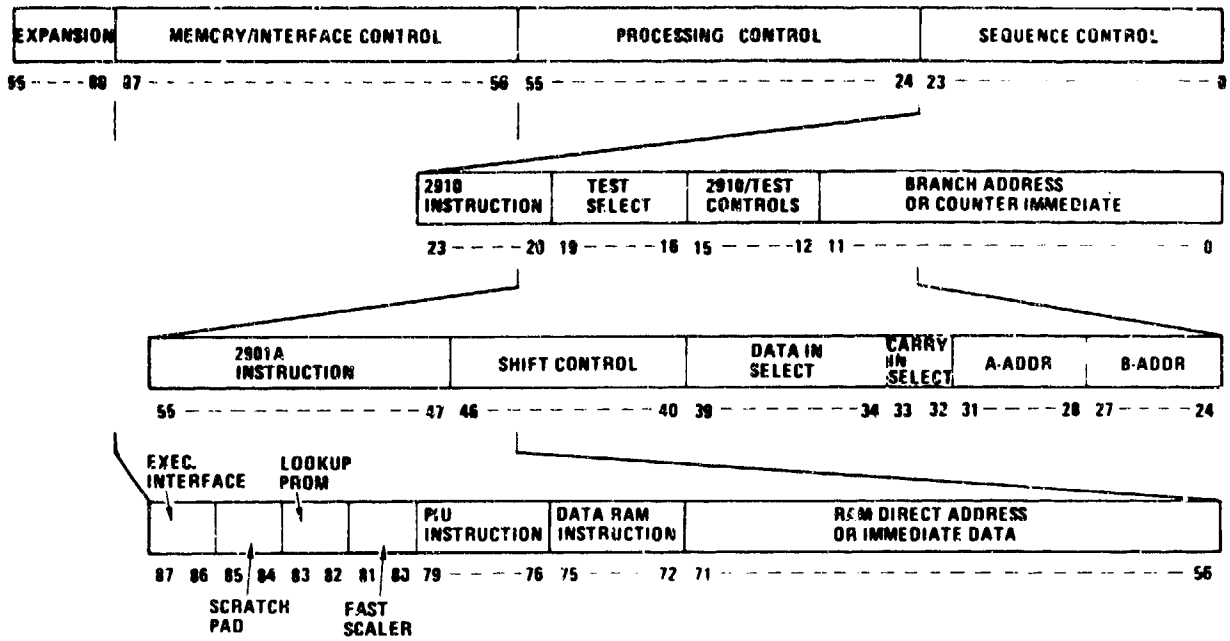


Figure 5. Microcode Format

The processing logic is based on the AMD 2901A microprocessor slice. This device is a 4-bit wide ALU and storage-register slice which can be cascaded for any data-word size. The device contains an 8-function arithmetic logic unit (ALU), 16-word two-port RAM register file with shifter, an additional storage register (Q) with shifter, and associated decoding and multiplexing circuitry. The AMD 2901A can be programmed to select two of five data sources to the ALU, one of eight ALU functions, and data storage in the RAM or Q-register with or without shifting. A complete read from RAM, modify in ALU and shifter, and write back to RAM can be executed in one clock cycle. Data can be enabled onto the output bus from either the ALU or directly from the RAM register file.

The ALU and register file section is assembled from six AMD 2901A slice devices (three in the input processor) with supporting multiplexers to control external inputs on data left or right shifts. The RAM and Q-shifters are interconnected to facilitate double-precision shifts and accumulations. This is particularly useful in accumulate-and-shift algorithms used during multiply or divide operations. Data scaling and power-of-two multiply operations also benefit from this configuration. Other support circuits provide programmed selection of the ALU carry input (for increment, two's complement and round-off operations), and a storage register for ALU status bits.

Additional processing functions can be easily implemented using standard microprocessor support devices. Special-function logic is provided to facilitate key algorithm computations which are not efficiently executed in the basic ALU and register file. A 1k x 16-bit lookup table PROM is provided as a special function in the two algorithm processors. The PROM is divided into two 512-word sections, each addressed by a 9-bit data word. One section of this PROM is used for a table of squares to support correlation and t-statistic subtracker computations. The PROM output is loaded into a pipeline register to allow overlap between access and other ALU operations. The PROM address input can be either from the ALU output bus or from the data memory output register, providing another level of pipeline overlap. This configuration along with data memory address-control logic, is essential for correlation and t-statistic subtracker real-time computations. The least mean square correlator operation of taking the difference between two corresponding pixels, squaring and accumulating can be programmed into a three-instruction loop. The basic mean and variance operations of accumulating pixel amplitudes and their squares can be programmed into a two-instruction loop.

A power-of-two fast scaling circuit is provided as a special function in the input processor. A multiplexer array can be programmed for scale factors of 2^{-1}, 2^{-2}, 2^{-3}, or 2^{-4}. This is an essential function to enable real-time image input gain correction.

Input image picture elements (pixels) are stored in the common data memory. Both algorithm processors share memory cycles in a two-phase clocking scheme with no reduction in either processor's speed. The input processor has a lower priority due to its less frequent need for memory cycles, and therefore, operates on a cycle-available basis. The input processor has a smaller dedicated working RAM (4k word), and only requires data memory cycles when new pixels are ready to be stored. The two algorithm processors each have data memory address and data output pipeline registers to provide for maximum overlap.

A special data memory pixel-indexing-unit (PIU) is provided to facilitate algorithm stepping through rectangular target areas in the image field of view. The PIU maps two-dimensional target pixel positions to data memory addresses. The PIU can be programmed to step through a desired portion of each image line (i.e., from left boundary to right boundary), and to generate an end-of-line flag at the end of each line. The algorithm processor software can then enter a tight program loop to load image pixels across each line, breaking out of the loop only at the end of each line to index the column pointer. The PIU can also be used in an automatic mode where both row and column pointers are automatically indexed, and an end-of-column flag is generated at the end of the last line. The PIU can also be programmed to generate data memory addresses for target positions displaced from a reference position in correlation computations.

The tracker multi-microprocessor architecture has been reduced to a brassboard assembly for field test and evaluation of the AI^{2}S system. Conservative design practices have provided for reserve program and data memory, data precision and general flexibility to allow for system growth. The brassboard hardware assembly, shown in Figure 6, is packaged in a 7-inch high standard rack-mountable enclosure. The logic is assembled on three wirewrap planes functionally partitioned to simplify modular growth. Standard connections for all Executive interfaces allow for the interchanging or addition of slice processors. A front panel includes indicator lamps to display tracker functional status (active modes) and failure alarms. Repackaging studies have established the feasibility of meeting tactical missile space and power requirements using LSI and hybrid techniques.

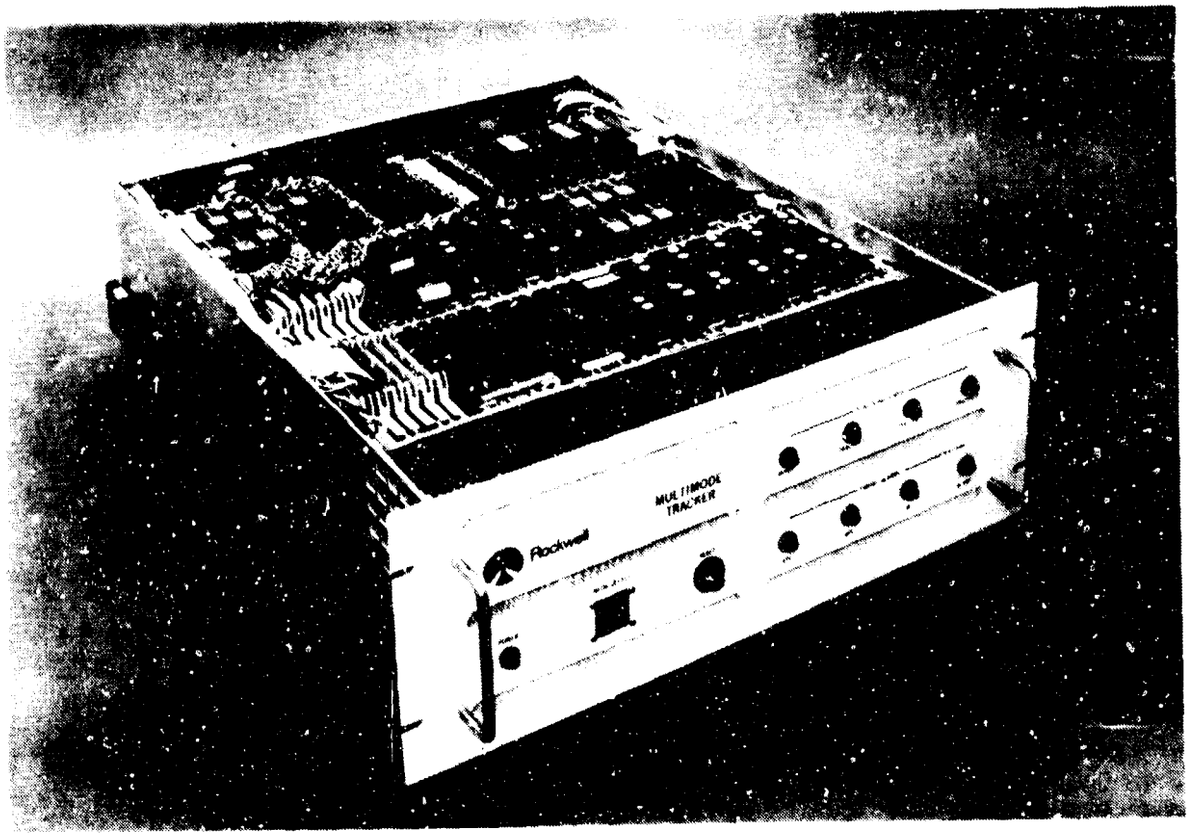


Figure 6. Tracker Brassboard Assembly

4. Firmware Implementation

The AI^{2}S tracker simulation produced an algorithm which was naturally suited to parallel processing. Ideally, an individual processor would be dedicated to each of the three modes - t-statistic, correlation and moving target. An Executive processor could then accept inputs from each mode and determine the final tracker output weighted by the confidence levels that the modes provide. An additional processor would handle image input and any global preprocessing that is necessary. There are, however, several factors which make this less than an optimal design and, in fact, incapable of performing some of the required calculations.

First, the three tracker modes all make different CPU demands upon the processor and so require different amounts of CPU time. The calculation time is also a complex function of the target size. Thus, at one stage of the tracker operation a particular mode may require only one-fourth of the computation time available from a single dedicated processor during an image frame. Three-fourths of the processor time would be unused. Also, the situation frequently arises when a particular mode requires the capabilities of more than one processor during a frame time.

For these reasons, the AI^{2}S tracker algorithm processors were not viewed as being dedicated to one particular mode, but rather were seen as resources on which the Executive could draw for extended, high-speed calculations. Each processor must be capable of performing any requested task on file. Thus, with the exception of data memory biases, the algorithm processor programs are identical. With this configuration, the Executive might request processor number 1 to perform a search for the new target position of one size, and at the same time request processor number 2 to perform the search using another projected size. In this way, the Executive can devote all of the available processor time to one particular type calculation if the situation, e.g., loss of track, merits it. Also, with this viewpoint, one is unrestricted in specifying the actual number of high-speed algorithm processors that are provided in hardware. Of course, a sophisticated Executive structure is required to handle the multiple, parallel processing tasks. What has resulted is, in fact, a software Executive program which has many characteristics of general purpose computer operating systems.

The Z80 Executive program's function is to divide the tracker operations into distinct computing tasks, allocate these tasks to available algorithm processors, accept the task results, and assimilate the results to provide flight control commands. To organize the tasks that must be completed during the 1/60th of a second frame time and to monitor their status, the Executive maintains a task table shown in Figure 7. The task table is formed by calls to a task manager which queues tasks into the table. Each time that a task is queued, the task manager program checks to see if any algorithm processor is

idle or if its input first-in-first-out (FIFO) memory buffer is empty. If so, the task message consisting of from 3 to 16 bytes is loaded into the FIFO and a "task ready" control bit is set. When the bit slice processor is in its idle loop, it will monitor the "task ready" bit, read the task from the FIFO when it is raised, execute the designated task, place an output message consisting of 2 to 16 bytes into the output FIFO, and signal the Executive by setting a "task done" bit. The setting of this bit interrupts the Executive. Processing of the output data continues in a completion routine which was designated for the task at queing time. The status of any task handed to the task manager is also contained in the task table and may be: (1) ready to start in an available processor, (2) started in a processor but not yet complete, or (3) completed. Note that a task which has been started may actually be in the input FIFO awaiting bit slice processor attention, or may currently be in the execution phase, or may be in the output FIFO awaiting the Z80 Executive's attention.

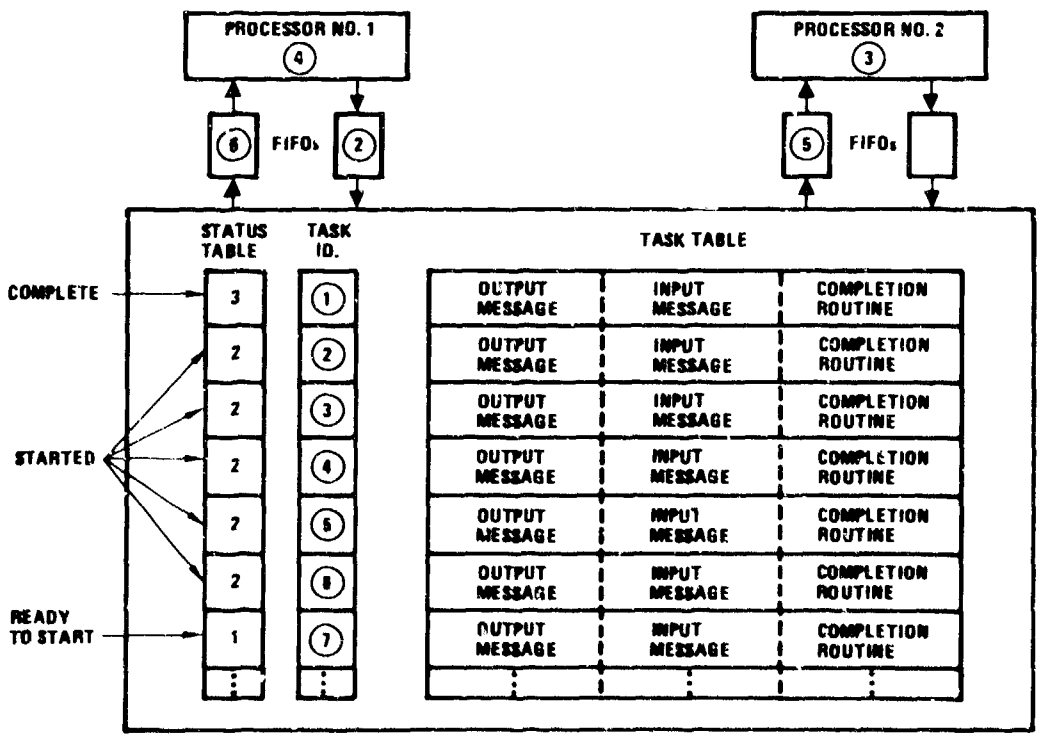


Figure 7. Executive Multi-Tasking Table

The format of a typical task is illustrated in Figure 8. A byte count containing the number of bytes remaining in the message is followed by a task ID, program number and from 0 to 13 data bytes. The task ID identifies the task and its location in the task table. The task ID is echoed by the algorithm processor when he supplies task results to the Executive. A program number indicates which task that the algorithm processor is to perform, e.g., zero an image slot, correlate two image sections, etc.

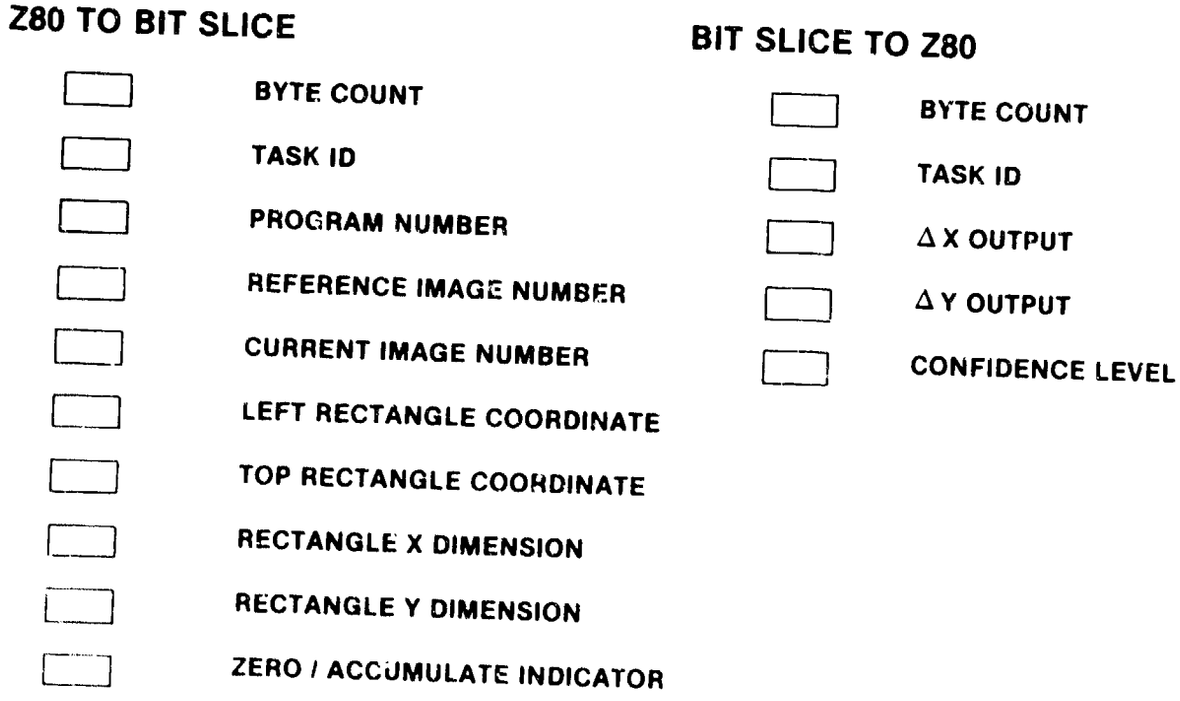


Figure 8. Typical Task Message

As mentioned above, the Z80 Executive is interrupted by the bit slice processor when it completes a task. After the task manager has initiated any further tasks which have been queued, and has executed the completion routine, the Executive returns to queue other tasks and then begins his own target position estimates. When all tasks have been completed and the Executive has assimilated the output, he enters an idle loop awaiting the next image. One of the tasks that the Executive queued to the input processor indicated where the next image was to be located in RAM as well as if bias and/or gain parameters should be updated. The notification that a new image is in and ready to be processed is indicated in the same way that any other "task done" signal is made, i.e., the Executive is interrupted. Since the image source is externally timed, the tracker firmware was, in effect, independent of frame time. This proved useful during hardware and software checkout when images were provided at a reduced frame rate.

A Data General Nova 840 minicomputer was used to simulate the imaging sensor interfaces during tracker development and test. The same images used in the earlier Fortran level algorithm simulations were used for firmware checkout resulting in a significant time-saving. The Nova 840 also provided complete peripheral support to two microprocessor development systems and operator control for test runs. The firmware development and tracker test configuration is shown in Figure 9.

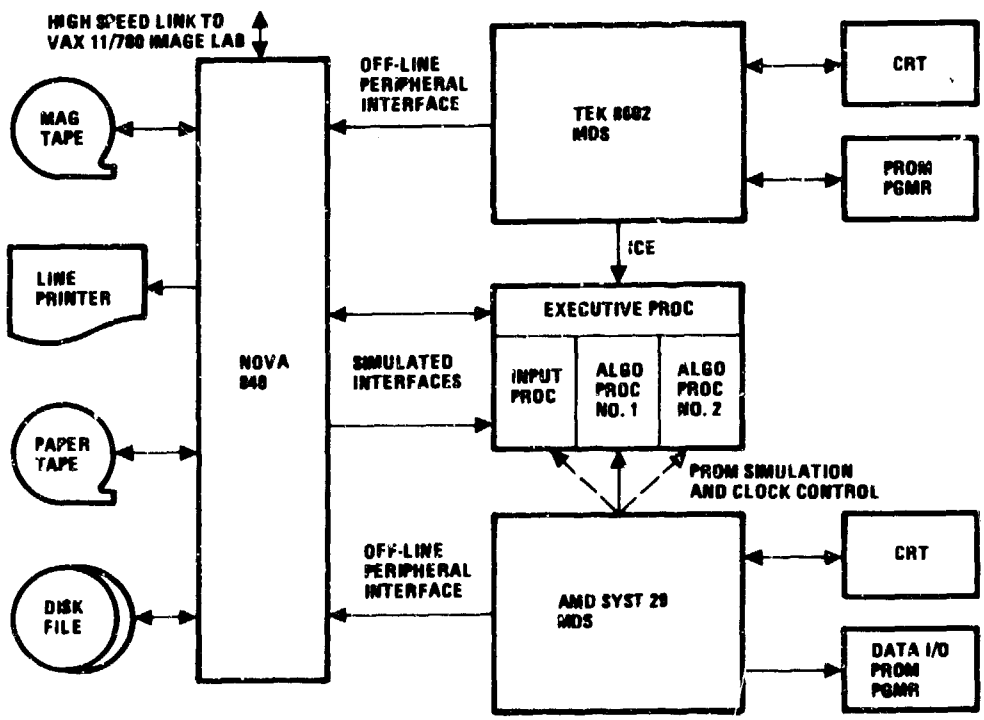


Figure 9. Firmware Development/Test Configuration

A Tektronix 8002 microprocessor development system was used for Z80 Executive firmware development, and a Advanced Micro Devices System 29 was used for algorithm and input processor firmware development. Both systems were used extensively for firmware development and integration into the tracker multi-microprocessor configuration.

5. Conclusions

The AI^2S multimode tracker has passed initial real target ground tests and will soon be flight-tested. The capabilities of the tracker hardware, however, have only been partially utilized. The tracker software implements a lock-on before launch algorithm which in current research is now serving as the tail-end processor for automatic target detection and acquisition algorithms. The modular hardware and software design will allow the introduction of additional number-crunching algorithm processors to meet the demands of detection, feature extraction and classification in addition to tracking of multiple targets with larger detector arrays. In addition to serving as an invaluable real-time algorithm development tool, LSI and hybrid packaging techniques promise a very small volume, low power, and inexpensive implementation of the multimode tracker hardware.

Paper No. IIA-4, Presented at the Workshop on Imaging Trackers and Autonomous Acquisition Applications for Missile Guidance, 19-20 November 1979, Redstone Arsenal, Alabama.

HYBRID CORRELATION ALGORITHMS--A BRIDGE BETWEEN FEATURE MATCHING AND IMAGE CORRELATION

Joseph A. Ratkovic
The Rand Corporation
Santa Monica, CA 90406

ABSTRACT

Up to the present time there have been two basic classes of map matching algorithms--those based on feature matching techniques and those based on image correlation. This paper describes a new class of hybrid correlation algorithms which incorporate features as an integral part of the matching process. These algorithms can be implemented such that it is not necessary to extract features from the sensed image. This paper concludes by showing the domains in which each class of matching algorithm (feature matching, image correlation, and hybrid algorithm) is most appropriate.

INTRODUCTION

The map matching problem has been in search of an "optimal universal" matching algorithm since its inception. Because of difficulty in (1) defining a performance criteria for both accuracy and probability of correct match, and (2) in knowing a priori the distributions associated with all map errors, most researchers have resorted to the use of "ad hoc" algorithms. These have generally been divided into two classes--feature matching and correlation.

The image matching problem, as shown in Fig. 1, is a two-phase problem. In phase 1, the acquisition phase, one is concerned with locating, somewhat grossly, the area in which the match point is centered and avoiding false matches. In phase 2, one is concerned with refining the accuracy with which the match location can be determined. In general, no one algorithm can possibly be suited for solving both the acquisition and accuracy problems, and it is probably necessary to develop algorithms separately for each phase of the problem.

The overall matching problem, shown in Fig. 2, involves four major components: (1) error sources, (2) the scene, (3) preprocessing, and (4) matching algorithms. Before discussing algorithms and describing some algorithm techniques, it is necessary (to provide background for the algorithm discussion which follows) to briefly describe scenes, errors, and preprocessing.

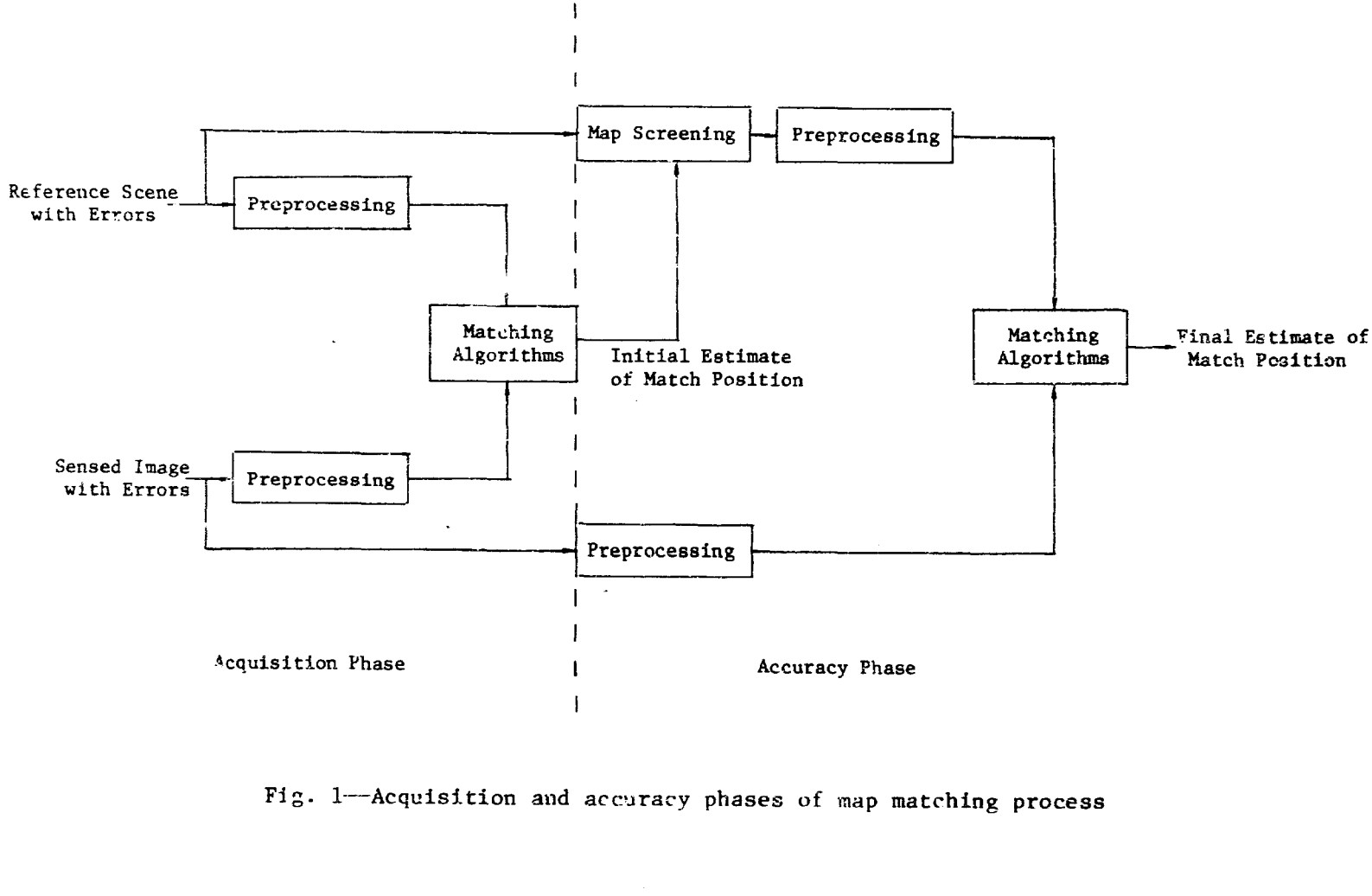


Fig. 1—Acquisition and accuracy phases of map matching process

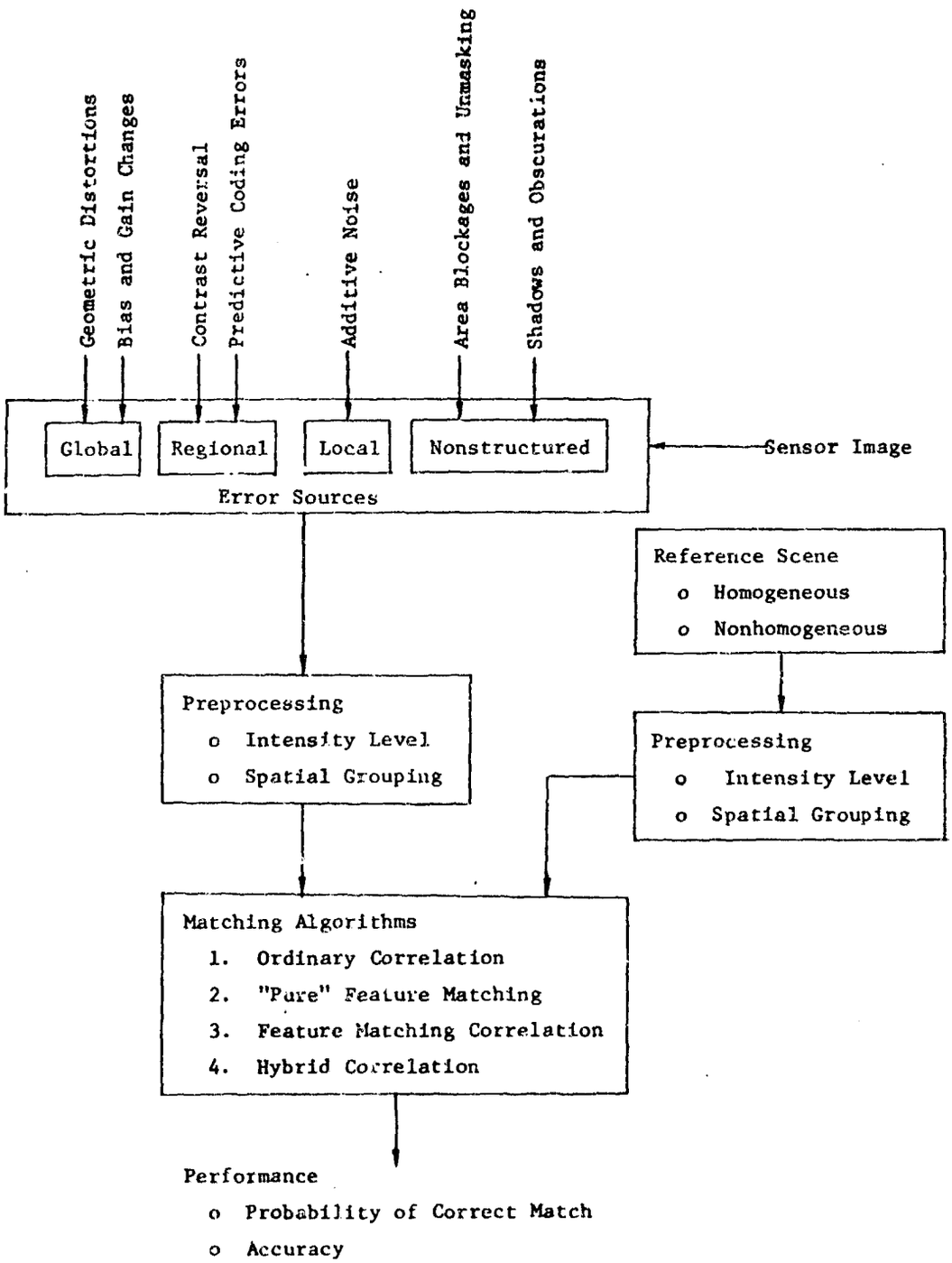


Fig. 2—Generic overview of map matching process

The Scene--Its Composition

The scene is the most complex component of the map matching problem and the most difficult to model. In the discussion that follows we shall examine the question of "scene composition" (relative to both a visual and statistical representation of a scene), and methods for decomposing the scene.

Scenes can be described in the visual domain by the eyeball process as being composed of a set of features. Let us consider as an illustrative example the simple scene shown in Fig. 3. Here, for example, the window feature consists of a set of four panes enclosed by a frame.

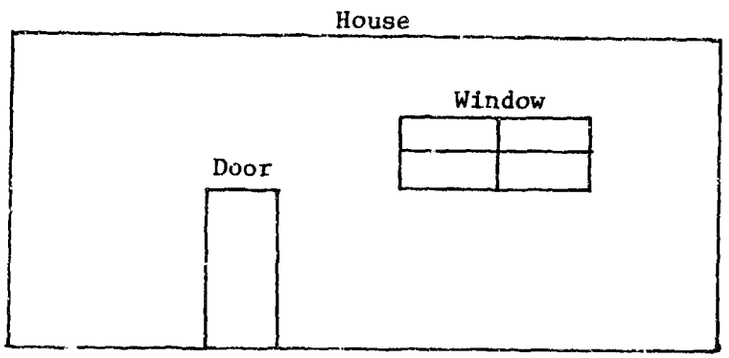


Fig. 3—Example of features consisting of a set of homogeneous regions

In dealing with actual sensor data, picture elements (pixels) are described by a set of intensity values, as indicated in the agricultural scene of Fig. 4. In dealing with intensity values, there are regions in the scene which can be considered analogous to features in the visual domain. These are homogeneous regions within the scene. We shall define a homogeneous region to be a set of spatially connected pixels or elements which possess the statistical property of at least first-order* stationarity and possibly second-order stationarity** and will assume that homogeneous regions are equivalent to features (as a feature can be defined by a single homogeneous region or a set of homogeneous regions).

In Fig. 4 we have identified four homogeneous regions and tagged each pixel (indicated at the bottom portion of the figure) as belonging to one of the four regions. Examining each region we see that the

---

*Mean intensity level constant over the region.

**Mean and variance constant and the autocorrelation independent of position.

REGION 12 USED FOR REFERENCE SCENE

```
20. 21. 22. 21. 22. 22. 22. 22. 22. 21. 17. 15. 15. 15. 16. 15. 15. 20. 24. 25.
20. 20. 21. 21. 21. 21. 21. 21. 21. 21. 18. 14. 13. 13. 13. 14. 14. 17. 22. 23.
22. 20. 21. 22. 22. 22. 21. 21. 22. 21. 20. 17. 15. 15. 15. 14. 14. 16. 21. 23.
21. 21. 21. 21. 21. 22. 21. 21. 22. 21. 22. 18. 14. 15. 14. 14. 14. 15. 21. 23.
20. 20. 22. 22. 21. 21. 21. 22. 22. 22. 21. 18. 14. 14. 14. 13. 13. 14. 21. 24.
21. 21. 21. 21. 21. 21. 21. 22. 21. 21. 21. 18. 14. 14. 14. 13. 13. 15. 20. 24.
21. 21. 22. 22. 22. 22. 22. 23. 23. 22. 21. 18. 15. 15. 14. 14. 14. 16. 18. 19.
20. 20. 21. 21. 21. 21. 21. 21. 21. 20. 20. 18. 14. 14. 15. 14. 14. 14. 15. 16.
20. 20. 21. 22. 22. 21. 21. 21. 21. 20. 21. 21. 19. 14. 14. 14. 14. 14. 15. 16.
21. 20. 19. 21. 21. 21. 22. 22. 22. 22. 22. 22. 20. 14. 13. 14. 14. 14. 15. 15.
15. 15. 17. 22. 22. 23. 23. 22. 21. 21. 23. 24. 22. 14. 12. 13. 14. 14. 14. 15.
14. 14. 15. 21. 22. 23. 23. 22. 22. 22. 23. 24. 22. 15. 12. 13. 13. 14. 14. 14.
15. 15. 17. 22. 24. 24. 23. 22. 22. 23. 24. 25. 23. 16. 13. 13. 15. 14. 15. 16.
14. 14. 15. 20. 22. 22. 22. 21. 21. 22. 23. 23. 22. 16. 13. 13. 14. 14. 14. 15.
15. 14. 14. 19. 21. 23. 21. 22. 21. 21. 22. 23. 23. 22. 15. 13. 13. 15. 15. 15.
15. 14. 14. 15. 21. 23. 22. 23. 22. 22. 23. 24. 23. 23. 16. 13. 13. 14. 15. 15.
14. 14. 14. 14. 20. 23. 23. 23. 22. 22. 22. 22. 21. 21. 17. 14. 13. 14. 14. 14.
14. 14. 14. 14. 19. 23. 23. 23. 22. 22. 22. 23. 21. 19. 16. 15. 16. 19. 21. 22.
15. 15. 16. 18. 20. 23. 24. 23. 21. 19. 21. 22. 23. 24. 23. 23. 23. 24. 23. 25.
17. 17. 17. 17. 19. 19. 18. 17. 18. 19. 22. 24. 25. 23. 23. 23. 23. 23. 24. 24.
```

REGION NUMBERS FOR REFERENCE SCENE

```
2 2 2 2 2 2 2 2 2 2 3 3 3 3 3 3 3 4 4 4
2 2 2 2 2 2 2 2 2 2 2 3 3 3 3 3 3 3 4 4
2 2 2 2 2 2 2 2 2 2 2 3 3 3 3 3 3 3 4 4
2 2 2 2 2 2 2 2 2 2 2 2 3 3 3 3 3 3 4 4
2 2 2 2 2 2 2 2 2 2 2 2 3 3 3 3 3 3 4 4
2 2 2 2 2 2 2 2 2 2 2 2 3 3 3 2 3 3 4 4
2 2 2 2 2 2 2 2 2 2 2 2 3 3 3 3 3 3 4 4
2 2 2 2 2 2 2 2 2 2 2 2 3 3 3 3 3 3 3 3
2 2 2 2 2 2 2 2 2 2 2 2 2 3 3 3 3 3 3 3
2 2 2 2 2 2 2 2 2 2 2 2 2 3 3 3 3 3 3 3
1 1 1 2 2 2 2 2 2 2 2 2 2 3 3 3 3 3 3 3
1 1 1 2 2 2 2 2 2 2 2 2 2 3 3 3 3 3 3 3
1 1 1 2 2 2 2 2 2 2 2 2 2 3 3 3 3 3 3 3
1 1 1 2 2 2 2 2 2 2 2 2 2 2 3 3 3 3 3 3
1 1 1 1 2 2 2 2 2 2 2 2 2 2 3 3 3 3 3 3
1 1 1 1 2 2 2 2 2 2 2 2 2 2 3 3 3 3 3 3
1 1 1 1 2 2 2 2 2 2 2 2 2 2 3 3 3 3 3 3
1 1 1 1 2 2 2 2 2 2 2 2 2 2 3 3 3 2 2 2
1 1 1 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
1 1 1 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
```

Fig. 4--4 km · 4 km agricultural scene

intensity value of a given pixel does not vary significantly from the mean value and that there are distinct boundaries (defined by differences in the mean intensity level) between regions.

Thus far we have shown that scenes are composed of homogeneous regions which may be considered equivalent to features. From a physical standpoint homogeneous regions are areas in which the signature (emissivity for visual and IR, reflectivity for radar, and altitude for terrain contours) is expected to remain fairly uniform, e.g., a grassy field in which all the elements in the region are expected to have the same mean value but this mean value may change as a function of time.

Having established that a scene is composed of homogeneous regions, is there a further subdivision by which we can characterize homogeneous regions? Returning to Fig. 4 we see that there are small variations in the intensity level within a homogeneous region. Some of this variation can be attributed to sensor noise but, neglecting this possibility for the moment, one can consider the variation to be due to some perturbation in the signature of the region. For instance, one can consider the grassy field not to be uniform, but instead to have a few fallen tree trunks and shrubs dispersed within it. If the ground resolution of the sensor is of the same magnitude as the size of the shrubs and tree trunks, then we would expect variations in the intensity level of the grassy region due to these objects, presuming, of course, that the signature of the objects was different from the grass at the wavelength of the sensor. Thus, we can further categorize a homogeneous region in the physical domain by the number of objects which contribute to a signature variation, and in the statistical domain by the number of statistically independent elements* which comprise the region.

The "scene resolution" provides a useful concept in analyzing the statistical variation of a region. We shall define the "scene resolution" as the number of sensor resolution elements or pixels required to make up one independent element in the scene. If there are N pixels within a homogeneous region and N_I independent scene elements ($N_I \leq N$) then the average "scene resolution" for the region would be given by N/N_I. Returning to the grassy field example, if

*Statistical independence is different from the property of homogeneity. For instance, one can generate a completely random map from a single distribution which will have the property of homogeneity but will also have all the elements independent. One can imagine a homogeneous region containing a number of independent elements, e.g., a desert area in which the shrub patterns (depending on resolution) constitute the independent elements. It is a difficult procedure to test for and locate independent elements in a scene. Reference 1 describes a short-cut method for estimating this parameter by working backwards from the statistics of the correlation surface and assuming a homogeneous scene with all elements being independent.

the field were completely uniform with no variations in intensity level, then it could be considered to contain only one independent scene element and the scene resolution would be given by the total number of sensor elements in the region, N. In this particular case one could not expect to resolve any features within the region due to the uniformity of the region; thus the scene resolution equals the size of the region (in terms of sensor elements). If, on the other hand, there had been a number of objects (with different signatures) such as tree trunks and shrubs within the grassy region, then we would expect the region to be statistically represented by several independent scene elements. It should be noted also that if the resolution of the sensor were to increase to the point that dimensions of objects within the grassy field were to cover several sensor resolution elements, then these objects would be considered homogeneous regions in themselves. If the resolution were to increase further, then areas within the objects (e.g., moss on the fallen tree trunks) would eventually become homogeneous regions and the process of identifying homogeneous regions could continue ad infinitum.

At this point we see that for a given sensor resolution it is possible to statistically describe a scene as being composed of a set of homogeneous regions with each region being described by a number of statistically independent elements.

Structuring the Errors

There are a number of error sources that affect the performance of the system. It would be desirable to lump these errors into generic categories in discussing system performance rather than treating each error source separately. Such a generic categorization should possess the following properties:

1. The error categories should be mutually exclusive.

2. They should be comprehensive.

3. There should be a positive relationship between the category and a specific preprocessing technique or correlation algorithm to accommodate all errors in that category.

Based on the types of errors that occur in the map matching process and the statistical description of the scene, the following generic categories of errors are proposed:

1. Global Errors--those errors which uniformly affect the intensity level of all scene elements equally. In this category the following errors would generally fit:

    - geometric distortions
    - bias and gain changes

2. Regional Errors--those errors where the change in the intensity levels occurs uniformly only within homogeneous regions or features within the scene. Examples would be:

    - region level shifts (contract reversals)
    - predictive coding errors

3. Local Errors--in this situation the errors are expected to affect each pixel or grouping of pixels (contained within an interpixel correlation length) independently. The primary example of this error source is additive noise.

4. Nonstructured Errors--this is a rather catchall category designed to fit those errors whose effect on the scene cannot be described as being global, regional, or local (e.g., a cloud cover over the target area casts a ground shadow which changes the signature in a nonstructured manner).

Although some errors may sometimes fit into more than one category, this generic categorization will normally accommodate all error sources as well as provide a convenient means of establishing guidelines for algorithms and preprocessing selection.

Preprocessing

The preprocessing of sensor imagery consists of either changing the intensity levels through the image or segmenting the scene spatially into groups of pixels. The intensity level preprocessing is designed to compensate for any biases or gain changes in the system; whereas, spatially grouping of elements is designed to accommodate geometric errors.

In general, preprocessing is designed to accommodate global errors that occur in the scene and which, by definition, effect all scene elements equally. Thus global errors such as gain changes and bias errors are handled by normalizing the intensity level and by zero meaning the data, respectively. As discussed previously, geometric errors also are global in nature and reduce the degree of congruence between sensed image and reference image. In order to reduce the effect on system performance, geometric errors always force one to work with smaller map sizes and, depending on the nature of the distortion (in azimuth and elevation), may also force one to shape the window of the sensed image or to search for a rotation or scale error. Thus, to accommodate this type of error, it is necessary at a minimum to spatially group the sensor map elements into a single (or number of) smaller map(s). If distortions are uneven in azimuth and elevation it will also be necessary to spatially group the elements so that the appropriate window shape may be obtained.

MATCHING ALGORITHMS

The matching algorithm is only one part of the overall matching process, as indicated in Fig. 5. To begin with, there are a number of system parameters which can be chosen to lessen or worsen the severity of the errors on system performance. These include the sensor orientation, resolution and wavelength, the reference map preparation, and the flight geometry of the vehicle. There are, as indicated in the figure, separate processes for accommodating each of the error sources. Global errors (e.g., geometric distortions, gain changes, etc.) are accommodated in the preprocessing by either reducing and shaping the map size or by normalizing the intensity level of the sensed image. They can also be accommodated by searching in the matching algorithm for rotation and/or scale factor errors. The scene composition problem involves checking to insure that the reference map contains a sufficient amount of independent information and that there are no "scene redundancy" problems within the reference map boundaries.

The algorithm itself is primarily designed to accommodate regional and local errors with nonstructured errors being more difficult to foresee and accommodate. The basic matching algorithm for accommodating regional and local errors can be categorized as belonging to a feature matching or image correlation class of algorithms. It should be noted that none of these algorithms have been mathematically derived to maximize system performance (probability of correct match or accuracy) and, therefore, must be considered in a sense to be "ad hoc."

It is first necessary for the "feature matching" procedure to extract the features from the scene. The first part of the feature extraction process involves locating the edges or boundaries of features. Thus, the scene can be reduced to a set of lines which are the boundaries of the feature. Next the line intersection points are located. In general, the number of lines emanating from each vertex is retained and used as part of the weighting criteria in the feature matching algorithms.

In image correlation there are two basic types of algorithms utilized--those which emphasize the degree of similarity between scenes such as the product, and those which emphasize differences between scenes such as the difference squared and MAD* algorithm.

The standard correlation process works on the gross characteristics of the scene and all preprocessing is done globally (i.e., the mean level when subtracted out is zero-meaned over the entire scene, and similarly when the scene is normalized by the variance, this is

---

*Mean absolute difference.

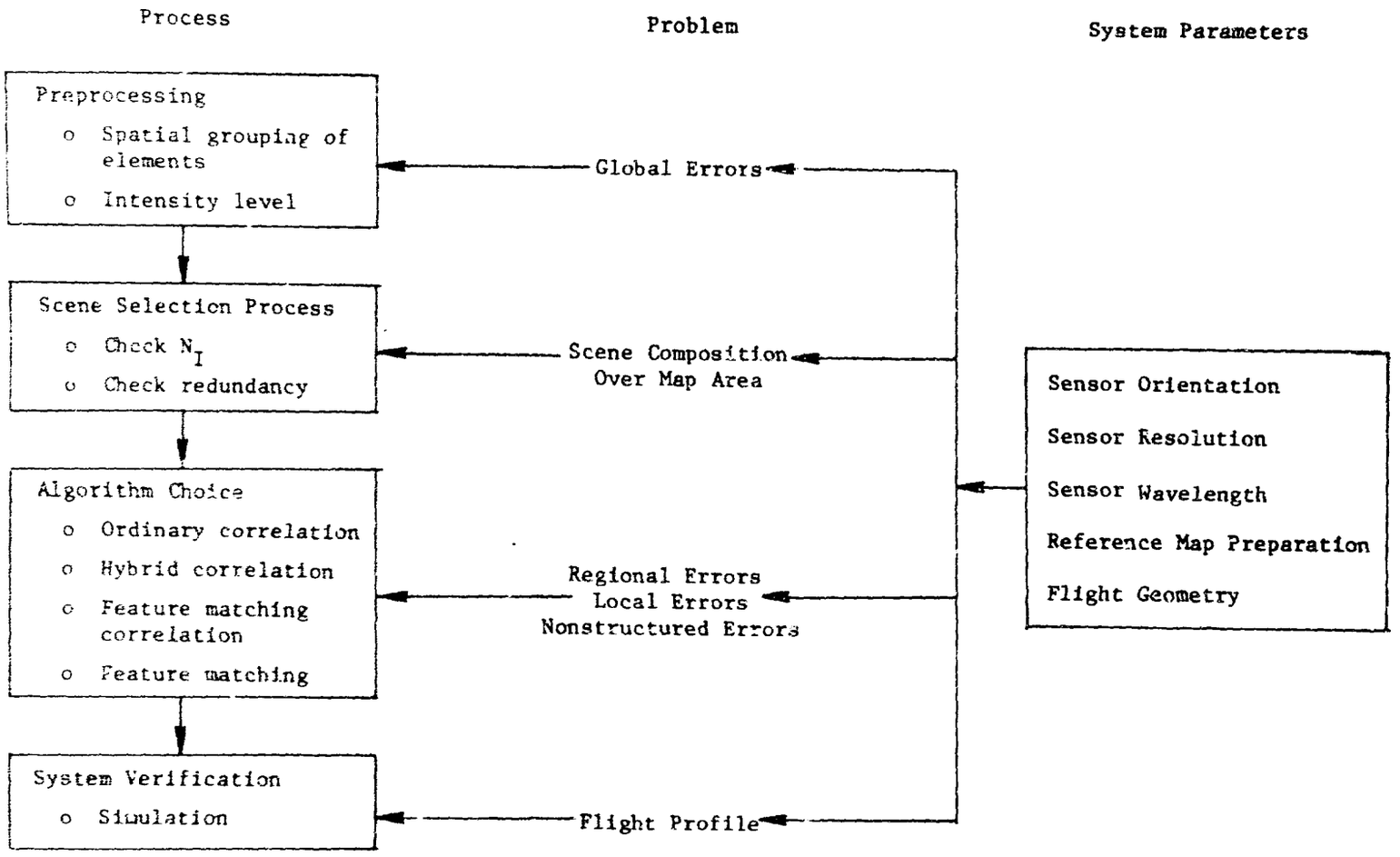


Fig. 5 Acquisition system design

done over the entire scene). In a sense the usual correlation process is designed to work on a homogeneous scene. There are two basic variations to the standard or usual correlation algorithm which are more specifically tailored to nonhomogeneous scenes and the errors associated with them. It should be noted that these variations, in the absence of nonhomogeneity in the scene, reduce to the usual correlation process. We shall denote these variations that deal with scene nonhomogeneities as (1) feature matching, and (2) hybrid algorithms.

One could introduce a feature matching algorithm into the correlation process by breaking up separately the sensor and reference maps into homogeneous subareas. Each of these maps would then consist of a set of homogeneous regions and all processing (rather than being on a global scale) would be performed separately on each homogeneous subregion. Thus, when maps are zero-meaned and normalized, the local mean and variance in each subregion is computed and used to perform the normalization.

After processing both the reference and sensor map on the basis of homogeneous regions, a standard correlation algorithm can be used to determine the position of match between the two maps. The major generic difference between this feature matching correlation algorithm and the "pure" feature matching algorithm (employing pattern recognition techniques) is the weighting given to homogeneous regions. In "pure" pattern recognition algorithms, edges are first extracted and used to identify line intersection points. These line intersection points or vertices then form the primary basis for matching two scenes. In a sense (since edges can be considered the boundaries of homogeneous regions, and vertices are formed by the intersection of edges) a pure feature, or pattern matching algorithm weight all homogeneous regions equally, whereas in the feature matching correlation algorithm, each homogeneous region would receive a weighting proportional to its size (measured in terms of the number of independent elements contained within). In summary then "pure feature matching algorithms can be viewed as being different from feature matching correlation in that different weights are assigned to the various homogeneous regions.

There is another adaptation of the standard correlation algorithm what has been developed at Rand which one can implement to accommodate homogeneous regions. We shall refer to this as a hybrid algorithm which processes only the reference scene into homogeneous regions. The principal idea here is that every position of comparison between the two images is assumed to be the correct one. Thus at each displacement position or comparison point the sensor scene is segmented identically as its counterpart reference map.* At the position at which the two maps correctly match the sensor scene will then be segmented almost perfectly, enhancing the match, and at all other positions the sensor map segmentation will essentially look like noise.

*For each displacement position the matching process consists of correlating each homogeneous region of the reference map and segmented sensor image separately, and combining additively the correlation in each individual region.

The objective of this correlation method is to avoid the errors associated with extracting homogeneous regions or features from the sensor image, and the additional processing requirements placed on the system. If the image is noisy, normal edge operators have difficulty in performing their feature extraction task and, as a compromise, the hybrid approach, which strictly is not as good as a "pure" feature matching or correlation feature matching algorithm, does possess significant advantages over the standard correlation approach at accommodating certain types of regional errors such as contrast reversals.

In Fig. 6 we show an example of this hybrid processing scheme. We have in the figure identified each reference pixel with a homogeneous region. Thus each reference pixel has both a region identification and an intensity associated with it. The template for the sensor map processing is shown for two map displacement positions. As indicated in the figure, the sensor map is segmented into homogeneous regions at each of these displacement positions in a manner identical to that of the reference map elements occupying the same spatial position. The sensor map elements are then processed by homogeneous regions (i.e., the mean intensity level subtracted out and possibly normalized by the intensity variation in the region) with the total correlation between sensed images and reference map being the sum of the correlation in each region at each displacement position. Thus we have identified four generic types of image matching methods:

1. Standard correlation algorithm
2. "Pure" feature matching algorithm
3. Feature matching correlation algorithm
4. Hybrid algorithm

The first two methods are the two basic approaches to image matching, while the latter two methods are variations of the standard correlation process designed specifically to accommodate nonhomogeneous scenes and the nonglobal errors associated with them.

SIMULATION RESULTS

Let us examine the effects of regional and local errors on the performance of matching systems for various classes of algorithms. First, let us examine the accuracy of the system measured in terms of the sharpness of the correlation peak. The general broadening of the correlation peak around the match point is caused primarily by the nonhomogeneous nature of the scene. Thus if we could process out the nonhomogeneous regions in the scene by a feature matching or hybrid algorithm we could expect a general sharpening of the correlation peak around the match point.

To illustrate these points we will decompose several Earth Resource Satellite (ERTS) maps into homogeneous regions and perform an autocorrelation between a sensor and reference map using the standard

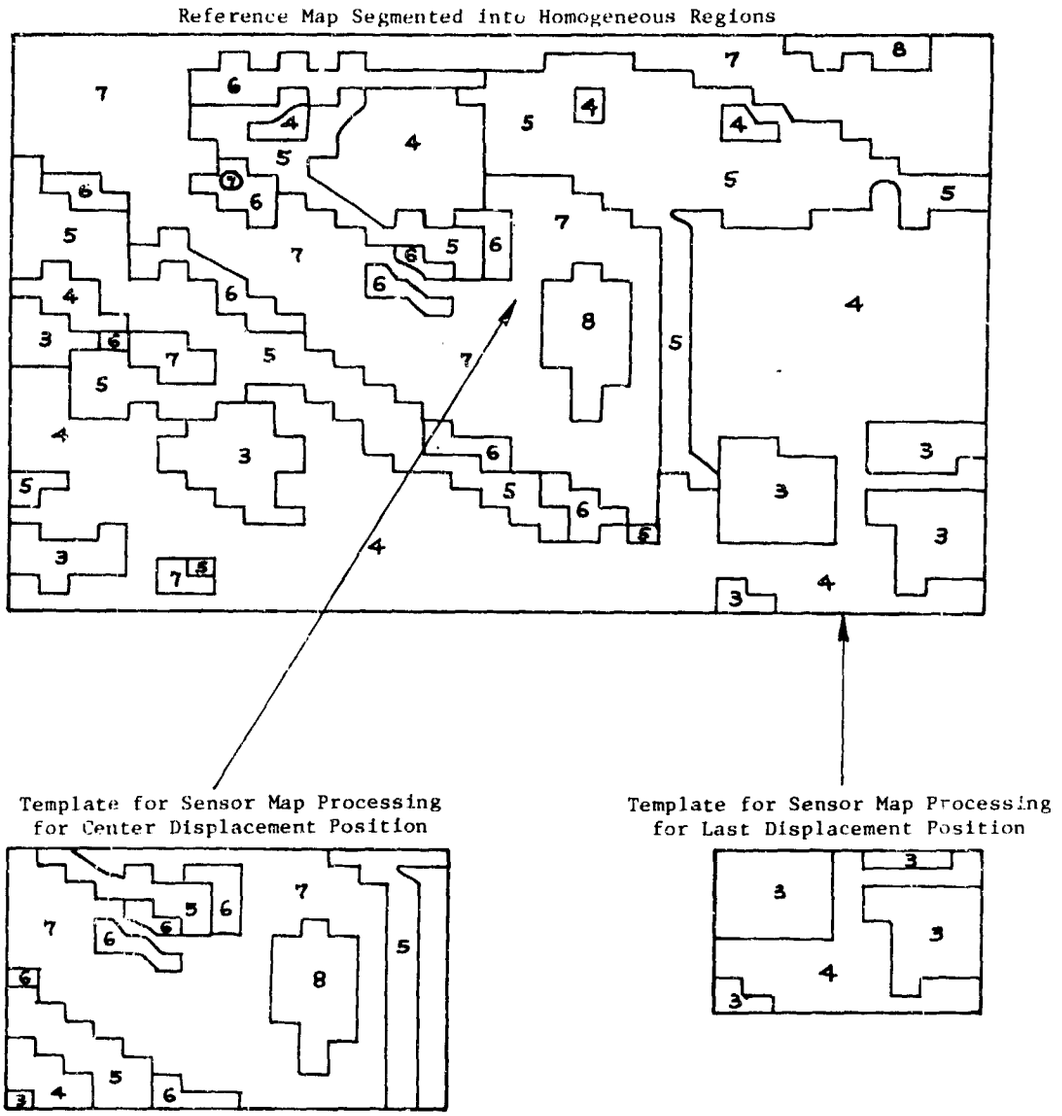


Fig. 6—Illustration of hybrid matching process

product algorithm, a feature matching algorithm, and a hybrid correlation matching algorithm which have been described previously. The feature matching algorithm essentially removes the effect of homogeneous regions since all homogeneous regions are zero meaned and normalized separately. The hybrid algorithm, on the other hand, takes out some but not all of the effects of the scene nonhomogeneity. Figure 7 shows the effect of using these three different algorithms upon the correlation surface for four different ERTS scenes. The normal autocorrelation process produces a spread out correlation peak, while the feature matching algorithm (homogenizing both the reference and sensor scene) produces the sharpest correlation peak, being limited

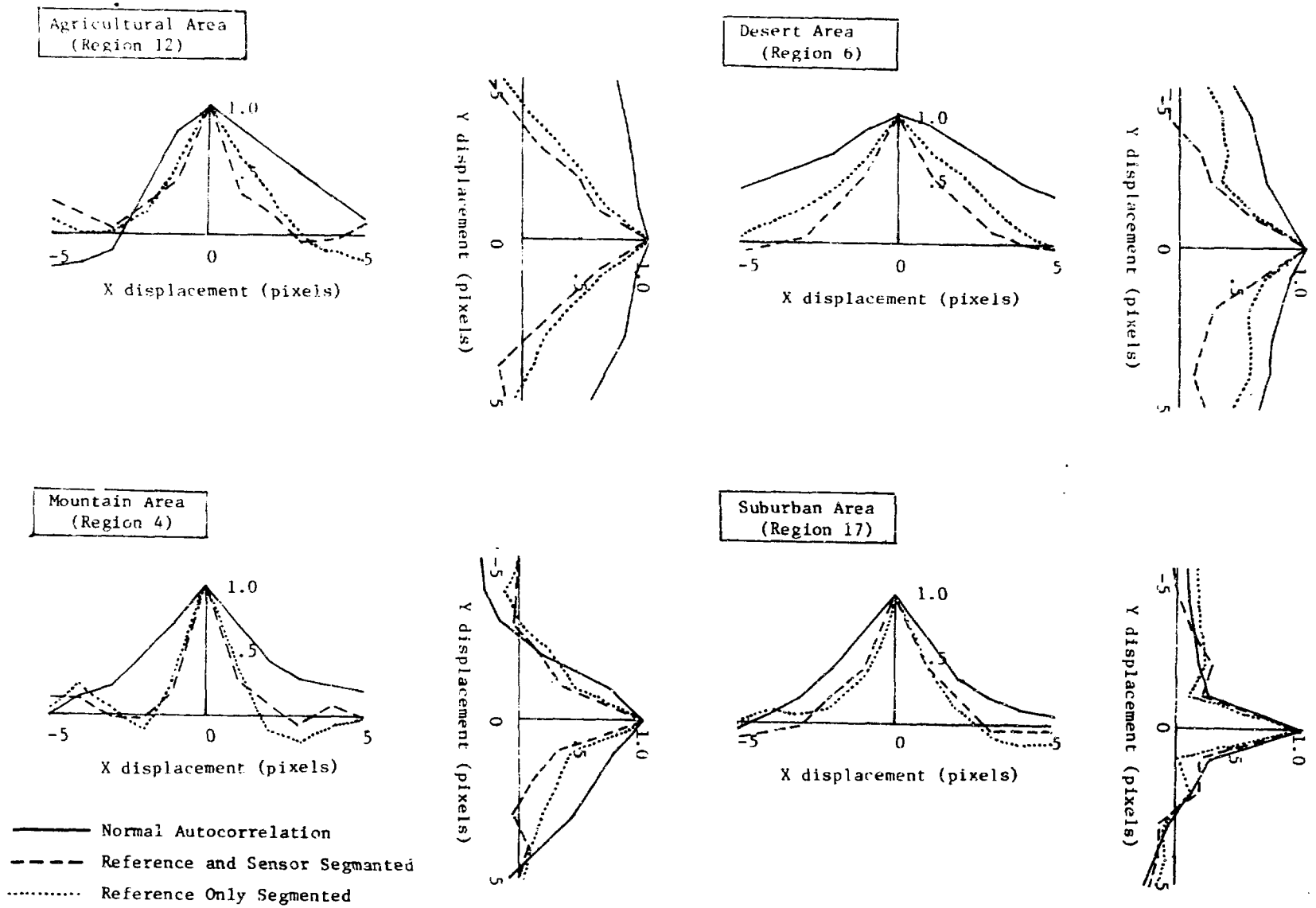


Fig. 7—Effects of homogeneous segmentation

only by the interpixel correlation. The hybrid algorithm produces a correlation surface between the two indicating that it does remove some but not all of the effects of scene nonhomogeneity. The remaining width of the correlation surface is due to interpixel correlations between nonindependent map elements contained within the homogeneous regions. To summarize, the slope of the correlation surface is dominated by the size and shape of the homogeneous regions composing the scene. Thus by utilizing feature matching or hybrid algorithms it is possible to filter out these low spatial frequency components and sharpen the correlation peak. The interpixel correlation and intensity variations between pixels, represented by the number and size of independent elements within the region, are only significant to the correlation process for completely homogeneous scenes (which are rare) and for scenes which have been homogeneously processed. Conversely, by homogeneously segmenting the scene, sharper correlation peaks can be produced whose widths are limited only by the interpixel correlation or the size of the independent elements.

The choice of matching algorithms for acquisition (P_c being major performance measure) will depend on the nature and magnitude of the regional and local errors. Some analysis has been performed in relating nonstructured errors to changes in system performance. In general the algorithm choice is not strongly dependent on the nature of nonstructured errors. Nonstructured errors are best accommodated in the mission planning phase of the operation. By proper route planning obscuration and masking errors may be avoided, and by timing and weather planning it may also be possible to reduce the diurnal and weather effects which can cause nonstructured errors. Thus the occurrence of nonstructured errors can be reduced by careful mission planning. Generally any residual nonstructured errors cannot be adequately modeled and thus one can only hope that they do not seriously degrade system performance.

The algorithm choice, then, in the extreme case of local errors only, tends toward ordinary correlation, whereas, in the other extreme (regional errors only) the algorithm tends toward pure feature matching. As one is generally never confronted by an either-or-situation, except in the case of Terrain Contour Mappping (where there are primarily local errors), it is necessary to weigh the relative magnitude of local and regional errors present in deciding upon the choice of algorithm.

Let us first consider the differences between the various categories of correlation algorithms when only local errors (additive noise) are present. To examine the effect, we took several 10 × 10 element sensor maps from the center of 20 × 20 reference scenes in various parts of an ERTS map. To these sensor scenes we added white Gaussian destributed noise such that the S/N ratio was 0.5. The simulation consisted of creating 25 different noisy sensor images and matching the reference and sensed imagery for different categories of algorithms (feature matching correlation, hybrid, and ordinary correlation algorithms) using the product algorithm. Table 1 shows the percent of successful matches (P_{SIM}) for each category of algorithm. The feature matching algorithm scored perfectly each time and is not shown in the table.

Table 1

MONTE CARLO SIMULATION RESULTS
Reference Map: 20 × 20
Sensor Map: 10 × 10

Terrain Type	Region	Type of Algorithm	Simulation Results (Product)
Mountain	2	Ordinary Correlation	0.96
Mountain	2	Hybrid	0.68
Suburbs	17	Ordinary Correlation	1.00
Suburbs	17	Hybrid	0.80
Desert	10	Ordinary Correlation	1.00
Desert	10	Hybrid	1.00
Desert	6	Ordinary Correlation	0.96
Desert	6	Hybrid	0.68
Agricultural	12	Ordinary Correlation	0.76
Agricultural	12	Hybrid	0.36

The homogeneous regions within the reference map boundary were defined manually. The homogeneous regions or features in the sensor image were also defined manually for the feature matching correlation algorithm. In the real world these regions must be extracted automatically so that the results for the feature matching correlation algorithm are, in a sense, an optimum case. In the real world, homogeneous regions are generally extracted through the use of edge operators. These systems generally do not perform well in the presence of local errors. Simulation results achieved for real-world scenes using pure feature matching approaches generally indicate results closer to or worse than those achieved by the hybrid algorithm are obtainable when automated edge finding feature extraction techniques are used.

To determine the change in system performance measured in terms of probability of correct match (P_c) due to regional errors interacting with the three different categories and types of algorithms described previously, we ran an experiment to test the effects of such errors. In an attempt to place regional errors into the correlation process we decided to see the effect of changing the mean values of the "intensity" levels in the homogeneous regions of the scene. For this experiment a sensor map (20 × 20) was chosen with a larger number of homogeneous regions (mountain area, region 4) and the mean level of each homogeneous region was changed by a random amount. The magnitude of the level change was drawn from a zero mean Gaussian distribution with three different standard deviations chosen to be 25, 50, and 100 percent of the dynamic range of intensity values in the scene. Two different algorithms (the normalized product and the difference-squared with the mean intensity value subtracted out) and three different processing schemes (both sensor and reference maps homogeneously segmented,

only the reference map segmented (hybrid) and no segmentation) were utilized. Additionally a small amount of noise was added to each pixel in the scene. The results are shown in Table 2. Shown in this table are the percent of successful correlations (out of 25), P_{SIM}, for each run using the different algorithm categories and types. Since we are using the "perfect" feature matching correlation algorithm we would not expect any change in performance with change in level and the results so indicate. On the other hand, there is a definite degradation in P_{SIM} for the ordinary correlation cases for all types of algorithm, with increasing changes among homogeneous levels in the scene. The hybrid algorithms, while generally having performance somewhat below that of the "perfect" feature matching algorithms, essentially do not degrade with increasing regional error.

Table 2

SIMULATION RESULTS WITH LEVEL CHANGES BETWEEN HOMOGENEOUS REGIONS
Mountain Area--Region 4
(20 × 20 Sensor Map, 40 × 40 Reference Map)

Process	Algorithm	Magnitude of Level Change		
		25 Percent P_{SIM}	50 Percent P_{SIM}	100 Percent P_{SIM}
Ordinary Correlation	Normalized Product	0.92	0.88	0.52
Hybrid	Normalized Product	0.72	0.72	0.68
Perfect Feature Matching	Normalized Product	1.0	1.0	1.0
Ordinary Correlation	Difference Squared (zero-meaned)	0.88	0.68	0.48
Hybrid	Difference Squared (zero-meaned)	0.96[a]	1.0	1.0
Perfect Feature Matching	Difference Squared (zero-meaned)	1.0	1.0	1.0

[a]The lower value relative to higher magnitude level changes is attributed to a statistical variation in only using 25 samples.

SUMMARY AND CONCLUSIONS

This paper described the image matching process as a two-phase process, with the first phase being concerned with the acquisition of the correct match area, and the second stage being concerned with accurately locating the match point. The major rationale for the failure of the system to acquire is described as being due to a combination of noise plus interscene redundancy (e.g., checkerboard), this latter problem being extremely difficult to model. Accuracy was shown to depend on two components of the scene structure--the size and magnitude of homogeneous regions in the scene and the interpixel

correlation (expressed in terms of an independent scene element)--and the amount of geometric distortion present.

It has been shown that accuracy can be improved by utilizing a hybrid or feature matching algorithm which segments the scene into homogeneous regions. This segmentation significantly sharpens the correlation. The residual spread in the correlation peak can be attributed to interpixel correlation.

The acquisition problem, described in Fig. 5, consists of determining the preprocessing requirements, developing a scene selection criteria, choosing an algorithm, and verifying the system via a simulation. As indicated in this figure, the first problem that must be accommodated is global errors. These errors are generally accommodated by either normalizing the intensity level or by spatially grouping the scene elements so as to reduce the susceptibility of the matching process to geometric distortion.

The scene selection process requires that two criteria be met. The first is that a sufficient amount of independent information must be contained in the map. Although not discussed, a number of methods have been proposed to measure the independent information contained within the scene. The correlation length appears to be a poor measure because of the ambiguity associated with the term. The number of "independent scene elements" appears to be a good measure to utilize for correlation processes, while the "number of vertices" appears appropriate for pure feature matching processes. The second scene selection process of importance is the avoidance of interscene redundancy (e.g., checkerboard patterns). The height of secondary correlation peaks using ordinary correlation does not appear to be as good a measure of scene redundancy as the height of secondary peaks using the hybrid algorithm. This hybrid class of algorithm assumes that at each displacement position the sensor image is segmented into homogeneous regions in an identical manner to the portion of the reference map against which it is being compared. Thus, this class of algorithm emphasizes the spatial structure of the scene and the few simulation results acquired to date indicate that secondary peaks on the autocorrelation surface associated with the hybrid algorithm are places where false matches are likely to occur due to an interscene redundancy.

Finally, in the acquisition process, an algorithm must be chosen from the generic class of ordinary correlation, hybrid correlation, feature matching correlation, and feature matching such that it can accommodate the amount of regional, local, and nonstructured errors that are anticipated. If only local errors are anticipated (e.g., TERCOM navigation system) then ordinary correlation algorithms are appropriate, whereas, if regional errors dominate, a feature matching or hybrid algorithm is demanded. Most real-world scenes have both regional and local errors superimposed. If the magnitude of the variation in the mean intensity levels between homogeneous regions in the area (that can be accounted for in the signature prediction) exceeds in value 50 percent of the intensity level difference between regions,

then it appears that one is forced to use a feature matching algorithm, with the hybrid algorithm looking as an attractive alternative to avoid the near real-time feature extraction process in the sensed image, while at the same time being able to deal with regional errors.

REFERENCES

1. Ratkovic, J. A., et al., *Estimation Techniques and Other Work in Image Correlation*, The Rand Corporation, R-2211-AF, September 1977.

2. Gupta, J. N., and P. A. Wintz, *Closed Boundary Finding Feature Selection and Classification Approach to Multi-Image Modeling*, Laboratory for Applications of Remote Sensing, Note No. 062773, Purdue University, W. Lafayette, Indiana, 1973.

3. Farag, R. F. H., "An Information Theoretic Approach to Image Partitioning," *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. SMC-8, No. 11, November 1978.

4. Kettig, R. L., and D. A. Landgrebe, *Classification of Multispectral Image Data by Extraction and Classification of Homogeneous Objects*, Laboratory for Applications of Remote Sensing, Purdue University, W. Lafayette, Indiana, 1975.

5. Robertson, T. F., et al., *Multispectral Image Partitioning*, Laboratory for Applications of Remote Sensing, Purdue University, W. Lafayette, Indiana, 1973.

6. Davis, L. S., "A Survey of Edge Detection Techniques," *Computer Graphics and Image Processing*, Vol. 4, pp. 248-270, September 1975.

7. Brice, C. R., and C. L. Fennema, "Scene Analysis Using Regions," *Artificial Intelligence*, Vol. 1, pp. 205-226, 1970.

8. Garnett, J. M., III, and S. S. Yaw, "Nonparametric Estimation of the Bayes Error of Feature Extractors Using Ordered Nearest Neighbor Sets," *IEEE Transactions, Computers*, Vol. C-26, pp. 46-54, January 1977.

9. Pratt, W. R., *Digital Image Processing*, Wiley & Sons, 1978.

10. Rosenfeld, A., and A. C. Kak, *Digital Picture Processing*, Academic Press, 1978.

11. Wacker, A. G., *A Cluster Approach to Finding Spatial Boundaries in Multispectral Imagery*, Laboratory for Agricultural Remote Sensing, Purdue University, W. Lafayette, Indiana, 1970.

12. Bailey, H. H., et al., *Image Correlation, Part I, Simulation and Analysis*, The Rand Corporation, R-2057/1-PR, November 1976.

13. Lahart, M. J., "Local Segmentation of Noisy Images," *Optical Engineering*, Vol. 18, No. 1, pp. 76-78, January-February 1979.

14. Merchant, J., *Address Modification, Image Technology Program*, Vol. I (Overview and Theory), Vol. II (Experimental Results), Honeywell Electro-Optics Center, Lexington, MA, September 1978.

15. Gerson, G. et al., *Image Sensor Measurements Program: Volume 1 Multiple Subarea Bi-Level Correlation Scene Matching System*, Contract F-30602-77-C-0049, Hughes Research Laboratories, Malibu, California, June 1979.

16. Hall, E., and R. C. Gonzalez, *Scene Content Analysis, Measurement, Refinement, and Verification*, Contract F4701-77-C-0072, DARPA/SAMSO, University of Tennessee, December 1978.

17. Kin, R. L., *Pattern Matcher Development Study*, DARPA Contract DAAK40-77-C-0017, Rockwell International, Missile Systems Division, Anaheim, California, June 1978.

18. Ratkovic, J. A., *Structuring the Components of the Image Matching Problem*, The Rand Corporation, N-1216-AF (to be published).

19. Ratkovic, J. A., *Performance Considerations for Image Matching Systems*, The Rand Corporation, N-1217-AF (to be published).

Paper No. IIA-5, Presented at the Workshop on Imaging Trackers and Autonomous Acquisition Applications for Missile Guidance, 19-20 November 1979, Redstone Arsenal, Alabama.

79Y133

Performance Comparison of a Laboratory Digital Tracker to the Human Eye

Written By:

James R. Hamer and Richard Satterfield
Northrop Corporation, Electro-Mechanical Division

ABSTRACT

Imaging tracking systems have no "benchmark" standard of performance to measure and compare against. This paper describes a laboratory experiment with Northrop's digital tracker system where test conditions were arranged in a manner similar to psychometric detection experiments of the human eye. Human detection performance is compared to the tracker acquisition signal to noise. Models and experimental data of both the human eye and digital tracker system are presented. Signal-to-noise, target size, and bandwidth considerations are presented and discussed. A video tape of tracking in cluttered environments is also presented.

Performance Comparison of a Laboratory Digital Tracker to the Human Eye

Written By:

James R. Hamer and Richard Satterfield
Northrop Corporation, Electro-Mechanical Division
500 East Orangethorpe Avenue, Anaheim, California 92801

Introduction

With the application of low-power, low-weight, and low-volume digital computers to airborne imaging tracker applications, the level of sophistication and performance of digital trackers has increased substantially. This growth is difficult to measure because an appropriate "benchmark" or standard of performance does not exist. The authors suggest that one of many such standards may be the limiting performance of the human eye. The advantage of using a psychometric data base for comparison is that a large number of detection experiments exist which have data readily available. The difficulty in use of this data base is in proper application and interpretation. Also, important factors such as tracking in clutter or moving targets are not included.

A brief review of the Northrop digital tracker system is next presented. The digital tracker laboratory facility and supporting electrical optical facilities are described. The problem of selecting a proper psychometric data base for comparison to the digital tracker is reviewed, and the Rosell and Wilson detection experiment is chosen as a proper data base for comparison to the digital tracker. The tracker laboratory experiment is described and results presented in parametric form and then compared to the noise-limited eye detection. Target acquisition and track Signal-to-Noise Ratio (SNR) are defined for the tracker as a function of target size. A tracker model and the Rosell model of "display SNR" are compared to highlight data similarities.

System Concept

The laboratory tracking system, shown in block diagram form in Figure 1, consists of a programmable general-purpose computer fitted with special peripheral devices to provide the capability of interfacing with real-time, video data streams, and high-speed digital control systems. Our technical approach has been to retain features of analog video processing which reduce the data rates to the computer and perform the tracking functions by digital computation. During every video field, the video processor formats the analog video into an n by n array of digital numbers, with each number representing the video scene in a rectangular area of the raster. Each area is called a bin and typically an 8-bin by 8-bin array is used for tracking, although there is no fundamental constraint to the use of an 8 by 8 array. The n by n array covers a variable aspect area of the scene controlled by the digital computer and nominally overlays the target, although a subspace of the target may be used. For large targets, special transformations are performed in the video processor to

prevent truncation or spatial quantitization from degrading tracking accuracy. This technical approach provides a constant data rate to the microcomputer and allows allocation of computer resources to tasks other than tracking, such as self-assessment.

Tracking Algorithms

The tracker generates position and size errors by use of a two dimensional, single integration, product correlator algorithm. The algorithm operates in two modes, a "point" track mode and an "area" track mode. In either mode and during every video field, the digital computer generates a set of tracking weights with which the incoming, live n by n bin array is integrated to form position errors. In point track the correlation reference template is a rectangle, and the template is stretched by the size algorithm to best fit the rectangular component of the target image being tracked. In area track mode the template is generated from the scene in the camera in the familiar "snap shoot" mode. The point track algorithm is illustrated in Figure 2. This approach differs from conventional trackers by having a size algorithm independent of range data or other independent estimators. A block diagram of the computer functions for the point track algorithm is shown in Figure 3.

Laboratory Facilities

The digital tracker software was developed by realistic, real-time simulation of closed loop system tracking problems, with a military camera-servo system (TISEO) mounted on an optical bench as shown in Figure 4. The TISEO servo-camera is integrated into the digital tracker and can be controlled from a hand stick for aircraft cockpit simulation, or from a digital data bus. A ten-to-one servoed zoom lens mounted on the optical bench simulates range closure to targets while tracking. Range closure rate can be controlled manually or under control of the laboratory computer. Target and background imagery may be presented independently and background contrast controlled by a back lighting technique. Target motion can be controlled on two dimensions from a motorized X-Y positioning mechanism. In the laboratory "special effects" are also simulated, for example, loss of one or more IR detectors of a parallel scan FLIR. Besides development of tracking software, this facility is being used for evaluation of track performance when totally integrated into the camera servo loop. A number of track scenes have been evaluated by recording the digital tracker output and later processing the data by Fourier transforms and statistical measures.

Psychometric Data Base

A wide range of psychometric data exists for human detection of line gratings and single targets under a variety of viewing conditions. Table I is a summary of the better known experiments. Direct view experiments are not applicable here because the eye is then contrast limited and the eye's limiting performance is dependent on the average brightness of the scene, target size, and target-background contrast. No noise is measurable and an SNR cannot be defined. This does not apply to indirect viewing such as from a Cathode Ray Tube (CRT) display where noise may be artificially controlled by the experimenter. The signal to noise can be carefully manipulated to make the human strictly noise limited in his detection performance. As first emphasized by the Coltman-Anderson experiment[1], the noise-limited observer's performance is independent of the contrast and brightness of the scene he views directly, the scene in the display, and, therefore, of CRT brightness and contrast manipulation.

Another feature of indirect viewing emphasized in their experiment is the relative independence of the size of the display. If the display size is increased and the video information bandwidth kept constant, there is no change in observer performance. If the display size is decreased and the observer is allowed to reposition himself closer to the display, there is also no change. His performance depends only on the relative image size in the display and the video SNR per unit video bandwidth. These features have been incorporated in all indirect viewing performance models.

A second fundamental consideration is the use of free choice versus forced choice detection methods, since both are often used. Forced choice psychometric experiments in human perception verify that the state of correct perceiving is not a discrete step, but probabilistic at low stimulus levels. Thus, models of human detection performances relate a perceptual SNR to a probability of correct detection. The performance curves are also a function of observer attention and fatigue level.

The probabilistic nature of human detection is brought out by forced choice psychometric experiments. The observer is forced to make a decision at a stimulus level where he would fail to decide. Normal viewing of scenes is not forced choice. Perception of targets is "free choice" and the observer is almost always correct when he makes a free choice decision. His perceptual SNR is greater than the SNR which corresponds to a detection probability of one. The range of subliminal stimulus levels is defined by that portion of the performance curve where performance is below a 0.95 probability level. At these stimulus levels the observers are unaware of their performance scores, which can be relatively high. The primary advantage of a forced choice experiment is the measurement of the absolute detection limits. Because of the advantage of CRT display parameter independence and the forced choice limiting sensitivity, the Rosell and Wilson experiments were selected as the data base for comparison with the digital tracker.

Noise Limited - Forced Choice Psychometric Performance

The experimental apparatus used to perform the psychometric experiments is shown in Figure 5 and described more completely.[2] A target rectangle is electronically generated and mixed with a white noise system band limited to five megahertz. The image displayed on the CRT appears in any quadrant and always in the same position in the quadrant. The observer is asked to choose the quadrant in which the image is located and the video SNR and the image locations are randomized. The observer specifies the image location every trial and the observation time per trial is 10 seconds. The observer distance from the 8-inch-high CRT was 28 inches, and the display background luminance was either 0.2-0.3 or 1 foot-Lambert. The television monitor was operated at 30 frames/second with a 525-line scan in the vertical.

Digital Tracker Threshold Performance

The laboratory facility shown in Figure 4 was modified to that shown in Figure 6 for the tracking experiment. A high-contrast square target against a plain background was placed before the TISEO camera. The high SNR target video was attenuated by a precision attenuator and then mixed with a white noise signal equal in bandwidth to the TISEO camera (20 megahertz). The signal plus noise was then presented to the digital tracker for processing track errors. Before each experiment the servo gains were adjusted to two system bandwidths, 3.6 and 0.9 radians per second. At each system bandwidth two curves were generated, a target acquisition SNR threshold curve and

a target track loss SNR threshold curve as a function of target size. Both curves are defined by a procedure using the self-assessment track loss propositions of the digital tracker which are posted during every video field. For target acquisition SNR, the TISEO camera was locked onto a square target and the video SNR reduced until a track loss proposition began to read true. The SNR was then increased slightly and the target reacquired several times to assure reliable lockup upon acquisition; the SNR and target size were then recorded and the experiment repeated for a different sized target. The track loss threshold curve is defined by monitoring the track loss propositions at reduced SNR from the acquisition SNR. When the track loss propositions are flashing true fifty percent of the time, the tracker will maintain lock; but this level of confidence is arbitrarily defined as the impending loss lock SNR. The SNR and target size are noted and the experiment then repeated for a different size target. Figure 7 shows the results for a system bandwidth of 3.6 radian per second. The upper curve is the acquisition curve, the lower is the track loss curve. Note the uniform displacement of the curves. For comparison with the human eye the 0.9 radian system bandwidth curve was used because of the long 10-seconds viewing time. The Rosell and Wilson SNR data was adjusted from their 5 megahertz noise bandwidth to the 20 megahertz noise bandwidth of our experiment and their data replotted for square targets. The results are shown in Figure 8. The 95 and 50 percent eye detection threshold curve is plotted. The 95 percent curve corresponds to the level at which an observer will begin to transfer from forced choice to free choice. Note the similarity in shape of all the curves.

Tracker-Eye Model Summary

The similarity of the data prompted a comparison of the Rosell display SNR model to Northrop's tracker. For an "ideal" sensor, such as the one used in the psychometric experiment, the perceptual SNR (Rosell calls this "display SNR") is related to the video SNR by:

$$\text{Perceptual SNR} = \left[\frac{\Delta f a t}{A}\right]^{1/2} \text{Video SNR}$$

Δf = Video noise bandwidth

a = target area

A = raster area

t = eye integration time

The eye spatial integration across the target reduces the effective noise bandwidth by a factor A/a and can be interpretted as a spatial matched filter improving system SNR. The temporal filter of the eye further reduces the effective bandwidth by the factor 1/t, thus two filters can account for the improvement in performance in the eye detection model as shown in Figure 9. There the tracker model is also shown and the eye detection model tied to it through the CRT display. The tracker also is a spatial matched filter which generates position estimates at a rate of 60 samples per second. The servo may be considered as a temporal filter having a bandwidth that

determines the extent of averaging the high sample rate. The video noise filter and whitener has a parallel in the human eyeball where gradient operations are known to exist. Thus there are many similarities between the digital tracker and human eye which are suggested by the experiment.

A modification to the eye detection model is suggested to account for contrast-limited versus noise-limited performance of the eye. If one assumes an eye with an internal noise source (N_o) dependent on scene brightness, the SNR at the input to the eye spatial filter is given by:

$$(S/N)_o = \frac{GS}{\sqrt{N_o^2 + G^2 N_v^2}}$$

N_v = Video noise

G = CRT gain

The contrast limited eye is defined when $N_o^2 >> G^2 N_v^2$, then adjusting the CRT contrast knob will improve target detection. The noise-limited eye is defined when $G^2 N_v^2 >> N_o^2$, then the SNR becomes independent of contrast gain and equal to the video SNR out of the camera. These conclusions further reinforce the decision to use only the noise limited performance of the eye to compare to the digital tracker. The contrast limited performance cannot be compared to the tracker because of our ignorance of the eye's internal noise source, N_o. Fortunately the noise limited performance is a minimum boundary and therefore a good baseline for comparison of tracker performance.

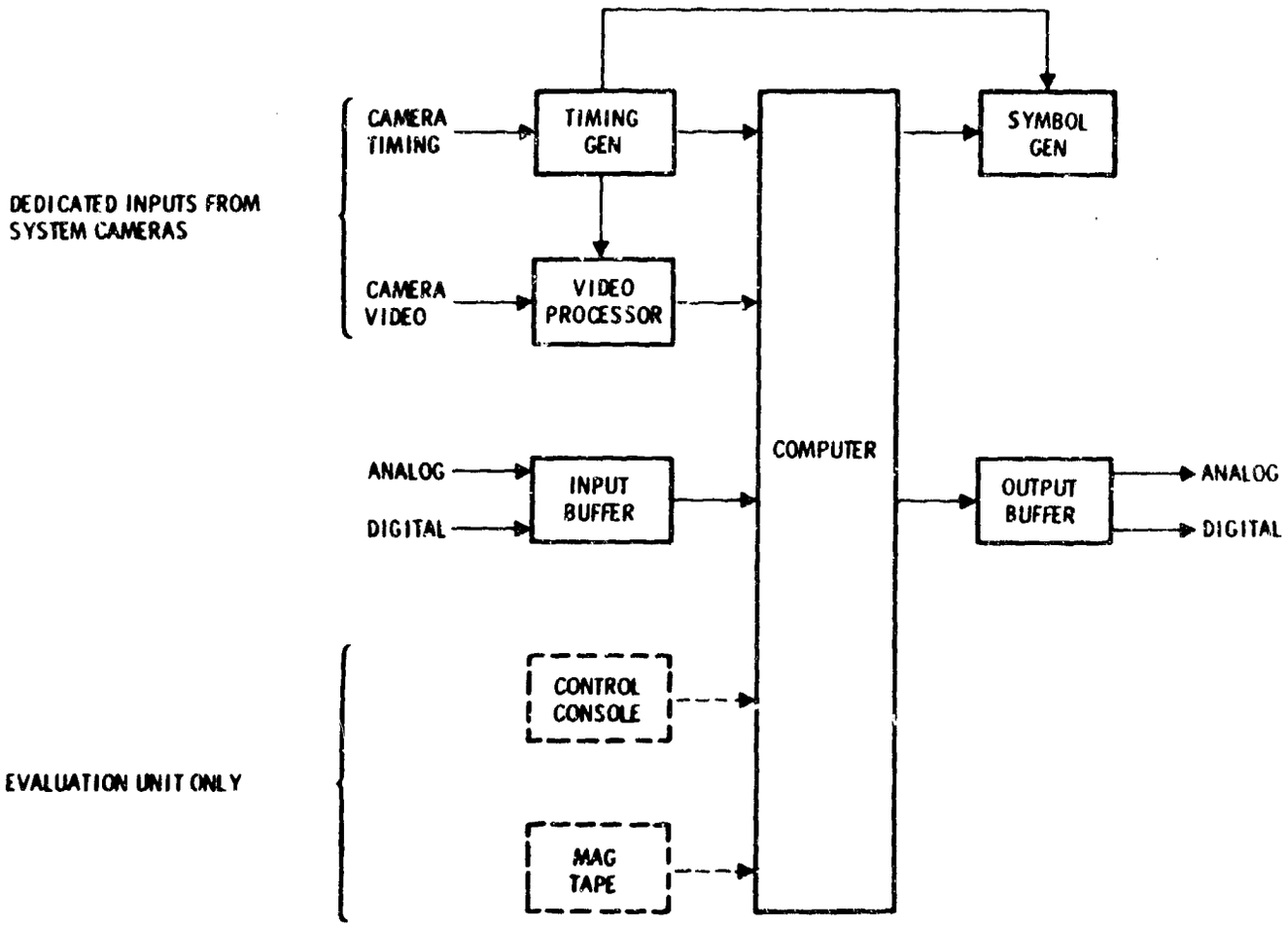


Figure 1. NORTHROP CPU TRACKING SYSTEM

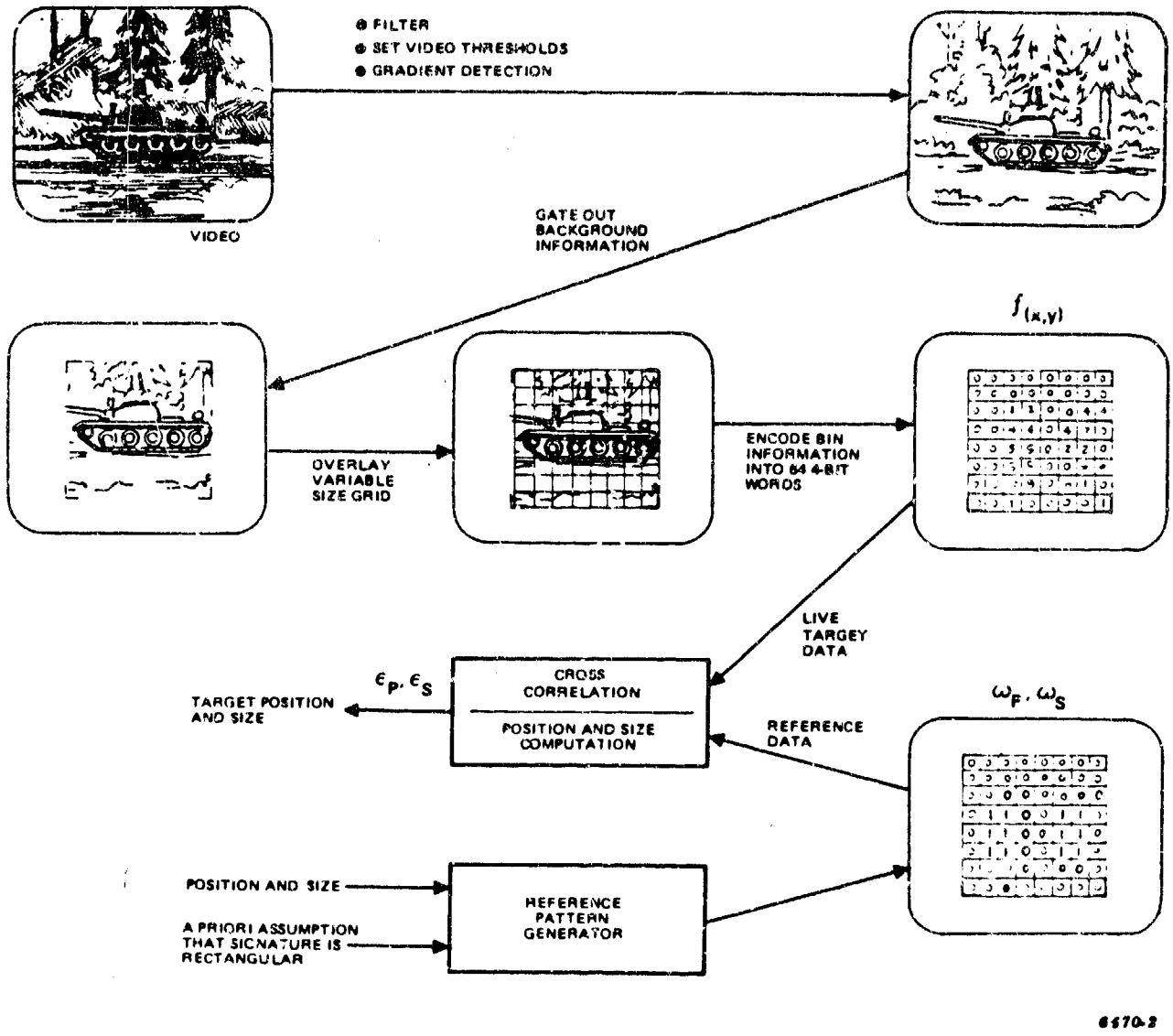


Figure 2. POINT TRACK ALGORITHM

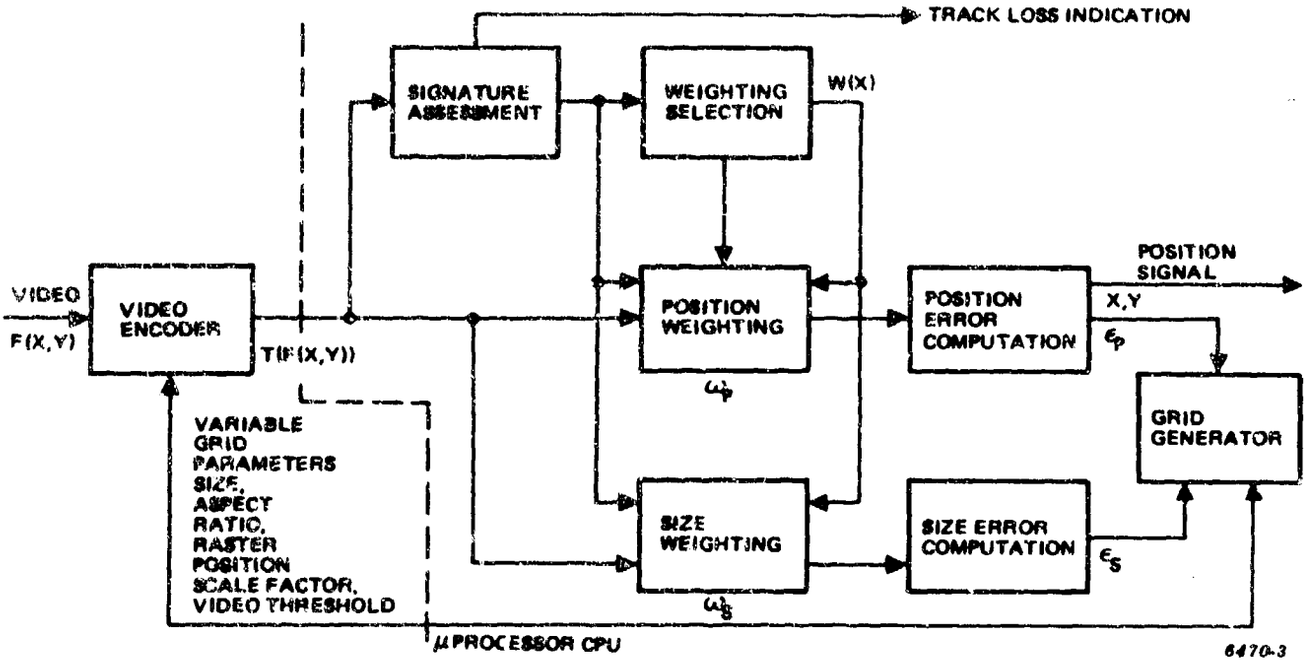


Figure 3. BLOCK DIAGRAM OF COMPUTER FUNCTIONS FOR THE POINT TRACK ALGORITHM

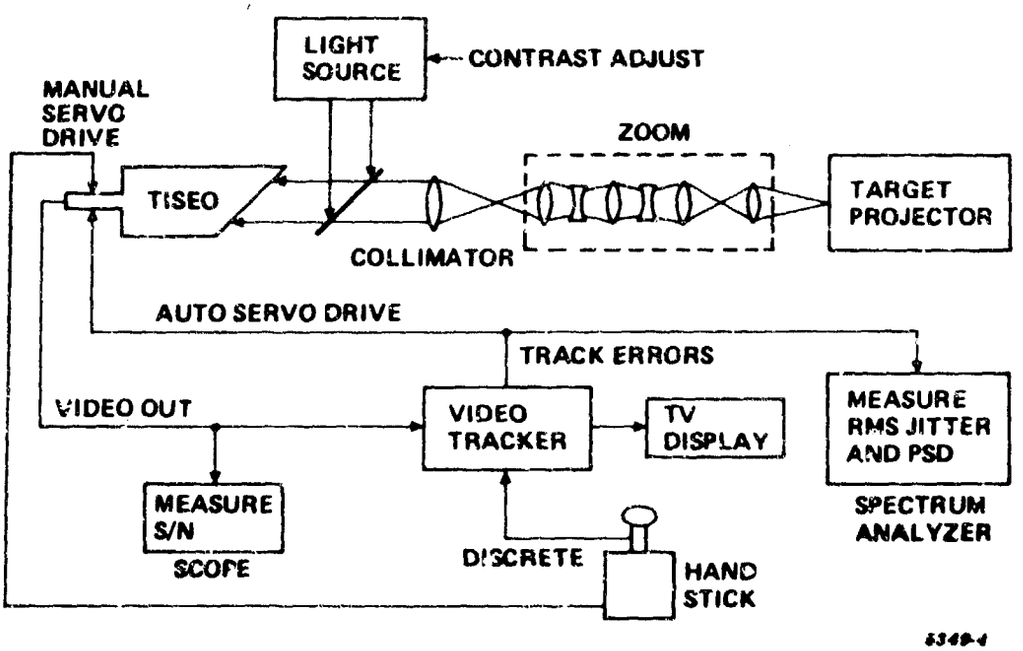


Figure 4. LABORATORY TRACKER EVALUATION

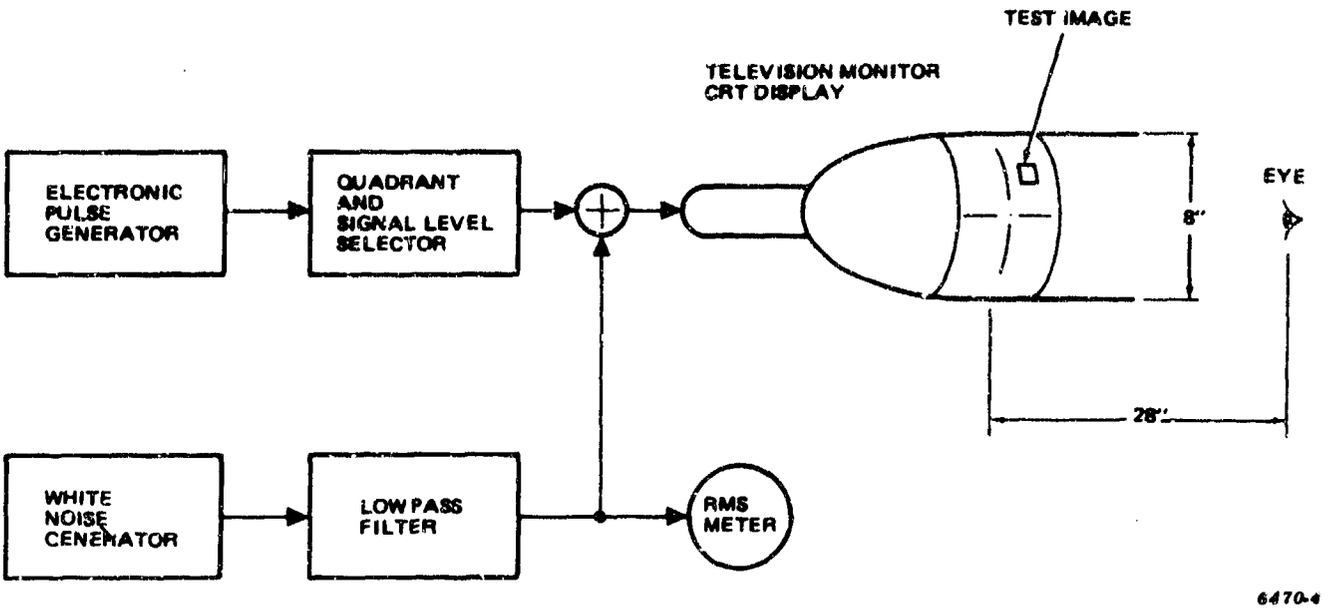


Figure 5. NOISE-LIMITED TARGET DETECTION EXPERIMENT

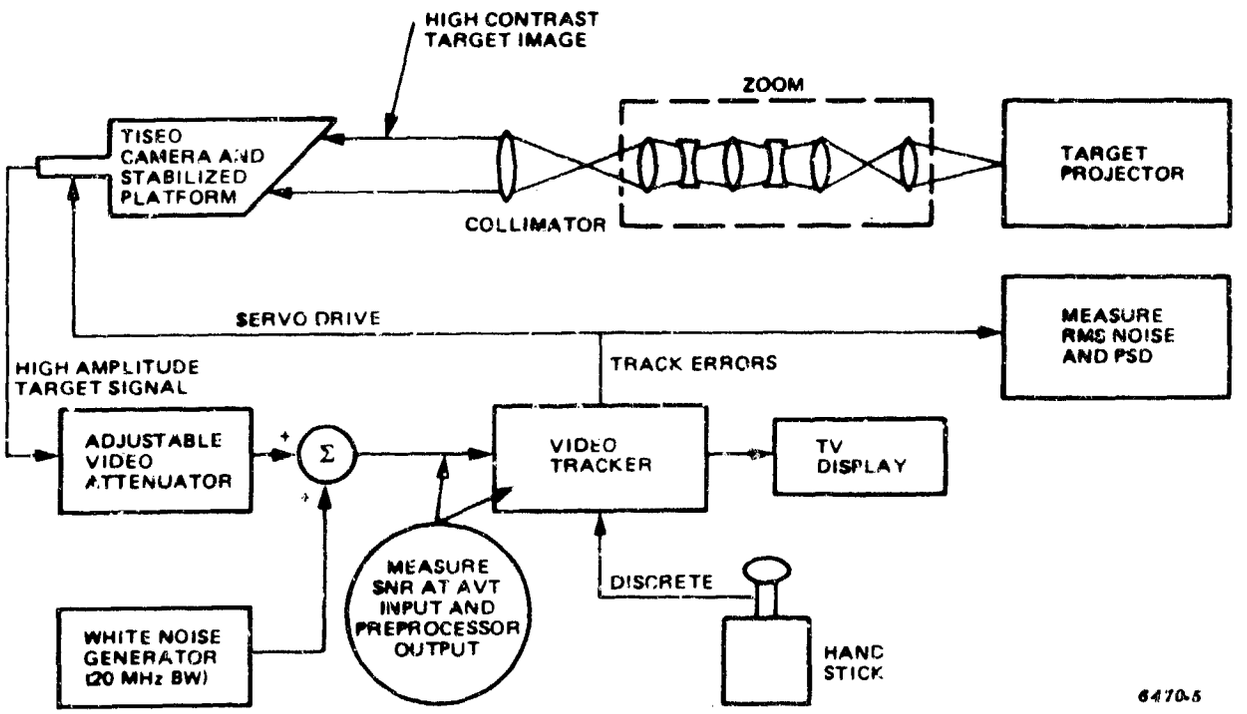


Figure 6. LABORATORY TRACKER EXPERIMENT FOR MEASURING TRACK THRESHOLDS

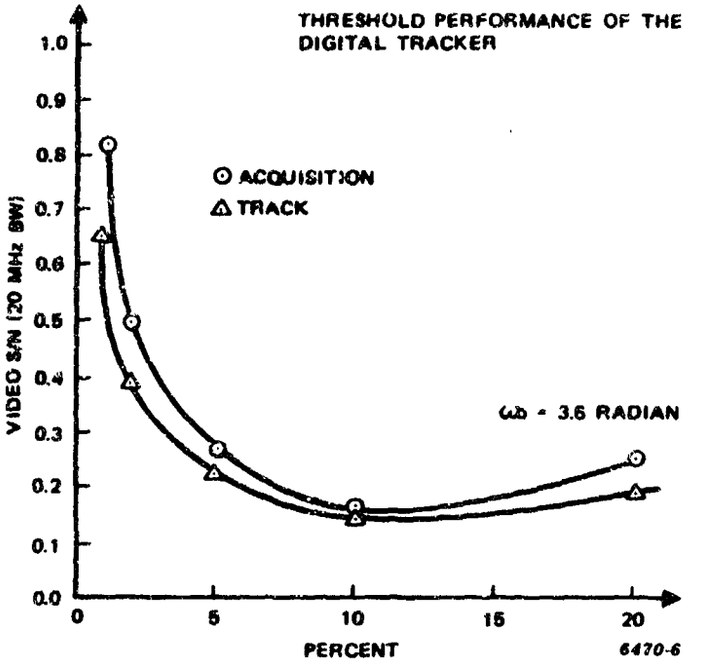


Figure 7. TARGET SIZE - PERCENT FOV

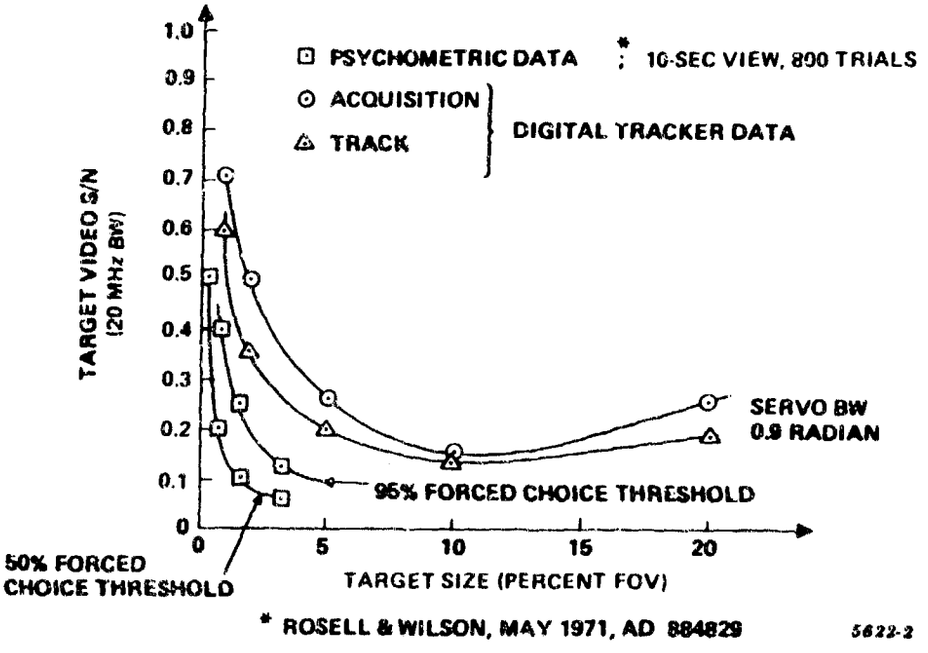


Figure 8. EYE DETECTION THRESHOLD AND THRESHOLD TRACKING PERFORMANCE (SQUARE, STATIC TARGETS)

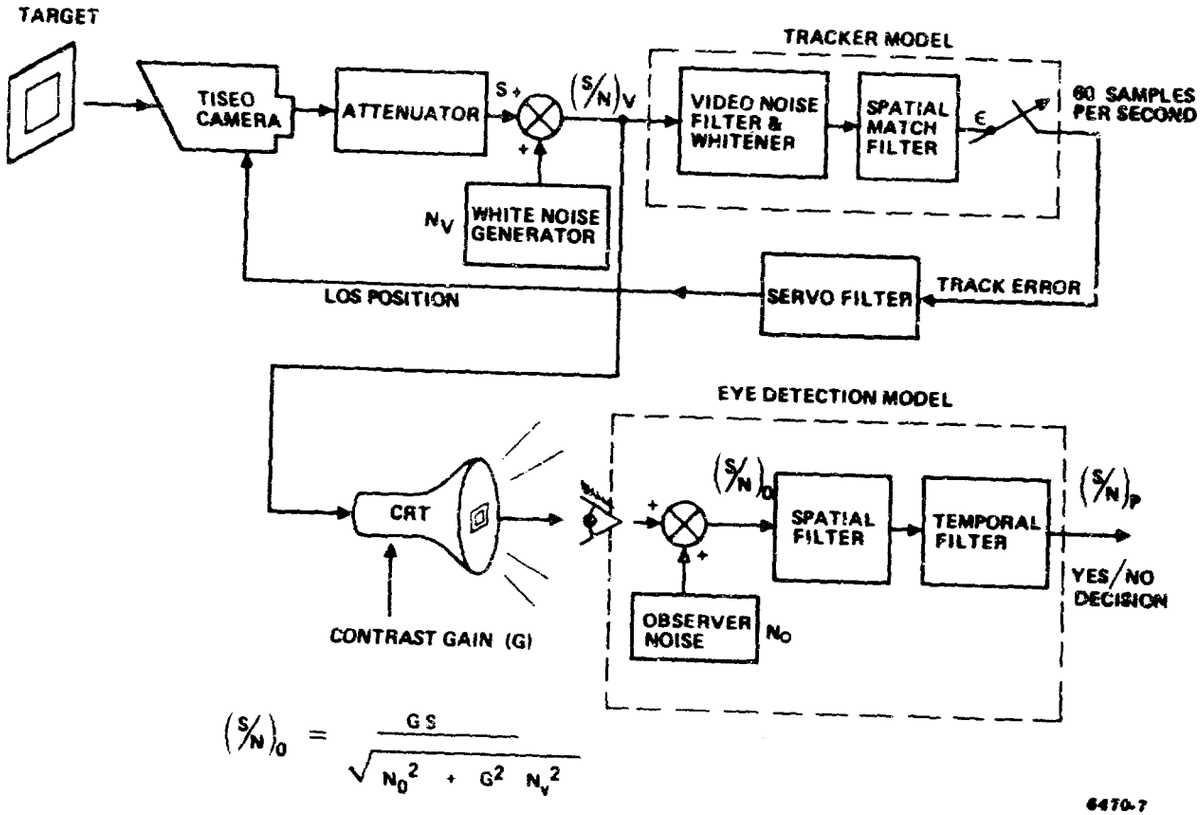


Figure 9. TARGET DETECTION AND TRACKING EXPERIMENT

Table I. PSYCHOMETRIC TEST SUMMARY

VIEWING CONDITION	SCENE DYNAMICS	NOISE VISIBILITY	OBSERVER TEMPORAL INTEGRATION	TARGET/METHOD BAR CHART - FREE CHOICE SINGLE TARGET - FORCED CHOICE	PSYCHOMETRIC DATA	OSNR THRESHOLD SIGNAL-TO-NOISE
DIRECT VIEWING	STATIC	NONE	1 SEC	BAR CHART		
				ISOLATED TARGET	BLACKWELL	
	DYNAMIC	NONE	0.2 SEC			
INDIRECT VIEWING ACTIVE (DISPLAY)	STATIC	NONE	> 1 SEC	ISOLATED TARGET		
				BAR CHART	CAMPELL	
		NOISE	> 1 SEC	ISOLATED TARGET	ROSELL	3
				BAR CHART	COLTMAN-ANDERSON	1.2
	DYNAMIC	NONE	0.2 SEC			
		NOISE	0.2 SEC			
INDIRECT VIEWING PASSIVE (PHOTOGRAPH)	STATIC	NONE	0			
		NOISE	0			
				ISOLATED TARGETS	CHAMBERS	5

6470-8

REFERENCES

[1] J. W. Coltman and A. E. Anderson, Noise Limitations to Resolving Power in Electronic Imaging, IRE Proceedings 48 (May 1960) 858

[2] F. S. Rosell and R. H. Wilson, Performance Synthesis of E-O Sensors, AFAL-TR-71-137, May 1971, AD 884829

Paper No. IIA-6, Presented at the Workshop on Imaging Trackers and Autonomous Acquisition Applications for Missile Guidance, 19-20 November 1979, Redstone Arsenal, Alabama.

IMAGE PROCESSING FOR SELECTIVE IMPACT

R. B. Merrill and J. J. Hwang
Nichols Research Corporation
Huntsville, Alabama 35802

ABSTRACT

An algorithm is described for detecting and classifying a tactical target from infrared sensed imagery. A point in the target with distinct features has been used as a reference point to extract the target region. Then edge features in the target region are transformed into a polar coordinate space and target matching is performed in this space. The experimental results from sixteen IR images indicate that the orientation and the size of the target can be accurately calculated by this method. Comparisons with the moment invariants method for target matching and the Hotelling transformation for target orientation calculation are also presented.

INTRODUCTION

The purpose of this paper is to present an algorithm for selecting an impact location from an infrared sensed image of the target. The difficulties in target identification are that the size and the orientation of the target are unknown. These vary according to the relative location and orientation of the target and sensor. Correlation of the reference image with the sensed image with different size and orientation is particularly difficult, since various sizes and rotations of templates must be used. A better approach is to preprocess the sensed image and calibrate it to the correct size and orientation. This also eliminates the need for an interpolation technique to fill in the missing information in the digital rotated image. The information of the target orientation is usually embedded in the shape features of the target. Since the infrared image displays the thermal emission of the target and the background, and the temperature distribution on the surface of the target is usually not uniform, the edges obtained by a local gradient type operator are usually broken and are difficult to use for shape description. To classify a target with different orientation and size, several scene matching algorithms based on global analysis of the local feature in the IR image may be appropriate. Among these algorithms, the moment invariants method shows success in many different applications (3,4,5). Also, the Hotelling transformation has been found useful for object rotation (1,2).

A new matching algorithm for detecting and classifying a tactical target in IR sensed images is presented in this paper. A point in the target with distinct features has been used as a reference point to extract a potential target region. Then the edge features in this region are transformed into the polar coordinate space and target matching is performed in this space. The experimental results from a set of sixteen images indicate that the orientation and size of the target can be accurately calculated by this method. Comparison with the moment invariants method in target matching and Hotelling transformation in target orientation calculation are also presented in this paper.

POLAR TRANSFORMATION

A typical infrared image is shown in Figure 1. In this image, the target is setting on a textured background. The high frequency information in this thermal image makes it very difficult to outline the entire target for shape description. Figure 2 shows the edge processed image of an extracted target region to illustrate the high frequency information of the thermal emission in the tank image. A distinct feature can easily be observed in the three dimensional graph of Figure 2, which is shown in the isometric plot in Figure 3. One can easily detect the cluster of points with very large edge values. By using the cluster center as a reference point, the edge points in the rectangular space may be transformed into a polar space in which target matching may easily be performed.

The concept of the polar transformation is shown in Figure 4. Five edge points are shown in both the reference and sensed images. Consider point "a" as the extracted reference point in the reference image and point "b" to be the extracted reference point in the sensed image. By using the X_a axis and the X_b axis as the references axes for the reference image and the sensed image, the edge points may be transformed into a polar coordinate space. It is obvious that if one correlates the reference image to the sensed image along the σ axis, the location of the correlation peak will indicate the relative angular orientation of the sensed image to the reference image.

Suppose there are N quantized levels for the radial components and M quantized levels for the angular components. Then an image in the polar space may be represented by a column vector with dimension NM. If the image is scanned in a vertical yaster fashion in the polar space or is sampled radially for every θ_s (sampling angle) with respect to the reference point in the rectangular space, then a $[(m-1)*N+n]$th component is set to 1 if an edge point is detected at location (M,N) in the polar space or location $(n*\cos(m-1)*\theta_s,\ n*\sin[(m-1)*\sigma_s)]$ in the rectangular space, otherwise, the value is 0. Let the image vector for the reference image be $\underline{A}$ and the image vector for the sensed image be $\underline{B}$. Then the correlation measure R may be represented by

$$R = \frac{\underline{A}^T \underline{B}}{||\underline{A}|| * ||\underline{B}||} \qquad (1)$$

Since both $\underline{A}$ and $\underline{B}$ are binary vectors of edge features, R is a measure which indicates the number of matching edge feature points at the same angle and radial distance, normalized by the geometric average of the number of edge feature points in the reference and the sensed images.

The radial distance of an edge feature point is proportional to the size of the target. In the ideal case, assuming no noise edge feature points have been extracted, it is reasonable to consider the size ratio of the reference target to the sensed target as $\bar{S}_r = \bar{\rho}a/\bar{\rho}b$, where $\bar{\rho}a$ is the mean radial distance of the edge points in the reference image, $\bar{\rho}b$ is the mean radial distance of the edge points in the sensed image and $\bar{S}_r$ is the mean size ratio.

In the noisy case, the size ratio may be calculated by

$$S = \max_{\underline{X}} R_s$$

where $\underline{X}$ is the set of vectors corresponding to different scale changes of the reference image.

By incrementing or decrementing $\bar{S}r$, the size ratio of the reference target to the sensed target is selected as the one which maximize the correlation coefficient at the orientation using $\bar{S}r$.

A computer synthesized IR image may be used as an example. Figure 5 shows the IR image. Figure 6 shows the edges extracted by a Sobel operator. Figure 7 shows the polar transformed edge image in a 64 × 64 grid. Figure 8 shows the autocorrelation result of the polar transformed edge image along the θ axis. Note that the correlation peak is at 0 degrees and the size ratio is 1 as expected.

Sixteen other images were used for orientation and size calculations using the polar transformation. Table 1 shows the experimental results. P_0 is used as a reference image. P_1 through P_{12} were the sensed images with different sizes and orientations assuming that the sensed targets were perfectly segmented from the background. L_1 through L_4 were four images with the same size and orientations as the reference image but are located in a noisy texture background. The edge feature points of images L_1 through L_4 were segmented using different threshold levels such that different amounts of noisy edge points were also presented in the sensed edge image for comparison. Figure 9 shows four edge images segmented by the Sobel operator using different thresholds. Figure 10 shows the corresponding polar transformation edge images. Figure 11 shows the correlation result using the IR image of Figure 6 as the reference image. In this limited laboratory test, the experimental results seem promising. The average error for the size calculation is generally less than 5%. When the size of the sensed image is smaller to the reference image by a factor of two, such as in images P_8 and P_{10}, the size calculation error is higher than 10%. This is due to the fact

that the resolution is lower for the smaller size sensed target and some of the feature points are merged together. The correlation mainlobe and sidelobe ratio is used to measure the goodness of the correlation result. All the correlation peaks are discriminable except for the smaller size sensed targets (P_8, P_{10}, P_{12}) for which the mainlobe to sidelobe ratios are less than 1.2. The orientation calculation result also seems promising. The errors in the orientation calculations of the sensed images P_1 and P_2 are due mainly to the quantizing error of the polar transformation. Since only 64 sampling angular intervals are used, the accuracy in the orientation calculation will never be better than 5.625 degrees.

COMPARISON WITH OTHER ALGORITHMS AND CONCLUSION

Using moment invariants for target matching has been successful in many applications (3,4,5). The mathematical foundation of invariant features is based on the theory of algebraic invariants. The theory deals with algebr ic functions of a certain class which remain unchanged under certain coordinate transformations. A set of seven moment invariants has also been calculated for the sensed target identification. The Euclidean distance of the moment invariants of the reference images to is computed to classify the sensed target. The Euclidean distance of the moment invariants of the reference image to those of the sensed images is plotted as shown in Figure 12. The sensed targets (P_8, P_{10}, P_{12}), which are a factor of two smaller than the reference target, are also difficult to identify by moment invariants.

A fast method for orientation computation is the Hotelling transformation (1,2). The covariance matrix of the spatially distributed edge feature points is calculated. Then the principal axis of the sensed images may be determined by finding the eigenvector with maximum eigenvalue. The relative orientation of the sensed target to the reference target may be determined from the principal axis of the reference image and sensed image.

Table 2 shows the orientation error calculated using the Hotelling transformation. For sensed images P_1 through P_{12}, the results are satisfactory since these images are perfectly segmented. In the noisy cases, the calculated principal axis may vary due to the presence of noisy edge points. A calibration step is required for the principal axis calculation assuming the background noise statistics is known. Figure 13(a),(b),(c) illustrate this procedure. The accuracy of the orientation calculation using the Hotelling transformation is very dependent on the segmentation results of the sensed target. For a well segmented sensed target, the Hotelling transformation is several orders of magnitude faster than the polar transformation method since the latter requires N correlation calculations of an MXN polar transformed edge image. However, in the noisy environment, the polar transformation is more robust.

Among the three algorithms discussed, the polar transformation has been shown to be the least dependent on the segmentation of the target and also has better tolerance to a noise environment.

REFERENCES

1. Hall, E. L., Computer Image Processing and Recognition, Academic Press, New York, In press.

2. Gonzalez, R. C. and P. Wintz, Digital Image Processing, Addison-Wesley, Mass., 1978.

3. Hu, M. K., Visual Pattern Recognition by Moment Invariants, IRE Transaction on Information Theory, IT-8, February 1962, pp. 179-187.

4. Dudani, S. and K. J. Breeding, Aircraft Identification by Moment Invariants, IEEE Transaction on Computers, Vol. C-26, No. 1, January 1977, pp. 39-45.

5. Wong, R. Y. and E. L. Hall, Scene Matching with Invariant Moments, Computer Graphics and Image Processing 8, 16-24, 1978.

TABLE 1. TARGET ORIENTATION AND SIZE CALCULATION USING POLAR TRANSFORMATION

IMAGES	SIZE	ROTATION DEGREES	CORRELATION PEAKS AT DEGREES	ORIENTATION ERROR IN DEGREES	MAINLOBE & SIDELOBE RATIO	CALCULATED SIZE	SIZE ERROR %
P0	1	-----	----	----	----	----	----
P1	1	-2.8125	-5.625	2.8125	1.22	1.01	1
P2	1	-5.625	-5.625	0	1.27	1.04	4.7
P3	1.01	-15	-16.875	1.875	1.56	1.03	2.47
P4	1.05	-22.5	-22.5	0	1.22	1.05	0.47
P5	1.07	-45	-45	0	1.23	1.08	1.02
P6	1	-90	-90	0	1.74	1.01	1.5
P7	0.8	0	0	0	1.55	0.81	2.37
P8	0.5	0	0	0	1.02	0.56	13.4
P9	0.856	-45	-45	0	1.31	0.81	4.79
P10	0.535	-45	-45	0	1.01	0.56	5.23
P11	0.8	-90	-90	0	1.40	0.82	2.87
P12	0.5	-90	-90	0	1.18	0.55	11.6
L1	1	0	0	0	1.90	1.02	2.8
L2	1	0	0	0	1.88	1.03	3.9
L3	1	0	0	0	1.84	1.04	4.3
L4	1	0	0	0	1.79	1.05	5.1

TABLE 2. TARGET ORIENTATION CALCULATION RESULTS USING HOTELLING TRANSFORMATION

IMAGES	TARGET ORIENTATION ERROR (DEGREES)	IMAGES	TARGET ORIENTATION ERROR (DEGREES)
P0	reference	P9	1.61
P1	0.2275	P10	0.93
P2	0.952	P11	2.978
P3	0.039	P12	1.714
P4	0.608	L1	7.714
P5	1.107	L2	1.922
P6	0.349	L3	0.411
P7	1.82	L4	0.713
P8	1.436		

FIGURE 1
AN IR SENSED IMAGE

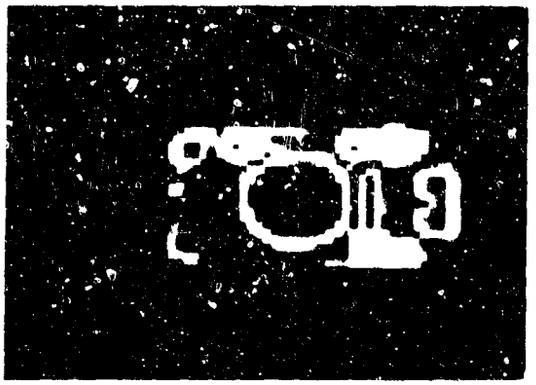


FIGURE 2
A SOBEL EDGE PROCESSED TARGET REGION

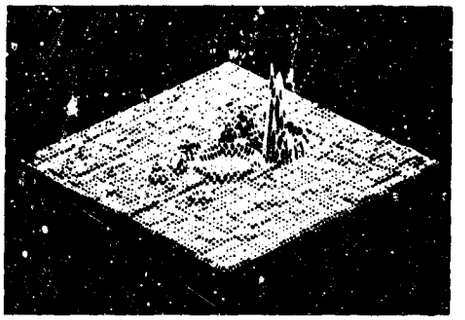
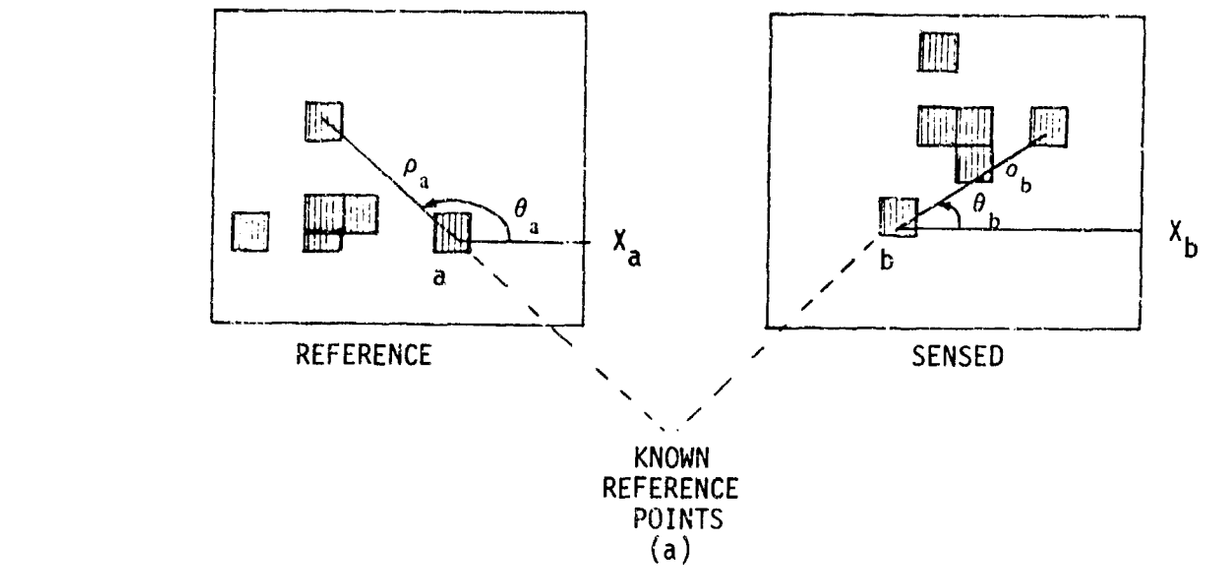
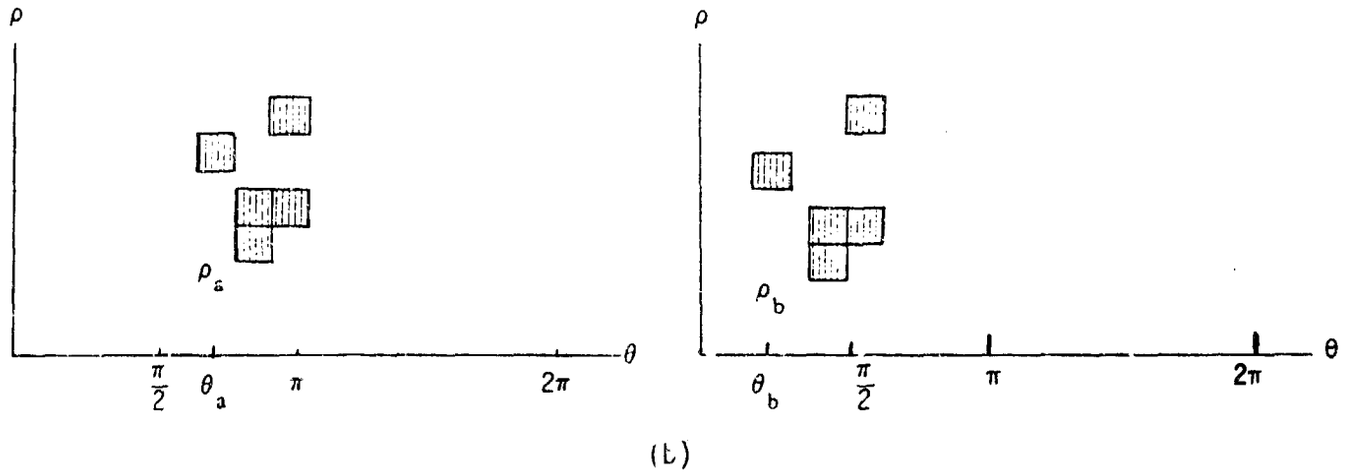


FIGURE 3
A THREE DIMENSIONAL PLOT OF
FIGURE 2





(b)

FIGURE 4. (a) THE EDGE EXTRATED REFERENCE IMAGE AND SENSED IMAGE POINT a AND POINT b ARE THE CALCULATED REFERENCE POINTS IN THE REFERENCE IMAGE AND SENSED IMAGE RESPECTIVELY.

(b) THE POLAR TRANSFORMATION EDGE IMAGES.

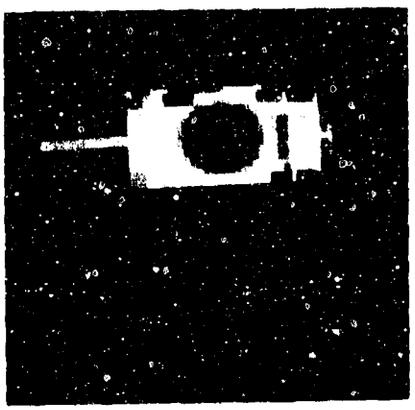


FIGURE 5
A COMPUTER SYNTHESIZED
IR IMAGE.

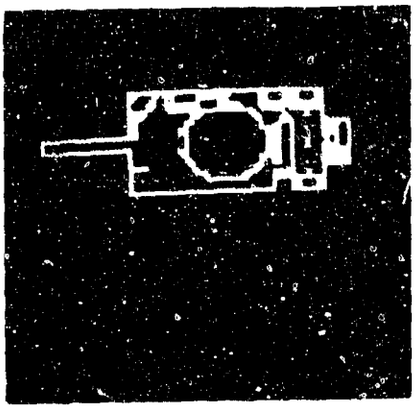


FIGURE 6
THE EXTRACTED SOBEL EDGES

FIGURE 7
THE POLAR TRANSFORMED
EDGE IMAGE

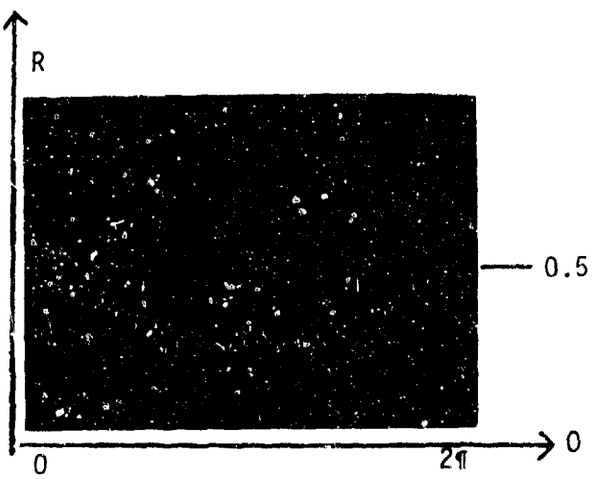


FIGURE 8
THE AUTOCORRELATION RESULT OF
FIGURE 7 USING EQUATION (1)

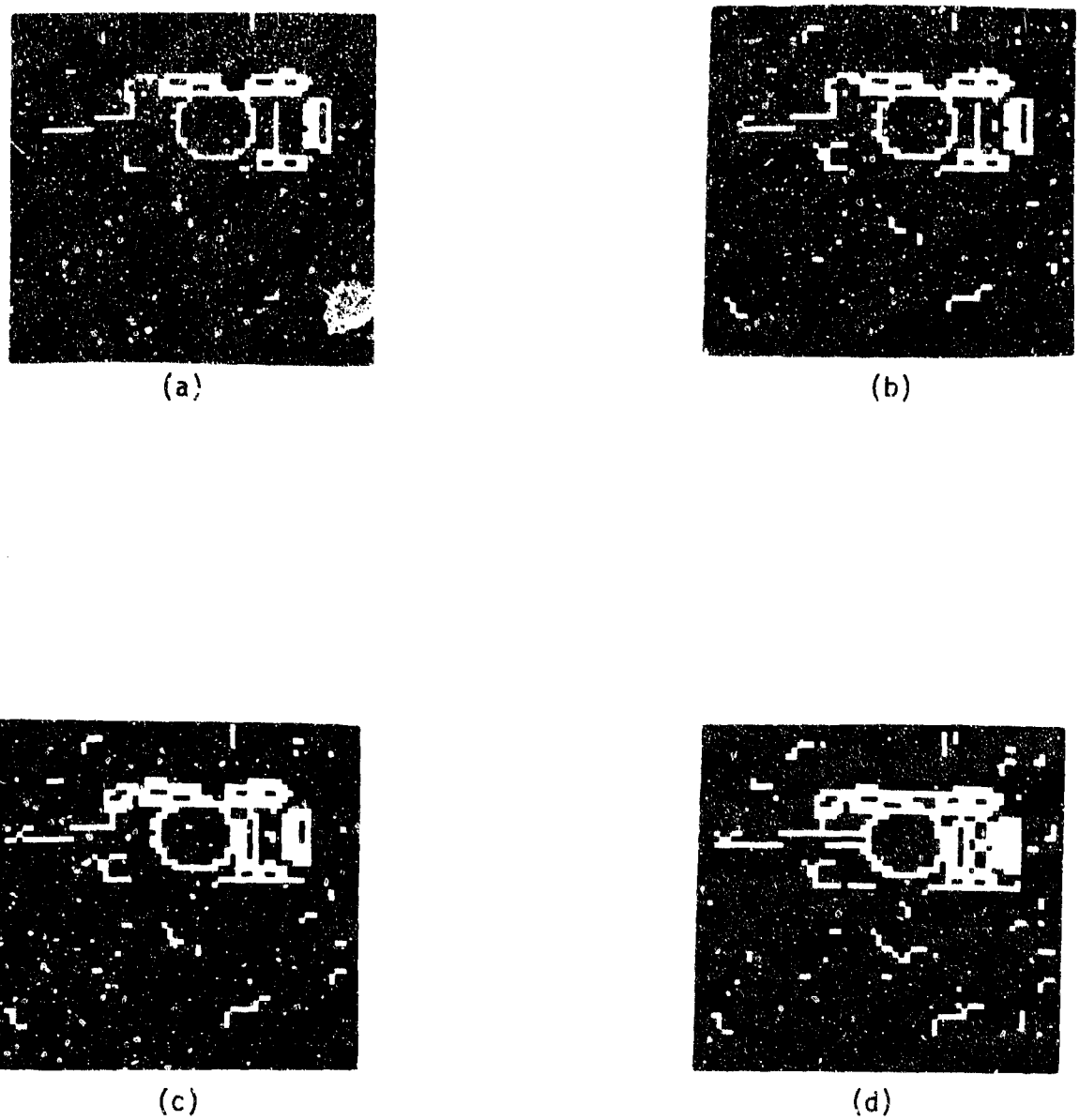


FIGURE 9. EDGES SEGMENTED BY SOBEL OPERATOR USING DIFFERENT THRESHOLD.
(a) THRESHOLD LEVEL IS 23, (b) THRESHOLD LEVEL IS 20,
(c) THRESHOLD LEVEL IS 18, (d) THRESHOLD LEVEL IS 16

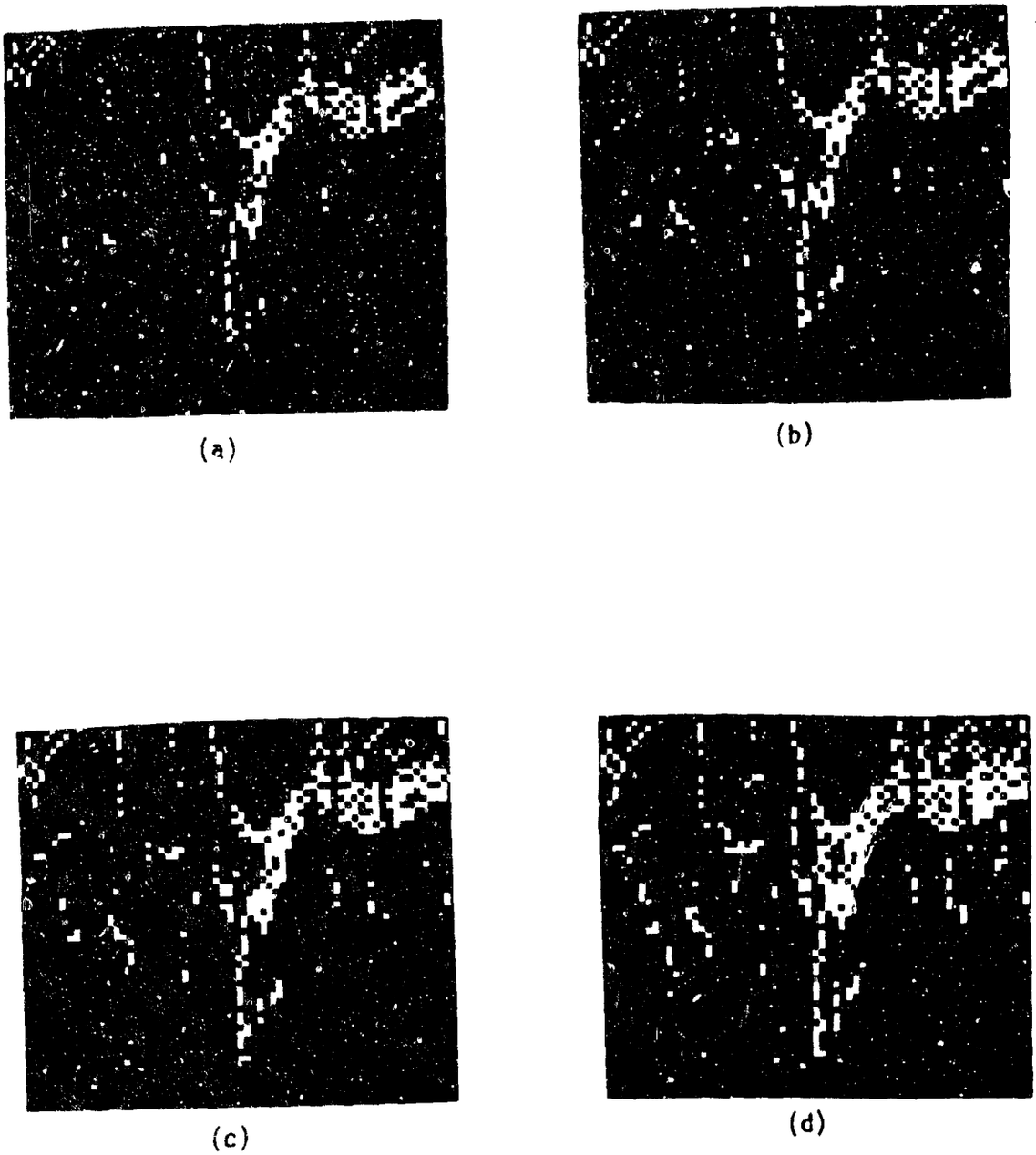


FIGURE 10. POLAR TRANSFORMATION OF FIGURE 9

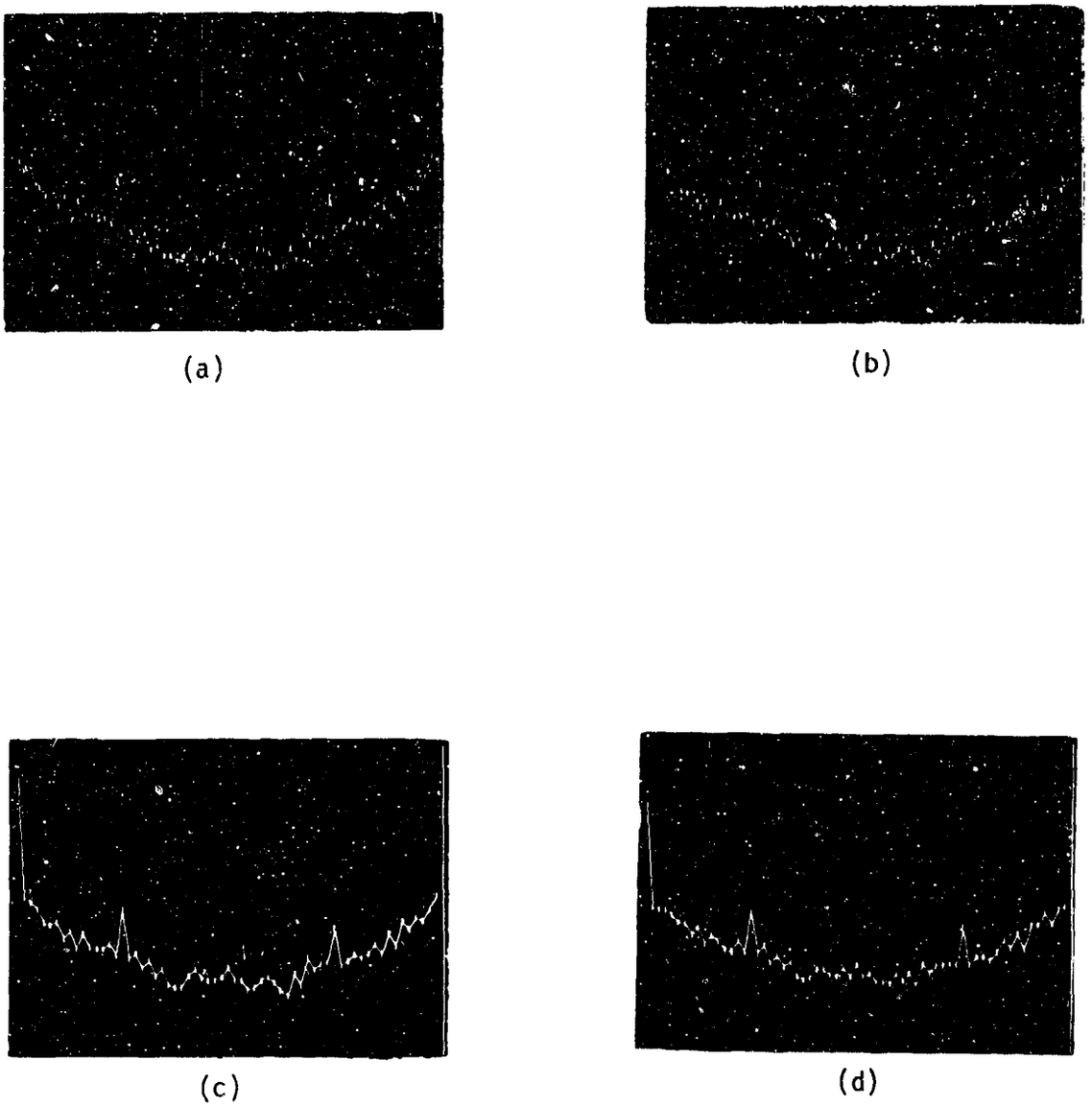


FIGURE 11. CORRELATION RESULTS OF FIGURE 10 USING FIGURE 6 AS REFERENCE IMAGE

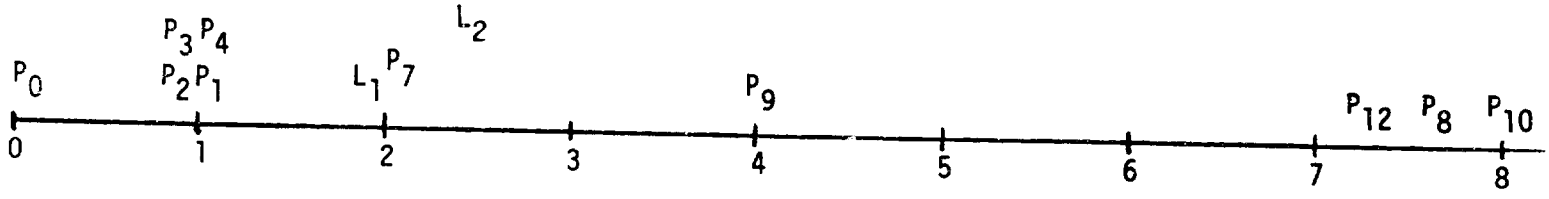
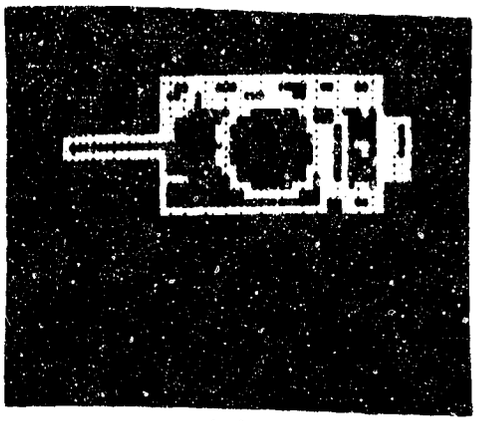
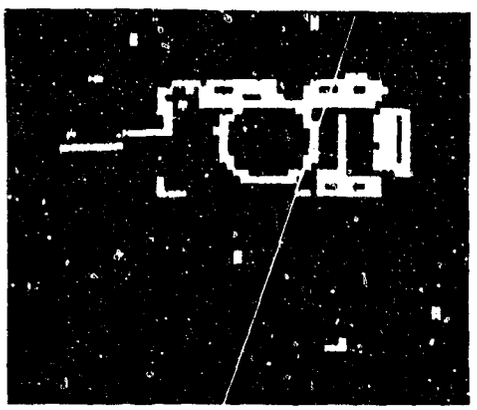


FIGURE 12. THE EUCLIDEAN DISTANCE OF MOMENT INVARIANT FOR THE SENSED IMAGES



(a)



(b)

(c)

FIGURE 13. PRINCIPAL AXES OF IMAGES
(a) PRINCIPAL AXES OF THE IMAGE OF FIGURE 6
(b) PRINCIPAL AXES OF THE IMAGE OF FIGURE 9(a)
(c) PRINCIPAL AXES OF THE IMAGE OF FIGURE 9(d)

SESSION IIB

ADVANCED IMAGING TRACKERS
FOR MISSILE GUIDANCE

Paper No. IIB-1, Presented at the Workshop on Imaging Trackers and Autonomous Acquisition Applications for Missile Guidance, 19-20 November 1979, Redstone Arsenal, Alabama.

AN ITERATIVE FEATURE MATCHING ALGORITHM

A. M. Savol, E. Noges, A. J. Witsmeer

Boeing Aerospace Company
P.O. Box 3999
Seattle, WA 98124

ABSTRACT

In autonomous missile guidance and navigation systems, position update is performed by matching the set of sensed features with a previously prepared reference set contained in the reference feature map.

A novel iterative classifier-feature matcher, MACHAL, is described. This algorithm optimizes the match between a set of sensed and reference features without the customary exhaustive correlation calculations between the two sets. The iterative clustering approach introduced shrinks the data set after each iteration resulting in an overall reduction of the computational burden. Although MACHAL is a general matching algorithm for feature vectors of any dimension, this paper describes its application to the low order image matching arising from autonomous missile navigation and guidance. The generalization of this approach to higher dimensions and other applications such as object recognition is discussed.

INTRODUCTION

Autonomous missile position updating involves a series of technical problems from a variety of disciplines. Among them are sensor type selection (such as imaging vs centroid-range type), feature extractions, reference generation and sensed-reference data matching. This latter problem has received attention at various intensities for several years and is still not an operationally mature discipline. Along with a gradual name change from correlation to pattern recognition, there has been a gradual abstraction of the features to be matched. In this light, correlation is the matching of primitive or basic features while pattern recognition generally means the matching of features which have been extracted from raw data. Traditionally, correlation has been performed exhaustively since local maxima of the correlation figure of merit had no guarantee of being global maxima. The matching of extracted features has generally been performed by associating a new measurement vector into its correct classification niche in the feature space either by proximity to a defining prototype in that n-space, or by dividing the n-space by some linear or curved hyperplanes. The herein described algorithm spans both of these approaches, depending on the sophistication, or its lack, of the extracted features.

After a formal description of the algorithm we apply it to the autonomous navigational problem, thereby demonstrating its simplicity and versatility. We detail the types of features we have tested and give empirical results we have generated from its implementation in our Terminal Guidance Lab facility.

THE ALGORITHM

The problem of matching a set of sensed and reference descriptors reduces to that of choosing a subset {R}* of a set of N reference feature vectors, {R}, which is the best match to the set of K vectors {S} obtained by feature extraction from the sensed data. The MACHAL algorithm implements this process of choosing in an iterative manner by successive exclusion of those reference feature vectors which are least likely candidates for a good match.

Consider a n-dimensional feature space. Let the j^{th} reference and sensed feature vectors be given by

$$R_j = [r_{1j}, r_{2j}, r_{3j} \cdots r_{nj}]^T \quad j = 1,2 \ldots N$$

$$S_j = [s_{ij}, s_{2j}, s_{3j} \cdots s_{nj}]^T \quad j = 1,2 \ldots K$$

Define a distance vector

$$D_{ij} = R_i - S_j = [r_{1i} - s_{1j},\ r_{2i} - s_{2j}, \cdots\ r_{ni} - s_{nj}]^T$$

$$= [d_{1ij}, d_{2ij} \cdots\ d_{nij}]^T$$

and a distance metric

$$M_{ij} = |D_{ij}| = \sqrt{\sum_{k=1}^{n} (d_{kij})^2} = \sqrt{\sum_{k=1}^{n} (r_{ki} - s_{kj})^2}$$

which represents the euclidean distance, or a measure of the mismatch, between the i^{th} reference object and the j^{th} sensed object. At the same time it represents a transformation from n-dimensional feature space to 1-dimensional distance space. $R^n \longrightarrow R^1$, in which our algorithm is defined.

The iterative feature matching algorithm consists of the following sequence of operations:

1. Calculate the matrix of distance metrics, $[M_{ij}]$, $1<i<N$ $1<j<k$

2. Perform clustering in M_{ij}

3. Exclude all members M_{ij} which are not members of either the largest cluster or a cluster with at least K members.

4. Perform "reference thinning" by excluding all members R_i which correspond to the excluded M_{ij} values.

5. Calculate the shifted sensed feature vectors

$$ {}_{1\Delta}S_i = [s_{1j}+\Delta, s_{2j}, s_{3j} \cdots s_{kj}]^T \quad \text{for all } i \underline{\geq} K $$

6. Repeat steps 1-5, using shifts of sensed feature vectors in other coordinate directions. Continue until a single largest cluster remains.

The set of sensed features $\{S_j\}$ are obtained from preprocessed sensor output signals by feature extraction. The noise in the sensed signal augmented by the noise in the preprocessing and feature extraction result in feature vectors containing noise components. The exact nature of this noise is dependent upon the sensor characteristics, scene properties as well as on preprocessor and feature extractor algorithms. For purposes of noise sensitivity studies, it is sufficient to assume that the resultant noise in each component of the sensed feature vector is uniformily distributed with zero mean and maximum amplitude $\pm\frac{A}{2}$. The noise variance is then given by

$$ \sigma_n^2 = \frac{A^2}{12} $$

With the presence of noise in the sensed scene, the resulting match position can be expected to contain errors. These errors are functions of noise as well as the threshold of the clustering portion of the algorithm. For comparative evaluation and parametric studies the absolute error measure E is defined as

$$ E_a = \sqrt{\sum_{j=1}^{n} e_j^2} $$

where e_j is the average absolute error in the j^{th} coordinate match taken over all corresponding feature vector pairs in $\{R\}^*$ and in $\{S\}$. However, the desired figure of merit for matching accuracy should incorporate the degradation caused by system errors, to provide a truer indication of the accuracy of match. This modification is

$$ E = \frac{E_a}{\sigma_n^2} $$

E now provides a measure of how accurately the matching was accomplished in spite of the corruption by noise.

APPLICATION

Our first application, and indeed the original motivation, for this approach was the autonomous navigational problem. The correlator or pattern matcher must take a two dimensional array of gray values, the sensed scene, and "locate" it within a previously prepared reference scene. As eluded to previously, the correlation approach would require the exhaustive testing of all possible placements of the sensed scene over the reference to find the global maximum of agreement. The more modern approach is the extraction of the important features from both sources followed by matching these features If properly executed, this latter approach offers two advantages.

(1) Each extracted feature is the resultant of operations on a neighborhood of pixels. Thus, if noise affecting these pixels is at least partially uncorrelated, the feature is more robust to these corruptions than each individual pixel.

(2) Again because each feature represents a collection of pixels, the computational burden of matching features is greatly reduced.

The second point needs clarification. It may be that the overall burden, including the reduction of source imagery to higher level features, may be greater for pattern recognition than for straight correlation. However, the matching portion of the computation now requires small resources. Since this matching must typically be done in real time and with the more limited resources available on board a missile, the advantage of this approach becomes clearer.

In our first application, optical and radar imagery provided the original reference material. Both from a theoretical and a practical viewpoint, most of the information in an image is concentrated in edges of dissimilar gray values or textures. From the practical viewpoint this seems reasonable because the various elements of a scene may be expected to respond differently to changes in their environment. However, the fact that these elements differ from each other, as seen by various sensors, tends to remain true. Thus the edges become the more stable features. These considerations lead to our decision of using the detected edges as the features to be matched by our algorithm.

Our preferred edge detection algorithm is actually a suite, embodying the "bottom up" approach of growing object boundaries from primitive individual edge elements. Its details and results have been reported previously [1,2] so need not be repeated here. Suffice to say that they reduce the original image, sensed or reference, to a collection of straight lines of various lengths. Figure 1 illustrates a typical final result. In our implementation, therefore, MACHAL was applied to the matching of two collections of straight lines.

In this context we now illustrate how the algorithm finds the best match without the normal exhaustive correlation. Figure 2 shows a larger reference scene and a superimposed sensed scene which have been reduced to their feature lines. Since, for navigational update, the heading and altitude are known, the illustration is devoid of zooming or rotation. However, these degradations can also be handled as will be expanded later. The algorithm now demands that we formulate a metric to quantify the degree of mismatch between features of the two images. For this application, euclidean distances certainly seem appropriate. For ease of illustration, the distance between centroids of the line segments was chosen as the error metric.

The algorithm next requires the computation of the matrix of the mismatch metric. Using the centroid euclidean metric and applying it to the illustrated edges, the matrix M_{ij} is also in Figure 2. Each element of the matrix

represents the degree of mismatch between that sensed element (row) and its associated reference element (column). One dimensional clustering, using integral values as a threshold, reveal that only two clusters, those of values 17 or 21, have sufficient elements to describe the correct match.

Next, the sensed image is moved to a new location and new values for the mismatch matrix must be computed. However, because only a few correspondences were viable candidates, only those values need to be computed. Thismove and its associated matrix are shown in Figure 3. Clustering in this reduced set quickly defines the best match. This example illustrates how this technique permitted the computation of the optimum match without exhaustive correlation. It should be noted that this was possible because of the sparsity of the features. If every pixel were a feature, the algorithm would deteriorate to normal correlation and M_{ij} would be of unmanageable proportions.

GENERALIZATION

Although the algorithm has been illustrated in a two dimensional feature space, its application may be to any dimensional space where a metric for mismatch of the features can be formulated. In the present example of navigational update, if the distortions of rotation or zooming, rather than mere translation, were to be addressed, they would result in either a higher dimensional mismatch matrix or a distance metric which incorporates rotation. This approach has been taken by other workers [3]. In our present formulation, the mismatch metric is a mapping into a 1-dimensional space, here real or integer numbers. The number of dimensions for matching are only limited by the availability of orthogonal feature spaces for the given problem at hand. Our own generalization into a 5-dimensional space, still to this application, is described in the next section.

In a broader sense, however, we believe this approach can be beneficially applied to any problem where the feature space is sufficiently sparse and a metric to quantify the degree of mismatch between feature vector elements can be formulated. The feature axes may represent levels of contrast or geometrical measurements. Therefore, this approach may have merit for object recognition, as well as the scene recognition application of this report. In object recognition, this approach may optimize the search through the associated feature space to classify the object.

IMPLEMENTATION

This algorithm was implemented, using extracted edges as features, and a Varian 72 minicomputer in our Terminal Guidance Lab. Since, at this stage of development, the importance of versatility dominates that of computational efficiency, its coding is in FORTRAN. A RAMTEK color graphics system provides a 2-dimensional display of its actions.

The final products of our edge detection suite are features in a 5-dimensional space. This is because, in addition to the two end points, each edge element has a magnitude associated with it. This magnitude is not its length, but rather a measure of the confidence that the detected edge is a true edge in the original image. The value of this magnitude is a function of the values of the primitive gradients and the linearity of those primitives, which combined to form the large edge element. The actual n-tuple describing each final edge element is

(centroid x, centroid y, angle, length, magnitude)

It thus became possible to apply MACHAL to varying degrees of dimensionality to test the utility of increasing dimensions. As would be expected, if they provide orthogonal information, as they do in this case, the more the better. This need for higher dimensions is primarily, although not exclusively, due to the presence of noise, or dissimilarity of the objects to be matched. Indeed, if the sensed scene is an exact duplicate of its reference, the matching problem becomes trivial. Our implementation therefore includes the measurement of correlation noise.

The two principal limitations to the development of autonomous imaging devices are its great computational burden and the susceptibility of the system to noise. As discussed previously, noise is any effect which causes the sensed scene to differ from the previously prepared reference. One can therefore speak of noise caused by climatic changes, orientation, scaling and inter-sensor noise. Because of this great variety of noise types, no analytical, unified model has been developed, although certain aspects have been addressed [4]. The pseudo-random noise of uniform distribution chosen for our simulations lends itself easily to quantification and may be expanded to model some observed corruptions by varying the noise ranges for the various elements of the feature vectors. In the experiments whose results we describe here, this weighting has not been implemented, although the ranges were normalized for each element type in the 5-tuple. In addition to this perturbation of feature vector elements, randomly generated, entire feature vectors were added to either the sensed or reference scene to simulate missing or additional artifacts. In these experiments, therefore, the live image was merely a chosen subset of the reference subjected to measured feature vector element perturbations to simulate system noise.

EXPERIMENTAL RESULTS

An edge image, the final result of our edge detection suite on an optical aerial view was taken as a reference. It contains 63 edge features. A small subarea, containing 10 edge features, was selected as the sensed scene. The testing then involved the measurement of match accuracy as a function of three independent variables -- the order of the feature space, the amount of additive noise, and the value of the clustering threshold.

Figure 4 is the family of curves generated by measuring the matching error as a function of the dimensionality of the feature space, tested at three noise levels. Even with the perplexing accuracy reversal, the advantage of higher dimensionality is obvious.

Figure 5 represents a study of the clustering threshold vs. noise. Again, matching error is the dependent variable for a family of curves at three noise levels. Because the results are such a mish mash of wildly fluctuating values, a table rather than curves is presented. About the only conclusion one could draw from this is that too large of a threshold may result in failure to find the correct match. The actual failure is that the defining cluster is too large, and cannot be reduced because of the generous clustering threshold. The algorithm vainly keeps circulating the sensed scene without reduction of the matching set (S).

Figure 6 is provided to give the reader a feeling for the great reduction of needed mismatch metric values with succeeding iterations of this algorithm. The curve is simply the means of the number of clusters at each iteration for all the runs reported herein which converged.

DISCUSSION

The details of an algorithm which efficiently matches the elements of a sensed scene with the elements of a reference scene, in the context of a sparse feature space, have been described. The application of this algorithm to autonomous navigational update has been described. Since, in that application, the algorithm obviates the need for normal exhaustive correlation, it results in a dramatic decrease of the computational burden associated with the process of scene matching itself. The reported results of computer simulations further indicate a robustness to noise and particularly, the advantage of utilizing higher dimensional spaces, even for scene matching. This then demonstrates another advantage of this algorithm -- its ease of adaptation to higher dimensional feature spaces. And finally, the adaptability of the algorithm to other applications such as scene recognition has been mentioned.

ACKNOWLEDGMENT

This work was supported by internal research funds of the Boeing Aerospace Company. The authors thank Nadine Parker for her dilligence in the preparation of this manuscript.

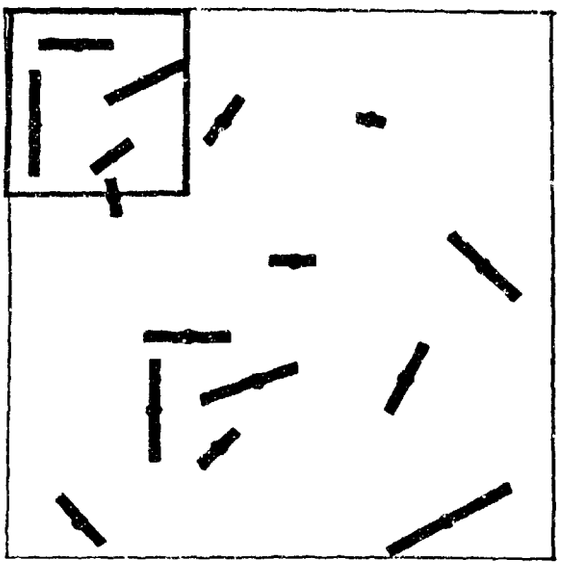
REFERENCES

1. "Development of an Onboard Navigation Update System Utilizing Pattern Recognition", A.M. Savol, E. Noges, A.J. Witsmeer, and J. Geros, Proceedings of the IEEE Computer Society Conference on Pattern Recognition and Image Processing, Chicago, Illinois, May 1978, pp. 91-95.

2. "Pattern Recognition as an Aid to Radar Navigation", A.J. Witsmeer and M. Savol, Proceedings of the National Aerospace and Electronics Conference, Dayton, Ohio, May 1978, pp. 26-32.

3. "Image Registration From Edge Content", G. Stockman and S. Kopstein, Eighth Annual Automatic Imagery Pattern Recognition Symposium Proceedings pp. 139-157, April 1978, Gaithersburg, Maryland.

4. "Quantization Effects on Target Handoff from TV to IR Digitized Scenes", J. S. Boland, III and H. S. Ranganath, Final Technical Report for Contract DAAK40-79-M-0104, under MICOM, September 1979.



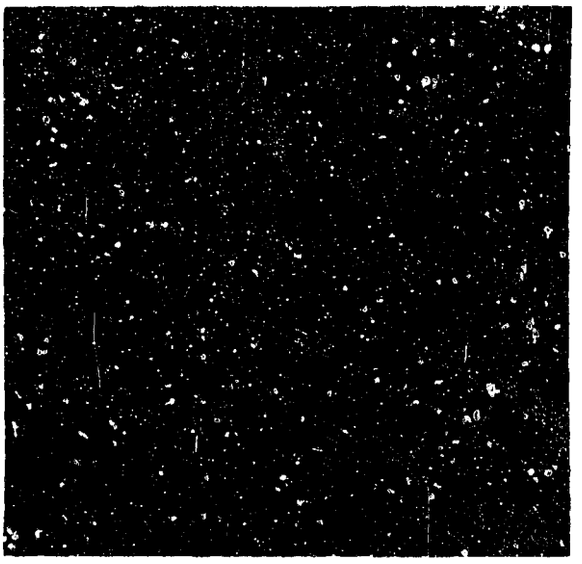


Figure 1

The end product of a typical edge detection process.

	1	2	3	4	5	6	7	8	9	10	11	12
1	6	10	18	16	25	15	18	25	(17)	22	△21	31
2	8	9	(17)	(17)	25	(17)	△21	26	20	26	23	33
3	6	4	12	13	△21	14	(17)	△21	18	24	△21	29
4	2	6	14	12	△21	11	14	20	14	20	(17)	27

Figure 2

Initial position of a sensed scene superimposed over a larger reference scene and the associated matrix of the mismatch metric values.

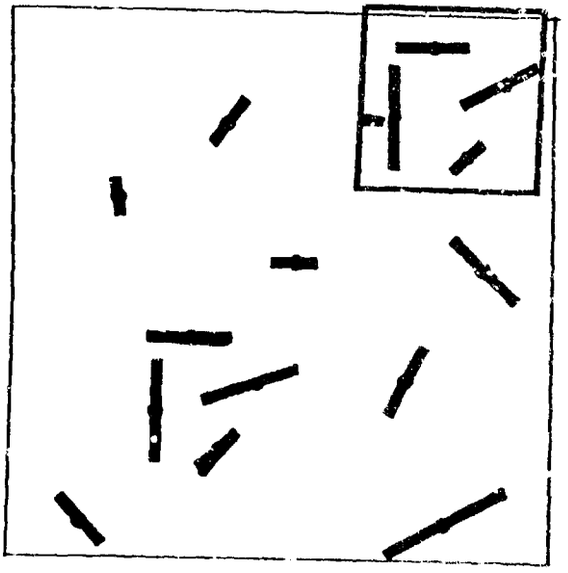


Figure 3

Second and final position of the sensed scene and the associated matrix indicating a correct match.

	1	2	3	4	5	6	7	8	9	10	11	12
1									(22)		20	
2			6	14		(22)	20					
3					10		(22)	17			25	
4					6						(22)	

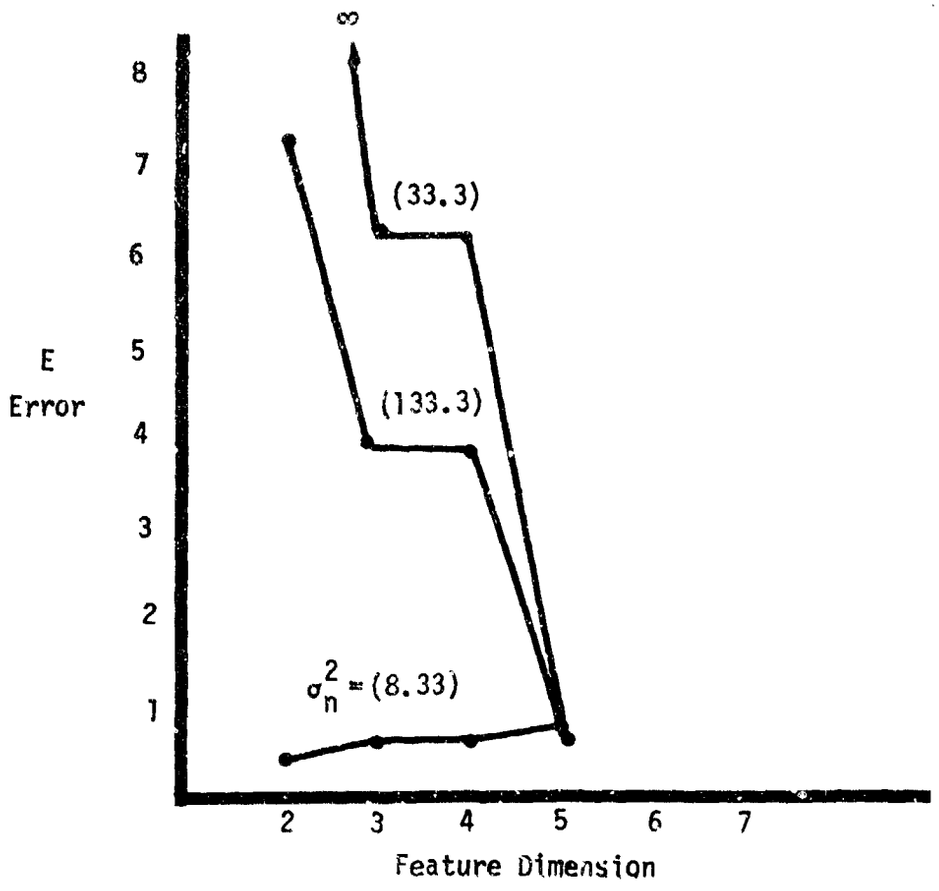


Figure 4 The Advantage of Higher Dimensional Feature Spaces

Noise σ_n^2	Cluster Threshold	Error
8.33	1.5	0.18
8.33	3.0	0.42
8.33	6.0	0.69
8.33	10.0	∞
33.33	1.5	77.01
33.33	3.0	∞
33.33	6.0	8.49
33.33	10.0	∞
133.3	5.0	28.3
133.3	10.5	0.73
133.3	20.0	2.75
133.3	30.0	∞

Figure 5 Clustering Threshold vs. Additive Noise

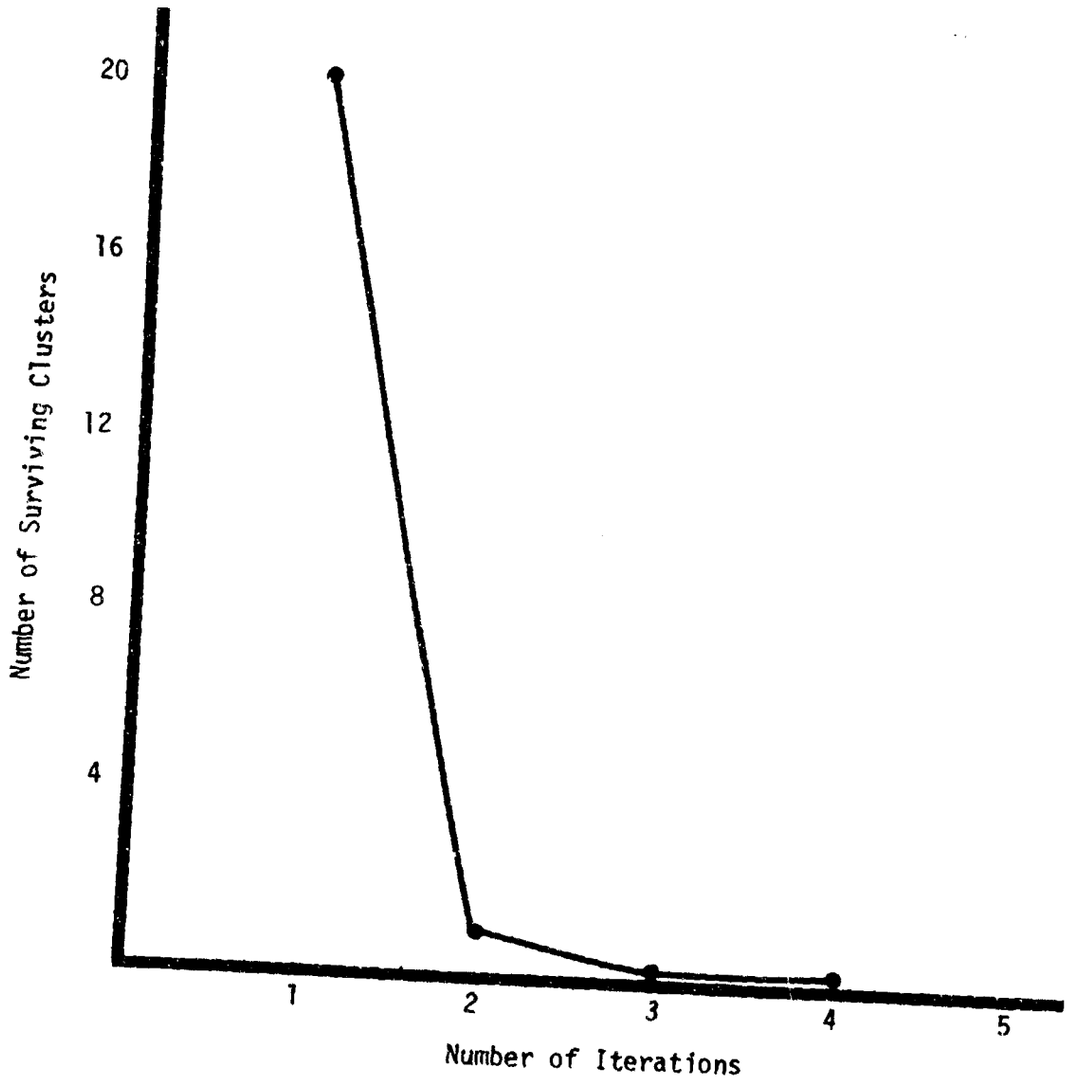


Figure 6 Cluster Reduction by Iteration

Paper No. IIB-2, Presented at the Workshop on Imaging Trackers and Autonomous Acquisition Applications for Missile Guidance, 19-20 November 1979, Redstone Arsenal, Alabama.

IMAGE PREPROCESSING FOR ATHOC IMAGING SENSORS

J. K. YOO, J. J. HOGAN

P. H. McINGVALE, W. W. MALCOLM

ABSTRACT. Low-cost, small Fire and Forget Missiles (F^2M) will provide a cost effective means of minimizing exposure of weapon delivery personnel to the enemy. F^2M requirements can be satisfied by means of automatic handing-off the target from an HRS to an LRS. This paper describes the methodology to select image preprocessing techniques for an Automatic Target Hand-Off Computer (ATHOC) and several preprocessing options. Experimental results utilizing real-world imagery are reported and evaluation procedures of the results to select a proper set of preprocessing techniques are discussed.

A. INTRODUCTION

In this paper, the methodology to select image preprocessing techniques for an Automatic Target Hand-Off Computer[1] (ATHOC) and several options are discussed. The objective of the study is to select a proper set of preprocessing techniques that will support all the ATHOC requirements.

The ATHOC is basically a microprocessor-based digital correlator. The ATHOC assembly consists of four sections: a microprocessor section to perform arithmetic and logical operations, a video preprocessor section to digitize and scale incoming video, a correlator section to perform real-time area correlation, and a signal conditioning section to generate scaled gimbal command signals. Detailed description of ATHOC hardware is presented in references.[1]

The ATHOC system was designed to initially operate with imagery from two TV sensors[2] With video from a TV High Resolution Sensor (HRS) and an IR Low Resolution Sensor (LRS), however, ATHOC must perform special preprocessing of the video before cross-correlating the two sensor images in order to accommodate various peculiarities of the two sensors, such as different resolutions, different scale factors, different spectral responses, different scan formats, etc. Cross-correlation is used to boresight the LRS to the target selected through the HRS.

J. K. Yoo and J. J. Hogan are with Goodyear Aerospace Corporation, Akron, Ohio, 44315. W. W. Malcolm and P. H. McIngvale are with the Guidance and Control Directorate, Technology Laboratory, MICOM, Redstone Arsenal, Huntsville, Alabama, 35809. This work was performed at Goodyear Aerospace Corporation and supported in part by U. S. Army MICOM under Contract DAAK40-79-C-0134, and by Goodyear Aerospace IR&D.

In order to accomplish hand-off using image matching techniques, the two images from the HRS and LRS sensors must be similar. However, the unprocessed images from the LRS and rescaled HRS are generally dissimilar. Under this condition, considerable image preprocessing is required to extract common information from those images. The basic technique exploited to accomplish the extraction process uses shape-based image matching techniques, which have been shown to be more desirable than signal-based matching techniques.

Computer simulation of image preprocessing techniques, also discussed here, has been developed for the testing and evaluation of ATHOC preprocessing options. The various preprocessing algorithms are examined by means of the simulation facility and real-world imagery. The preprocessing involves noise cleaning, contrast enhancement, edge enhancement, slicing, and other support functions.

Since the time required for the handoff also is of importance, consideration has been given to select preprocessing algorithms which are feasible in real-time and that can be programmed on a digital programmable central microprocessor in ATHOC.

B. OBJECTIVE OF IMAGE PREPROCESSING

In the study on image preprocessing for ATHOC, the central problem is how to extract common information from two sensor images, HRS and LRS. The irrelevant background information and contrast reversal problems must be handled accordingly to give an optimum efficiency to the image matching functions.

In general, the edge or feature extraction process from the given sensor images is trivial if the different objects can be identified easily by measuring the intensity differences. In order to extract edges or features from a scene, we must somehow single out and mark the pixels that belong to those features in a special way. In practice, however, the parts of images are not clearly contrasted, and it is not easy to select edges.

From the above consideration, it is natural to design preprocessing techniques which will enhance selected features against irrelevant data to aid in extracting edge portions. The preprocessing function will generate a picture F' from the original image F so that the edge can be extracted easily. In the new image F', the edges to be extracted have characteristic gray level ranges; hence we can use thresholding techniques by employing a proper gray level threshold. For example, if some local property such as the digital gradient or Laplacian[3] has a higher average gray value at points of the edges than at the background points, then we can use F' derived from the local property to obtain a threshold.

Another consideration given to the preprocessing function is how to handle noise if it exists in the picture. A technique[4] to remove noise is to compare the gray level I at one point with statistical gray level I_a at its neighboring points. If I does not satisfy certain relationships to I_a, we can consider this point as a noise point. But we should exercise care in applying this technique because, if applied indiscriminately, it tends to blur the picture, which is objectionable. This noise removal procedure requires several parameters that can be adjusted to suit the characteristics of the noise if they can be detected.

In summary, the study on image preprocessing for ATHOC imagery is intended to provide data that may answer the following questions: 1) Is noise cleaning beneficial? If yes, which algorithm is most efficient? 2) Which edge enhancement algorithm is most efficient? 3) Which slicing algorithm is most efficient?

C. EVALUATION METHODOLOGY

The answer to the above questions may be obtained by an organized series of tests. Figure 1 shows the main steps involved in the overall simulation test.

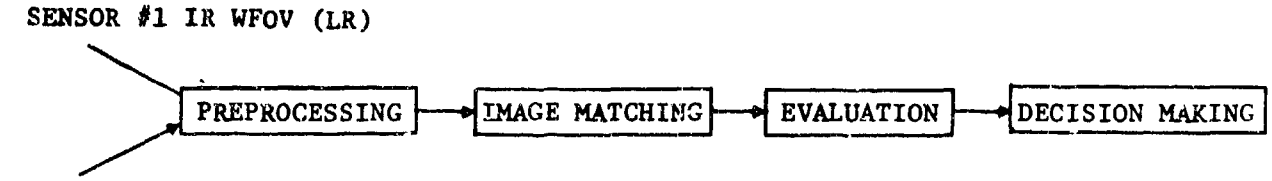


Figure 1. Preprocessing Evaluation for ATHOC Imagery

Preprocessing algorithms are applied in series and resulting images are correlated to see if the specific preprocessing algorithm improves the correlation performance. Figure 2 shows the flow for the preprocessing application. Two noise cleaning algorithms, five edge enhancement algorithms, and three slicing algorithms were tested. These and other algorithms have been incorporated in a software simulation program called GIPSY (Goodyear Image Preprocessing System).

D. PREPROCESSING TECHNIQUES

In this part, the preprocessing techniques studied are discussed.

1. Noise Cleaning Algorithms

a. Low Pass Filtering. An image may have noise from several sources including electrical sensor noise, channel errors, etc. These noise effects can be minimized by classical statistical filtering techniques available in the literature.[5]

Image noise appears as discrete isolated pixel variations that are not spatially correlated. Pixels that are in error often appear markedly different from their neighbors. Noise in an image generally has a higher spatial frequency spectrum than the normal components because of its spatial decorrelatedness. Hence, simple low-pass spatial filtering can be effective for noise smoothing.

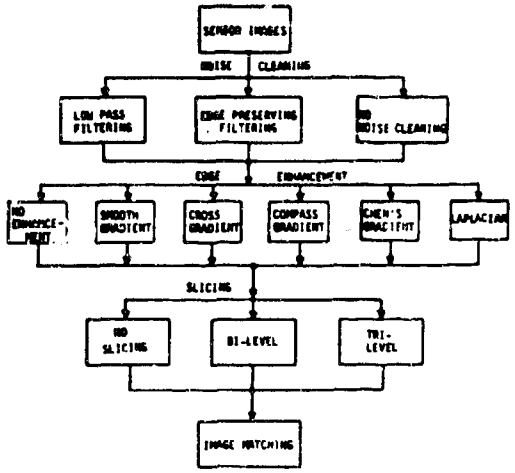


Figure 2. Flow Diagram of Video Preprocessing Test

A filtered output image F' is formed by discrete convolution of the input N x N image array F with the L x L convolution array H according to the relation

$$F'(m_1,m_2) = F(n_1,n_2) * H(m_1 - n_1 + 1,\ m_2 - n_2 + 1)$$

For noise smoothing, H should have a low-pass characteristic with all positive components. We used the following array for the present experiment:

$$H = 1/9 \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

b. Edge Preserving Noise Cleaning.[6] Such noise cleaning algorithms as low-pass filtering or smoothing have a basic difficulty that, if applied without care, tends to blur any sharp contrasted edges which are considered to be good information content. The edge preserving filter discussed here and used for experimentation was selected to resolve the conflict between noise elimination and edge degradation. It looks for the most homogeneous neighborhood around each point in an image, and then gives each point the average gray level of the selected neighborhood area. Noise in the image is removed by the usage of this method, while the edges remain sharp. The approach used in the experiment is as follows:

1. Compute four averages, a_1, a_2, a_3, and a_4 for four different neighborhood windows respectively.

   Those windows are defined as w_1 = (A, B, D, X), w_2 = (D, X, F, G), w_3 = (B, C, X, E), and w_4 = (X, E, G, H).

A	B	C
D	X	E
F	G	H

2. For four average values, compute $|a_j - X|$ and find a_j which gives minimum difference.
3. Then replace the old X by the value of selected a_j.

2. Edge Enhancement Algorithms

A variety of edge enhancement algorithms are available and implemented in GIPSY. Thus far, Laplacian, smooth gradient, cross gradient, compass gradient, and Chen's gradient have been applied.

a. Laplacian. A Laplacian mask can be used to sharpen edges without regard to edge direction. Several types of Laplacian masks are used in the literature.[3] We used the most common mask:

$$H = \begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$$

b. Smooth Gradient. A 3 x 3 nonlinear edge enhancement operator has been suggested by Sobel[7] as a bi-directional gradient operator.

In the experiment, we used a smooth gradient operator instead, which is very similar to the Sobel operator except for the equal weights.

$$\begin{bmatrix} 1 & 0 & -1 \\ 1 & 0 & -1 \\ 1 & 0 & -1 \end{bmatrix} \quad \text{and} \quad \begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix}$$

c. Cross Gradient. Instead of using a rectangular window, we used a cross gradient operator because of the simpler hardware implementation and computational procedure. The operator is defined as follows:

$[1\ 1\ 1\ \ldots\ 1\ 0\ -1\ -1\ -1\ \ldots\ -1]$ for the horizontal direction, and

$[1\ 1\ 1\ \ldots\ 1\ 0\ -1\ -1\ -1\ \ldots\ -1]^t$ for the vertical direction.

The number of 1's and -1's can be selected as an option. We used three 1's and -1.

d. Compass Gradient.[8] Two-dimensional discrete differentiations can be performed by convolving the original image array with the compass gradient masks. Several compass gradient masks are defined in the GIPSY. We used simple 5-level masks as shown.

North
$$\begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$$
Northwest
$$\begin{bmatrix} 2 & 1 & 0 \\ 1 & 0 & -1 \\ 0 & -1 & -2 \end{bmatrix}$$
West
$$\begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix}$$
Southwest
$$\begin{bmatrix} 0 & -1 & -2 \\ 1 & 0 & -1 \\ 2 & 1 & 0 \end{bmatrix}$$
South
$$\begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$$
Southeast
$$\begin{bmatrix} -2 & -1 & 0 \\ -1 & 0 & 1 \\ 0 & 1 & 2 \end{bmatrix}$$
East
$$\begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$$
Northeast
$$\begin{bmatrix} 0 & 1 & 2 \\ -1 & 0 & 1 \\ -2 & -1 & 0 \end{bmatrix}$$

e. Chen's Gradient.[9] This modified gradient operation takes the product of four conventional gradient operations in different directions. For a 16-point array, the Chen's gradient is

$$\sqrt[4]{a\ b\ c\ d} \quad \text{where} \quad a = .5\ (|F-K| + |J-G|),\ \ b = .5\ (|A-P| + |M-D|),$$
$$c = .5\ (|B-O| + |I-H|),\ \ d = .5\ (|C-N| + |E-L|).$$

A	B	C	D
E	F	G	H
I	J	K	L
M	N	O	P

3. Slicing Algorithms

a. Tri-Level Slicing. Based on recent studies[10] and experiments, use of a tri-level algorithm provides a desirable combination of acceptable performance and simple implementation. Improvements when using more than three levels are obtained only under special conditions. To improve performance by increasing the number of slice levels it is required that the average object size in the scene decrease as slice levels increase.

Following is a brief description of the logic used for tri-level slicing of sensor data. There are a number of possible thresholding or signal slicing methods that may be used to delete unwanted information or to emphasize desired information in the process of converting multilevel data to tri-level for subsequent matching. The basic method used is to measure the mean (μ) and standard deviation (σ) of the data and set threshold values that are proportional to these measurements. The relationships are:

$$\text{If } V(t) \geq \mu + (K_u)\sigma \text{ , then } b(t) = 2$$

$$\text{If } \mu + (K_u)\sigma > V(t) \geq \mu - (K_\ell)\sigma \text{ , then } b(t) = 1$$

$$\text{If } V(t) < \mu - (K_\ell)\sigma \text{ , then } b(t) = 0$$

where $V(t)$ = the instantaneous value of a multi-level signal

$b(t)$ = the corresponding instantaneous value of a sliced signal

K_u = the sigma multiplier for $V(t)$ greater than the mean; i.e., to establish the upper threshold

K_ℓ = the sigma multipler for $V(t)$ less than the mean; i.e., to establish the lower threshold

Values of $K_u = K_\ell = 0.43$ were used. This provides a uniform distribution of the three slice levels if the input scenes have a normal distribution. A limited survey of input scene histograms indicates that they do not have a normal distribution. However, this does not appear to have a significant effect on the distribution of slice levels.

b. Bi-Level. Bi-level slicing of sensor data is achieved by using only the mean (μ) computed from each rectangular window. The relationships are

$$\text{If } V(t) \geq \mu \text{ , then } b(t) = 1$$

$$\text{If } V(t) < \mu \text{ , then } b(t) = 0$$

c. Rayleigh Slicing. The gradient operation computes the gradient vector of a pixel from its X and Y components as discussed in the part of edge enhancement algorithms, here. Bi-directional enhancement algorithms, specifically, can use the following set of equations to combine both or select the maximum gradient:

(1) $X^2 + Y^2$; (2) $|X| + |Y|$; (3) $\text{MAX}\{|X|, |Y|\}$.

If we assume the original image is normally distributed, then applying the first equation will result in a gradient image with probability density function of Rayleigh distribution.[11]

$$f\left(\sqrt{X^2 + Y^2}\right) = f(r) = \frac{r}{\gamma^2} \exp\left(\frac{-r^2}{2\gamma^2}\right) U(r) \tag{1}$$

Now, in order to slice the gradient image into bilevel or trilevel images, the statistics of the image is used.

$$\text{Let } \mu_1 = \frac{r}{\gamma} \text{ in (1) ; Then } F(\mu_1) = 1 - e^{-\frac{\mu_1^2}{2}} \tag{2}$$

By setting for bilevel slicing as $F(\mu_1) = .5$ and solving Equation (2) we obtain

$$\mu_1 = \frac{r}{\gamma} = 1.18. \tag{3}$$

For trilevel slicing, setting Equation (2) to 1/3 and 2/3 for lower and upper level respectively, we obtain

$$\mu_1 = \frac{r}{\gamma} = 90 \tag{4}$$

$$\mu_1 = \frac{r}{\gamma} = 1.48 \tag{5}$$

To derive the relationship between the standard deviation σ and the value of γ, compute

$$< r > = \int_0^\infty f(r) \cdot r \cdot dr = \sqrt{\frac{\pi}{2}}\,\gamma \tag{6}$$

$$< r^2 > = \int_0^\infty f(r) \cdot r^2 \cdot dr = 2\gamma^2 \tag{7}$$

Now, from (6) and (7), $\sigma^2 = < r^2 > - < r >^2 = (2 - \pi/2)\gamma^2$. (8)

Therefore $\gamma = \sqrt{2.33}\,\sigma$. From Equations (3) through (5) and (8), we obtain the threshold values for bilevel and trilevel. $r = 2.75\sigma$ for bilevel, and $r = 1.38\sigma$ for lower level, and $r = 2.26\sigma$ for upper level in trilevel.

E. RESCALING ALGORITHM

Since the two sensor images are generally different in scale, rescaling is necessary prior to correlation. Assume original and rescaled images as $A(i,j)$ and $B(k,\ell)$ respectively. The scale factors in horizontal and vertical directions SX and SY are: SX = (resolution in B in horizontal direction)/(resolution in A in horizontal direction); SY = (resolution in B in vertical direction)/(resolution in A in vertical direction. The simplified rescaling algorithm will integrate pixels of SX by SY rectangular window from A and create a single pixel in B.

The following steps are taken for the rescaling and Figure 3 shows the definition of each parameter. The operator $\lfloor\ \rfloor$ indicates the truncation of the fraction.

1. $ISX = \lfloor SX + .5 \rfloor$, $ISY = \lfloor SY + .5 \rfloor$
2. Total number of pixels in B: $K = \lfloor I/SX \rfloor$, $L = \lfloor J/SY \rfloor$
3. Compute pixel B (k,ℓ), where $k = 1, 2, 3, \ldots, K$, and $= 1, 2, 3, \ldots, L$

$$B(k,\ell) = \sum_{i=j}^{p} \sum_{j=j}^{q} A(i,j)$$

where $i = \lfloor k * SX + .5 \rfloor - ISX + 1$, $j = \lfloor \ell * SY + .5 \rfloor - ISY + 1$, $p = \lfloor i + ISX - 1 \rfloor$, $q = \lfloor j + ISY - 1 \rfloor$.

Figure 3. Rescaling Algorithm

F. SCENE MATCHING ALGORITHMS

As pointed out previously, the method used for evaluating effectiveness of the image preprocessing is correlation of the video data. Previous work[10] at GAC provided a correlation program that here was adapted for application to the ATHOC tests. The correlation program consists of several algorithms that relate to coarse-fine search, number of levels correlated in the video data, and provisions for match point validation. The matching algorithms are discussed in this part, and the matchpoint validation algorithms will be presented in the forthcoming contract report.

a. Coarse-Fine Search. All tests used a coarse-fine search procedure. For coarse search the large and small images were both sampled to be smaller in each dimension by a factor of 2. For each test the coarse sampling was performed by simply using only the even-numbered rows and columns of the original imagery. Fine search was performed by generating a 9 x 9 match surface centered at th coarse search match peak location. The original live and reference images without sampling were used for fine search.

b. Multi-Level Correlation. Multi-level correlation requires 8-bit per pixel for correlation. The normalized absolute difference measure can be obtained by normalizing each correlation rectangular window first, then applying mean absolute difference computation. The normalization is achieved by the following procedure: Normalized Pixel = (pixel - mean)/ (standard deviation) where mean and standard deviation are statistics of each rectangular correlation window.

c. Bi-Level Correlation. Once a slicing or thresholding operation is applied to multi-level edge-enhanced images, a bi-level image is obtained. The absolute difference measure between two bi-level images does not need any normalization to the images before correlation computation.

d. Tri-Level Correlation. The internal characteristic of tri-level slicing supports normalization procedures. Therefore, simple absolute difference will do the correlation computation.

e. Normalization of Correlation Surface. The correlation surface amplitudes are normalized so that a perfect auto-correlation peak is unity while the average of off-match point values approaches zero. The necessary normalization expressions are derived below. The GAC correlation surfaces use the absolute difference between the reference and live images as a measure of the match between the two arrays. For example, a point i in the correlation surface has a per-pixel average difference, D_i, given by

$$D_i = \frac{1}{N} \sum_{j=1}^{j=N} |R_j - L_j| \tag{9}$$

where R_j and L_j represent the values respectively of the reference and live arrays of N pixels each at the point i. The difference D_i is normalized so that a perfect match (when $D_i = 0$) becomes unity and the average off-peak match when $D_i = E[D_i]$ becomes zero. Hence the normalized match ϕ_{Ni} is

$$\phi_{Ni} = 1 - D_i / E[D_i] . \tag{10}$$

The average off-peak match $E[D_i]$ is evaluated from the statistics of the live and reference values, R_j and L_j. Assuming pixels have a gaussian distribution with a nominally zero average $E[A]$ and a standard deviation, σ, the corresponding statistics of the individual pixel difference, $d_j = R_j - L_j$, are

$$E[d_j] = 0 \; ; \; \sigma_{d_j} = \sqrt{2}\,\sigma \tag{11}$$

The statistics of (11) assume off-match independence between the live and reference pixels, and that they have similar gaussian distribution with the same average and standard deviations. The desired average $E[D_i]$ is equal to the average pixel absolute difference; i.e.:

$$E[D_i] = E[|d_j|] = \frac{2}{\sqrt{2\pi}\,\sqrt{2}\,\sigma} \int_0^\infty d_j \exp\left(\frac{-d_j}{4\sigma^2}\right) dd_j \tag{12}$$

Solution of (12) produces: $E[D_i] = 2\sigma/\sqrt{\pi}$. The values of σ, the standard deviation of the reference, are evaluated for 3 cases, bi-level, tri-level, and multi-level surfaces. The bi-level and tri-level surfaces assume a uniform distribution of the 2 or 3 intensity levels, so that $\sigma = 1/2$ and $2/3$ for respectively the bi-level and tri-level surfaces. For the multi-level surfaces, σ is computed for each case from the actual intensity distribution.

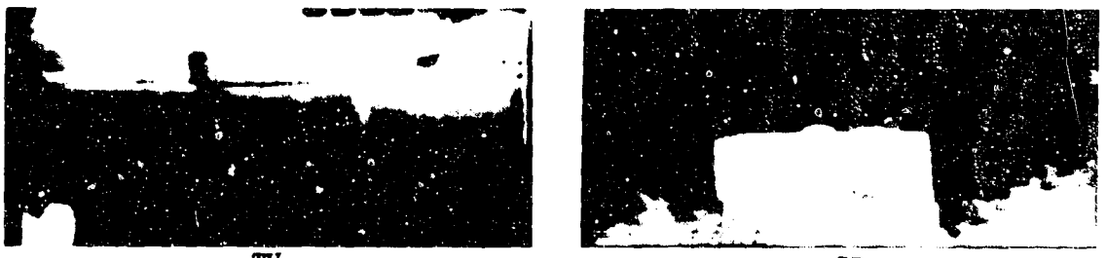
G. EXPERIMENTAL RESULTS

1. Sensors and Scenes

The HR TV imagery were obtained from the Stabilized Platform Airborne Laser System (SPAL) which contains a narrow field of view silicon vidicon. The LR infrared missile seeker input imagery was generated using an IR sensor unit. The characteristics of those sensor units are:

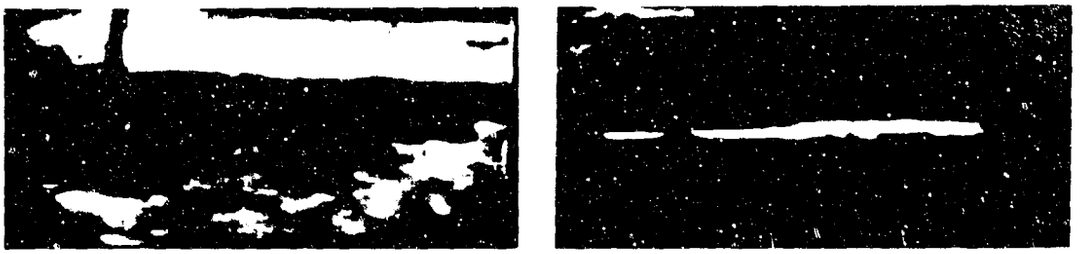
	HR VISUAL	LR IR
SENSOR:	SPAL NFOV TV	WFOV IR WH
FOV:	$.5^{o}$ x $.5^{o}$	2.25^{o} x 2.25^{o}
RESOLUTION:	Horizontal 33.56 μrad/pixel (5MHz)	Horizontal 159.6 μrad/pixel (independent)
	Vertical 36.36 μrad/TV line/field	Vertical 490 μrad/detector (independent) or 163.33 μrad/TV line/field

Six different scene pairs of low resolution IR and high resolution TV were studied. Two of them are shown in Figures 4 and 5.



TV IR

Figure 4. Scene 1 - NASA Tower



TV IR

Figure 5. Scene 4 - Parking Lot

The input imagery contained 512 x 480 pixels per frame (two fields). For use in the evaluation tests only one field and every other pixel in each line of video were used resulting in 256 x 240 pixels imagery.

2. Rescaling of Spatial Resolutions of HRS and LRS Images

The difference in resolution of HRS and LRS videos is caused by the differing fields of view, number of IR detectors, TV lines per frame, frame rate, aspect ratio, and sampling rate of the two sensor systems. The resolutions of the two images are rescaled to have the same spatial resolution in the following way: Case 1 Vertical scale factor between TV image and IR image is 163.33/ 36.36 = 4.49, and horizontal scale factor between TV image and IR image 159.64/ 33.56 = 4.76. Case 2 Another consideration is given to the scale reduction of

only independent IR detectors. Since each detector is read out three times in the IR sensors, the vertical scale factor is three times larger than that in Case 1: Vertical scale factor is 490/36.36 = 13.48, and horizontal scale factor is 4.76. Besides these scale reductions by scale factor of cases 1 and 2, every other pixel is used to equalize the scale in horizontal and vertical directions in one field of TV images. After the rescaling operation, case 1 gives 256x 240 for HRS and 53 x 53 for LRS and case 2 gives 256 x 80 for HRS and 53 x 17 for LRS. Figure 6 shows the case 1 rescaling of scene 1 (Figure 4).

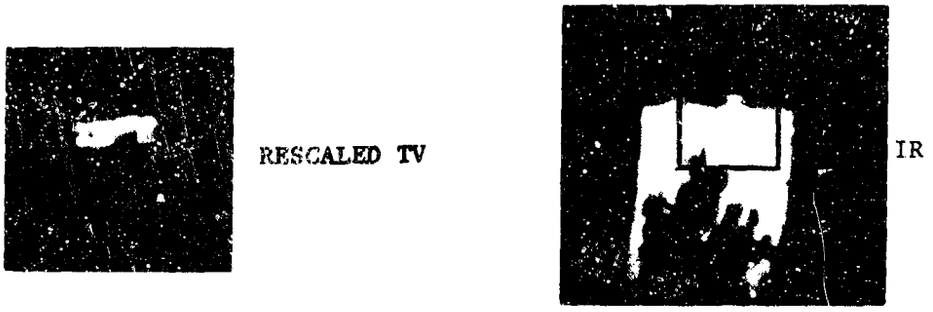


Figure 6. Example of Rescaled Image
(Framed portion in IR is the estimated target window)

3. Target Designation and Estimation of Target Location

The following steps were taken to designate target locations in high resolution TV images and locate estimated target position in each corresponding los resolution IR images.

1. (a) Designate a specific point which is easily identifiable in both LR and HR scenes, or (b) Designate a target as a center of the HR TV image.
2. Measure the corresponding target location in the LR IR image.

According to the procedure 1(a), we obtained the following set of designated target (x,y) and estimated match location (x,y) as shown in Table I (a). If only independent detector lines are used following the procedure 1(b), the locations of the designated and the estimated targets will be one-third of the values given in the Table I(a) as shown in Table I(b). The two tables show the locations measured in TV frame resolution scale.

TABLE I(a). TARGET COORDINATE

	DES. TARGET		EST. TARGET	
SCENE NO.	x	y	x	y
1	21.7	18.3	127.3	128.7
4	19.8	36.2	85.0	136.1

TABLE I(b). TV IMAGE CENTER COORDINATE

	DES. TARGET		EST. TARGET	
SCENE NO.	x	y	x	y
1	27	27	132	138
4	27	27	92	127

Although the specific target points selected in the HRS TV images generally were points that could be recognized with respect to a known object in the scene, it turned out to be nearly impossible to accurately identify the corresponding point in the LR images. Both the resolution difference and the contrast difference contributed to the difficulty. This inability to precisely pinpoint the selected target points in the two different sensor images may have caused some displacements in the correlation test outputs, the values of which were a measure of the inaccuracy of the operation. The values obviously were a function of scene characteristics. In some cases the displacements were within a few pixels.

4. Preprocessed Images

Figure 7 (two sheets) depicts the preprocessing results. Scene No. 1 is used as an example. For each scene, photographs were made of CRT displays of both the HR TV images and the LR IR images. The matrices were organized to illustrate in picture form the effects of the application of preprocessing algorithms to the scaled imagery. The picture in each upper left corner has had no intensity or amplitude preprocessing. Pictures in the left column resulted from edge enhancement preprocessing of the original image using only the smooth gradient algorithm, only the cross gradient algorithm, etc. Pictures in the second column illustrate the effect of applying the same edge enhancement algorithms to the original image after noise cleaning by low-pass filtering. And pictures in the third column illustrate the effect of applying the same edge enhancement algorithms to the original image after noise cleaning with the edge preserve algorithm. The CRT displays were enhanced in most cases to accommodate the photo process requirements.

5. Correlation Results

The ATHOC simulation facility includes a host digital computer system, an associative array processor, and reference image generation equipment. The host Sigma 9 digital computer has 128K 32-bit words, four 86M byte disk systems, four IBM-compatible 800/1600 bpi 9-track magnetic tape drives, and remote time sharing terminals.

Most of the simulation software is written in Sigma 9 FORTRAN language utilizing time sharing terminals. Non-real time scene matching simulation is implemented within the scope of this digital computer system. In the rest of this part, a typical correlation result is presented. Figure 8 shows the bi-level correlation result between rescaled HRS and LRS images of NASA Tower (Figures 4 and 6). Coarse and fine search sequence was applied. For coarse search, 46 x 44 rescaled HRS TV image and 224 x 220 LRS IR image were used. The estimated target location was (132, 138) in (x,y) coordinates. The answer obtained from the coarse search was (134.4, 138.9) with correlation amplitude of .597. For the fine search, 46 x 44 HRS and 54 x 52 LRS images were used to obtain 9 x 9 correlation surface. The result was (134.7, 138.9) with a correlation score of .607.

6. Performance

Test results are discussed in this section. Tests are performed by using several search modes: (1) Extended area/Limited area Search; (2) Multi-/Bi/Trilevel Search; (3) Coarse-Fine/Fine Search; and (4) TV Rate/Independent Detector Lines Search.

Limited search only searches limited area of large array around the estimated target location. Extended search searches all over the large scene. The limited search is tried to save computer time and it indicates the effect of preprocessing operations on the correlation performance. Multilevel, bilevel, and trilevel correlations are performed to investigate the effects of preprocessing algorithms on each slicing method. Coarse-Fine Search combination is tried to save computation time. Coarse search uses only every other pixel in every other line. After a coarse search, the location with the highest correlation score among the correlation results of three slicing algorithms is selected as a center of search for the fine search. The search window is 9 x 9 around the coarse answer. This approach is taken because higher correlation score indicates better match quality in principle. However, this method could prevent from applying the correct matchpoint with lower score in another slicing method.

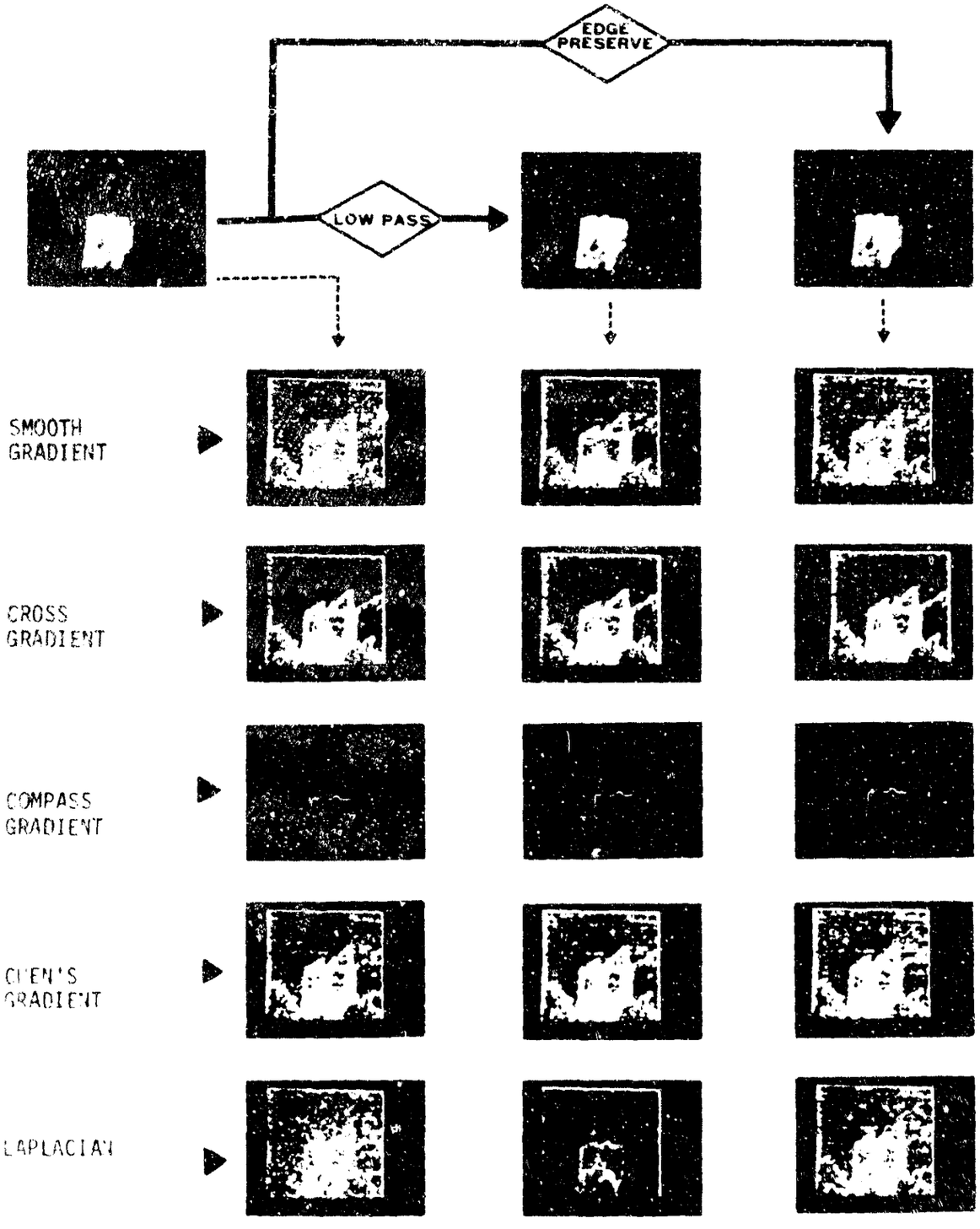


Figure 7(a). Scene No. 1 (NASA Tower) - IR Image Preprocessing Results

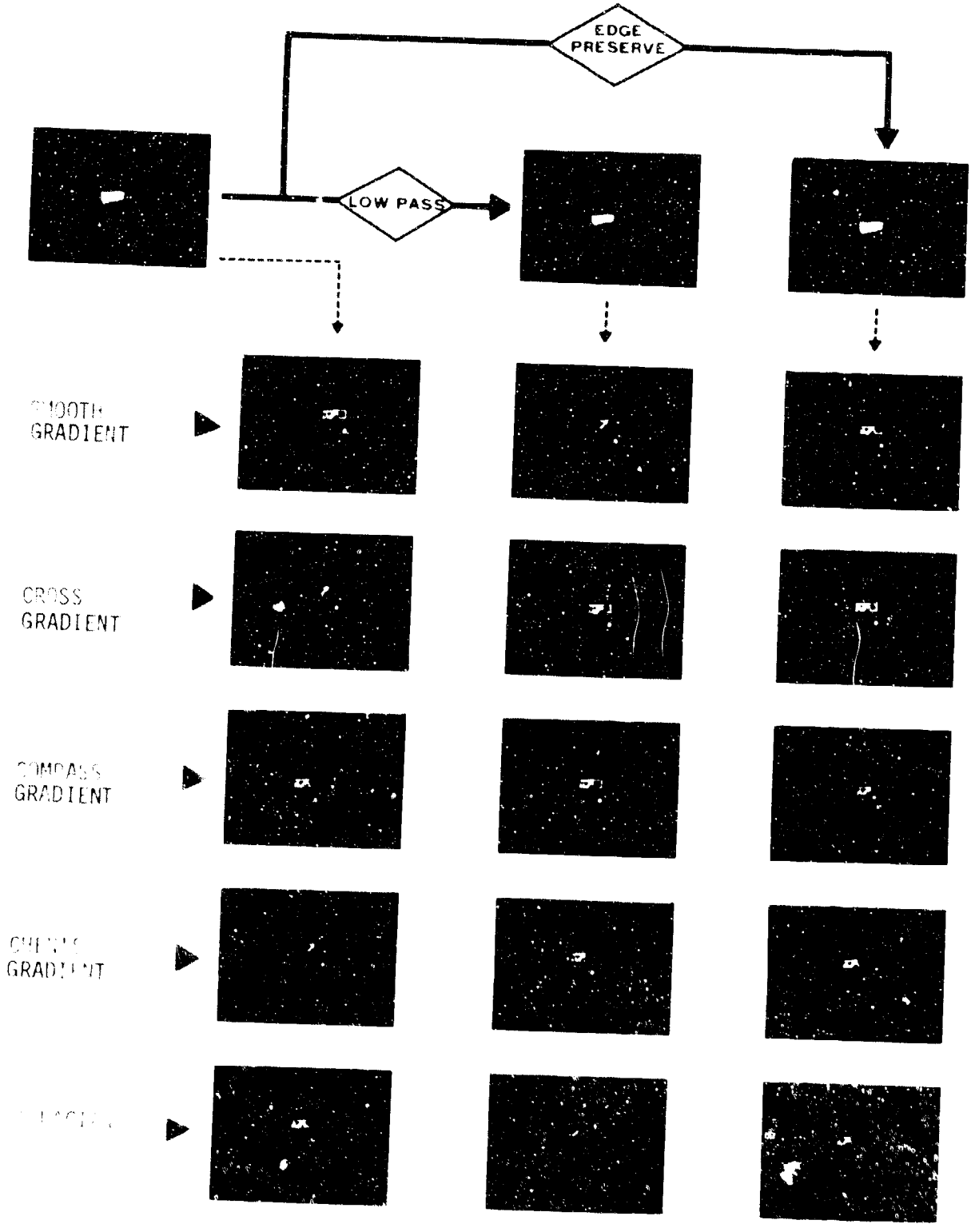


Figure 7(b). Scene No. 1 (NASA Tower) - Rescaled TV Image Preprocessing Results

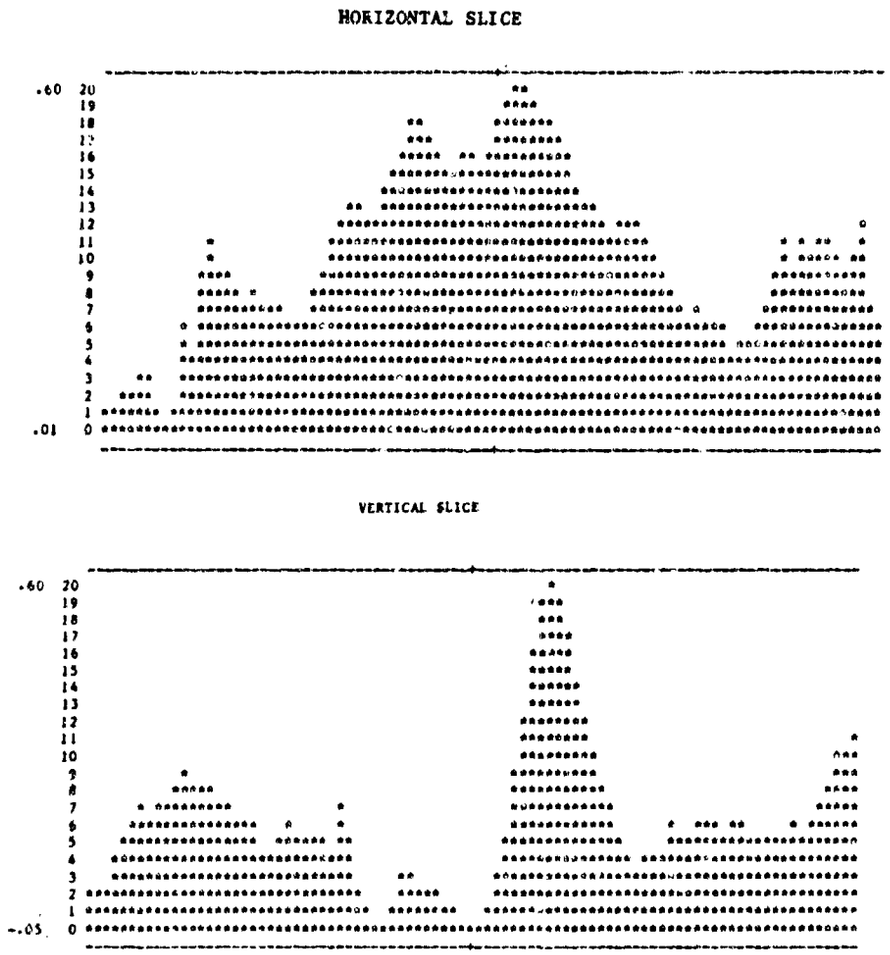


Figure 8 (a). Coarse Search Results

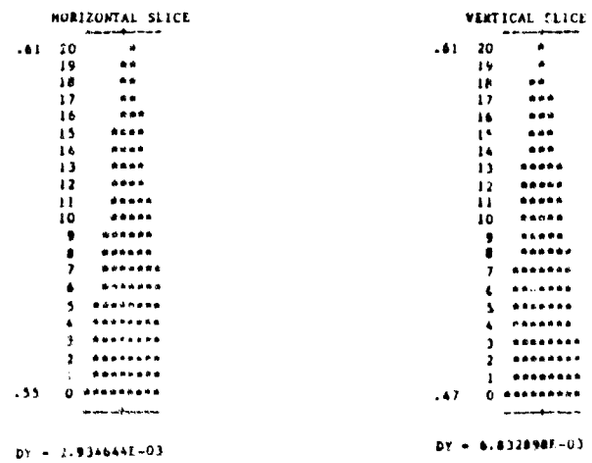


Figure 8 (b). Fine Search Results

Searches with images of TV rate and Independent detector line only are performed in order to investigate the effects of the preprocessing methods on those images.

a. Limited Area/TV-rate Search (Table II)

Table II shows the results of multilevel, bilevel, and trilevel correlation of scenes No. 1 and No. 4. All searches are based on limited, TV rate, and coarse-fine search mode. In the table, scene number, small image window sizes for coarse (C) and (F) searches are indicated. In radial error column, the parenthesized values are ones exceeding ±5 pixel error.

Multilevel correlation results show that all preprocessing methods give reasonably good results except Laplacian operation. Bilevel correlation results show some degradation in images with edge preserving filtering for scene No. 4 except cross gradient method. Correlation score is upgraded from those in multilevel correlation. Trilevel correlation shows same trends as in bilevel.

b. Extended/TV-rate Search (Table III)

Table III shows the results of multilevel, bilevel, and trilevel correlation of scenes No. 1 and No. 4. All searches are based on extended, TV-rate and coarse-fine search mode.

Laplacian did not give good results in this multilevel correlation, either. Cross and compass worked with reasonable results. Low-pass filtering helped to improve the performance. Smooth gradient with scene No. 1 failed except with low-pass filtering. Chen's gradient failed with scene No. 4 except with low-pass filtering. In bilevel, cross and compass worked well as in the multilevel. However, with scene No. 4, compass was failed. Cross showed strong improvement when it applied after both noise cleaning operations, especially with scene No. 4. Trilevel correlation results do not show any specific trend except that cross with scene No. 4 and smooth with scene No. 1 show good correlation results.

TABLE II. LIMITED SEARCH

		(a) MULTI-LEVEL			(b) BI-LEVEL			(c) TRI-LEVEL		
		RADIAL ERROR			RADIAL ERROR			RADIAL ERROR		
	FILTER	NONE	LOW PASS	EDGE PRESERVE	NONE	LOW PASS	EDGE PRESERVE	NONE	LOW PASS	EDGE PRESERVE
SCENE 1	SMOOTH	3.4	2.2	2.8	2.6	2.3	2.6	3.3	2.1	2.7
	CROSS	2.8*	1.6	2.4	2.6	2.6	2.7	(----)	2.4	2.9
C=32*32	COMPASS	3.0	2.1	2.5	2.8	2.8	2.5	2.7	2.3	2.9
F=16*16	CHEN	3.6	1.9	2.3	2.6	2.6	2.6	2.9	2.4	2.6
	LAPLACE	(----)	(5.1)	3.7	(5.3*)	4.7	2.3	(5.8*)	(5.3)	2.3
SCENE 4	SMOOTH	2.2*	1.8	2.24*	4.9*	2.4	(5.4*)	4.0*	2.3	(47.4*)
	CROSS	1.7	2.7	1.7	2.1	1.9	1.9	2.2	2.2	2.5
C=32*32	COMPASS	2.8*	2.2	2.5*	(5.4*)	2.7	(5.2*)	4.9*	2.5	2.6*
F=16*16	CHEN	2.2	2.2	(15.1)	3.3	3.0	(9.1)	3.2	2.7	(5.0*)
	LAPLACE	(----)	(14.8*)	(5.5)	4.5*	2.5	(5.8)	5.9	2.9	(5.6*)

TABLE III. EXTENDED SEARCH

		(a) MULTI-LEVEL			(b) BI-LEVEL			(c) TRI-LEVEL		
		RADIAL ERROR			RADIAL ERROR			RADIAL ERROR		
	FILTER	NONE	LOW PASS	EDGE PRESERVE	NONE	LOW PASS	EDGE PRESERVE	NONE	LOW PASS	EDGE PRESERVE
SCENE I	SMOOTH	(----)	2.3	(----)	(5.0)	14.3	3.5	3.7	3.1	3.2
	CROSS	2.3	1.9	2.5	2.7	2.8	2.9	(10.6*)	3.7	(10.5)
C=46*44	COMPASS	2.5	1.7	2.3	4.9	4.6	3.2	3.4	2.4	(72.5)
F=46*44	CHEN	2.5	1.8	2.2	(14.3)	(13.0)	(9.4)	2 9	(11.5)	2.4
	LAPLACE	(72.7)	(71.2*)	(75.8)	(19.3)	(70.9)	(12.5)	(38.1*)	1.7*	(30.1)
SCENE 4	SMOOTH	2.8	1.6	2.8	(59.0)	(17.3)	(60.1)	(94.8)	(5.2*)	(94.1)
	CROSS	1.7	1.6	1.7	(15.3)	1.5	1.5	1.6	1.6	1.7
C=32*32	COMPASS	.9	1.5	1.5*	(22.5*)	(32.1)	(22.4*)	(93.7)	(89.3)	(94.3*)
F=32*32	CHEN	(26.7*)	1.8	(13.2)	(79.1*)	(16.3)	(78.6)	1.9	(6.0*)	1.8
	LAPLACE	(92.7)	(3.8*)	(49.6)	(21.4*)	(124.6)	(20.0)	(63.5)	4	(91 4)

c. Limited/TV-rate/Raleigh Slicing (Table IV)

Table IV shows the results of bilevel and trilevel correlation of scenes No. 1 and No. 4. All searches are performed under the modes of limited, TV-rate, coarse-fine, and Rayleigh slicing algorithms. The purpose of Rayleigh slicing application is intended to compare the performance between Gaussian and Rayleigh distribution assumptions. The threshold values are selected as 2.74σ for bilevel and 2.10σ and 3.45σ for lower and upper threshold in trilevel respectively. In bilevel, compass gradient for scene No. 1 and cross gradient for scene No. 4 show good results. In trilevel cross and compass gradients show good results for scenes No. 1 and No. 4.

TABLE IV. RAYLEIGH SLICING (LIMITED SEARCH)

		(a) BI-LEVEL			(b) TRI-LEVEL			
		RADIAL ERROR				RADIAL ERROR		
	FILTER	NONE	LOW PASS	EDGE PRESERVE	FILTER	NONE	LOW PASS	EDGE PRESERVE
SCENE I	SMOOTH	1.9	(13.8)	(7.5)	SCENE I SMOOTH	.9	(15.5)	(12.5)
	CROSS	(6.3*)	(8.8*)	(9.8*)	CROSS	2.7*	1.9*	2.6*
C=32*32	COMPASS	1.7	1.6	1.6	C=32*32 COMPASS	1.7	.8	1.1
F=16*16	CHEN	(6.9*)	.8	(9.9*)	F=16*16 CHEN	(14.5*)	1.1	(17.2*)
	LAPLACE	(7.6)	4.04	3.1*	LAPLACE	3.0*	(13.8*)	3.5*
SCENE 4	SMOOTH	(----)	(12.8)	(6.8*)	SCENE 4 SMOOTH	2.8*	(12.9)	(9.96)
	CROSS	3.7	3.3	3.1	CROSS	(5.5)	3.0*	2.7
C=32*32	COMPASS	(7.8)	4.2	4.7	C=32*32 COMPASS	(----)	(----)	2.5
F=16*16	CHEN	(11)	4	(12.7)	F=16*16 CHEN	(----)	2.8*	(12.3*)
	LAPLACE	(12.1)	(----)	(14.7*)	LAPLACE	(10.2)	(----)	(17.4)

d. Extended/Independent Detector Lines (Table V)

Every third line of the original image is selected to extract each detector line output. This test is intended to compare the effect of preprocessing algorithms on the TV-rate format and independent detector line format. Multilevel, bilevel, and trilevel slicing methods are applied.

In multilevel correlation, low-pass filtering operation for independent detector lines degraded the performance for scene No. 1. This is expected because the NASA tower has very sharp edges in the original image. On the other hand, edge preserving filtering performed well for both cross and Chen's gradients. Scene No. 4 shows better results with low-pass filtering than with edge preserving filtering. Same trend is shown for bilevel with scene No. 1 as with multilevel. Scene No. 4 does not show good performance. It is noticed that for trilevel the cross gradient with edge preserving filtering for scene No. 4 shows good results. That is the only good result with scene No. 4. Scene No. 1 shows same trend as in multilevel and bilevel.

TABLE V. INDEPENDENT PIXEL (EXTENDED SEARCH)

		(a) MULTI-LEVEL			(b) BI-LEVEL			(c) TRI-LEVEL		
		RADIAL			RADIAL ERROR			RADIAL		
	FILTER	NONE	LOW PASS	EDGE PRESERVE	NONE	LOW PASS	EDGE PRESERVE	NONE	LOW PASS	EDGE PRESERVE
SCENE I	SMOOTH	(50.3*)	(20.8*)	(22.3*)	(62.0*)	(61.6*)	(61.5*)	(----)	2.7*	(----)
	CROSS	1.2	(20.4*)	.94	.8	(58.4)	1.3*	.7	(60.6)	.84
C=32*12	COMPASS	(38.1*)	(20.4*)	(38.0*)	(----)	(61.0)	(61.8*)	(----)	(61.0)	(----)
F=32*12	CHEN	.1	(20.6*)	.14	.5	(59.7)	.82	3.0	(61.3)	.9
	LAPLACE	(30.0*)	(32.4)	1.0	.76	(34.4)	1.3	2.2	(33.0)	1.22
SCENE 4	SMOOTH	(77.3)	3.7*	(28.0*)	(73.6)	(31.2)	(10.7*)	(34.2)	(27.3)	(81.8*)
	CROSS	(35.6)	(35.9)	3.8	(32.3)	(32.8)	5.0	(33.1)	(32.7)	4.0
C=32*12	COMPASS	(18.6*)	3.4*	(84.9)	(101.4*)	(28.3)	(82.0*)	(103.0)	(29.8)	(81.1)
F=32*12	CHEN	(10.8*)	3.7*	(22.9*)	(----)	5*	(70.1)	(71.7*)	(70.5*)	(70.2)
	LAPLACE	(92.9)	(119.7)	(34.9*)	(96.4)	(29.6)	(30.3)	(27.6*)	(31.9)	(29.0)

e. Extended/Independent Detector Line/Fine Search (Table VI)

Table VI shows the fine search results. These tests are intended to compare the performances of coarse-fine sequence versus fine-only search method. The computation time takes approximately four times longer than coarse-fine search sequence.

In multilevel, edge preserving filtering performed well giving higher correlation amplitudes. Low-pass filtering degraded performance for this independent detector line images. Preprocessing with noise cleaning also shows good performance. Compass and Chen's gradients worked better in scenes No. 1 and No. 4. Smooth gradient for scene No. 1 also shows good performance. Bilevel compass and smooth gradient operations give good results. Also, edge preserving filtering operation improved the performance with almost all edge-enhanced images. In trilevel, smooth, Chen's and cross gradient operations for scene No. 1 performed well. For scene No. 4, Chen's gradient worked well. Edge preserving filtering shows better performance than low-pass filtering, and no-noise cleaning also gives good results.

TABLE VI. (FINE SEARCH ONLY) INDEPENDENT PIXEL EXTENDED SEARCH

		(a) MULTI-LEVEL			(b) BI-LEVEL			(c) TRI-LEVEL		
		RADIAL			RADIAL			RADIAL		
	FILTER	NONE	LOW PASS	EDGE PRESERVE	NONE	LOW PASS	EDGE PRESERVE	NONE	LOW PASS	EDGE PRESERVE
SCENE I	SMOOTH	.82	(20.9)	.5	1.01	1.4	2.8	2.5	2.3	2.5
	CROSS	(20.85)	(29.9)	(21.)	.8	(30.8)	(29.8)	.7	1.3	.8
	COMPASS	.31	(20.8)	.4	.5	1.2	1.5	1.0	(30.6)	.6
F=32*12	CHEN	6	.7	1	1.0	(0.3)	.8	1.7	1.2	.9
	LACE	(27.2)	(36.0)	1	(6.2)	(21.0)	1.3	(24.4)	(31.0)	1.2
SCENE 4	SMOOTH	4.0	3.9	3.1	3.5	(120.8)	3.3	3.5	(55.6)	3.2
	CROSS	4.0	(67.9)	(24.1)	(16.9)	(18.7)	(31.4)	4.	(12.7)	(17.4)
	COMPASS	3.9	4.0	3.1	(12.8)	(----)	3.6	3.3	(55.3)	3.2
F=32*12	CHEN	3.4	3.9	3.3	3.4	(32.)	3.5	3.4	4.3	3.4
	LAPLACE	(74.1)	(28.2)	(8.0)	(20.8)	(29.6)	3.2	(21.)	(46.2)	3.4

I. CONCLUSIONS

Several preprocessing algorithms have been studied in application to the two-sensor boresighting problem. These included two noise cleaning algorithms, five edge enhancement algorithms and three slicing algorithms. After each preprocessing application, images were correlated to see if the specific preprocessing algorithm improved the correlation performance. Based on tests run thus far, the following conclusions are evident:

1. Limited area search revealed that, even in small area search around the estimated target location, Laplace enhancement does not improve the correlation performance. However, with scene No. 1 the edge preserving filtering combined with Laplacian showed good results.

2. Extended area search indicated that cross gradient operation produces good correlation results with or without noise cleaning operations. Specially, bilevel correlation with scene No. 4 showed that noise cleaning operation with cross gradient enhancement improved correlation results. Compass and Chen's gradients also worked well in this situation.

3. Images with Rayleigh slicing algorithms indicated that cross and compass gradient enhancement algorithms with noise cleaning algorithm work well for correlation tests.

4. Extended area search with images of independent-detector pixels only did not work well when low-pass filtering was applied. However, cross enhancement algorithm with edge-preserving filtering showed good performance with consistency.

5. To compare the results from coarse-fine search sequence with the results from the direct fine searches of extended area , tests were performed. The results showed that edge-preserving filting definitely improves the correlation performance. Smooth, compass, and Chen's gradients performed well in multilevel and bilevel searches. Smooth, cross, and Chen's worked well for trilevel correlation tests.

We know it is necessary to investigate reasons why false peaks were obtained in certain tests. It will be possible to refine the sequence of preprocessing algorithms and/or to improve correlation algorithms to adjust to the effects of preprocessing operations.

J. ACKNOWLEDGMENTS

The authors wish to thank, Mr. W. L. Steiner, Project Engineer at Coodyear Aerospace Corporation for his support and encouragements of the present work.

REFERENCES

1. P.H. McIngvale, M.Pitruzzello, and T. W. Iler, MICOM, "Design and Evaluation of an Automatic Handoff Correlator", to be presented at Army Missile Command Workshop on Image Trackers and Autonomous Acquisition for Missile Guidance, November 19-20, 1979
2. J. S. Boland, et al, Correlation Algorithm Development, 30 April 1979, Contract No. DAAK40-79-M-0016, Engineering Experiment Station, Auburn University, AL.
3. W. K. Pratt, Digital Image Processing, John Wiley and Sons, 1978, PP 482.
4. G. P. Dinneen, "Programming Pattern Recognition", Proc. of WJCC, March 1955, PP 94-100.
5. W. K. Pratt, Digital Image Processing, John Wiley and Sons, 1978, PP 320.
6. M. Nagao, et al, "Edge Preserving Smoothing", CGIP, Vol. 9, 1979, PP 394-407.
7. R. O. Duda, et al, Pattern Classification and Scene Analysis, John Wiley and Sons, 1973, PP 271.
8. G. S. Robinson, et al, "A Real-Time Edge Processing Unit", IEEE Workshop on Picture Data Description and Management, Chicago, IL., April 1977, PP 155-164.
9. C. H. Chen,"Note on a Modified Gradient Method for Image Analysis", Pattern Recognition, Vol. 10, 1978, PP 261-264.
10. Yoo, J. K., and Berry, J. E., "Transform Image Matching, "Proc. of Workshop on Computer Analysis of Time-Varying Imagery, sponsored by IEEE Computer Society and the Machine Intelligence and Pattern Analysis Technical Committee, Philadelphia, Pa., April 5-6, 1979.
11. A. Papoulis, Probability, Random Variables, and Stochastic Processes, McGraw-Hill, 1965, PP 65.

Paper No. IIB-3, Presented at the Workshop on Imaging Trackers and Autonomous Acquisition Applications for Missile Guidance, 19-20 November 1979, Redstone Arsenal, Alabama.

A DIGITIZED VIDEO SLICING TECHNIQUE FOR CORRELATION PROCESSING

D. PASIK*, H. R. DESSAU*, R. WALTER†

SUMMARY

A closed loop video processor is described which slices analog video from radar imagery into tri-level white, grey and black. The slicing levels are designed to track the video modulation so as to discriminate the scene features from noise and spurious modulation.

ABSTRACT

Slicing of analog video from a radar image is an essential preprocessing step prior to digital correlation. A tri-level white, grey and black slicing scheme converts the analog signal into a two-bit digital word. The encoded image content is highly dependent on the slicing level's equilibrium steady-state values and on their time constant responses. For example, a slicing level set too high fails to pass low amplitude scene content; conversely, if it is set too low it fails to discriminate higher amplitude scene modulation. Similarly, a time constant set too slow will not track fast changing contrast trends while if it is set too fast it will tend to track noise.

A design procedure is presented for determining the modulation-amplitude dependent time constants and equilibrium values of the slicing levels in terms of the modulation probability distribution.

BACKGROUND AND INTRODUCTION

Terminal guidance in the Pershing II system is based on a continuous inertial navigation process, from which long term errors are reduced through independent determination of vehicle position by a radar map-matching or area correlator.

For an area correlator to be implemented with a digital computational algorithm it is clearly necessary that both the prestored reference scene and the real time observed or live scene be quantized both spatially and with respect to signal amplitude or, equivalently, scene brightness. The Pershing area correlator slices scene brightness into a tri-level format designated as white, grey and black.

This work was performed under sponsorship of contract number DAAK40-79-C-0064.

*D. Pasik and H.R. Dessau are with Martin Marietta Aerospace, Orlando, Florida

†R. Walter is with Goodyear Aerospace, Akron, Ohio

A real time process derived from radar presents several constraints and options. Figure 1 illustrates one sequential allocation of functions for a 360° scan PPI radar. The points A, B, C represent three choices at which video analog brightness might be quantized to the required three levels.

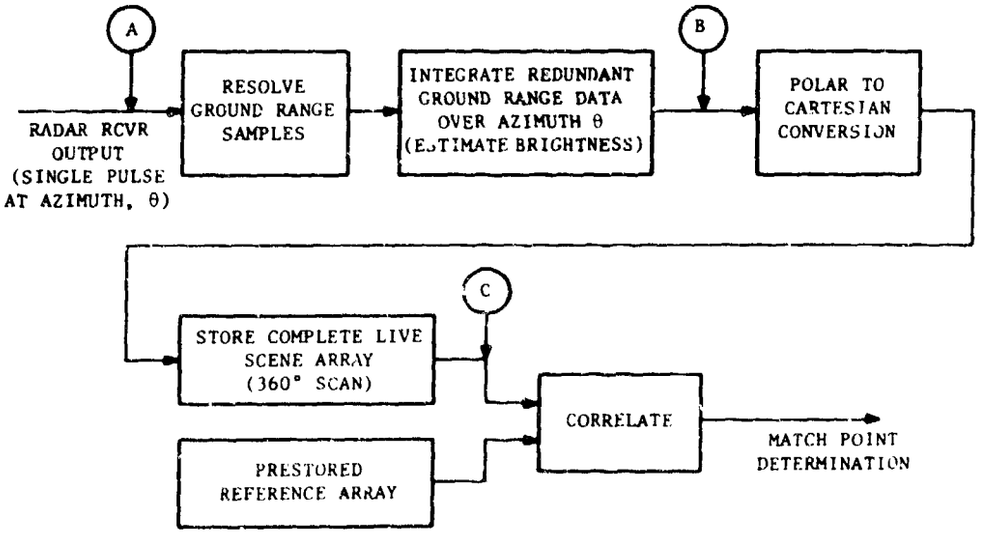


Figure 1. A, B, C are Alternative Tri-Level Video Slicing Points

Delaying the quantization to point C has the potential advantage that the entire scene information might be used as a basis for the slicing algorithm; however, the computational burden would be excessive. Conversely, quantizing and slicing at points A or B has the computational advantage that tri-level slicing can be reduced to two independent bi-level operations, resulting in a two-bit word. The drawback is that the slicing thresholds must be established in real time by a dynamical system which accepts each pulse return immediately. This approach is discussed below.

Two implementations of slicing logic have been developed and tested for the Pershing correlator. The first, depicted in Figure 2, is the Open Loop Video Processor (OLVP). Instantaneous slicing levels are based on short term observed values of average signal level and predesignated fractions of the positive and negative excursions about that average. The choice of level constants K_1, K_2 is dependent on à priori assumptions on the signal distribution.

The second implementation, described below, is an extension of a concept described by Brokl et al [1], shown in Figure 3. This Closed Loop Video Processor (CLVP) performs two parallel bi-level slicing operations directly on the raw analog video at the comparator junctions. Subsequent postprocessing and integration attenuates noise and sets the threshold levels in terms of predesignated fractions of the signal modulation amplitude. The slice levels track changes in signal modulation amplitude according to a dynamical model derived in the analysis.

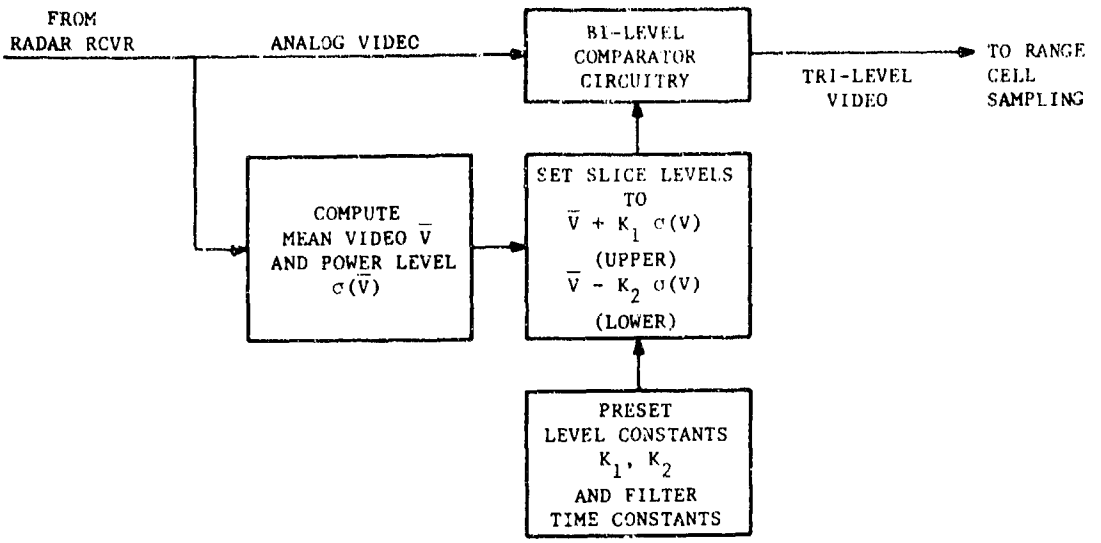


Figure 2. Open Loop Video Processor Concept

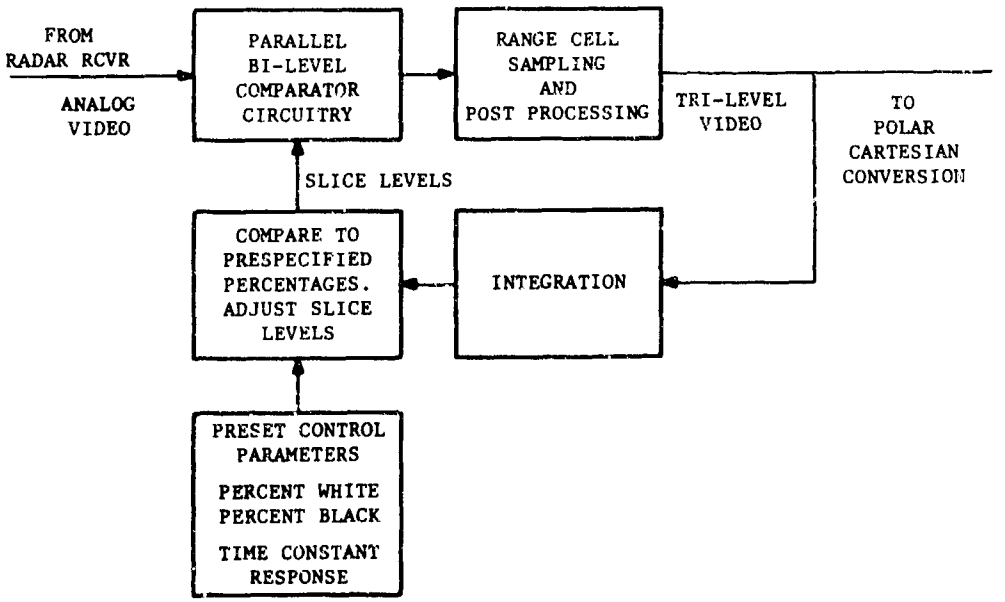


Figure 3. Closed Loop Video Processor Concept

CIRCUIT MODEL

The closed loop video processor (CLVP) must slice the analog video into three levels and remove spurious modulation and noise. This is accomplished by two similar processors acting in parallel, each of which generates a separate reference level. Figure 4 illustrates the functional block diagram for the upper half. The slice level, V, separates white from nonwhite. The digitized video output is +1 for detected white and -1 for detected nonwhite.

This is logically combined with digitized video from the bottom half representing detected black or nonblack to encode a tri-level white, grey or black.

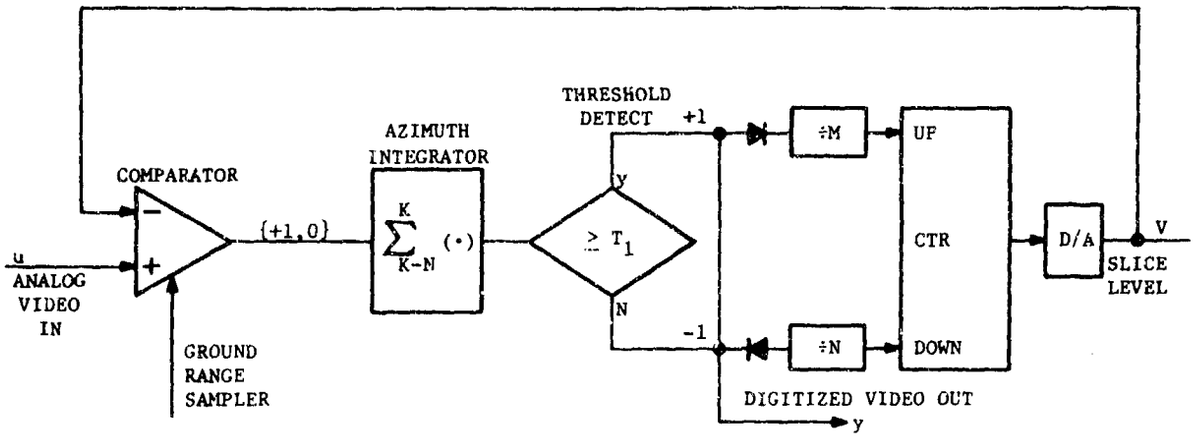


Figure 4. Positive Half of CLVP

The function of the azimuth integrator is to filter out high frequency noise. Its effect is tc introduce a delay of N/2 samples. Because of the very high sampling rate the net effect is insiguificant in the overall operation.

Under the simplifying assumption of no high frequency noise, the threshold detects white when the analog video exceeds the slice level, V, and nonwhite otherwise. This is shown in the analog model of Figure 5. The counters are approximately by integrators and the D/A by a simple gain, K.

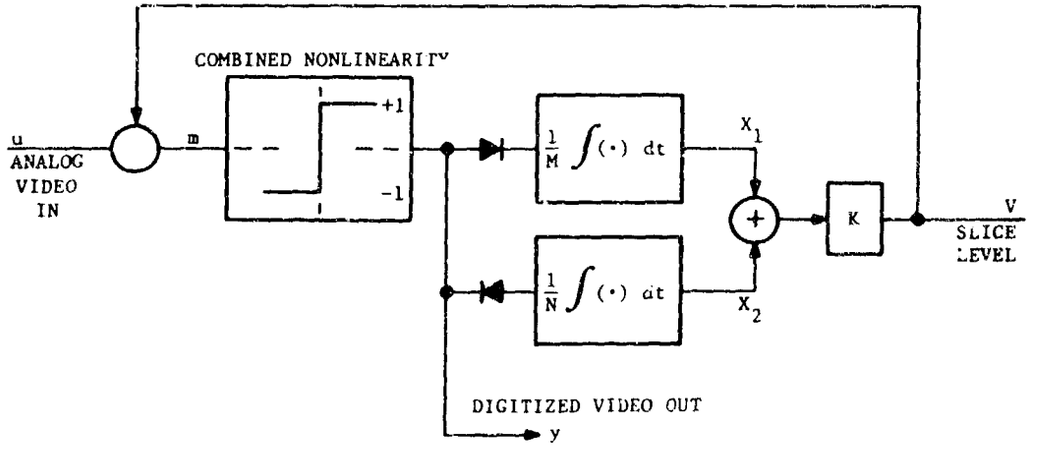


Figure 5. Analog Model of Figure 4

SLICING LEVEL DYNAMICS

From Figure 5 we can derive the fundamental equation for the slice level V. Define I_1 to be the fraction of time white which corresponds to the fraction of time that $u \geq V$. Define also J_1 to be the subinterval set of the closed interval [o, t] during which white is detected. Take $I_2 = 1 - I_1$ and $J_2 = [o, t] - J_1$. Then from Figure 5

$$x_1 = \frac{1}{M} \int_{J_1} dt = \frac{I_1 t}{M} \tag{1}$$

$$x_2 = -\frac{1}{N} \int_{J_2} dt = \frac{I_2 t}{N} \tag{2}$$

and

$$V = V_o + Kt \left(\frac{I_1}{M} - \frac{I_2}{N} \right) \tag{3}$$

It follows that the equilibrium condition is

$$\frac{I_1}{M} = \frac{I_2}{N} \tag{4}$$

$$I_1 = \frac{M}{M+N} , \quad I_2 = \frac{N}{M+N} \tag{5}$$

This is the first major result. It is more convenient to express V as a differential equation. Substituting $I_2 = 1 - I_1$ in (3) and differentiating yields the second major result.

$$\frac{dV}{dt} = K \left(\frac{M+N}{MN} I_1 - \frac{1}{N} \right) \tag{6}$$

DISTRIBUTION FUNCTION DEPENDENCE

The analog video u is modelled as modulation about a trend, u_o, whose dynamic range is $2M_1$. The modulation frequency is less than the noise but otherwise high compared with the trend. It will be shown below that the distribution function for the modulation about the trend determines both the equilibrium slice level and the instantaneous time constant of the slice level response to changes in the trend.

For the variables x and X define the density function f(x) and the cumulative distribution function F(X) where

$$x \triangleq \frac{u-u_o}{M_1} \; ; \; X \triangleq \frac{V-u_o}{M_1} \tag{7}$$

$$F(X) = P(x \leq X) \; ; \; f(x) = \frac{dF(x)}{dx} \tag{8}$$

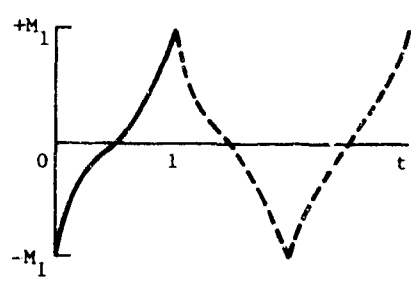
Then

$$I_1 = P(u>V) = 1 - F(X) \tag{9}$$

Thus the equilibrium slicing level follows from (5) and (9) from the relation

$$I - F(X_o) = \frac{M}{M+N} \; ; \; F(X_o) = \frac{N}{M+N} = \frac{1}{1+M/N} \tag{10}$$

This implies the filter tracks a fixed percentage of white. Some representative distributions with their associated canonical waveforms are shown in Figure 6 for scaled time.

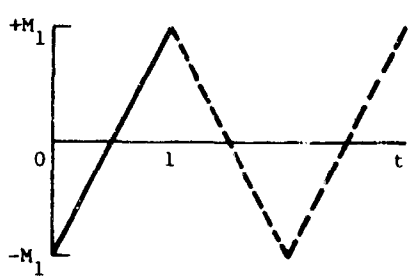


SINUSOIDAL DISTRIBUTION

$F(X) = \frac{1}{2} + \frac{1}{2} \sin \frac{\pi}{2} X$

CANONICAL WAVEFORM

$u = \frac{2M_1}{\pi} \sin^{-1} (2t-1)$

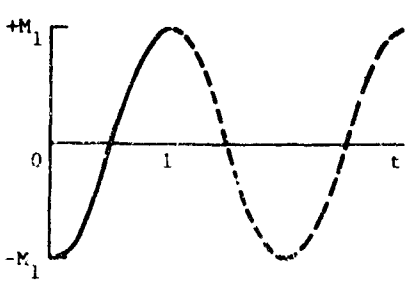


UNIFORM DISTRIBUTION

$F(X) = \frac{1}{2} + \frac{X}{2}$

CANONICAL WAVEFORM

$u = M_1(2t-1)$



ARC SIN DISTRIBUTION

$F(X) = \frac{1}{2} + \frac{1}{\pi} \sin^{-1} X$

CANONICAL WAVEFORM

$u = M_1 \sin \frac{\pi}{2} (2t-1)$

Figure 6. Distributions and Canonical Modulation Waveforms

Substituting these distributions into (10) yields

SINUSOIDAL DISTRIBUTION

$$X_o = \frac{2}{\pi} \sin^{-1} \frac{N-M}{N+M}, \text{ or for } u_o=o,\ V = \frac{2M_1}{\pi} \sin^{-1} \frac{N-M}{N+M} \tag{11}$$

UNIFORM DISTRIBUTION

$$X_o = \frac{N-M}{N+M}, \text{ or for } u_o=o,\ V = M_1 \frac{N-M}{N+M} \tag{12}$$

ARC SIN DISTRIBUTION

$$X_o = \sin \frac{\pi}{2} \frac{N-M}{N+M}, \text{ or for } u_o=o,\ V = M_1 \sin \frac{\pi}{2} \frac{N-M}{N+M} \tag{13}$$

Note the inequality

$$\sin \frac{\pi}{2} y \geq y \geq \frac{2}{\pi} \sin^{-1} y, \forall y \tag{14}$$

Thus the slicing level for the arc sin distribution is highest while that of the sinusoidal is lowest.

A complete description of the slicing level dynamics may be obtained by substituting (9) into (6). Then

$$\frac{dV}{dt} = K \left(\frac{M+N}{MN} (1-F(X)) - \frac{1}{N} \right) \tag{15}$$

This is, in general, a nonlinear first order differential equation. For the special case of the uniform distribution the equation is linear and (15) becomes

$$\frac{dV}{dt} = \frac{-K(M+N)}{2MN} X + K \left[\frac{M+N}{2MN} - \frac{1}{N} \right] = \frac{-K(M+N)}{2MN} \frac{V}{M_1} + \frac{K(M+N)}{2MN} \frac{u_o}{M_1} + \frac{K(N-M)}{2MN}$$

$$= - \frac{V}{\tau} + \frac{u_o}{\tau} + C \tag{16}$$

Hence

$$V = (u_o + C\tau)\ (1 - e^{-t/\tau}) \tag{17}$$

The time constant τ is given by

$$\tau = \frac{2MN\ M_1}{K(M+N)} \tag{18}$$

and note that Ct gives the equilibrium determined by (13).

LOCAL TIME CONSTANT

For the general equation given by (15) a local time constant, τ_L, can be defined in terms of a small slice level change from V_0 to V_1. This might result for example from a step change in u_o. This leads to an instantaneous displacement in X at t = o from its new equilibrium value. We have the following result:

THEOREM:

The local time constant, τ_L, is given in terms of the density function, f, which is evaluated at the equilibrium condition, (10), by

$$\tau_L = \frac{MN\ M_1}{K(M+N)\ f(X_o)} \tag{19}$$

PROOF:

For small slice level changes about the equilibrium expand F(X) in a Taylor's series yielding

$$F(X) = F(X_o) + f(X_o)\ \Delta X_o \quad + \quad +$$

$$= F(X_o) + f(X_o)\ \frac{(V-V_o)}{M_1} + \quad + \tag{20}$$

where the result for ΔX_o above follows (7) and the assumption that u_o is constant for $t > 0$. Substituting (20) into (15) yields a linear equation.

$$\frac{dV}{dt} = -\frac{K(M+N)\ f(X_o)}{MN\ M_1}\ V + \text{other terms}$$

$$= -\frac{V}{\tau_L} + \text{other terms} \tag{21}$$

REMARK:

At the equilibrium point X_o, the substitution of (10) into (19) yields

$$\tau_L = \frac{MN\ M_1}{K}\ \frac{F(X_o)}{f(X_o)}$$

This indicates that time constant τ_L is a function of M while equilibrium X_o (Eq 10) depends on the ration M/N.

Consider the following examples:

SINUSOIDAL DISTRIBUTION

$$f(X_o) = \frac{\pi}{4} \cos \frac{\pi}{2} X_o \quad ; \text{ from (11)}$$

$$= \frac{\pi}{4} \cos \left(\sin^{-1} \frac{N-M}{N+M} \right) \tag{22}$$

$$= \frac{\pi}{4} \sqrt{1 - \left(\frac{N-M}{N+M} \right)^2}$$

$$\tau_L = \frac{4MN\, M_1}{\pi\, K(M+N) \sqrt{1 - \left(\frac{N-M}{N+M} \right)^2}} \tag{23}$$

UNIFORM DISTRIBUTION

$$f(X_o) = \frac{1}{2} \tag{24}$$

$$\tau_L = \frac{2MN\, M_1}{K(M+N)} \quad ; \text{ consistent with (18)} \tag{25}$$

ARC SIN DISTRIBUTION

$$f(X_o) = \frac{1}{\pi \sqrt{1 - X_o^2}} \quad ; \text{ from (13)}$$

$$= \frac{1}{\pi \sqrt{1 - \sin^2 \left(\frac{\pi}{2} \frac{N-M}{N+M} \right)}} \tag{26}$$

$$\tau_L = \frac{\pi\, MN\, M_1 \sqrt{1 - \sin^2 \left(\frac{\pi}{2} \frac{N-M}{N+M} \right)}}{K(M+N)} \tag{27}$$

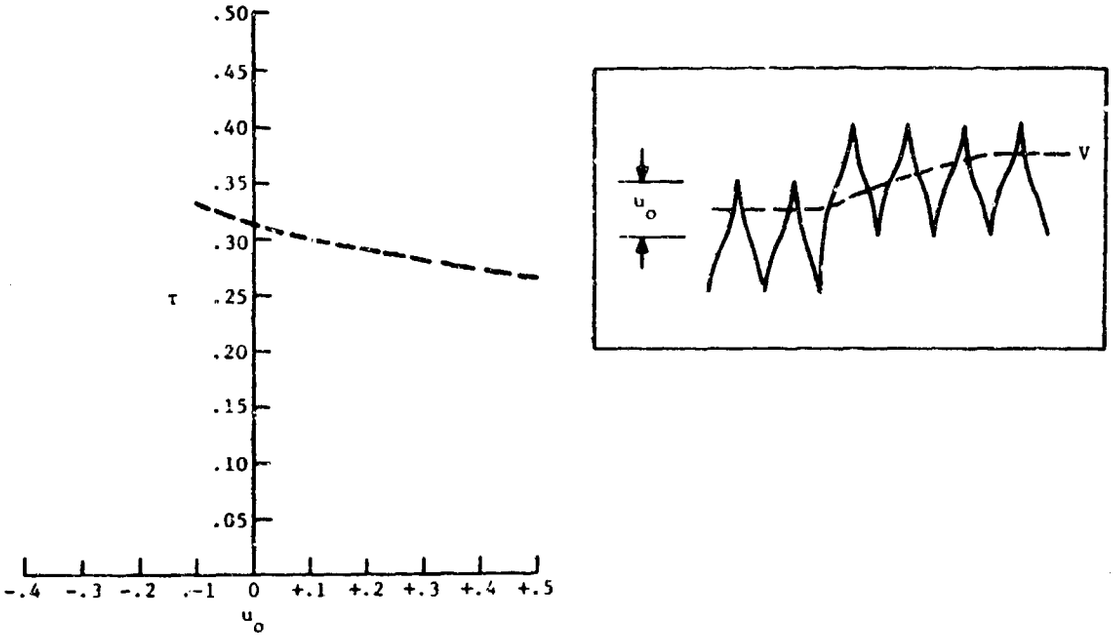
For values K = 405.405, N = 64, M = 16, M_1 = 0.625. There results

τ_L (SINUSOID) from (23) = 31.406 ms

τ_L (UNIFORM) from (25) = 39.466 ms

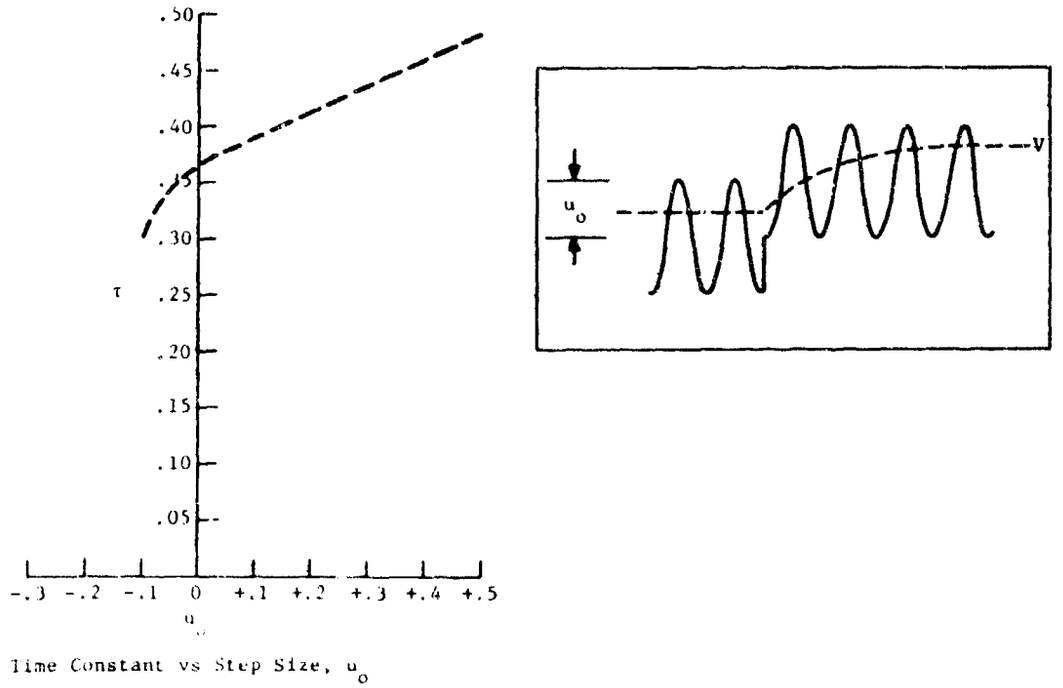
τ_L (ARC SIN) from (27) = 36.4 ms

A computer simulation of Eq (15) within the unsaturated region $0 \leq I_1 \leq 1$ was performed for various input steps u_o. The results are shown in Figure 7 for the sinusoidal distribution and in Figure 8 for the arc sin distribution. These results are consistent with the following extended interpretation of



Time Constant vs Step Size, u_o

Figure 7. Sinusoidal Distribution



Time Constant vs Step Size, u_o

Figure 8. Arc Sin Distribution

local time constant: Let X_1 be any point in the distribution not necessarily the equilibrium. Then Eqs (20) and (21) remain valid for an expansion about X_1 so that

$$\tau(X_1) = \frac{MN\ M_1}{k(M+N)\ f(X_1)} \tag{28}$$

For the sinusoidal distribution a positive step, u_o, which increases percent white, reduces X_o increases f(x) and from (28) reduces τ. A negative step, u_o, reduces f(X) and increases τ. For the arc sin distribution the converse is true. The uniform distribution has an invariant time constant.

Suppose now equal values of N and M are assigned to both halves of the CLVP. Then Figure 7 implies that for a positive step input, u_o, the upper slicing threshold will have a faster time constant while the lower slicing threshold will be slower. The converse holds for a negative step. On the other hand Figure 8 indicates that for a positive step input, u_o, the upper slicing threshold is slower while the lower is faster. Again the converse holds for a negative step. Consequently, for equal N and M settings the time constants of the two halves of the CLVP are never for finite u_o except in the case of the uniform distribution. The difference in time constant between the upper and lower halves might then be used as a measurement for the distribution function.

Finally the above arguments indicate that the CLVP circuit will tend to adaptively reject unlikely noise pulses which pass the azimuth integrator. In fact positive and negative pulses are filtered differently so as to bias the noise in the most likely direction. Consider the sinusoidal distribution. A positive noise pulse drives the threshold high into the region where f(X) is decreasing. From (28) the time constant gets larger thus attenuating the pulse. Conversely a negative pulse drives the threshold low into the region where f(X) increases and the corresponding smaller time constant tends to pass the pulse. Similarly a negative pulse is rejected and a positive pulse passed in the case of the arc sin distribution. There is no preferential filtering for the uniform distribution.

CONCLUSIONS

The Closed Loop Video Processor performs tri-level slicing based on predetermined percentages of the analog video modulation. Hence it tracks changes in modulation amplitudes in addition to signal level trends. Slice levels and time constants depend on the modulation distribution; however, if this is known they can be preset in terms of the parameters N and M. Note that time constants depend directly on the modulation level, M_1, so that if the level M_1 varies over the scene some additional mechanism may be needed to control time constant.

REFERENCE

[1]Three-Level Signal Samples has Automatic Threshold, Stanley S. Brokl, et al., NASA Tech Briefs, Summer 1977

Paper No. IIB-4, Presented at the Workshop on Imaging Trackers and Autonomous Acquisition Applications for Missile Guidance, 19-20 November 1979, Redstone Arsenal, Alabama.

AUTOMATIC HAND-OFF FROM FLIR ACQUISITION DEVICE TO IMAGING IR SEEKER

John A. Knecht
Naval Weapons Center
China Lake CA

ABSTRACT

This paper describes the background and the current efforts to develop an aircraft sensor correlation device (ASCD) to automatically hand-off from a Forward Looking Infrared (FLIR) acquisition sensor to an Imaging IR (IIR) seeker. This type of system implementation makes it possible for the pilot/operator of an attack aircraft to acquire and identify the target at long range using the high resolution and sensitivity of the FLIR and then automatically hand-off to an imaging IR seeker at a range which the seeker, if used alone, could detect but not identify the target. Additionally the reduction in operator work load and time line provides a high probability of first pass attack. The hand-off device described uses high speed digital technology to perform real time video-cross-correlation and, via the FLIR-seeker servo loop, continuously align the seeker to the FLIR.

I. INTRODUCTION

The objective of the Aircraft Sensor Correlation Device (ASCD) is to provide an automatic hand-off from a FLIR acquisition sensor to an imaging IR missile seeker. The development effort has been directed toward working within current Navy plans for aircraft, avionics, and missiles. Specifically, this means A-6, A-7, A-18 aircraft, the FLIRs already in development for these aircraft, and Maverick variant weapons of the lock-on-before-launch type.

The utility of the ASCD hinges on its ability to maximize both the weapon release range and the probability of first pass attack. These requirements are necessary to reduce aircraft attrition which can easily become the dominant factor in assessing the cost of killing targets.

First pass attack demands a minimum time to locate a target and ready a weapon to fire. It implies first, a FLIR and second a means of automatically handing-off targets from the FLIR to the missile seeker. That means is provided by ASCD.

II. SYSTEM CONCEPT

a. The operational problems facing Naval attack aircraft include first pass attack, including off-axis targets and the capability of rapid multiple fire. The ASCD by tying the FLIR and seeker together provides a system which solves these problems. By enabling the operator to rapidly find and lock up a target, an ASCD based system can initiate a first pass attack and perform a rapid multiple fire. The continuous alignment of seeker to FLIR allows easy off-axis acquisition. The ASCD approach has the additional advantages of reducing the operator work load which when added to the reduced time to launch the weapon results in increased aircraft survivability. This enables even a single seat aircraft to successfully utilize a weapon of this type. Use of the FLIR as an acquisition aid for the IR weapon takes advantage of the higher resolution, sensitivity and pointing accuracy of the FLIR to acquire targets at greater ranges under more degraded conditions. Also the larger gimbal angles and larger field-of-view of the FLIR enable the operator to acquire the target with a higher degree of probability.

Human factor studies have shown[1] that it takes an operator 3 to 5 seconds more time to locate and designate a target on the FLIR display and then

[1] Naval Weapons Center. "Feasibility Study of a FLIR/Imaging Seeker System, by Jeffrey D. Grossman. January 1977. (NWC TP 5909).

manually re-locate the target again on the seeker display when compared to an automatic hand-off system. This study was for an operator in a dual seat aircraft and time savings for single seat aircraft would be significantly greater. This additional time usually comes at a point where the aircraft is exposed to the targets defenses. The shortening of the time line provided by automatic hand-off directly results in an increase in aircraft survivability.

A typical acquisition and launch sequence for an ASCD based system is as follows:

The operator performs an automatic coarse alignment of the system while enroute to the target. This coarse alignment removes gross static misalignment between the seeker and FLIR due to mechanical mounting of the missile on the wing station. The operator then initiates a continuous fine alignment which removes dynamic misalignment due to wing flexure and non-linear position transducers. The aiming symbol on the FLIR display now indicates where the seeker is pointed. Next the operator acquires the target on the FLIR display, places the aiming symbol over the target and enables the track trigger. After verifying the target, the weapon is ready to launch.

III. BACKGROUND

The idea for an ASCD was originally conceived out of necessity during the early stages of the Night Attack Program. The ASCD developed during this program was used to align a modified S3-A FLIR with a non-imaging circular scan seeker on a lock before launch missile. Since the seeker was by design non-imaging, an ASCD was the only option available to perform the hand-off. As shown in Figure 1, the second generation ASCD built by Raytheon Co. digitized a "snap shot" of FLIR and seeker video. This was then processed by a normalized product 15 level correlator implemented as a pipeline processor. Using the seeker video as a reference and searching the array of FLIR video the point of highest correlation (matchpoint) was found. The position of the matchpoint was then used to generate a boresight offset signal which was fed to

the seeker position controller to correct the position of the seeker which was coarsely slaved to the FLIR via gimbal angles. Time required to perform the coarse alignment was on the order of one second. The normalized product algorithm and 15 level quantization were used in the correlator because of the limited amount of video data available from the non-imaging seeker.

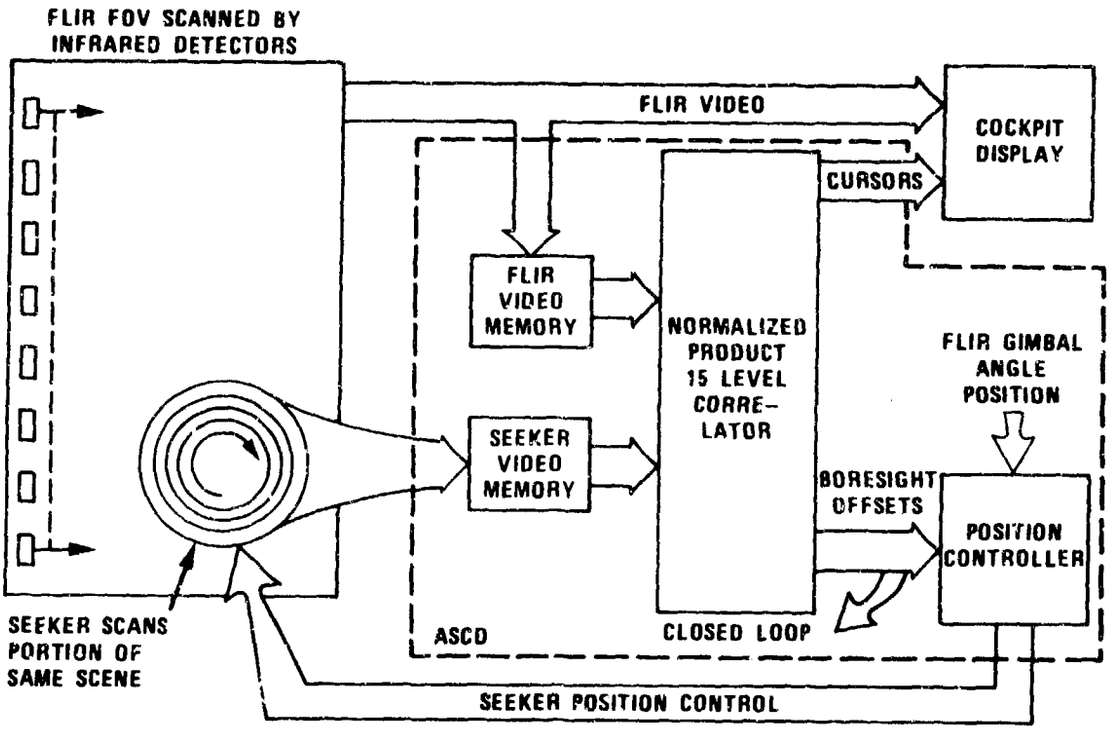


FIGURE 1. Night Attack Aircraft Sensor Correlation Device Block Diagram

The ASCD was flight tested on an A-6 aircraft in conjunction with the Night Attack system. Five flights were flown and the ASCD performance was evaluated using scenes such as urban areas, mountains, desert, clouds and farmland. ASCD performance during the first flights was degraded and after timing problems were identified and corrected performance improved. On the last flight the ASCD was correctly aligning the seeker to the FLIR 72% of the time. In addition to these flight tests the ASCD was used in the missile

validation portion of the Night Attack program. In this phase of the program the ASCD provided automatic hand-off from FLIR to seeker for a total of three missile launches resulting in three direct hits. In summary, the ASCD demonstrated proof of concept, documented the merits of the automatic hand-off approach and demonstrated fast and accurate hand-off.

IV. CURRENT EFFORT

Since the previous ASCD development effort was for a non-imaging seeker application, the current effort has been redirected to develop an ASCD to work with imaging IR seekers which are due to be introduced into the Navy inventory. The first step is to define an ASCD to work with a baseline system. Concurrent with this is an investigation of non-correlation alignment techniques such as precision gimbal angle slaving. The baseline system used is the A6-E TRAM FLIR and the Imaging IR Maverick seeker.

Areas of the ASCD to be defined included: Preprocessing of video data, number of digitization levels, effects of S/N ratio, seeker roll, scene complexity, reference size, hardware implementation and sensor interfaces. The first step in defining the ASCD is to analytically derive the relationship between probability of correlation, P_c, reference size, signal to noise ratio, and number of quantization levels. Figures 2, 3 and 4 show the behavior of the probability of correlation function, P_c. In the case of Figure 2 a S/N ratio greater than 5 is needed to optimize correlator performance when 30 independent pixels (M_i) are used. Figure 3 shows how the number of independent pixels affects the probability of correlation. As one would expect, the more pixels used the better the performance up to about 30 to 40 pixels. It is important to remember that the pixels being referred to here are independent pixels. That is, that the value of any independent pixel can not be determined by looking at its neighbors. Thus a reference of 30 independent pixels may spatially have more than 30 pixels; and is dependent on the complexity or detail of the particular scene. In Figure 4 the P_c is shown as a function of range to a ship target. The ship target case is of interest because, of the expected targets (land and sea), it represents the worst

case in terms of the lack of scene detail, low S/N ratio and long range. In all graphs it is significant to note that there is a large jump between the curves for two levels and those for three through infinity. This tends to indicate that an optimal system would be three levels in terms of maximum performance for minimum number of quantization levels.

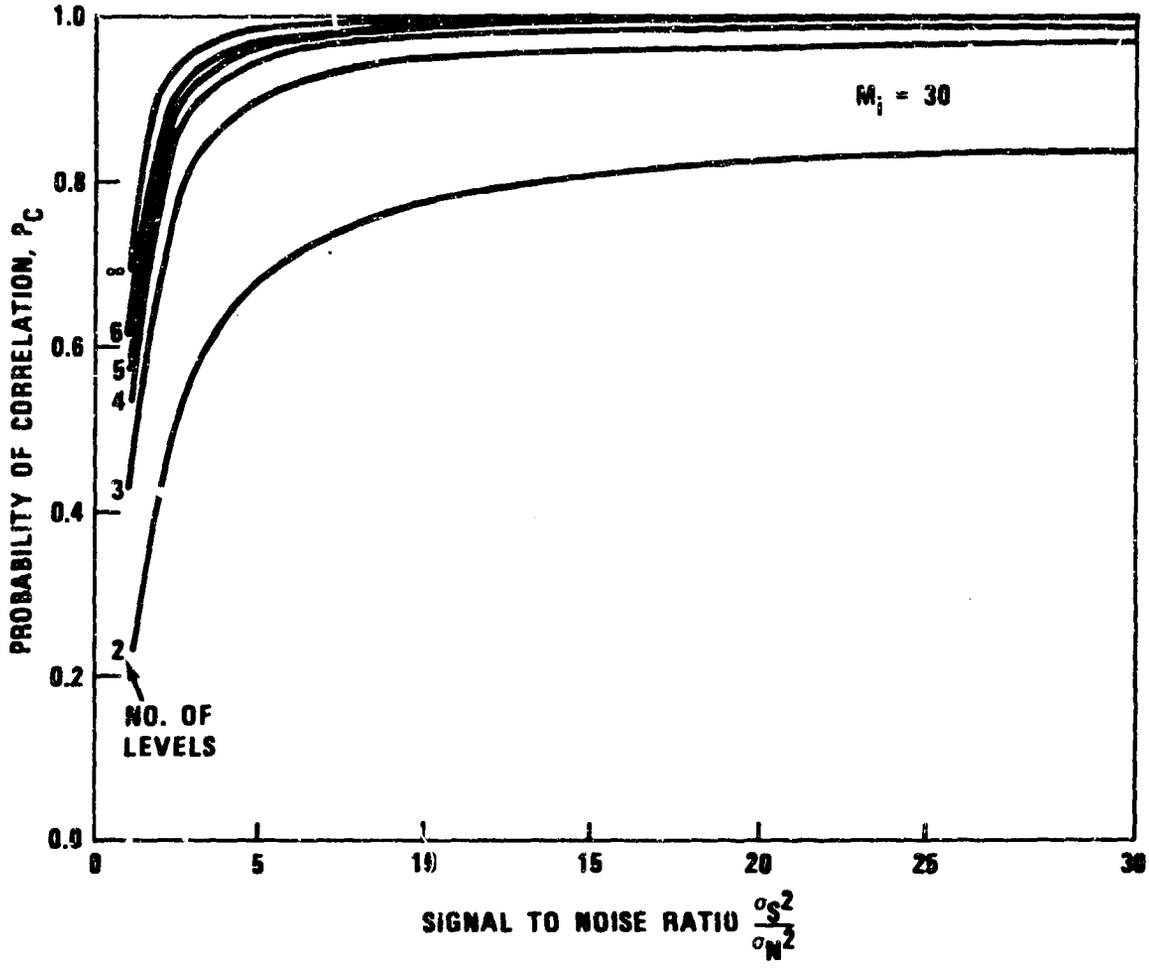


FIGURE 2. Probability of Correlation vs Signal-to-Noise Ratio

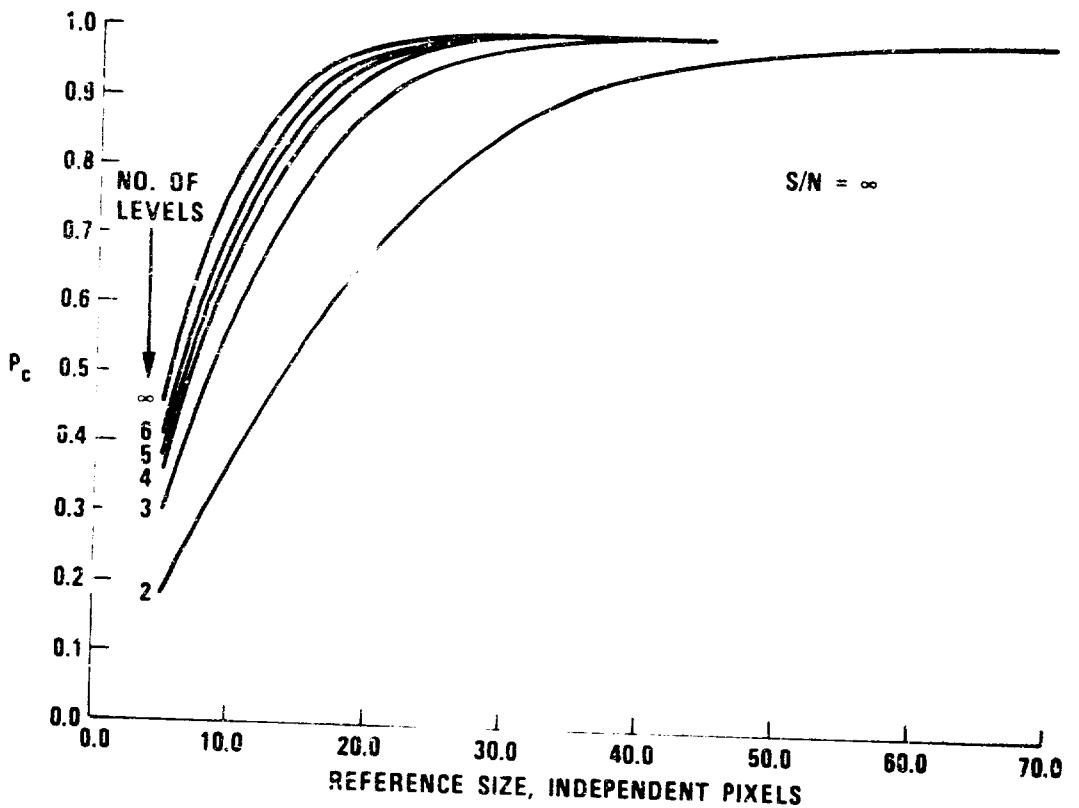


FIGURE 3. Probability of Correlation vs Reference Size

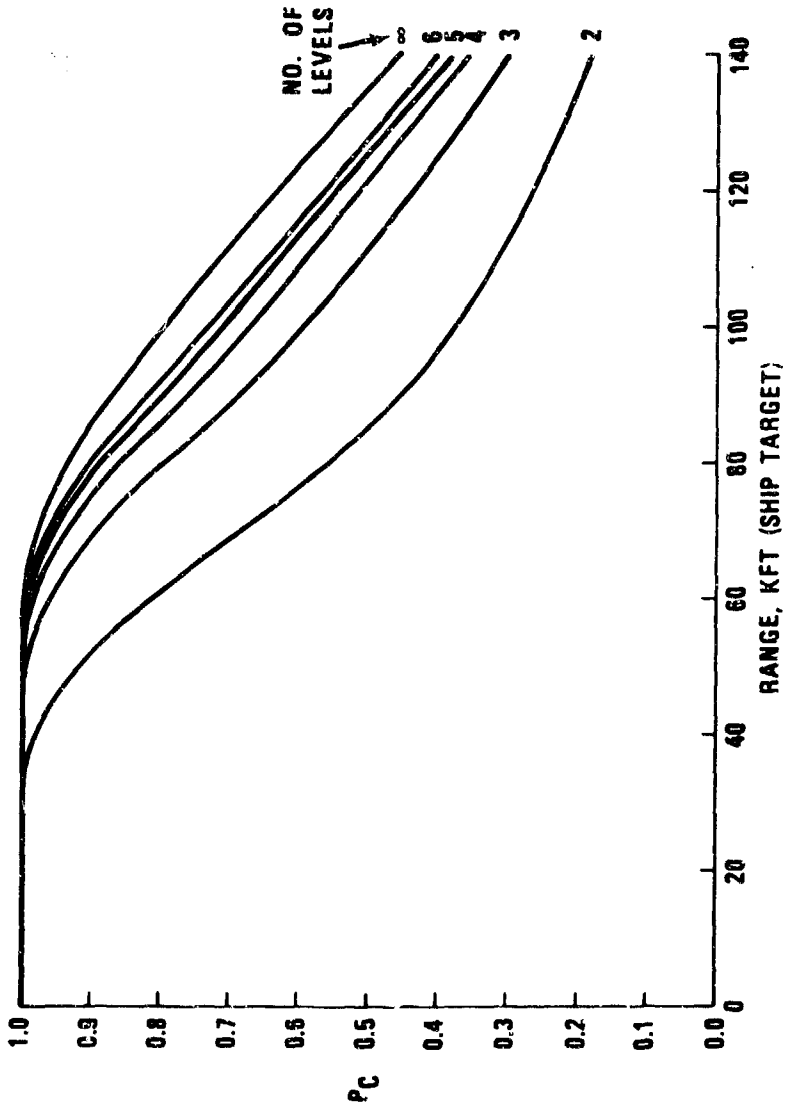


FIGURE 4. Probability of Correlation With Various Quantization Levels

To validate the analytical results a simulation using real IR Maverick video data was run. In the simulation two different TV frames of each ship image were digitized and run through the correlation algorithm using different quantization levels. For each correlation map produced a Figure of Merit, ΔP, was calculated. The peak to side lobe ratio (ΔP) is the ratio between the highest correlation peak and its next highest side lobe. It is a measure of the goodness of match between the two correlated data sets. An arbitrary rule of thumb based on experience, is that a $\Delta P > .1$ will yield reliable correlation. Figure 5 summarizes the results of the simulation and shows that for reliable correlation three or more levels are required. As a point of reference, Figure 5 shows that reliable correlation using a three level system can be performed out to a range of 78K feet. Comparing this range with Figure 4 we see that this corresponds to a probability of correlation of $P_c \approx .8$ which is in good agreement. The results of the analysis and simulation are as follows:

1. A correlator should be able to correlate reliably out to ranges of 80K feet against ship targets (assumed 471 feet long and 4:1 aspect ratio).

2. The most sensitive parameter to correlator performance is the number of independent pixels in the reference image.

3. Correlation performance improves with finer quantization levels and beyond three quantization levels improvement is slow.

With these results in mind one can conclude that a correlator using three levels will meet performance requirements for the baseline system and at the same time have a moderate hardware complexity. By looking at Figure 6 one can get an intuitive feel for why three level performance is significantly better than two levels. In setting a positive and negative threshold symetrically about zero the low amplitude high frequency signals, which may be considered "noise", are hidden in the zero level and make no contribution to the correlation function. This is in contrast to two level quantization with a zero threshold in which each transition if sampled would yield a digital transition and degrade the correlation process by inflating the value of the correlation function.

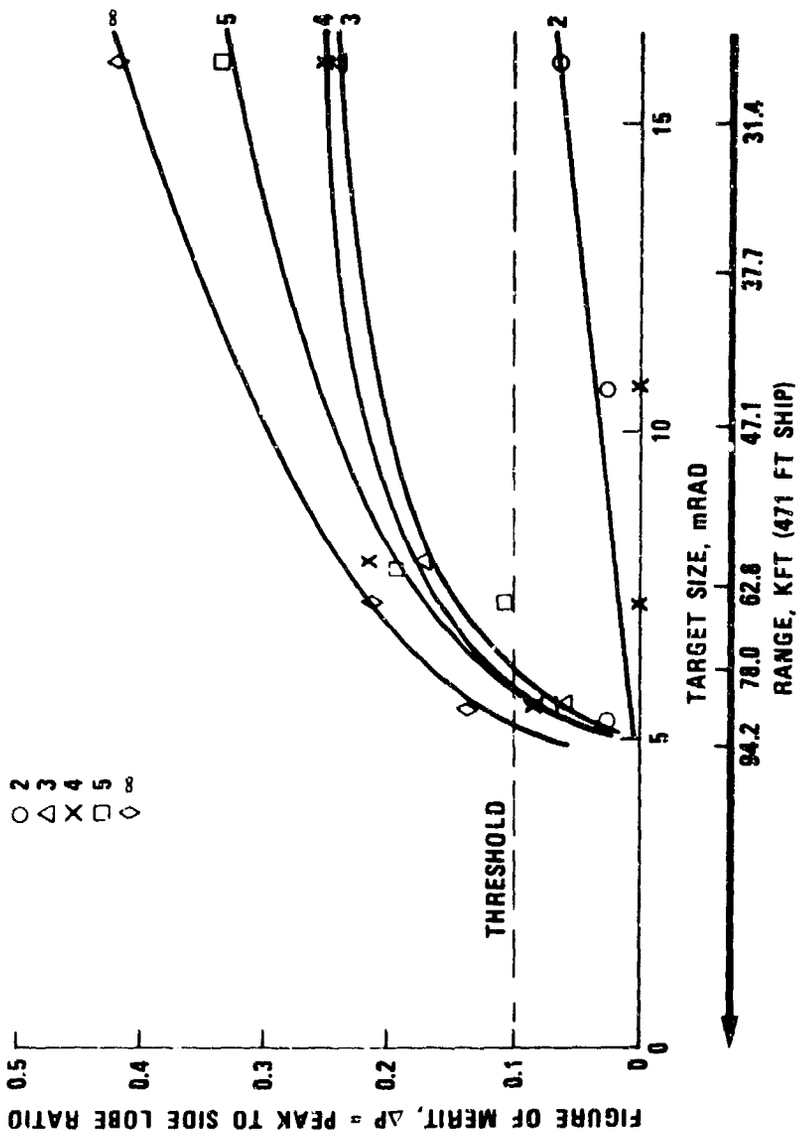


FIGURE 5. Figure of Merit, Δ P. vs Target Size

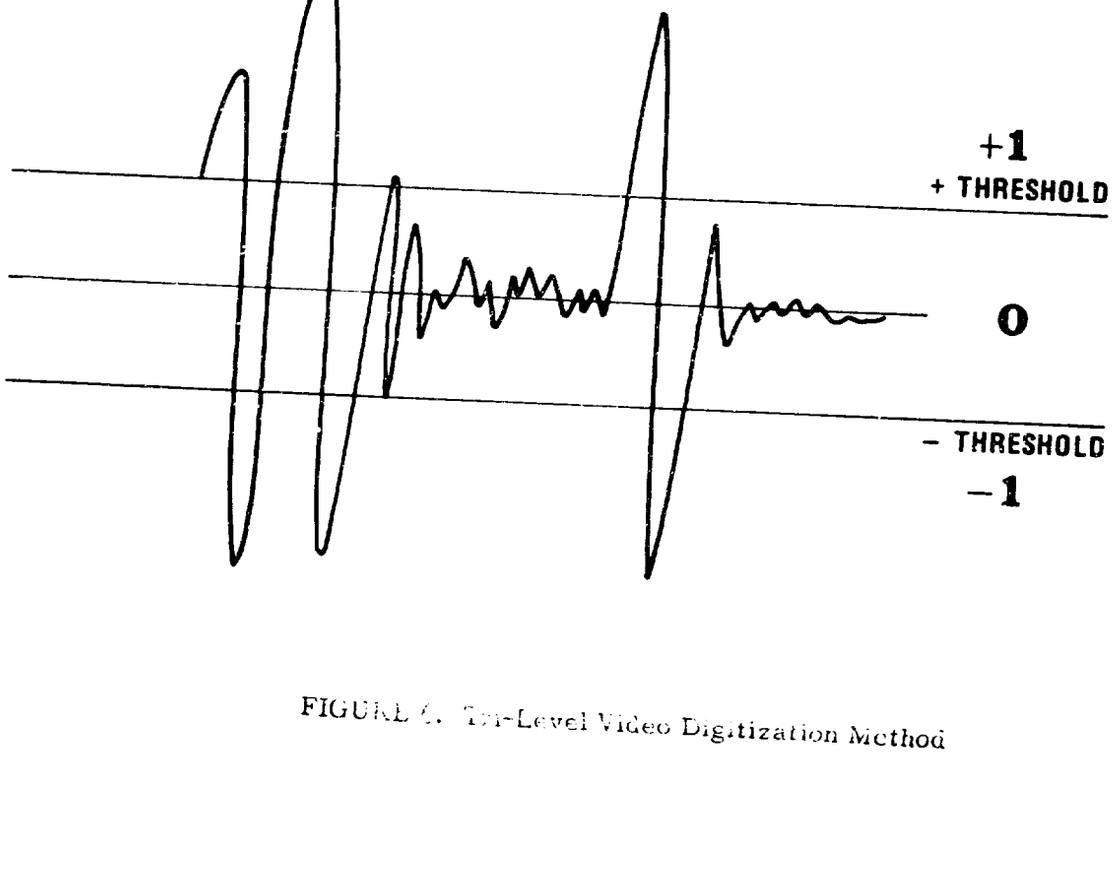


FIGURE 4. Tri-Level Video Digitization Method

Now let us look at the hardware aspect of a three level normalized product correlator. The normalized product correlation function (Figure 7) when implemented in hardware requires many summations and multiplications. It would be ideal to have the complete correlation and search process done at TV frame rates which are on the order of 1/30 of a second. This requires that the summations and multiplications be performed extremely fast. Fast summations are generally not a problem but fast multiplications are. To get around this problem one can code the multiplication in two bits of two's compliment arithmetic as shown in Figure 7. The multiplication result can then use combinatorial logic. That is, the most significant bit (MSB) of the product is a simple combinatorial function of the MSB's and LSB's of the multipliers. The least significant bit (LSB) of the product can be found in the same way. This allows two bit multiplication to be performed at a speed limited only by the logic family used.

This method of two bit multiplication is similar to the one bit multiplication which has been used in bi-level correlators. Making a rough extrapolation then one can estimate the hardware complexity of the three level correlator should be about twice that of a bi-level correlator. Thus for an increase in hardware complexity of two, one can obtain the significant increase in performance as shown previously by using a three level correlator instead of a two level.

V. SUMMARY

Thus one can conclude that an ASCD using a three level normalized product correlator will yield reliable performance and can be constructed to run real time with moderate hardware complexity. Future development efforts will be directed toward an ASCD to go with an IR weapon on the A6-E TRAM aircraft.

NORMALIZED PRODUCT CORRELATION FUNCTION

$$C_j = \frac{\sum_i S_i F_{i+j}}{\sqrt{\sum_i S_i^2 \sum_i F_i^2}}$$

S_i = i^{th} SEEKER VIDEO PIXEL

F_i = i^{th} FLIR VIDEO PANEL

F \ S	+1	0	−1
+1	+1	0	−1
0	0	0	0
−1	−1	0	+1

3 LEVEL TRUTH TABLE

F \ S	01	00	11
01	01	00	11
00	00	00	00
11	11	00	01

TRUTH TABLE CODING

FIGURE 7. Algorithm and Coding

Paper No. IIB-5, Presented at the Workshop on Imaging Trackers and Autonomous Acquisition Applications for Missile Guidance, 19-20 November 1979, Redstone Arsenal, Alabama.

IMPROVED METHOD FOR CORRELATION OF TV SENSOR IMAGES†

J. S. Boland, III & H. S. Ranganath
Electrical Engineering Department
Auburn University
Auburn, AL 36830

W. W. Malcolm
US Army Missile Command
Redstone Arsenal, AL 35809

Abstract

The problem of accurately aligning the line of sight (LOS) of an imaging seeker with the LOS of a precision pointing and tracking system (PTS) using correlation techniques is considered in this paper. A new method of locating a target, which is in the center of the higher resolution PTS, within the lower resolution seeker image is presented. The new method greatly improves the correlator accuracy and reliability. Simulation results using several typical digitized scenes are given to justify the conclusions.

Introduction

The particular application of scene matching considered in this paper is that of locating a reference image, obtained from a high resolution day-TV sensor, within a larger image, obtained from a lower resolution day-TV sensor. The high resolution system is located on one stores wing or in the nose of an attack helicopter and the imaging seeker is in a missile located in a stores rack mounted on the other stores wing of the helicopter. A high resolution (HR) system, usually referred to as the PTS, is used to acquire, recognize and automatically track potential targets such as tanks, personnel carriers, etc. When in the tracking mode, the target is in the center of the PTS field of view (FOV). The reference image is obtained by extracing a KxL array after preprocessing from the center of the PTS FOV.

The problem considered in this paper is that of locating the reference image, which contains the target, within the seeker image. The LOS of both the PTS and seeker sensors are inertially stabilized. Furthermore, it is assumed that the two lines of sight have been aligned either on the ground or previously in flight and that the seeker gimbals are slaved to the PTS gimbals. However, due to gyro drift, stabilization errors, helicopter flexure, etc., the target will not be at the center of the seeker FOV and therefore, must be located. After the target is

†This work was supported by the U.S. Army Missile Command, Huntsville, AL under contract DAAK40-79-M-0104.

located, error signals are generated and fed to the seeker gimbal torquers such that the seeker LOS is aligned with the PTS LOS (or such that the target is in the center of the seeker FOV). Once this is accomplished the seeker tracker locks on to the target, the missile is fired, and the helicopter can remask.

An algorithm for correlation of two images obtained from sensors sensitive in the visual spectrum (day TV sensors) has been demonstrated to work satisfactorily by simulations and hardware in a US Army Missile Command technology program. The new algorithm presented in this paper greatly improves the reliability and accuracy when correlating images obtained from similar sensors. Simulation results are given to justify the conclusions.

Image Preprocessing

Both the PTS and IR seeker are 525 line video imaging systems with a 30 Hz frame rate, 60 Hz field rate, and 4:3 aspect ratio. There is an approximately four-to-one ratio of the two FOV, however. Because of the above difference in the sensors, the two images must first be preprocessed such that they have the same spatial resolution. An algorithm to accomplish this is given in reference 1.

After the spatial resolutions of the two images are equalized, a number of correlation or matching methods can be investigated. For the remainder of the paper the dimensional relationships between the two images will be as shown in Figure 1.

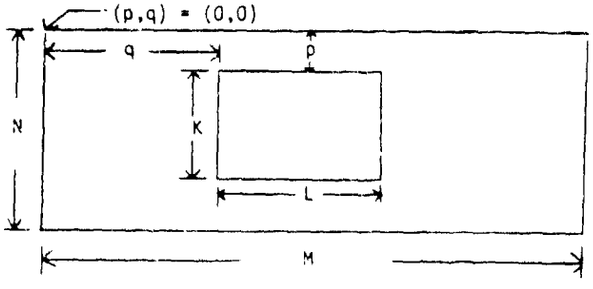


Figure 1. K X L HR image located at position (p,q) of N X M LR image.

The missile seeker image, referred to as the LR image, is represented by a N X M array of pixels. The values of N and M are determined from the choice of sampling rate and number of TV lines of the missile seeker system. Since the correlation is accomplished on each TV field and there are 240 active lines in a field, N is 240. Also, when sampling at 5 MHz there are approximately 260 samples during the 52 μsec active portion of the video line. M is equal to 256 in this paper. The PTS image or HR image is represented by a K X L array and might be all or only a portion, containing the target, of the PTS image after its spatial resolution has been converted to that of the missile seeker image. The p and q dimensions

in Figure 1 give the vertical and horizontal position of the HR image in the LR images. These indices start in the upper left corner of the LR image, where $p = q = 0$.

The classical approach to the problem of determining where two signals match is correlation. The correlation integral of two functions $f_1(t)$ and $f_2(t)$ is defined to be

$$C(\tau) = \int_{-\infty}^{\infty} f_1(t)\, f_2(t+\tau)\, dt \qquad (1)$$

where τ is allowed to take on values between $-\infty$ and $+\infty$. The value of τ which maximizes $C(\tau)$ in equation 1 is the correlation peak and is defined to be the match point between the two signals. It is obvious that determining the correlation peak consists of multiplying one signal by the other signal shifted by τ and then evaluating the area under the resulting curve.

The two TV images are first sampled and preprocessed to match spatial resolution and then stored in arrays. Since the HR image is a K X L array and the LR image is a N X M array, a two dimensional discrete correlation algorithm is given by

$$R(p,q) = \frac{1}{KL} \sum_{n=1}^{K} \sum_{m=1}^{L} HR(n,m)\, LR(n+p,m+q) \qquad (2)$$

for $0 \leq p \leq N - K$

$0 \leq q \leq M - L$

where $R(p,q)$ is the correlation function, and the division by KL is a scaling factor. Equation 2 is referred to as the Direct Method in this paper.

Using the algorithm of Equation 2, the selected K X L array of HR points is compared to each array of LR points of dimension K X L in the total N X M LR array. The algorithm produces the correlation array $R(p,q)$. In most situations the maximum value of the correlation function indicates image registration or match. However, in the present case since the LR image spans a wider field-of-view, and is obtained from a different sensor, $R(p,q)$ is actually a cross-correlation, and therefore it is possible that the maximum value of the correlation function does not indicate a target match between the HR and LR image. In order that the maximum value of the correlation function indicate target location in the LR image, both image arrays must be normalized. Normalization can be accomplished as shown in Equation 3.

$$R(p,q) = \frac{\sum_{n=1}^{K}\sum_{m=1}^{L} HR(n,m)\, LR(n+p,m+q)}{\left[\sum_{n=1}^{K}\sum_{m=1}^{L} HR^2(n,m)\right]^{\frac{1}{2}} \left[\sum_{n=1}^{K}\sum_{m=1}^{L} LR^2(n+p,m+q)\right]^{\frac{1}{2}}}$$

This obviously involves considerable more computation time than the unnormalized method of Equation 2.

In order to implement the algorithm in Equation 2 or 3 both the HR and LR video signals must be digitized. The process of digitizing continuous signals can be thought of as two separate steps. The first is sampling at discrete instants of time and the second is quantization. The sample rate was chosen to be 5 MHz in order to give approximately equal horizontal and vertical resolution to each pixel in a video field. The effects of quantization on the mean square signal-to-noise ratio have been reported in the literature [1-5]. A one-bit or two-level correlator is used in this report. When using a bi-level correlator, the normalized correlation of Equation 3 reduces to Equation 2 [1].

Two methods which have been successful with TV-to-TV correlation are based on quantizing to one when the signal level is above some local mean signal value and to zero otherwise [1]. One such local mean value is a running mean of the video based on a portion of the line immediately preceding the pixel being quantized. Another local signal average is based on the mean of an array of pixels about the pixel being quantized. These two methods, referred to as line averaging and area averaging, have been shown to work for TV-to-TV correlation [1].

Improved Correlation Method

Without any a priori knowledge about the scene being correlated, it has been shown that the reference image should be quantized to an equal number of zeroes and ones [1]. For optimal correlation results, each k x L subarray in the N x M low resolution image shown in Figure 1 should also be quantized to an equal number of zeroes and ones. To do this, however, would require requantization of a K x L subarray for each value of p and q, a task which cannot realistically be done with existing hardware. To overcome this problem the LR video is quantized only once using either a line averaging or area averaging technique. If the length of the line being averaged is L or the size of the sub-array being averaged is K x L, then any K x L subarray within the LR image should have approximately an equal number of zeroes and ones.

Using the method outlined in the paragraph above leads to the occurrence of false peaks in the correlation surface in some cases. The true peak for the scenes used in the simulations reported in this paper always appeared as one of the four highest peaks. The second highest peak was

obtained by masking out a 9x9 pixel area centered at the highest peak and then searching the remaining correlation surface for its highest peak. The third and fourth highest peaks were obtained similarly. In some cases, where the first peak was very broad and/or dominant, the second highest peak found was actually part of the highest peak. These cases can be spotted very quickly because their row and/or column location differs from the highest peak location by only five pixels.

A new method to bring out the true peak and to increase the ratio of the true peak to the next highest peak is outlined below and will be referred to as the improved method.

1. The reference from HR video, containing an equal number of zeroes and ones, is correlated with LR video, quantized using either the line or area averaging technique.

2. A predetermined number of highest peaks and coordinates of their occurrence are identified from the cross correlation surface. In this simulation the first four peaks are used because the true peak appears as one of them in all cases. Let, (I_1,J_1), (I_2,J_2), (I_3,J_3) and (I_4,J_4) be the coordinates of the first four peaks.

3. Then a sub-array of size $(K+k) \times (L+\ell)$ beginning at $(I_1-k/2, J_1-\ell/2)$ is chosen. (In this simulation $k = \ell = 6$.) The $(K+k) \times (L+\ell)$ sub-array is then quantized to zeroes and ones about the mean of this subarray. Cross correlation surface of size $(k+1) \times (\ell+1)$ is computed by correlating the reference of size $K \times L$ with the sub-array of size $(K+k) \times (L+\ell)$. The peak correlation value and its coordinates are identified. Let this be $R(I_1',J_1')$. $R(I_2',J_2')$, $R(I_3',J_3')$ and $R(I_4',J_4')$ are computed by repeating the above procedure using $(K+k) \times (L+\ell)$ sub-arrays corresponding to (I_2,J_2), (I_3,J_3) and (I_4,J_4), respectively.

Simulation results presented in the following section show that the improved method increases the probability of finding the true peak and reduces the probability of false peaks.

Simulation Results

Tables 1 and 2 contain the simulation results for a 32 by 32 reference array using the line average quantizer and the area average quantizer, respectively. Similar results for a reference array size of 16 x 16 are tabulated in Tables 3 and 4.

In order to implement the above method, one field of LR video must be stored in memory. In spite of the additional memory requirement, the following advantages make the improved method worthwhile.

1. Using the improved method on the four sub-arrays of LR video corresponding to the four highest peaks obtained by the initial correlation yielded the true peak as the highest peak every time when using a 32 by 32 or a 16 by 16 reference array. The case where the original correlation process yielded a false peak is marked with an asterisk in Table 3. In all of the simulations the first peak was higher using the improved method.

2. One measure of performance of a correlation technique is the ratio of true peak to the second highest peak. Simulation shows that in all but five of the 24 cases this ratio is higher after using the improved analysis. These ratios before and after the improved analysis are tabulated in Tables 1 through 4.

3. The difference in correlation values between successive peaks increases which indicates better signal-to-noise ratio. Figures 2, 3 and 4 show plots of the first four peaks for three of the scenes using the line and area average quantizers with reference array sizes of 32x32 and 16x16. The solid lines show the original correlation results and the dashed lines show the improved correlation method results.

The improvement in correlator performance is obvious from the figures. Consider Figure 4(c) which is a plot of the first four peaks for the NASA tower scene using the line average quantizer and a 16x16 reference array. The peak was expected at (33, 21), but when correlated using the line average quantizer, the true peak appeared as the second highest peak. The highest peak occurred at (105, 77). The difference between the first and fourth peak is only 9. However, after using the improved method, the true peak appeared as the highest peak, with the previous false peak at (105, 77) now being the fourth highest peak. The difference between the first and second peak is 28 and the difference between the first and fourth peak is 64.

Conclusions

From the above simulations and analysis it is concluded that this improved method yields significantly better correlation results than the previously reported correlation method. This method yields a higher probability of finding the true peak and then reduces the possibility of false peaks by limiting the dynamic search range.

Table 1. Line average quantizer with 32 x 32 reference array.

Scene	Method	True Peak	1st Peak	2nd Peak	3rd Peak	4th Peak	Ratio of First Peak to Second Peak
Jeep in front of the fence	Initial	(23,26)	(23,28) 777	(20,23) 667	(77,66) 631	(3, 5) 625	1.2314
	Improved		(23,28) 837	(20,23) 698	(77,66) 560	(3, 5) 623	1.4946
Jeep behind the fence	Initial	(39,37)	(39,37) 743	(27,59) 650	(8,56) 648	(13,30) 640	1.1430
	Improved		(39,37) 898	(13,30) 589	(8,56) 533	(27,59) 432	1.5246
Jeep in parking lot	Initial	(30,44)	(30,44) 751	(36,46) 636	(43,11) 627	(60,79) 625	1.1978
	Improved		(30,44) 914	(43,11) 555	(36,46) 513	(60,79) 464	1.6468
Parking lot	Initial	(28,35)	(28,35) 785	(60,45) 655	-	-	1.1984
	Improved		(28,35) 879	(60,45) 542	-	-	1.6217
NASA tower	Initial	(30,22)	(31,22) 683	(9,41) 648	(16,37) 642	(22,29) 630	1.0540
	Improved		(31,22) 886	(22,29) 841	(16,37) 752	(9,41) 670	1.0535
Water tower	Initial	(33,23)	(32,23) 703	(37,25) 640	(27,26) 615	(49,24) 599	1.1736
	Improved		(32,23) 812	(27,26) 752	(37,25) 679	(49,24) 568	1.4296

Table 2. Area average quantizer with 32 x 32 reference array.

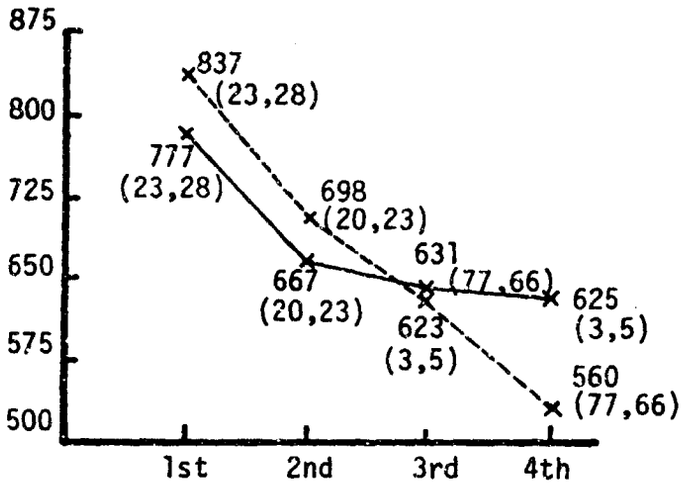
Scene	Method	True Peak	1st Peak	2nd Peak	3rd Peak	4th Peak	Ratio of First Peak to Second Peak
Jeep in front of fence	Initial	(23,26)	(23,28) 790	(25,33) 682	(23,23) 673	(13,55) 650	1.2154
	Improved		(23,28) 897	(23,23) 846	(25,33) 840	(13,55) 670	1.3388
Jeep behind the fence	Initial	(44,42)	(45,42) 800	(45,47) 701	(45,37) 689	(46, 8) 686	1.1662
	Improved		(45,42) 873	(45,47) 817	(45,37) 721	(46, 8) 718	1.2159
Jeep in parking lot	Initial	(25,49)	(24,49) 754	(24,44) 671	(25,27) 664	(10, 6) 653	1.1355
	Improved		(24,49) 914	(24,44) 764	(10, 6) 678	(25,27) 626	1.3481
Parking lot	Initial	(28,35)	(28,36) 810	(28,31) 743	(27,41) 696	(28,24) 693	1.1638
	Improved		(28,36) 862	(28,31) 804	(27,41) 802	(28,24) 751	1.1478
NASA tower	Initial	(25,27)	(26,28) 662	(8,30) 592	(78,75) 589	(26,66) 589	1.1182
	Improved		(26,28) 886	(26,66) 868	(8,30) 807	(78,75) 594	1.0207
Water tower	Initial	(28,28)	(28,27) 683	(50,56) 594	(66,59) 589	(2,50) 585	1.1498
	Improved		(28,28) 795	(50,56) 719	(2,50) 714	(66,59) 681	1.1057

Table 3. Line average quantizer with 16 x 16 reference array.

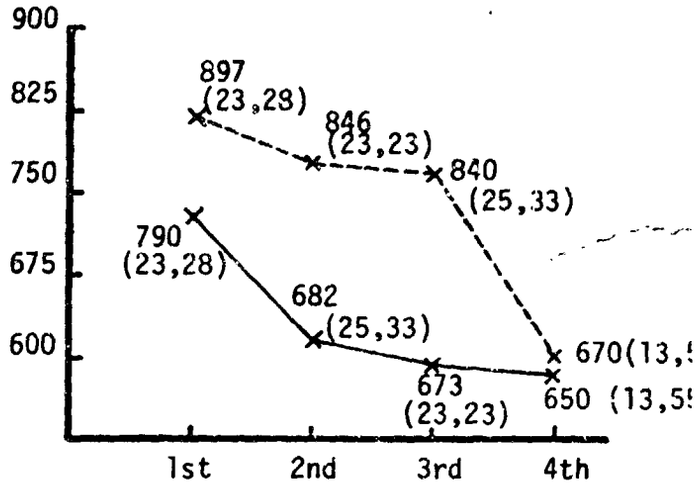
Scene	Method	True Peak	1st Peak	2nd Peak	3rd Peak	4th Peak	Ratio of First Peak to Second Peak
Jeep in front of the fence	Initial	(37,27)	(38,30) 207	(5,17) 193	(14,40) 192	(94,73) 189	1.0725
	Improved		(38,30) 227	(5,17) 187	(94,73) 139	(14,40) 120	1.2139
Jeep behind the fence	Initial	(47,48)	(47,49) 191	(63,33) 180	(21,43) 175	(6,69) 172	1.0611
	Improved		(47,49) 191	(63,33) 178	(6,69) 172	(21,43) 164	1.0730
Jeep in parking lot	Initial	(41,50)	(40,50) 218	(75,35) 202	(64,30) 197	(53,19) 194	1.0792
	Improved		(40,50) 219	(75,35) 176	(53,19) 175	(64,30) 168	1.2443
Parking lot	Initial	(48,47)	(48,48) 206	(46,22) 196	-	-	1.0510
	Improved		(48,48) 218	(46,22) 144	-	-	1.5138
NASA tower	Initial	(33,21)	(105,77)* 208	(34,22) 206	(12,43) 205	(19,38) 199	1.0097
	Improved		(34,22) 226	(19,38) 198	(12,43) 185	(105,77) 162	1.1414
Water tower	Initial	(37,30)	(37,30) 194	(32,37) 181	(103,48) 180	(28,58) 178	1.0778
	Improved		(37,30) 218	(32,37) 197	(28,58) 191	(103,48) 141	1.1414

Table 4. Area average quantizer with 16 x 16 reference array.

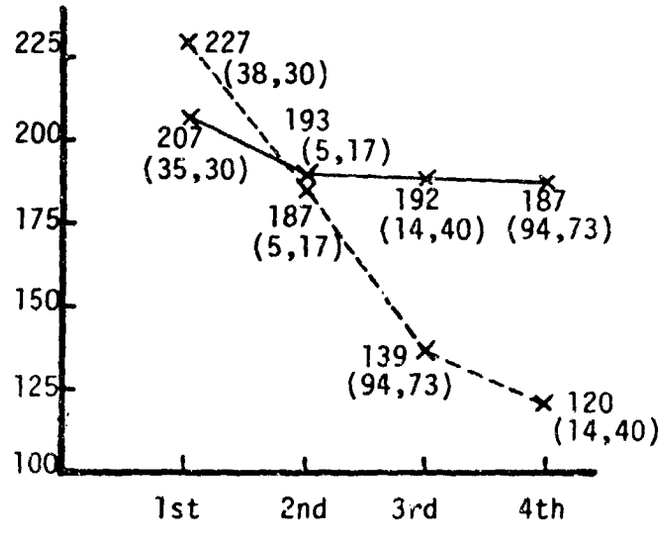
Scene	Method	True Peak	1st Peak	2nd Peak	3rd Peak	4th Peak	Ratio of First Peak to Second Peak
Jeep in front of fence	Initial	(32,32)	(33,35) 223	(23,61) 205	(34,40) 196	(23,45) 195	1.0878
	Improved		(33,35) 227	(23,61) 187	(33,40) 186	(23,45) 167	1.2139
Jeep behind the fence	Initial	(52,53)	(53,54) 201	(40,43) 192	(28, 8) 179	(54,26) 178	1.0468
	Improved		(53,54) 207	(40,43) 186	(54,26) 175	(28, 8) 167	1.1129
Jeep in parking lot	Initial	(36,55)	(36,55) 213	(13,67) 193	(12,59) 189	(13,75) 187	1.1036
	Improved		(36,55) 229	(13,75) 175	(13,67) 172	(12,59) 167	1.3085
Parking lot	Initial	(28,35)	(28,36) 216	(40,68) 203	(28,31) 200	(47,85) 198	1.0640
	Improved		(28,36) 232	(28,31) 212	(47,85) 158	(40,68) 152	1.4684
NASA tower	Initial	(28,27)	(29,28) 189	(92,62) 175	(27,91) 171	(27,42) 170	1.0800
	Improved		(29,28) 220	(27,96) 212	(27,42) 180	(92,62) 89	1.0377
Water tower	Initial	(32,35)	(32,34) 179	(8,23) 177	(54,83) 173	(53,68) 170	1.0112
	Improved		(32,34) 218	(53,68) 166	(8,23) 157	(54,83) 139	1.3132



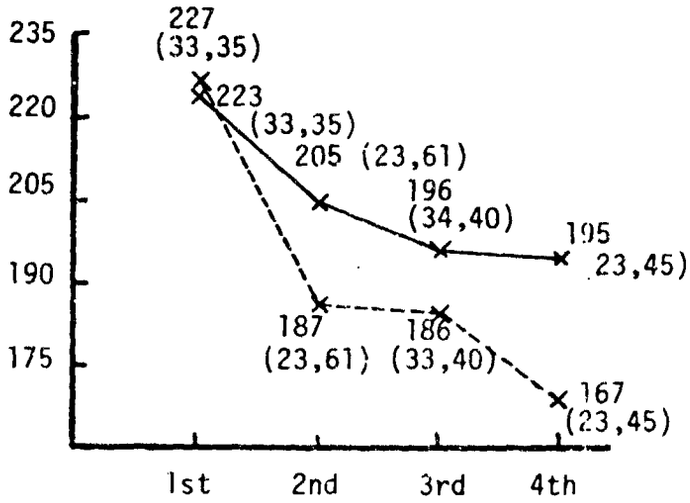
a) Line average quantizer (32 x 32 reference).



b) Area average quantizer (32 x 32 reference).

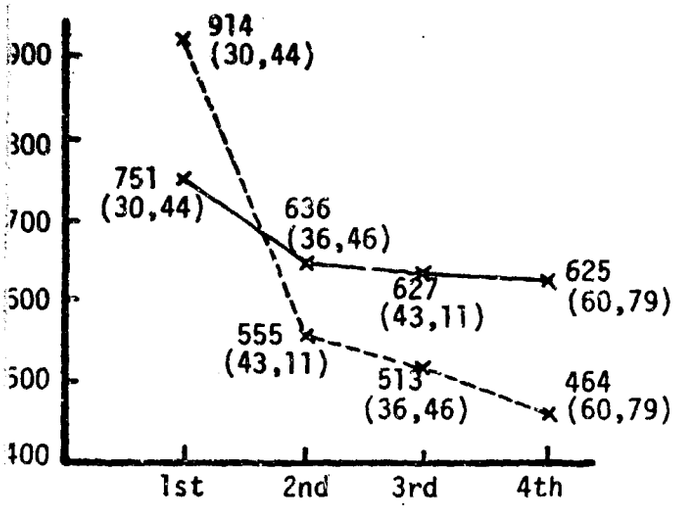


c) Line average quantizer (16 x 16 reference).

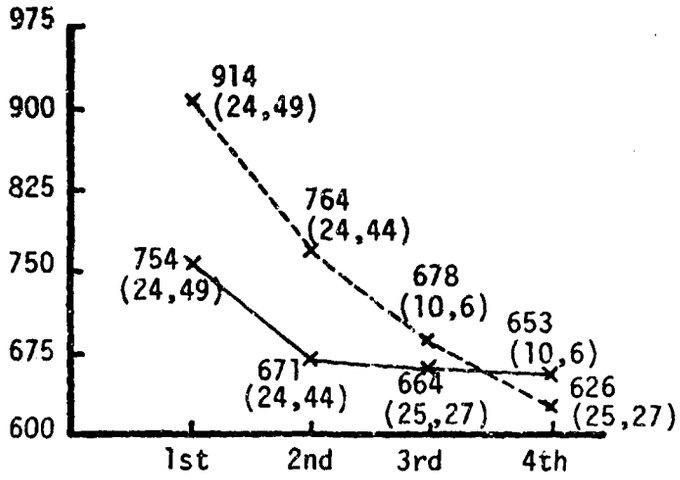


d) Area average quantizer (16 x 16 reference).

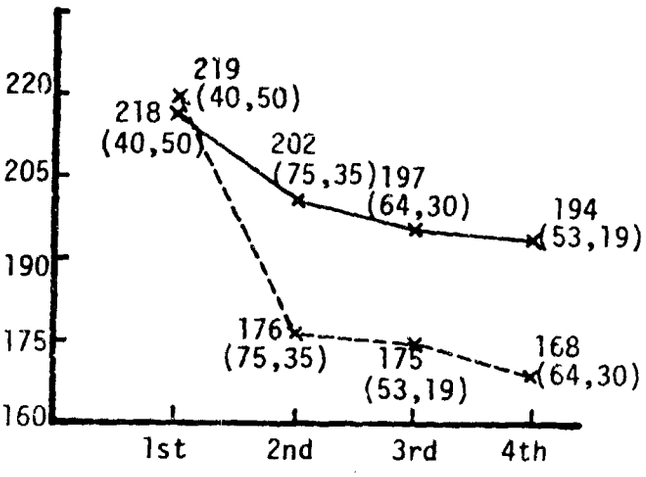
Figure 2. Correlation values of first four peaks before (solid lines) and after (dashed lines) improved analysis for jeep in front of fence.



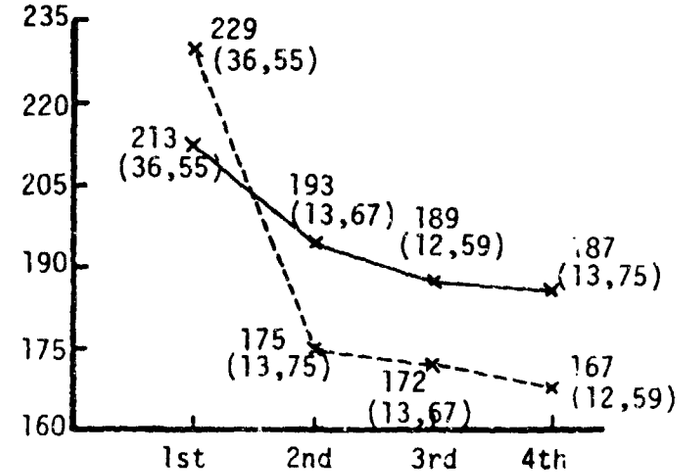
a) Line average quantizer (32 x 32 reference).



b) Area average quantizer (32 x 32 reference).

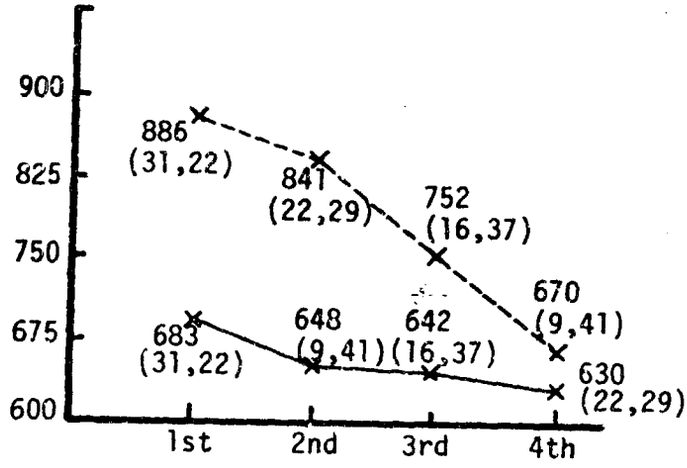


c) Line average quantizer (16 x 16 reference).

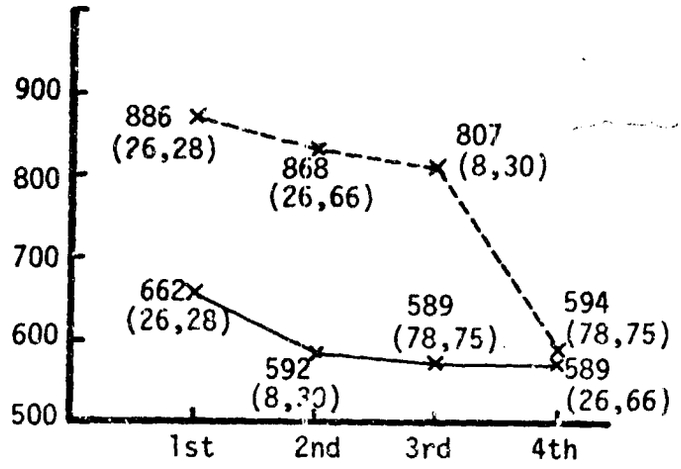


d) Area average quantizer (16 x 16 reference).

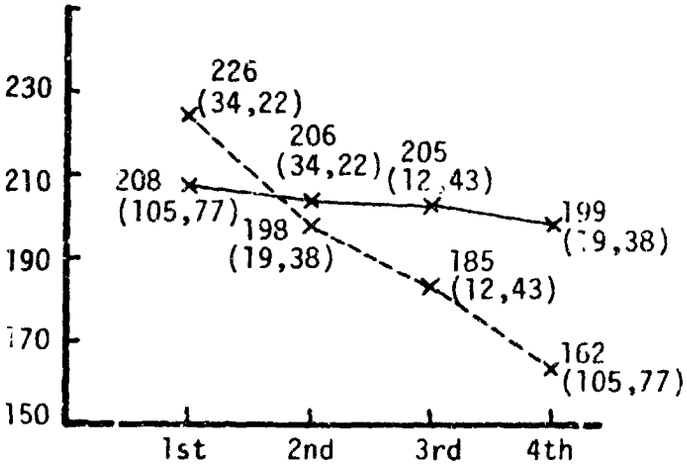
Figure 3. Correlation values of first four peaks before (solid lines) and after (dashed lines) improved analysis for jeep in the parking lot.



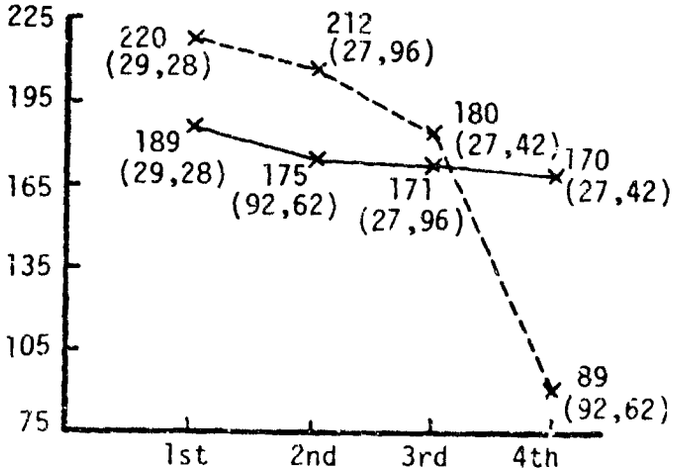
a) Line average quantizer (32 x 32 reference).



b) Area average quantizer (32 x 32 reference).



c) Line average quantizer (16 x 16 reference).



d) Area average quantizer (16 x 16 reference).

Figure 4. Correlation values of first four peaks before (solid lines) and after (dashed lines) improved analysis for NASA tower scene.

References

1. Boland, III, J. S., Pinson, L. J., Kane, G. R., Honnell, M. A., and Peters, E. G., "Automatic Target Hand-Off Using Correlation Techniques," Final Technical Report Contract DAAHO1-76-C-0396, U.S. Army Missile Research and Development Command, Redstone Arsenal, AL, 31 January 1977.

2. Bowers, F. K. and Klinger, "Quantization Noise of Correlation Spectrometers," Astronomy and Astrophysics Supplement," Vol. 15, 1974, pp. 373-380.

3. Cooper. B. F. C., "Correlations with Two-Bit Quantization," Aust. J. Physics, Vol. 23, 1970, pp. 521-527.

4. Hagen, J. B. and Farley, D. T., "Digital Correlation Techniques in Radio Science," Radio Science, Vol. 8, Aug.-Sept., 1973, pp. 775-784.

5. Peters, E. G., Boland, III, J. S., Pinson, L. J., and Malcolm, W. W., "Quantization Effects on Signal Matching Functions," IEEE Transactions on Information Theory, Vol. IT-24, No. 3, May 1978, pp. 395-398.

Paper No. IIB-6, Presented at the Workshop on Imaging Trackers and Autonomous Acquisition Applications for Missile Guidance, 19-20 November 1979, Redstone Arsenal, Alabama.

DESIGN AND EVALUATION OF AN AUTOMATIC HAND-OFF CORRELATOR

BY

T. W. ILER - GOODYEAR AEROSPACE CORPORATION

M. PITRUZZELLO - MICOM, REDSTONE ARSENAL

P. H. McINGVALE - MICOM, REDSTONE ARSENAL

ABSTRACT

In order to make the utilization of fire-and-forget missiles practical in a heliborne environment, it is necessary to rapidly and accurately handoff to the lower resolution missile seeker a target which was recognized in the high resolution target acquisition system. MICOM and Goodyear Aerospace Corporation developed the Automatic Target Hand-Off Correlator (ATHOC) to perform this function for two television sensors. The ATHOC employs digital area correlation techniques to continuously compare the "seeker" video to the target acquisition system video (which has the target centered in its tracking gate). The error between the actual location of the target in the "seeker" video and its desired location (the center of the field-of-view) is used to generate error signals to drive the seeker gimbals so as to center the desired target. An exhaustive engineering evaluation program was conducted on the ATHOC at MICOM. This included developing techniques for system evaluation which were then used to quantify the critical internal and external system parameters.

INTRODUCTION

The design of the Automatic Target Hand-Off Correlator (ATHOC) system (see Figure 1) required a unit which could be airborne and would achieve the sensor boresighting and target hand-off in less than one second. This time constraint dictated the use of real-time digital area correlation techniques. Since the target of interest could be moving through background information which could add correlation noise, the correlation reference aperture size had to be programmable.

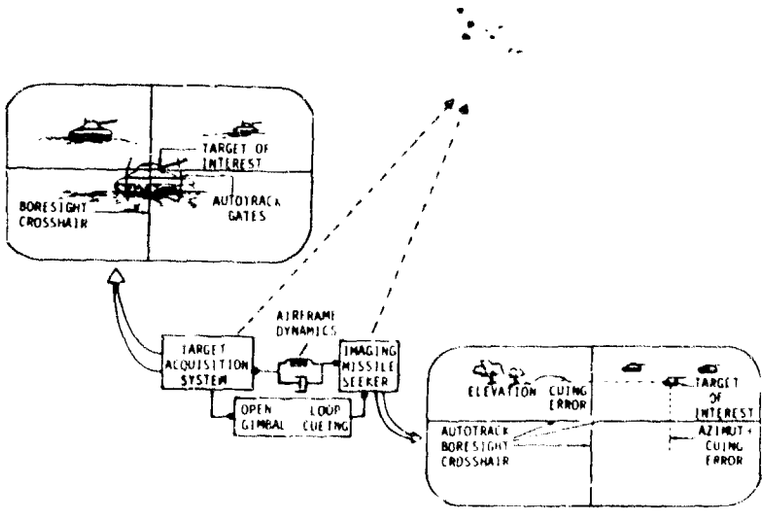


Figure 1. Imaging Missile Seeker Target Hand-Off Problem

Additionally, the anticipated scale factor variation from missile to missile required a means of matching the scale factor of the designation system to that of the seeker system. Finally, the ATHOC had to signal the fire-control system when a satisfactory boresight sequence was completed.

The real-time operation of all these functions required the use of an interrupt driven custom designed bit-slice microprocessor. Also a trade-off study between system performance and system complexity resulted in the selection of a two-bit trilevel Mean Absolute Difference correlation algorithm that could be realized with real-time hardware. For the laboratory mode of operation, a general purpose microprocessor was required to handle the data formatting and general I/O.

DESCRIPTION OF HARDWARE

The ATHOC system (see Figure 2) consists of airborne and laboratory control units, a correlator unit, and power supply unit. The airborne control unit is used by the weapon delivery personnel for in-flight control of the ATHOC system. It contains a mode control switch, a display control switch, and indicator lamps.

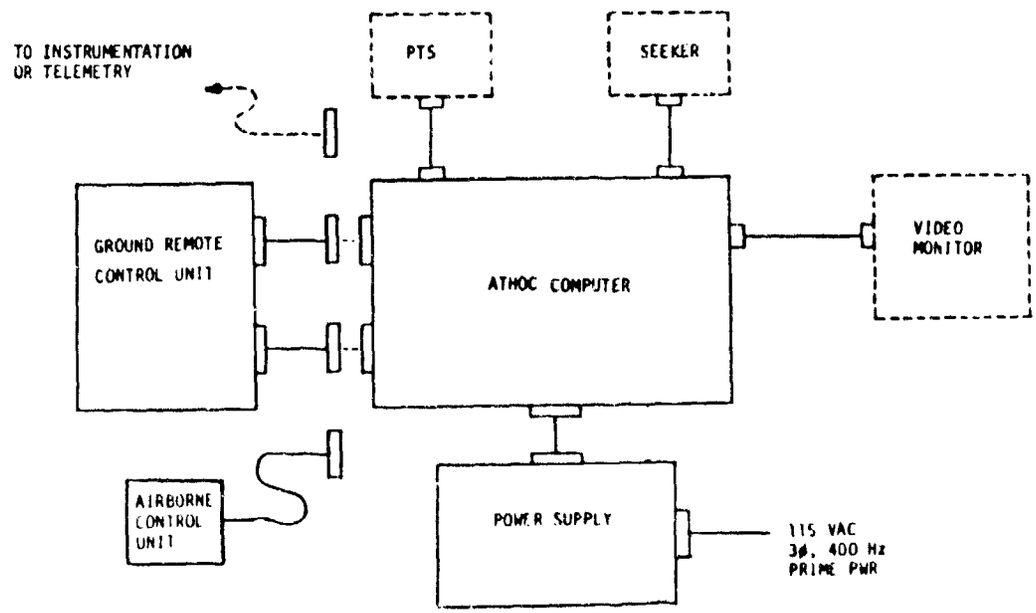


Figure 2. ATHOC System Block Diagram

The power supply unit contains seven individual modular power supplies and all the necessary control and distribution wiring to supply the ATHOC voltage/current requirement. Primary power is 115 VAC, 400 Hz, 3-phase configured for "Y" connected operation.

The laboratory control unit (see Figure 3) provides for operator control of certain processing parameters during laboratory tests. In addition to the functions contained on the airborne control unit, it contains a section for display of the correlator error signals, scale factor signals and correlation quality index. A data entry section provides for input from either a keyboard or cassette tape. Another section provides for monitoring of the microprocessor address and data bus and the priority interrupt signals.

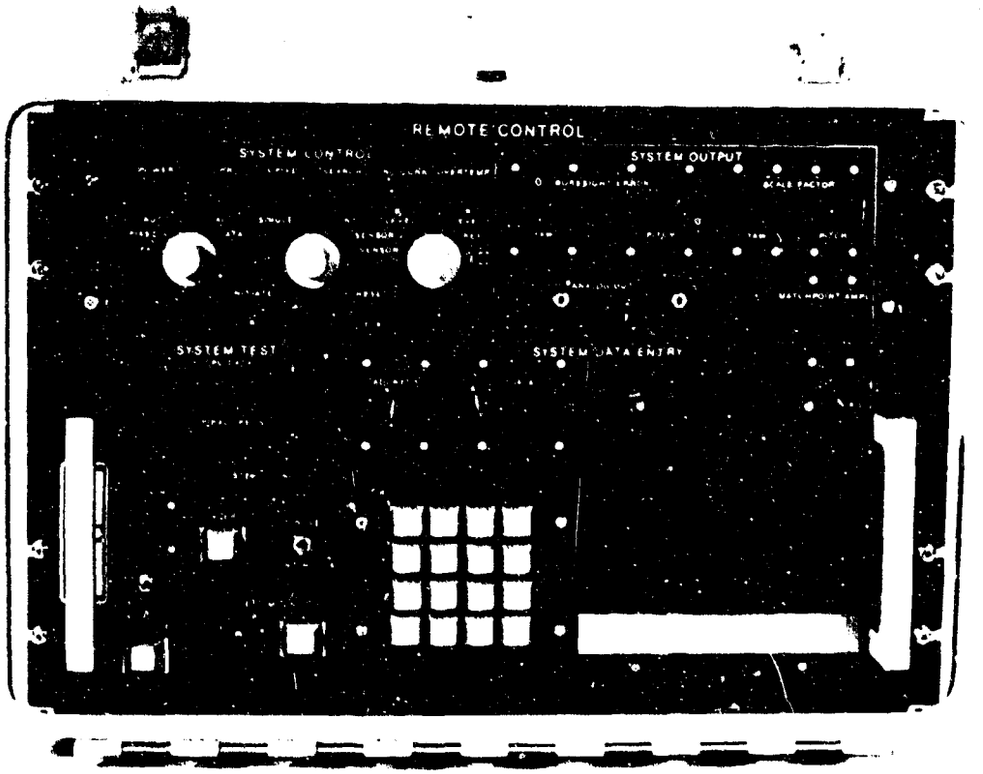


Figure 3. Ground Remote Control Unit

The correlator unit contains six subsections: video preprocessing, image reformatter, correlator array, position processor, interface and sequence control, and microprocessor. Each of these subsections will be described in the following section.

Figure 4 shows the system block diagram. Starting at the left, the video processor must switch to the selected video source, strip the composite sync from the selected composite video and finally digitize the selected video signal into two bits. Proper video digitization can be achieved only by using an adaptive slice.

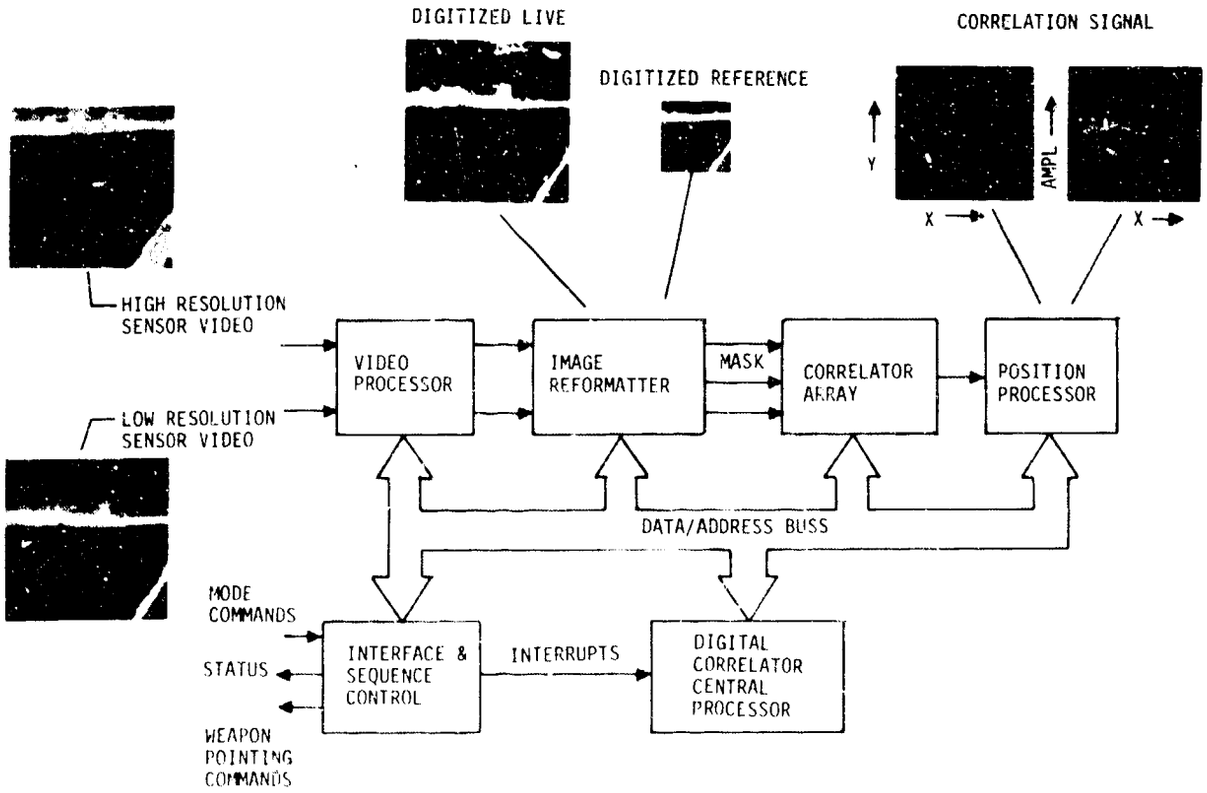


Figure 4. System Block Diagram

First a programmable low-pass filter is utilized to reject unwanted noise and sampling frequency aliasing. Then the effect of ramp shading and targets larger than the selected reference aperture size were minimized by using a programmable high-pass filter. The standard deviation of the video is approximated by smoothing the absolute value of the filtered video. A portion of this approximated standard deviation of the video is utilized to set both the negative and positive slice threshold. Table I lists the two-bit trilevel format.

TABLE I. DIGITIZED VIDEO FORMAT

INPUT VIDEC SIGNAL	NEGATIVE BIT	POSITIVE BIT	VIDEO CODE
WHITE	0	1	1
GRAY	0	0	0
BLACK	1	0	-1

The scaling of the digitized video is achieved in the image reformatter by using a digital integrator and predetermined sampling pattern. A running average of from 1 to 16 lines, depending on the desired scale factor, is computed in real-time. The running cell average is performed in a similar manner. The digital video is converted back to trilevel video by using digital magnitude comparators. The actual scale factor matching is achieved by masking the integrated video with the predetermined sampling pattern. The non-masked video sampled are then clocked into the correlator array for use as the digitized reference.

In order to achieve real-time processing, the correlation results from 32 adjacent lines of video had to be available simultaneously. A 32-line by 64-cell correlator array simultaneously performs the MAD correlation for all 2048 elements of the reference with the input video. The correlation signal is outputted as an 8-bit digital word at a rate of 5 MHz.

Using the standard 525 TV format, each field is digitized into 256 pixels per each of the 240 lines. Table II lists the available search limits versus reference array size.

TABLE II. SEARCH LIMITS VERSUS ARRAY SIZE

HORIZONTAL		VERTICAL	
REFERENCE SIZE (PIXELS)	SEARCH LIMIT % FOV OF SEEKER	REFERENCE SIZE (PIXELS)	SEARCH LIMIT % FOV OF SEEKER
64 *	± 38		
32	± 44	32	± 43
16	± 47	16	± 47
8	± 48	8	± 48

*This reference size assumes the ratio of the respective fields-of-view of the two sensors is less than four to one.

Referring to the right side of Figure 4, a digital peak detector which is located in the position processor determines the highest peak of the correlation surface. The value of horizontal and vertical hardware coordinate counters is stored in RAM every time a higher peak of the correlation surface is detected. At the end of each video field, the microprocessor firmware program is started via a hardware interrupt.

The highest peak amplitude of the correlation surface and the horizontal and vertical location of that peak are read from the position processor RAM. The peak amplitude when compared to a threshold value is used to accept or reject this field of correlation results. When the selectable number of good correlation results is obtained, the digital position error data is

converted to an analog voltage which is used to drive the seeker gimbal to a target boresight condition. When the target is within ±3 pixels of a boresight condition, a ready signal is sent to the fire-control system.

The above microprocessor requirements were easily met by using a customized bit-slice machine which features a 16-bit data bus, 1K of instructions words, and a 4 MHz execution rate. The interrupts were generated from the TV vertical sync signal by the sequence control.

The ATHOC (see Figure 3) has three major modes of operation with respect to scale factor and a fourth mode where system parameters can be changed. In the FIXED mode, the microprocessor simply initializes the system with the scale factor data which is contained in the non-volatile memory.

In the AUTO mode, the microprocessor initializes the system to the initial reference scale factor data, and a selected number of correlations are performed. The reference scale factor is then incremented by the delta which is contained in non-volatile memory. This process is repeated until the reference scale factor limit is exceeded. Subsequent correlations are performed with the reference scale factor which results with the highest correlation amplitude during the scale factor search.

The CALIBRATE mode is similar to the AUTO mode except that the horizontal and vertical scale factors are incremented independently. This mode is required when the aspect ratios of the respective sensors are not the same.

In the DATA mode, the system parameters can be changed. Table III lists the system parameters which are stored in non-volatile memory. These parameters can be altered only with the aid of the remote control unit's keyboard or digital cassette data entry section.

TABLE III. SYSTEM PARAMETERS

SAMPLE RATE (5 MHz or 2.5 MHz)
BILEVEL/TRILEVEL SELECT
REFERENCE VIDEO PROCESSOR PARAMETERS
LIVE VIDEO PROCESSOR PARAMETERS
HORIZONTAL AND VERTICAL BIAS
HORIZONTAL AND VERTICAL REFERENCE SCALE FACTOR
INITIAL, DELTA, LIMIT & RATIO
NUMBER OF CORRELATIONS FOR VALID MATCH
MATCHPOINT THRESHOLD
REFERENCE UPDATE RATE
REFERENCE SIZE
INPUT ANGLE SCALE FACTOR
OUTPUT ANGLE SCALE FACTOR
POSITION LIMIT FOR VALID MATCH

EVALUATION OBJECTIVE

The objective of MICOM's ATHOC test program was to quantify the effects of internal and external system parameters on system performance. Parameters considered critical were:

1. Internal parameters - reference size, reference white/black pixel ratio, and the ability of the ATHOC to correlate regardless of scene content (i.e., scene dependence of performance).

2. External parameters - field-of-view errors, sensor roll misalignment, problems associated with slaving a seeker to the ATHOC.

EVALUATION APPROACH

One key to the success of this test program was in the mixed laboratory and field tests which took place. The approach used was to record many and varied "real life" scenes in the field using a two field of view gimballed TV system. The TV system and recorders were mounted in an enclosed van and driven to several elevated test sights to simulate the slant range views seen in typical helicopter imagery. The recorded scenes were then played into the ATHOC in the lab where the various critical parameters could be varied at will. Thus the recorded scenes became a constant rather than a variable in the evaluation process. Also, freed from many of the problems associated with "live" testing (such as flight schedules, range schedules, etc.), it was possible to run a statistically significant number of tests.

Another important key to the program's success was the development of a reliable means to separate "good" from "false" correlations. The method is based to a certain extent on the peak-to-sidelobe ratio that has been used for years, but it is more reliable in the case of the real time correlator described here. The method developed was to study the statistics of the field-to-field variation in correlation peak position. If the correlation peak was always significantly higher than any sidelobes, the field-to-field variations in peak position would be small. However, if the peak is not always significantly higher than all the sidelobes, video noise can cause one or more of the sidelobes to temporarily exceed the amplitude of the true peak. Since false peaks are usually randomly distributed about the correlation surface and the area of the surface occupied by the true peak is small, there is a good probability that the distance between the true peak and the false peak will be more than several pixels. Thus, if a false peak condition exists, the field-to field variations in peak position will be significantly higher than for a true peak condition.

TEST RESULTS

The test results are summarized in Figures 5 through 10. In Figure 5, it can be seen that for category I scenes (i.e., scenes in which the reference and live images come from the <u>same</u> scene), a relatively few number of peak

positions is needed to produce a highly reliable measure of the "goodness" of the correlation peak. For category II scenes (i.e., scenes in which the reference and live images are taken from different scenes), more peak positions are necessary to produce highly reliable results. Figure 6 clearly shows the relation between probability of correlation (P_c) and reference image size for several aspect ratios. Figure 7 clearly shows the effects of white/black pixel ration on P_c. Figure 8 shows the effects of field-of-view scaling errors on P_c for a 3.11:1 field-of-view scale factor. Figure 9 is a similar graph except the scale factor is 1:1. Figure 10 demonstrates the effects of roll misalignment between the two sensors. Table IV shows the effects of visibility on P_c.

TABLE IV . SUMMARY OF BASELINE PROBABILITY OF GOOD CORRELATION TESTS

VISIBILITY	NUMBER OF TEST SCENES	PROBABILITY OF GOOD CORRELATIONS FOR THE FOLLOWING DATA SAMPLES		
		600	20	5
GOOD (10 kM or GREATER)	40	.95	.95	.95
FAIR (5 kM - 10 kM)	21	.81	.81	.76
BAD (LESS THAN 5 kM)	44	.23	.18	.25

CONCLUSIONS

A TV-to-TV automatic hand-off correlator was built using state-of-the-art technology. This correlator performed successfully under a wide variety of situations typical of a field environment. The critical parameters affecting correlator performance were identified and their effects were quantified. In addition, a new method for judging "goodness" of correlation for real-time correlators was developed which offers promise for future lab and field tests.

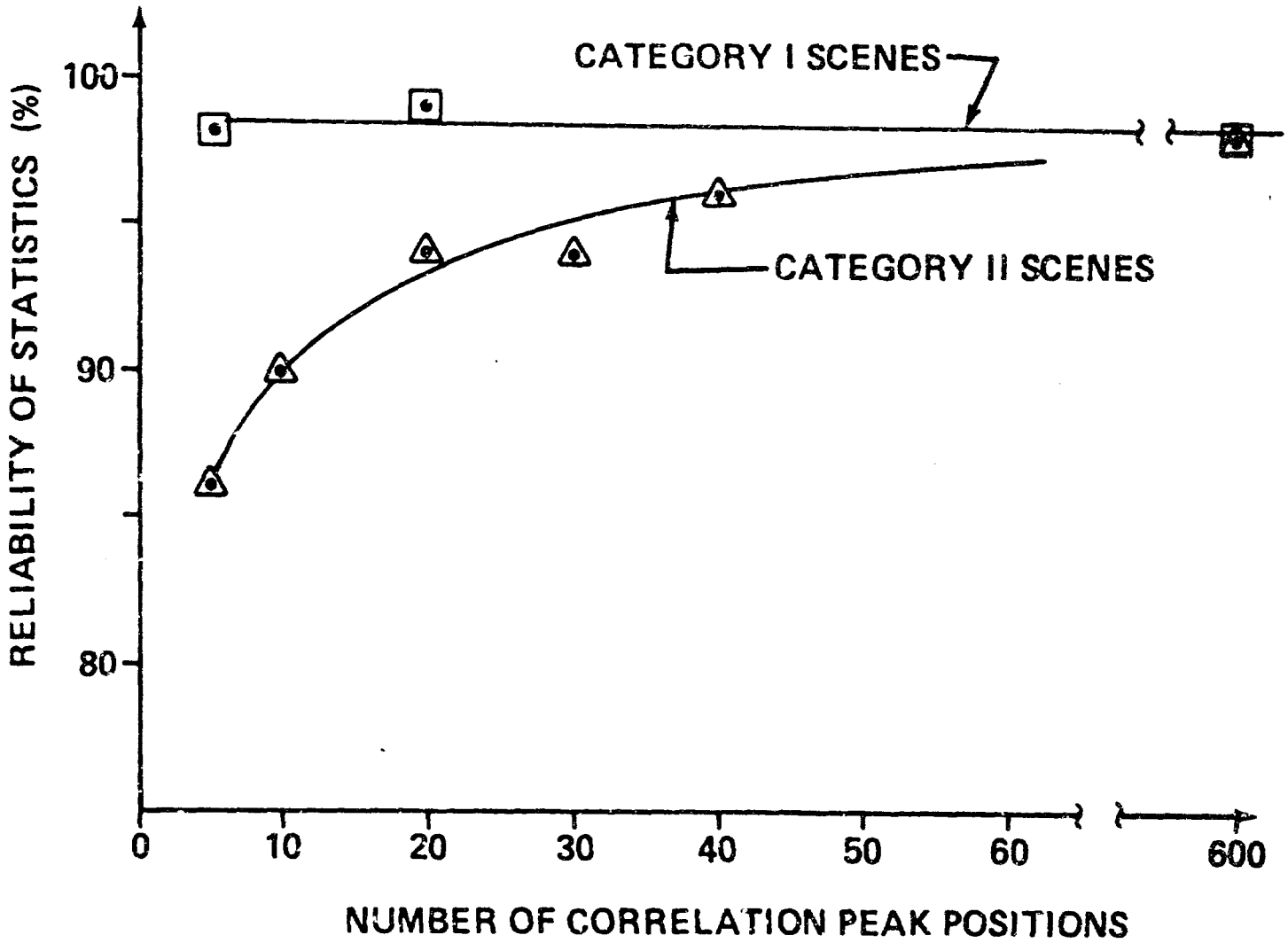


FIGURE 5. RELIABILITY OF THE STATISTICAL METHOD AS A FUNCTION OF THE NUMBER OF PEAK POSITIONS

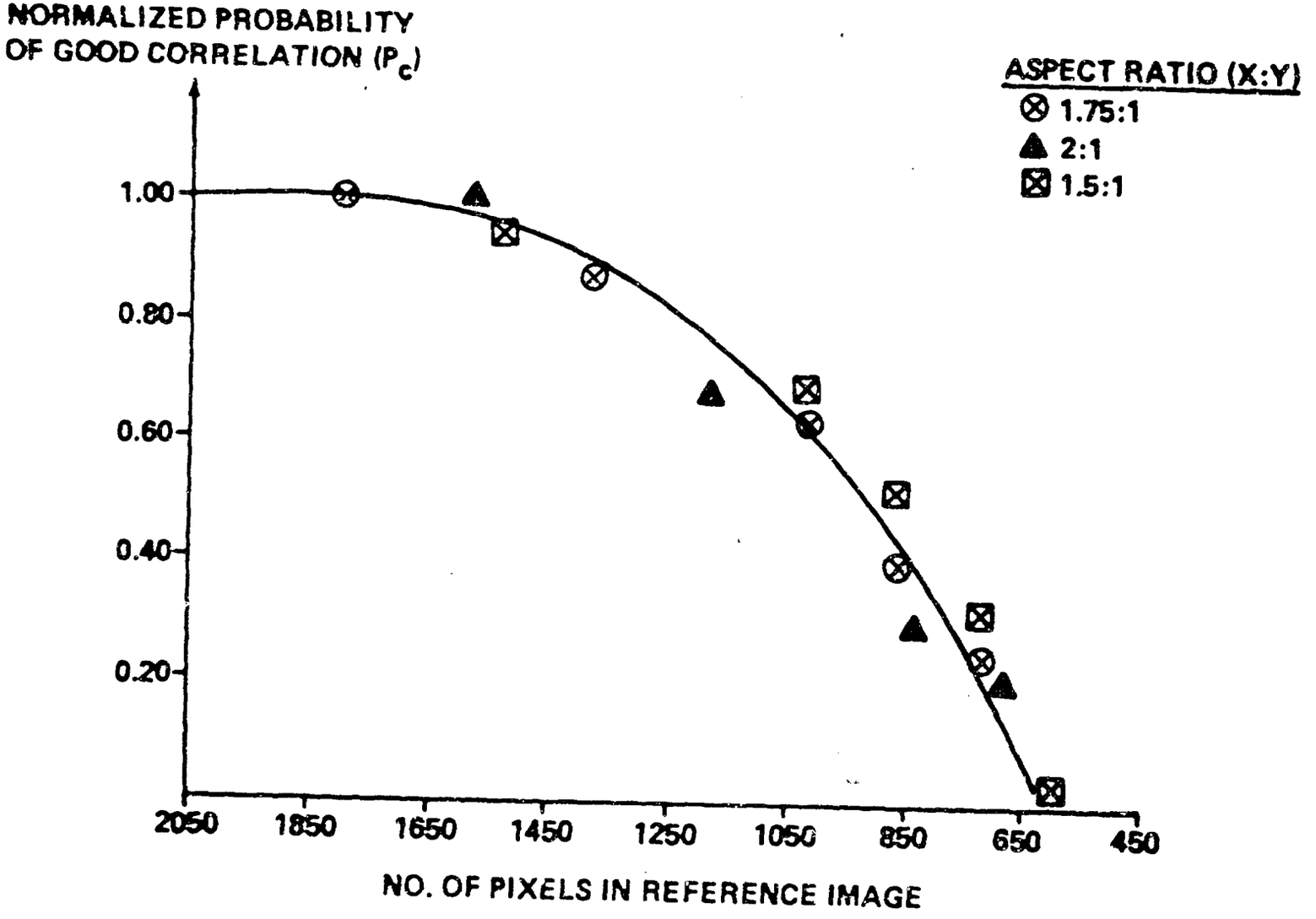


FIGURE 6. PROBABILITY OF GOOD CORRELATION AS A FUNCTION OF REFERENCE IMAGE SIZE

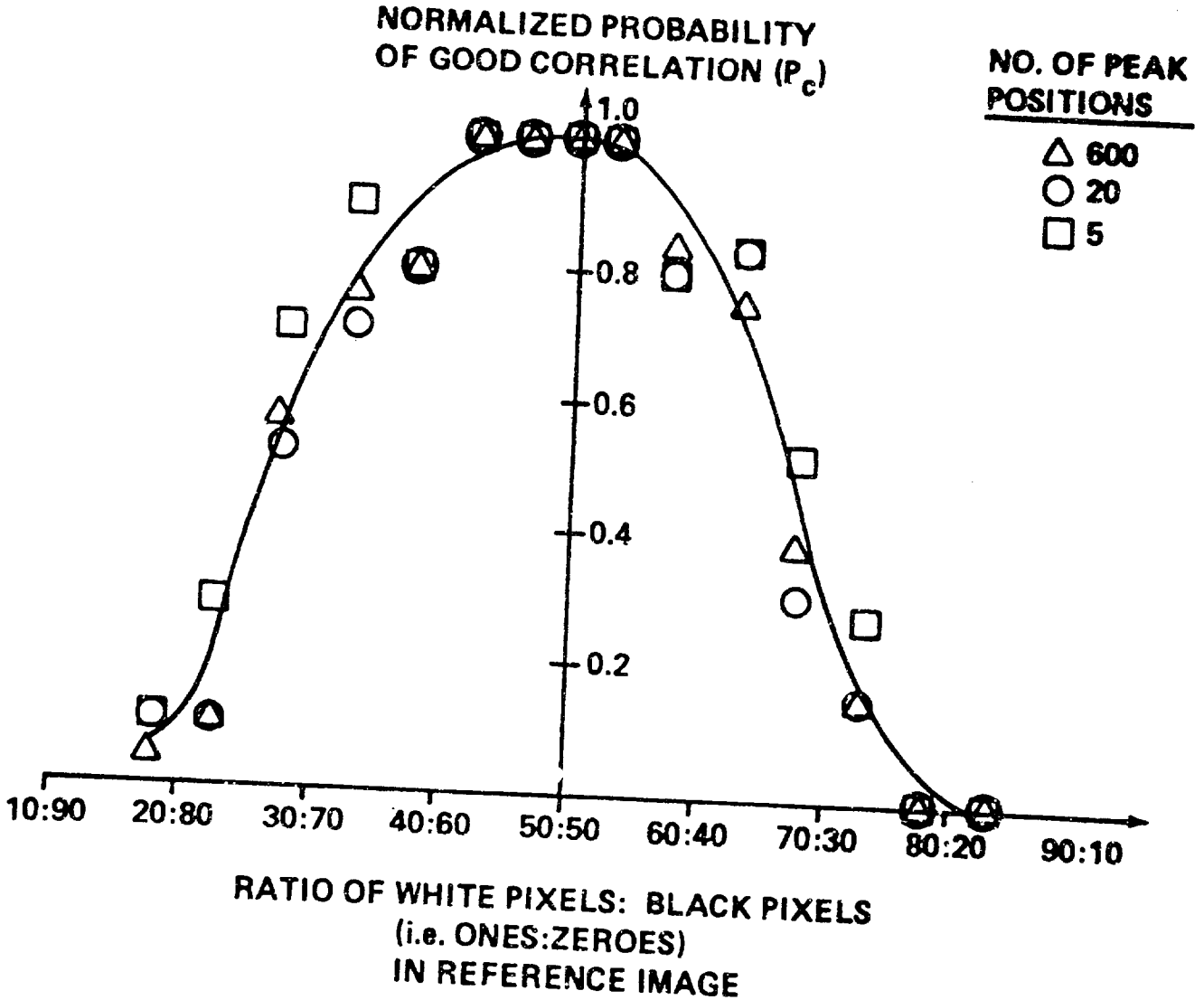


FIGURE 7. PROBABILITY OF GOOD CORRELATION AS A FUNCTION OF REFERENCE WHITE/BLACK PIXEL RATIO

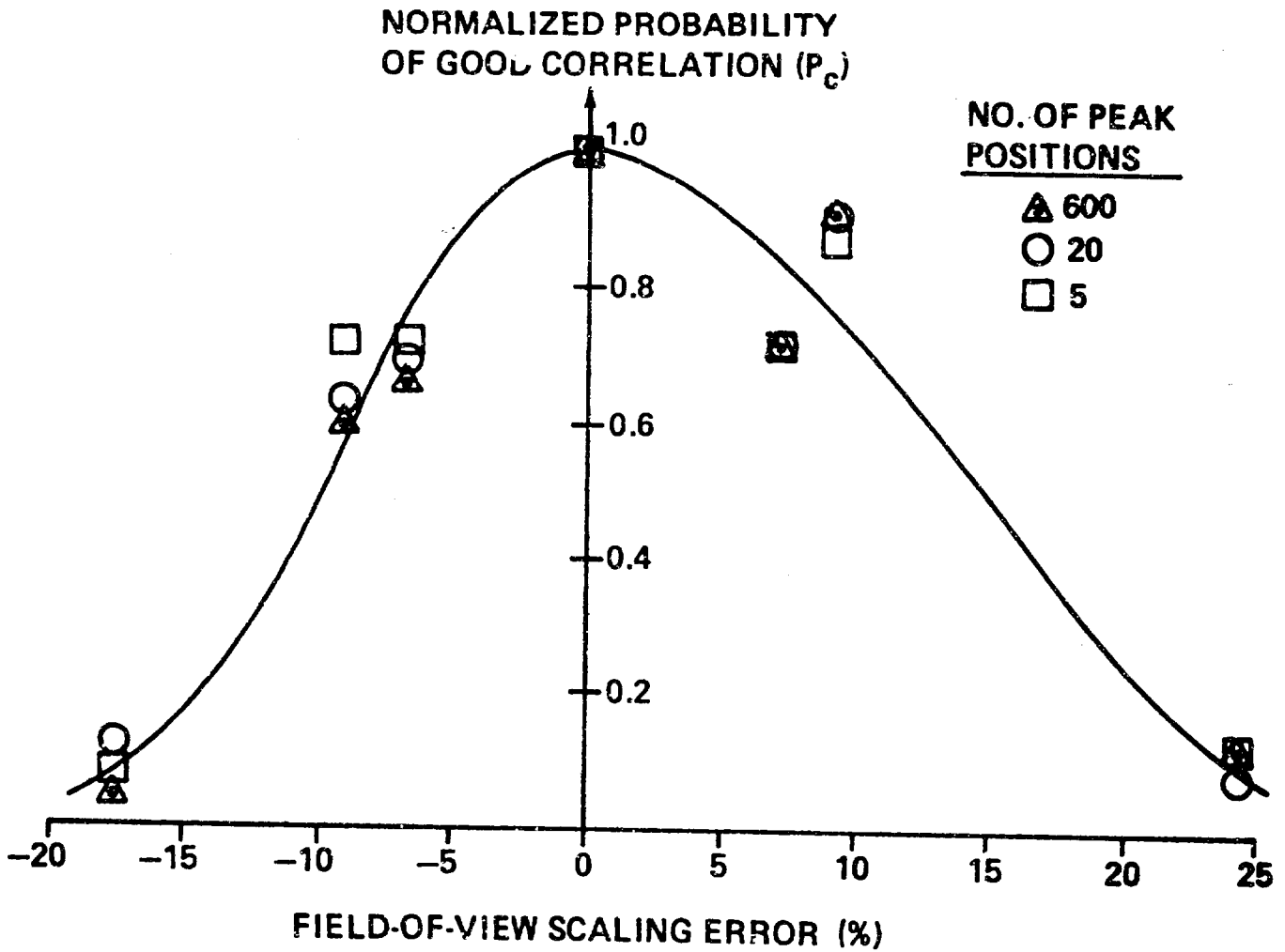


FIGURE 8. PROBABILITY OF GOOD CORRELATION AS A FUNCTION OF FIELD OF VIEW SCALING ERROR(3.11 TO 1 NOMINAL SCALE FACTOR)

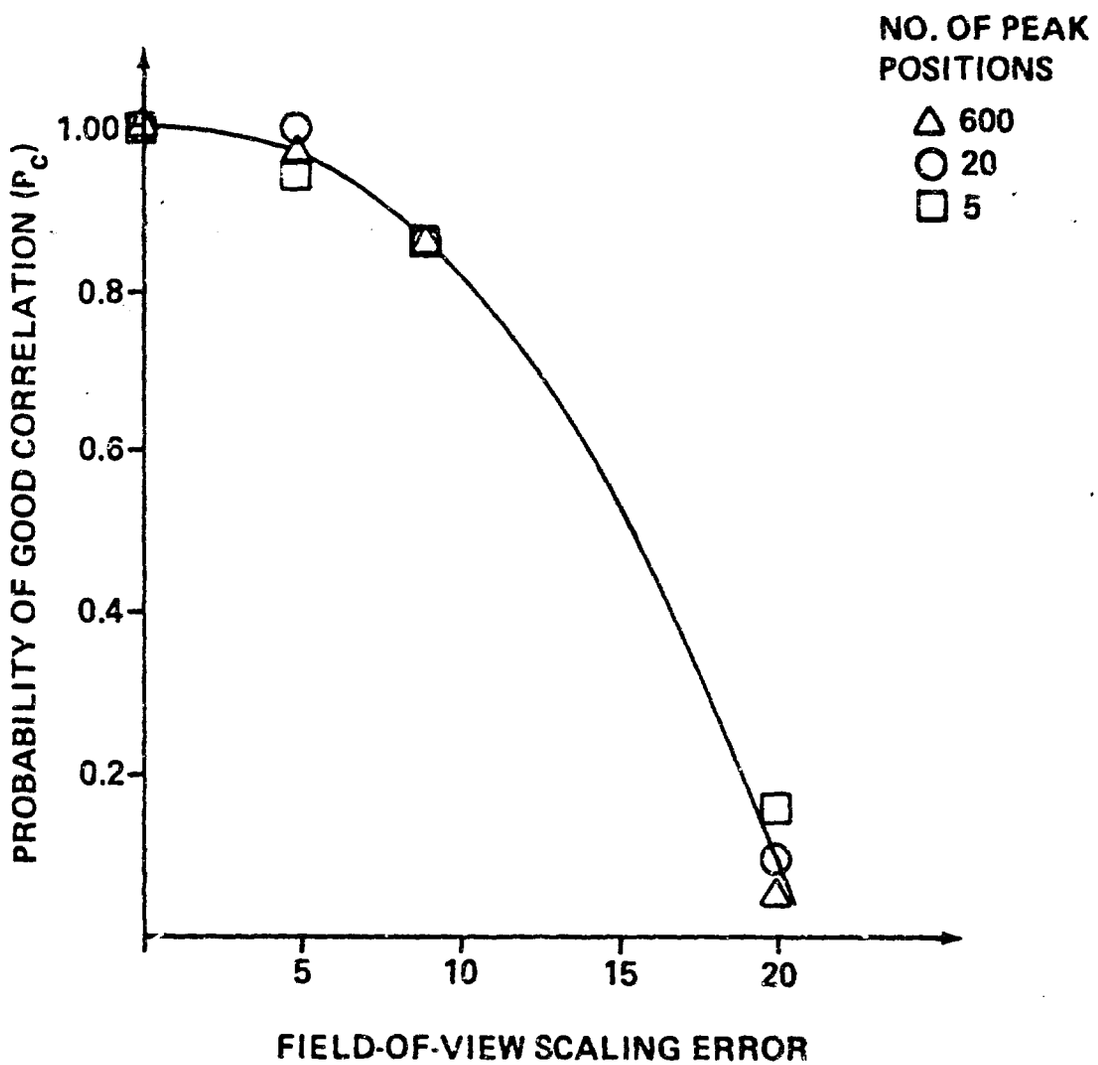


FIGURE 9. PROBABILITY OF GOOD CORRELATION AS A FUNCTION OF FIELD OF VIEW SCALING ERROR (1.0 TO 1.0 NOMINAL SCALE FACTOR)

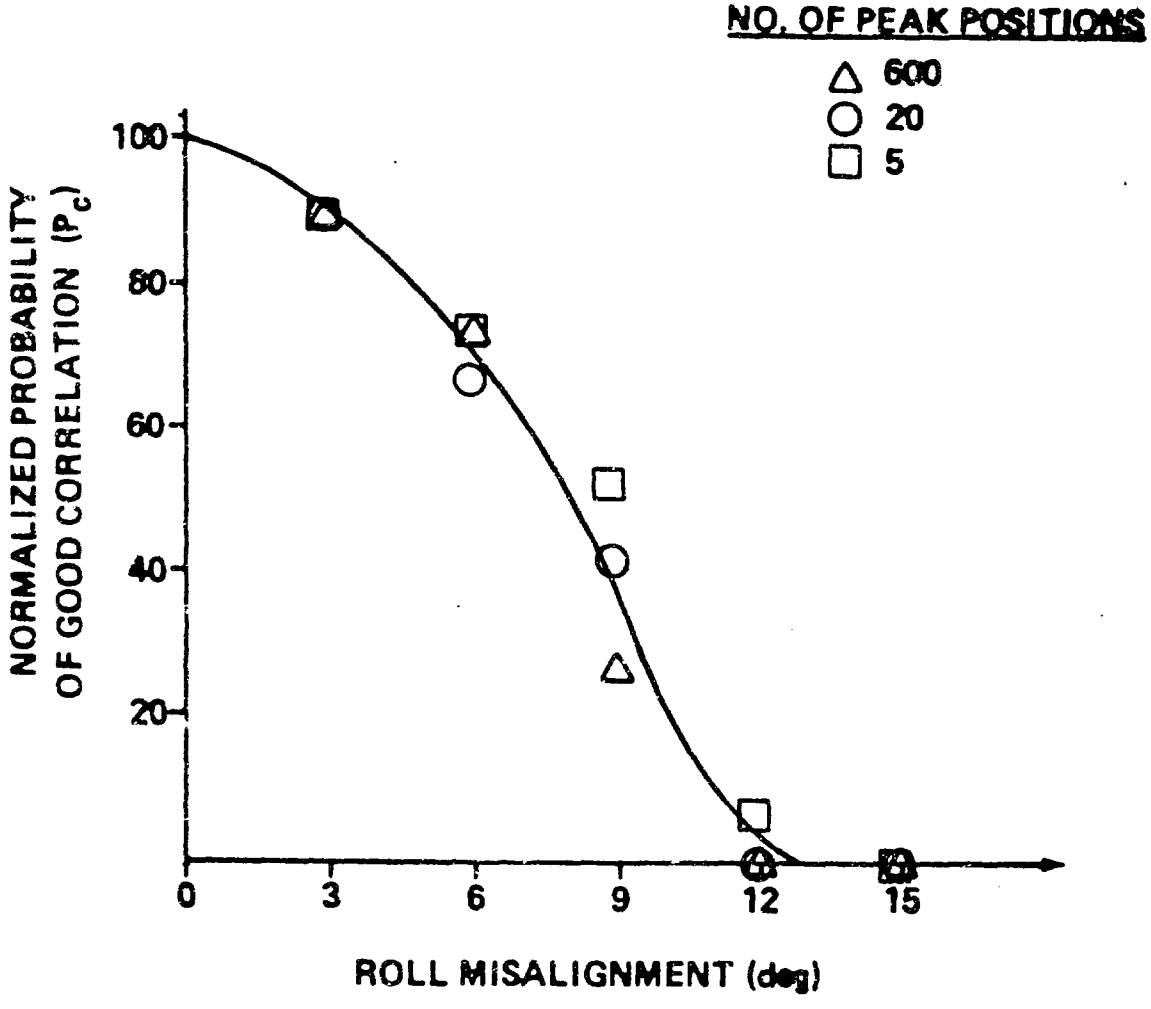


FIGURE 10. PROBABILITY OF GOOD CORRELATION AS A FUNCTION OF SENSOR ROLL MISALIGNMENT

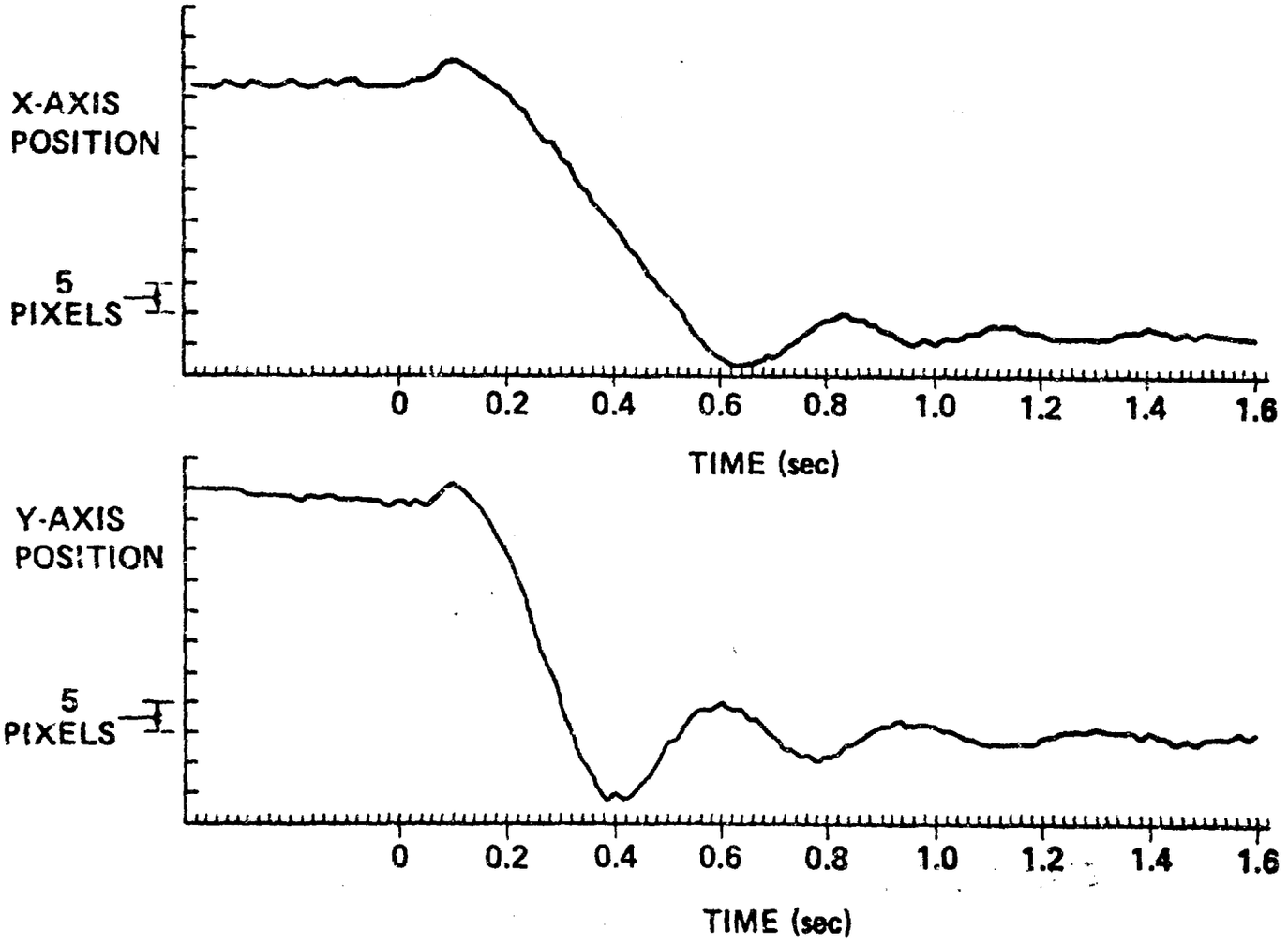


FIGURE 11. TARGET POSITION VERSUS TIME FOR A TYPICAL AUTOMATIC HANDOFF

SESSION IIIA

IMAGE UNDERSTANDING AND AUTONOMOUS ACQUISITION FOR MISSILE GUIDANCE

Paper No. IIIA-1, Presented at the Workshop on Imaging Trackers and Autonomous Acquisition Applications for Missile Guidance, 19-20 November 1979, Redstone Arsenal, Alabama.

APPLICATION OF IMAGING SENSOR TECHNOLOGY TO PRECISION CRUISE MISSILE GUIDANCE*

Viggh, M.E., Ormsby, C.C. and Edge, E.R.
The Analytic Sciences Corporation, Reading, MA 01867

ABSTRACT

The development of imaging sensors for use in precision cruise missile guidance must take into account a wide variety of different factors including vehicle constraints, mission scenarios, scene matching considerations (e.g., signature requirements and predictability) and the state of sensor technology. The Autonomous Terminal Homing (ATH) Program has recently selected two sensor concepts for brassboard development and flight testing. This paper will review the sensor options considered prior to sensor selection and the methodology used for concept development/comparison. In addition, the advantages and disadvantages of the most promising candidates will be outlined and a summary of the selected design principles presented.

1. MISSION SCENARIOS

The Defense Advanced Research Projects Agency (DARPA) is currently funding the development of a second generation cruise missile guidance system as part of the Autonomous Terminal Homing Program (ATHP). The performance goals for this system include:

- Sufficient precision for effective nonnuclear strike
- Autonomous operation from launch
- Night and adverse weather operation.

In addition to these primary goals, several growth capabilities are being contemplated including Bomb Damage Assessment (BDA), Terrain Following and Obstable Avoidance, as well as Doppler Navigation.

The primary penetration aid will be stealth. This implies low altitude flight and small radar cross-section, as well as emitting a mini-

*This work was supported by the Defense Advanced Research Projects Agency under Contract No. DAAK40-78-C-0032.

mum of readily detectable radiation. Coarse position updates will be provided during midcourse flight by TERCOM. As the target is approached, more accurate updates will be provided by area correlation between sensed images and stored reference data for selected scenes along the flight path. It is critically important that these sensed images be obtained without radiating signals which significantly increase the risk of detection.

During the terminal phase of the mission, precision guidance is the most important factor. Imaging the area surrounding the target should provide the highest potential for delivery accuracy, while also offering the potential for increasing force effectiveness through the use of a BDA capability.

It cannot be assumed that reliable predictions of the weather enroute or near the target are available at the launch site either for purposes of making launch decisions or for use in reference preparation (i.e., predicting weather dependent scene signatures). Furthermore, even if that information could be obtained, strike requirements would not generally allow delay of launch until favorable weather develops and current capabilities in signature prediction are not adequate to justify the inclusion of weather dependent signature characteristics. For these reasons, sensors must be designed to ensure adequate performance under unknown, adverse weather conditions.

2. SYSTEM CONTEXT AND OPERATING MODES

An overview of the proposed weapon system configuration is shown in Fig. 2-1. The delivery vehicle is assumed to be a low altitude, subsonic cruise missile, equipped with a TERCOM-aided inertial system for mid-course navigation. The function of the imaging sensor is to provide images of the target area, or of an intermediate offset aimpoint, which are compared to pre-stored reference images by the scenematching algorithm to produce precision guidance updates. This information must be provided at a range which allows sufficient time to correct residual cross-track error caused by the limited accuracy of the midcourse navigation system. There is thus a direct system design trade between:

- Magnitude of cross-track errors
- Missile maneuverability
- Imaging sensor operating range.

Furthermore, the accuracy of the imaging sensor must be sufficiently high to ensure target destruction by a non-nuclear warhead. The most reliable approach towards meeting this objective is to generate images of the target area, since any uncertainty in the location of an off-set

aimpoint relative to the target ("mapping error") will contribute to miss distance.

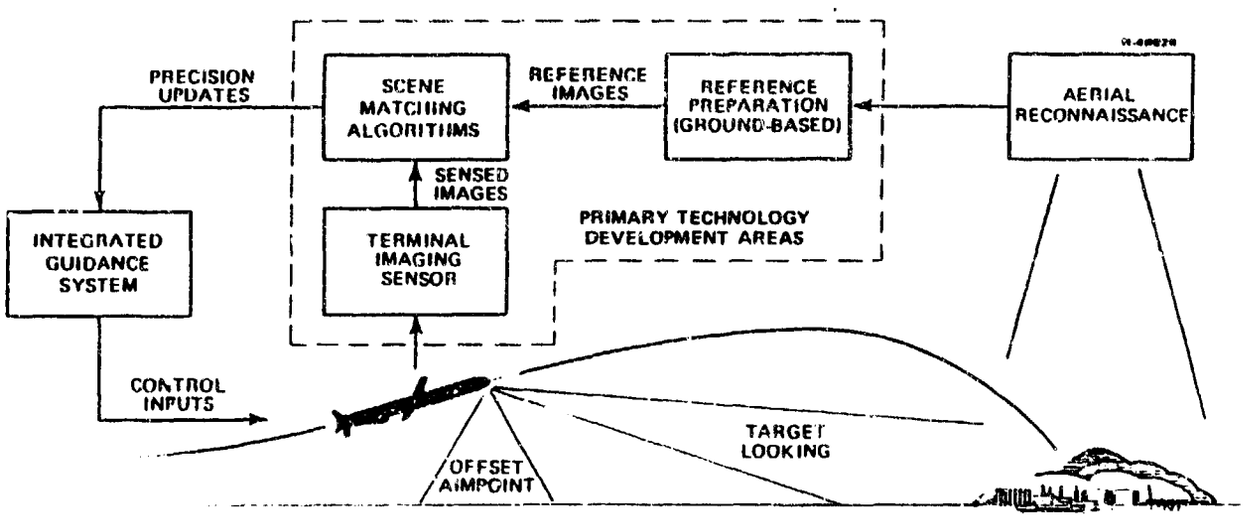


Figure 2.1 Overview of ATH System Context

A sensor designed exclusively for targetlooking must obtain the first image of the target at a range which permits correction of cross-track errors. Crosstrack maneuvers are accomplished by generating aerodynamic forces which are perpendicular to the missile velocity vector. The achievable trajectory curvature (given by a_{max}/V^2, where a_{max} is the maximum acceleration and V is the velocity) can be very small for high velocity cruise missiles, which have relatively low acceleration capabilities. The situation is further aggravated by the fact that cruise missiles use a roll-to-steer configuration. In order to maneuver, the missile must first roll to obtain a component of its lift force in the desired crosstrack direction. However, the missile is designed for minimum radar cross-section to reduce the probability of detection. This requirement leads to lifting body configurations with small elevon surfaces and severe adverse roll-yaw coupling effects. In order to control these adverse aerodynamic effects, the combined airframe-autopilot roll response may be very sluggish and the maximum roll rates low. The end result of these maneuverability limitations is to place a severe penalty on targetlooking only sensors which do not have sufficient range capability in adverse weather.

If the first image were obtained with a sidelooking or downlooking sensor, there is no direct relation between distance to the target and imaging range. Since mapping errors generally are small relative to the midcourse navigation system uncertainty, most of the cross-track error can be corrected on the basis of correlation against an offset aimpoint. A downlooking or sidelooking sensor could thus provide most of the corrections needed, with a minimum requirement on imaging range and with that range being independent of vehicle characteristics.

The imaging range is an important parameter. Virtually every type of sensor that can be used for ATH suffers from some degree of sensitivity to adverse weather. A shorter imaging range thus always results in a higher probability of mission success for a given sensor. Furthermore, increasing the range of an active sensor usually implies increased radiated power and thus higher probability of detection. A dual mode imaging sensor combines the advantages of targetlooking and down/sidelooking sensors, for best possible performance.

3. INITIAL SENSOR SELECTION

The selection of sensor concepts for ATH was performed in two phases. In the first phase, a number of potentially applicable sensor types were identified and subjected to a preliminary evaluation. Those found to be most promising for the ATH application were studied in more detail during the second phase, after which a final selection was made.

Ranges for major performance parameters were established, based on the mission scenario and desirable operational modes outlined in Sections 1 and 2. The most critical aspects of sensor performance are:

- Adequate range in adverse weather
- Resolution consistent with accuracy requirements
- Sufficient number of resolution cells in each image to permit correlation
- Frame times allowing position updates to be made at desired intervals.

The forwardlooking mode is the most demanding, particularly in terms of range and resolution. In addition, adapting a single sensor for both downlooking and targetlooking operation presents considerable difficulties for many sensor types.

Table 3-1 lists the generic sensor candidates which were initially considered for evaluation. The table also summarizes major advantages and disadvantages which were identified during the early phases of this effort. Some of these inherent disadvantages caused elimination of several candidates from further consideration. For example, millimeter (mm) wave radiometers were removed from the list of potential alternatives because of inadequate sensitivity. Even with minimum requirements on resolution, number of cells per image and frame time, the available integration time per cell is about an order of magnitude shorter than that needed for acceptable sensitivity.

With the exception of Synthetic Aperture Radar (SAR), the angular resolution of radar sensors is determined by antenna aperture size, measured in wavelengths. Available space and required resolution dictate use of 94 GHz or higher frequencies for such sensors. With the present state of the art, this precludes use of electronic scanning, e.g., phased array antennas. Furthermore, it was established that mm wave reflector or lens antennas large enough to provide desired resolution could not readily be mechanically scanned along a raster pattern at rates needed to obtain acceptable frame times. This limits the types of mm wave radars under consideration to those using range-azimuth scanning, employing rotating fan beam antennas.

TABLE 3-1

SENSOR CANDIDATES CONSIDERED

SENSOR TYPE	ADVANTAGES	DISADVANTAGES
MM Wave Radiometer	Passive System Low Technical Risk	Marginal Resolution and Sensitivity No Ranging Capability
MM Wave Radar	Ranging Capability Low Weather Sensitivity Low Technical Risk	Marginal Angular Resolution High Detectability
Synthetic Aperture Radar	All Weather Capability High Resolution	Complex Processing No Forwardlooking Capability
Passive Visible-Light Imager	Passive System High Resolution Low Technical Risk	Weather Sensitivity No Night Operating Capability No Ranging Capability
Passive Mid-Far IR Imager	Passive System High Resolution Significant Weather Capability	Moderate Technical Risk No Ranging Capability Susceptible to Dense Fog
Active Near IR Imager	Ranging Capability Low Detectability High Resolution	Moderate to High Weather Sensitivity
Active Coherent Far IR Imager	Ranging Capability Low Detectability High Resolution Low to Moderate Weather Sensitivity	Moderate to High Technical Risk

SAR also remained among the sensor candidates to be evaluated further in the second phase.

To achieve a raster scan with required resolution and frame time, it proved necessary to consider wavelengths significantly shorter than 3 mm. However, the wavelength region from about 15 μm to 1 mm is unsuitable for imaging because of atmospheric absorption. The infrared portion of the spectrum, from 1 μm to 15 μm, is thus the most viable alternative if raster scan imaging is to be performed.

Initial evaluation of the various infrared sensors listed in Table 3-1 gave strong indications that active systems operating in the near infrared region (0.9 to 1.1 μm wavelength) would not have the range capability required for the targetlooking mode, except under favorable weather conditions. On the other hand, both Nd:YAG (1.06 μm) and GaAs (0.9 μm) laser imagers should perform reasonably well in the downlooking mode, even in adverse weather. The option of using a GaAs laser system for downlooking only was retained as a back-up, in the eventuality that no satisfactory solution for an active dual mode sensor would emerge. Following the initial evaluation, the following candidates remained

- Two radar options, mm wave range-azimuth radar and SAR
- Passive infrared imagers, operating in either of two bands: 3 to 5 μm or 8 to 14 μm
- Active infrared imager using a CO_2 laser (10.6 μm wavelength)
- Active infrared imager using a GaAs laser (0.9 μm wavelength) for downlooking operation only (back-up alternative).

Further evaluation of these options was undertaken, including the development of several point designs to establish sensor feasibility and obtain detailed performance predictions (e.g., operating ranges, field of view, frame times and resolution).

4. FINAL SENSOR SELECTION

The initial evaluation and selection allowed judgements and decisions to be made on the basis of major incompatibilities with relatively hard requirements. As the selection process continued, an increasing number of mission related factors had to be considered. A full description of the final selection process is not possible within the framework of this paper, but the following summary attempts to relate those criteria and factors which were most important in arriving at the final recommendations.

4.1 RADAR SENSORS

A point design of a range-azimuth, 94 GHz radar was generated to determine feasibility and to estimate possible performance capabilities. It proved impractical to design this type of sensor for both downlooking and targetlooking operation. Position updates could be based on sidelooking and forwardlooking images, but reliable prediction of intensity signatures versus range is difficult for mm waves. Other undesirable features of this sensor alternative include

- Marginal azimuth resolution even with largest possible antenna aperture
- High radar cross-section
- High detectability.

Increasing the frequency to 140 GHz (next higher frequency band with relatively low atmospheric attenuation, see Fig. 3-1) would allow the use of a smaller antenna for the same resolution or better resolution with the same antenna size. However, with current transmitter and receiver technology there is a significant power budget penalty for increasing the frequency above 100 GHz. In fact, even if the aperture size is held constant, the maximum imaging range would most likely be less at 140 GHz than at 94 GHz. While a mm wave sensor was not chosen for further development as part of ATHP, it is believed that additional research could effectively reduce many of the identified shortcomings of mm wave sensors and that a 3rd generation mm wave guidance system with greater weather penetration capability may be possible.

Synthetic Aperture Radar offers high resolution with relatively small physical antenna aperture; typically on the order of 10 wavelengths. It would thus be possible to use a frequency somewhere in the 10 to 40 GHz range without violating space constraints, thereby avoiding most of the adverse weather restrictions occurring at higher frequencies. However, SAR cannot be employed for imaging along the line of flight, which poses a serious problem for the targetlooking mode. After evaluation of several approaches, including off-set aimpoints and various terminal phase maneuvers (which would allow imaging of the target area,) it was concluded that the disadvantages of these approaches largely outweigh the advantages of the SAR sensor. Since SAR also requires extensive and costly signal processing and scene signatures in complex cultural areas are difficult to predict, this alternative was not among those finally selected.

4.2 PASSIVE INFRARED SENSORS

The initial evaluation of passive IR sensors indicated a preference for either of the "atmospheric windows" 3 to 5 μm or 8 to 14 μm, as opposed to shorter IR or visible wavelengths. During the second evaluation phase,

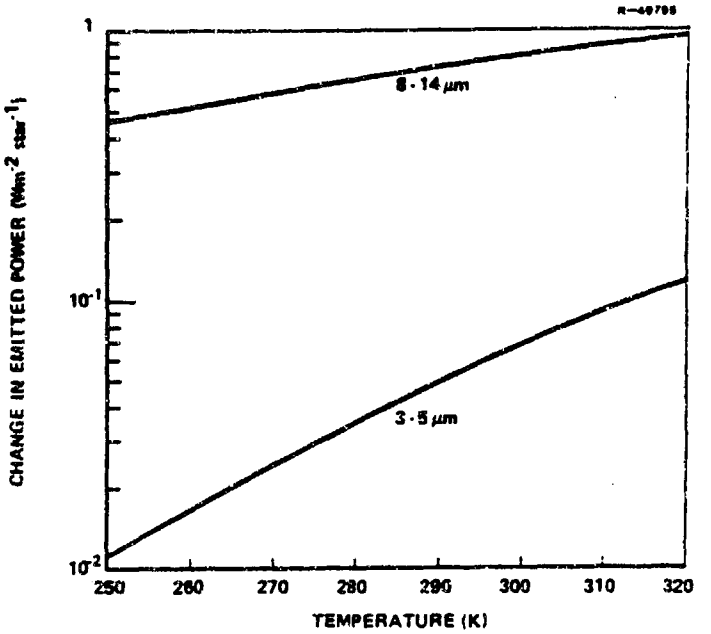


Figure 4-1 Change in Emitted Power Caused by a Temperature Increase of 1K

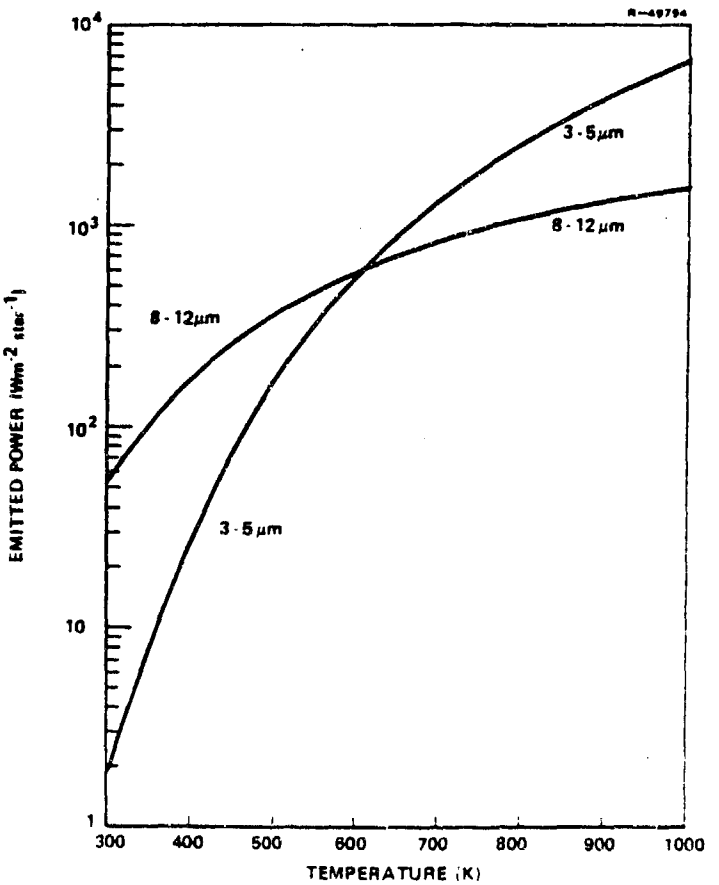


Figure 4-2 Emitted Power as a Function of Temperature

a choice between these two alternatives had to be made, based on considerations such as

- Scene quality, i.e., the information content in the imaged area
- Propagation in adverse weather, haze and smoke
- Availability of hardware, cost and development risk.

The physical temperature of most objects within imaged scenes can be expected to fall in the range from 250 K to 320 K, depending on season, time of day, solar irradiation and other climatic factors. As shown in Fig. 4-1, the change in radiated power per unit area, for a temperature change of 1 K within this temperature range, is considerably larger in the 8 to 15 μm band than for 3 to 5 μm. If the objective were to detect isolated, hot objects against a relatively cold background, the 3 to 5 μm band would be more suitable, see Fig. 4-2. However, for area correlation it is more important to reliably resolve small differences in temperature and emissivity throughout the imaged scene than to locate a few isolated, warm objects.

Attenuation caused by precipitation (rain, snow or hail) and aerosols (fog, haze or smoke) depends primarily on drop or particle size relative to the wavelength. Falling raindrops, snowflakes or hailstones are typically larger than 15 μm in size, which results in approximately equal attenuation for the two wavelength bands considered.

Fog may contain drops which range in diameter from less than 1 μm to more than 100 μm. In fogs where most of the water is contained in drops smaller than about 5 μm (e.g., radiation fog in the formative stage), the attenuation is significantly lower in the 8 to 14 μm band than for 3 to 5 μm radiation. On the other hand, in stabilized advection fogs, a large percentage of the water may form drops with diameters larger than 10 μm, in which case the attenuation is virtually independent of wavelength (λ) for $\lambda \leq 14$ μm.

Haze is normally dominated by particles which are smaller than 10 μm in size. The same is true for most types of smoke, particularly those commonly used for obscuration or blinding on the battlefield ("smoke screens"). For both haze and smoke, radiation within the 8 to 14 μm wavelength band will thus tend to be attenuated less than radiation in the 3 to 5 μm region.

Hardware availability does not appear to be a significant factor in chosing between the 3 to 5 μm and 8 to 14 μm bands for passive imaging. Neither do development risk and cost seem to be major considerations, even if the 3 to 5 μm technology is more mature. The choice of 8 to 14 μm was thus based primarily on higher scene quality and lower atmospheric attentuation under certain adverse weather conditions.

Another important design feature is scanning format, which in turn is closely related to detector configuration. The two alternatives initially considered for the downlooking mode are illustrated by Figs. 4-3 and 4-4. The "pushbroom" arrangement shown in Fig. 4-3 employs a linear array of detectors for cross-track coverage, while the line scanner in Fig. 4-4 uses a single detector and mechanical cross-track scanning. In both cases, down-track scan is accomplished by vehicle motion.

Either of the two configurations shown in Figs. 4-3 and 4-4 could be converted to targetlookers by introducing a galvanometer mirror to "fold" the optical path in the forward direction and provide elevation scanning. However, the required angular scan range in azimuth during targetlooking is considerably smaller than the desired angular cross-track coverage in the downlooking mode. Some form of angular "scan expander" will thus be needed for downlooking operation.

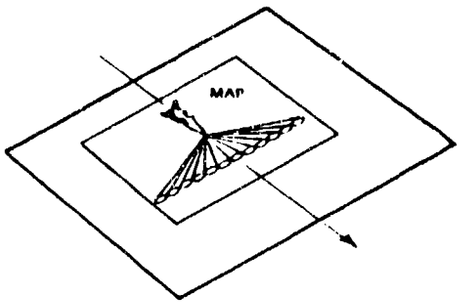


Figure 4-3 "Push-Broom" Scan, Using Linear Detector Array

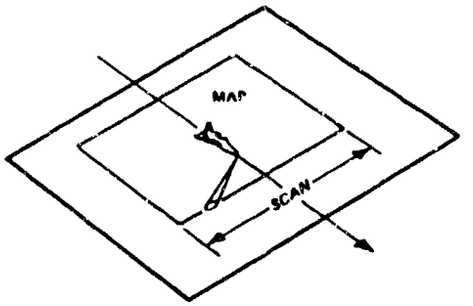


Figure 4-4 Line Scan, Employing Single Detector

Sensitivity is primarily determined by integration time per resolution cell. A linear array offers obvious advantages in this respect, over a single detector. For a given integration time per cell, the frame time is reduced by a factor at least equal to the number of detectors in the array. Conversely, for a certain frame time, the integration time per cell can be increased by the same factor. However, separate amplifiers are needed for each detector and unless all detector/ amplifier combinations have identical characteristics, the sensed image will contain pattern noise. This is avoided if the same detector/ amplifier assembly is used for all cells, but at the expense of shorter integration time and thus lower signal-to-noise ratio per cell.

A third option is a compromise between the two alternatives illustrated in Figs. 4-3 and 4-4. An array containing a relatively small number of detectors could be used in a configuration similar to that shown in Fig. 4-4. For each cross-track (or azimuth) scan, each detector would generate one scan line. This provides for sufficient integration time per cell to obtain the desired sensitivity, acceptable frame time and a manageable number of detector/amplifier combinations for which sensitivity and gain equalization must be performed.

4.3 ACTIVE INFRARED SENSORS

As mentioned in Section 3, active sensors employing GaAs (0.9 μm) or Nd:YAG (1.06 μm) lasers were ruled out for targetlooking operation because of high attenuation in adverse weather and smoke. The CO_2 laser provides a source for high power radiation which more readily penetrates haze, most types of smoke and some fogs (see Section 4.2). This section reviews the feasibility of designing a dual mode, active sensor operating at a wavelength near 10.6 μm.

An active imaging sensor using a CO_2 laser as a transmitter can be designed for either direct or coherent detection. In the latter case, the local oscillator signal may be derived from the transmitter (homodyne), if the Doppler shift ($\sim$200 kHz per m sec^{-1} velocity) is sufficiently high for the intermediate frequency to fall above the 1/f -noise "knee" of the amplifier. Coherent detection provides high sensitivity (equivalent noise figure typically <20 dB) and effective suppression of background radiation, both of which prove to be essential for meeting the range requirements for the targetlooking mode.

In the passive case, discussed in Section 4.2, several detectors were used to provide sufficient integration time per resolution cell, while maintaining a short frame time. A similar approach for the active case would require that the transmitted power be spread over several resolution cells, which reduces the received power per detector by the same factor that the noise is reduced through longer integration time.

Another factor which must be taken into account is that the maximum effective collecting aperture is approximately the same as the transmitter aperture when coherent detection is used. Thus, if the aperture is made larger to collect more reflected power, the diffraction-limited beamwidth will be reduced. For given values of frame time and scanned field, the dwell time per resolution cell then decreases by the same factor as the power increases. The signal-to-noise ratio (SNR) stays constant, but resolution is improved.

For any reasonable aperture diameter, the diffraction-limited resolution is considerably higher than needed in the sensed image. At the same time, the SNR obtained with available transmitter power would be inadequate at the desired maximum range if the potential resolution were

fully realized*. To achieve a better balance between SNR and resolution, one can apply either or both of the following techniques

- Integrate over several diffraction-limited resolution cells to generate each "pixel" in the sensed image
- Space consecutive scan lines further apart than one beamwidth.

A combination of these approaches can be selected to provide for a nominally square grid in the sensed image, as well as integration of several independent samples to reduce the effects of fading.

The factor which ultimately limits the aperture diameter is the finite round-trip time required for laser energy to propagate to the scene and back to the sensor. If the beam is scanned a significant portion of one beamwidth during that time, some of the reflected energy will not reach the detector. This lag angle effect can be compensated for, but only within a limited range interval.

The modulation waveform must be device compatible and also provide the desired range accuracy. Either frequency or amplitude modulation could be used, but the latter appears to present the lowest development risk. One time-proven approach, employed in optical surveying instruments for decades, would be to use sinusoidal or square-wave amplitude modulation, implemented by means of a modulator located outside the laser cavity (to maintain a continuous local oscillator signal for the mixer). Range is determined by measuring the phase shift between transmitted and received modulation envelopes. To obtain desired accuracy it may become necessary to use a high modulation frequency, which can cause range ambiguities. These could be resolved by alternating between two different modulation frequencies.

5. SUMMARY

This paper has presented the sensor requirements for DARPA's Autonomous Terminal Homing Program and provided a summary of the advantages, disadvantages and design options of both radar and electroptical sensors. A passive, 8 to 14 μm dual mode imager and an active 10.6 μm coherent dual mode imager were identified as the most promising design option for meeting both accuracy and adverse weather/night requirements.

*Due to frame time limitations and the resultant limitation on dwell time per pixel for a high resolution system.

Paper No. IIIA-2, Presented at the Workshop on Imaging Trackers and Autonomous Acquisition Applications for Missile Guidance, 19-20 November 1979, Redstone Arsenal, Alabama.

EVALUATION OF SCENE MATCHING CONCEPTS FOR AUTONOMOUS ACQUISITION TERMINAL HOMING*

M. Svedlow
The Analytic Sciences Corporation
Reading, Massachusetts 01867

ABSTRACT

Advanced scene matching concepts for application to autonomous terminal homing have been investigated as part of the Autonomous Terminal Homing Program. ATH program objectives are to develop a precision terminal guidance system for fixed targets capable of operating successfully during both day/night and adverse weather conditions using synthetically generated references. Scene matching issues and a conceptual framework to address these issues are presented. Functional comparisons between different processing components are summarized, and suggested approaches to the development of a robust scene matching processor are presented.

INTRODUCTION

The Autonomous Terminal Homing Program is a multiphase program, one aspect of which is the development of advanced scene matching concepts for high accuracy, autonomous guidance during the terminal phase of flight. A sensed image collected during flight is compared with a synthetically generated reference image (or data set) of predicted scene signatures (prepared prior to the mission) to estimate vehicle position. This position estimate is used to update the inertial navigation system (INS) aboard the vehicle. It is expected that there will be several match updates during the terminal phase of flight. Initial updates (at distances far from the target) will be in a downlooking mode where the sensor scans the ground directly below the vehicle. As the vehicle nears the target, the sensor will switch to a targetlooking mode, imaging the target directly.

Imaging conditions and the mission scenario impose certain constraints on scene matching requirements. The system will be directed against fixed targets and will be required to operate during adverse weather and day/night conditions using synthetically generated reference images. The fixed target scenario relieves the requirement for scene matching against targets with unknown orientation since the INS will provide an approximate position estimate relative to the target and an accurate heading estimate. Adverse weather and day/night conditions coupled with the use of synthetic references require an insensitivity to signature prediction uncertainties which occur as part of the reference preparation process.

*This work was supported by the Defense Advanced Research Projects Agency under Contract No. DAAK40-78-C-0032.

The following sections summarize the conclusions of the investigation performed for the ATH program. Included are: issues to be addressed by the match processor, conceptual approaches to scene matching, a comparison of approaches considered during the study, and suggested scene matching approaches for fixed target, day/night, all weather, autonomous terminal homing.

SCENE MATCHING ISSUES

The system context requires that certain issues be addressed in the design of a scene matching algorithm. These are:

- Geometric distortions
- Contrast reversals/intensity mispredictions

Relative geometric distortions between the reference and sensed images are a result of the uncertainty in the vehicle's position relative to the target at the time the sensed image is collected. The INS will provide an accurate estimate of vehicle heading, so that angular scanning parameters of the sensor can be accurately estimated. However, only an approximate estimate of vehicle position will be provided, resulting in some uncertainty in viewing aspect. Generation of reference images from aspects differing from that at the time of imaging will yield relative perspective distortions between the images.

Contrast reversals/intensity mispredictions are characteristics of changes in scene signatures and the predictive process which is part of reference preparation. Relative intensity levels of different surfaces within a scene may change with time of day or environmental conditions resulting in contrast reversals between the different surfaces. This is particularly evident for a passive thermal band sensor (e.g., roofs of buildings may be warmer than the surrounding ground during the day, whereas at night the roofs may be cooler). Since reference preparation is a predictive process, it is possible that mispredictions of the intensity levels of surfaces in the scene will occur. Match algorithms should be designed to be insensitive to these image characteristics.

CONCEPTUAL APPROACHES

There are two fundamental approaches to the scene matching process:

- Correlation processors
- Feature matching processors

The correlation processor approach is a variation of the crosscorrelation procedure wherein one image is shifted relative to the other image and a correlation value is computed at each offset. The particular correlation function used may have one of any number of forms (e.g., correlation coefficient, mean square error, mean absolute difference). Feature matching

processors extract designated sets of features and their associated descriptors from the images (e.g., lines and their orientations). Matching proceeds using only the extracted features which may be stored in either an image or tabular format.

Both of these approaches are compatible with the conceptual representation of a scene matching processor shown in Fig. 1. This representation does not necessarily indicate the operational flow. The order of processing may differ and any iterative processing which may be performed is not shown. The functional blocks were selected to address key requirements of the match process. A brief description of each of the processing segments is provided in Table 1.

Figure 1. Conceptual Representation of Scene Matching Algorithm

TABLE 1
DESCRIPTION OF PROCESSING SEGMENTS

PROCESSING SEGMENTS	DESCRIPTION
Geometric correction*	Correction for relative perspective distortions between the reference and sensed images via image reconstruction.
Intensity processing	Compensation for contrast reversals/intensity mispredictions
Geometric compensation*	Compensation for residual geometric distortions
Match function	Similarity measure used for matching the images
Match location	Provision for subpixel estimation and an indication of the match quality (i.e., a match discriminant)
Guidance update	Mechanism for interfacing with and providing the update information to the INS

*Corrective measures actually correct for predicted distortions (e.g., generating reference images from hypothesized sensor positions for matching), whereas compensatory measures are used to desensitize the match processor to distortions (e.g., resolution reduction by filtering)

FUNCTIONAL COMPARISONS

Several organizations participated in the development of scene matching concepts (Refs. 1 to 8) for the ATH program. A functional comparison of the different approaches to each of the processing segments is presented in Table 2. This table summarizes the approaches used to correct and compensate for geometric distortions, compensate for contrast reversals/intensity mispredictions, and estimate the match position between reference and sensed images. An evaluation of the applicability and effectiveness of these approaches is indicated.

TABLE 2

FUNCTIONAL COMPARISON OF PROCESSING SEGMENTS

MATCH FUNCTIONAL COMPARISON		PREFERRED	ACCEPTABLE	QUESTIONABLE
Geometric Correction		• Deterministic Correction - Projection - Search		• Estimated Correction • None
Geometric Compensations*		• Filtering • Window Application • Match Tolerances†		• Multiple Subareas
Intensity Processing	Intensity	• Threshold |GRAD| • Line Extraction	• |GRAD| with Adaptive Normalization	• |GRAD| • None
	Range	• Height Conversion • Slant Range • Line Extraction†		• |GRAD|
Match Function/ Procedure		• Normalized Correlation • Line Match	• Phase Correlation	• Endpoint Match • Unnormalized Correlation

*Used with geometric correction

†Feature matching algorithm

Geometric Correction - Geometric processing is divided into two parts: corrective and compensatory measures. The suggested corrective processing approach is to use deterministic correction. This method uses all of the geometric information available (i.e., three dimensional reference model, hypothesized sensor positions, sensor angular pointing information and sensor range data (if available)).

With an active ranging sensor a single projection of the sensed image is generated for matching (Ref. 1). For this method both the reference and sensed images are projected onto a plane perpendicular to the nominal line-of-sight (projection with viewpoint at infinity (see Fig. 2)). This type of projection is possible with an active sensor since both range data and angular information are available. The advantage of this approach is that it provides a single-pass match procedure (minimizing computation time).

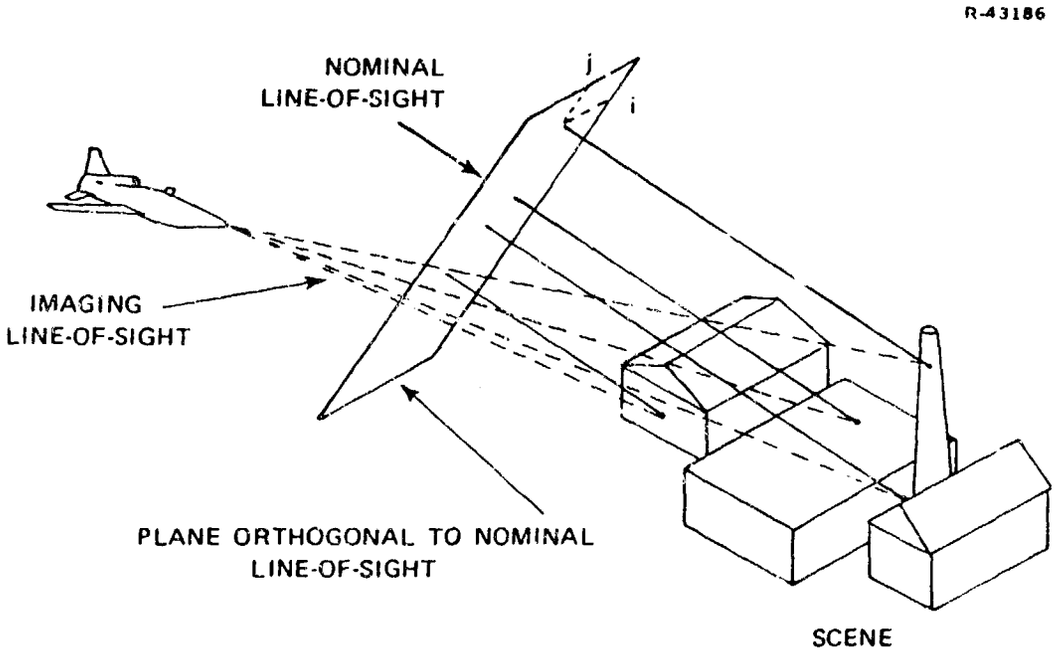


Figure 2 Sensed Image Projection to Plane Perpendicular to Nominal Line-of-Sight

With a passive sensor range data is not available* (precluding use of the single projection technique). Multiple reference images are reconstructed from a set of hypothesized sensor positions from different perspectives within the search basket using the three-dimensional scene model and a search is performed to determine the match update.

Estimated correction as a self contained approach is not suggested as a preferred approach. In the estimated correction approach proposed in Ref. 2 the distortions between the reference and sensed images are estimated using only the image data. No a priori three dimensional information is

*However, a technique for generating a range image using sequential passive sensor images has been proposed (Ref. 9).

used. However, it is possible that this procedure could be used as a supplemental processing segment to estimate range to the target for a passive sensor. Estimates of azimuth, elevation and range offsets are required for a position estimate. Match processors inherently provide estimates of the angular offsets for both active and passive sensors. With an active ranging sensor, range to the target is provided by the sensor range data. However, no range data is available for a passive sensor. Estimated correction could be used to estimate the relative scale difference between the reference and sensed images, and thus estimate the range to the target.

Geometric Compensation - This type of processing does not employ distortion correction procedures, but rather desensitizes the match processor to the distortions which are present. Geometric compensation is a procedure which may be used with geometric correction to desensitize the processor to residual distortions. Suggested approaches for correlation processors are filtering and application of a window function (Refs. 1 and 5). (Filtering is a convolution performed in the spatial domain and the window function is equivalent to convolution in the frequency domain.) Both methods reduce the effects of distortion on the match processor. For feature matching processors, tolerances between matching features can be specified to compensate for residual geometric distortions. Designated as questionable is a multiple subarea approach which does not use a priori three-dimensional information in the match process (Ref. 8). This approach assumes small distortions within the respective subareas (which may not be the case, particularly with low altitude downlooking sensed images) and a first order polynomial model for the geometric distortions between subareas (which is not necessarily sufficient for the perspective distortions that occur). However, the notion of a multiple subarea approach could be used as a supplemental processing segment to provide an estimate of range for a passive sensor (via estimating the relative scale difference between images).

Intensity Processing for Intensity Signatures - The objective of preprocessing intensity signature images is to compensate for contrast reversals and intensity mispredictions which occur as part of the reference image prediction process. Edge magnitude enhancement procedures address the contrast reversal problem by retaining those structures (edge magnitudes) that tend to be the most predictable. Thresholding (or adaptive normalization) of these edge magnitudes then compensates for the misprediction of intensity magnitudes. Suggested approaches for intensity processing are a thresholded magnitude of the gradient (edge enhancement) or line extraction scheme since both of these address the contrast reversal and intensity misprediction issues. Questionable approaches are no intensity processing and edge magnitude enhancement techniques without thresholding or normalization (which do not compensate for intensity mispredictions).

Intensity Processing for Range Signatures - Processing for range images differs from that of intensity images since range is a predictable signature. Suggested approaches for range processing depend on the type of algorithm and match processor used. Suggested range processing methods are conversion of slant range to height, retention of the slant range signature, and the extraction of lines (which is applicable to feature matching algorithms). The magnitude of the gradient as a sole processing approach

is questionable. Since the |grad| alone accentuates only far edges of scene structures and eliminates useable match signature information contained in the spatial direction of the range gradient.

Match Function/Procedure - The match function and procedure used are dependent on the type of match algorithm. Suggested are normalized correlation* (for a correlation processor) and line matching (for feature matching algorithms (Refs. 4 and 8)). Endpoint matching (Ref. 6) is not a suggested approach since it depends on a potentially noisy signature.

Additional Operational Issues - In addition to the corrective measures required for geometric distortion and contrast reversal/intensity misprediction, other operational issues should be addressed in the design of a high accuracy, missile compatible, closed-loop guidance match processor. These issues are listed below.

- Reference preparation requirements (intensity, surface shell, wire frame)
- Reference data storage and handling/image selection procedure
- Guidance update generation technique
- Pre-mission estimation of match performance and real-time evaluation of fix reliability/accuracy
- Computational/processor requirements

SUGGESTED ALGORITHM APPROACHES

Suggested approaches to algorithm development for both correlation processor and feature matching concepts are presented in Figs. 3 and 4. Both include processing for geometric correction and contrast reversal/intensity misprediction compensation. The additional operational issues outlined in the previous section are also important. Several of these are indicated in the figures at the appropriate functional positions.

Correlation Algorithm (Fig. 3) - The single projection and multiple reference search techniques are suggested for the geometric correction procedure with a threshold (or equivalently normalized) edge magnitude enhancement procedure for intensity signature preprocessing. Any required filter/window compensation is also suggested.

Suggested approaches for the match function are normalized correlation (achieved via image spatial normalization or normalized match func-

*Normalization may be equivalently achieved at the intensity processing stage via thresholding operators and adaptive normalization, rather than its explicit use in computing the match function.

tion) with a bounded penalty function for range data (to avoid overweighting spurious large range data errors). Also included are subpixel estimation, premission performance estimation (probability of false fix (P_{ff}) or equivalent indicator), real-time fix reliability/accuracy estimation and the guidance update procedure.

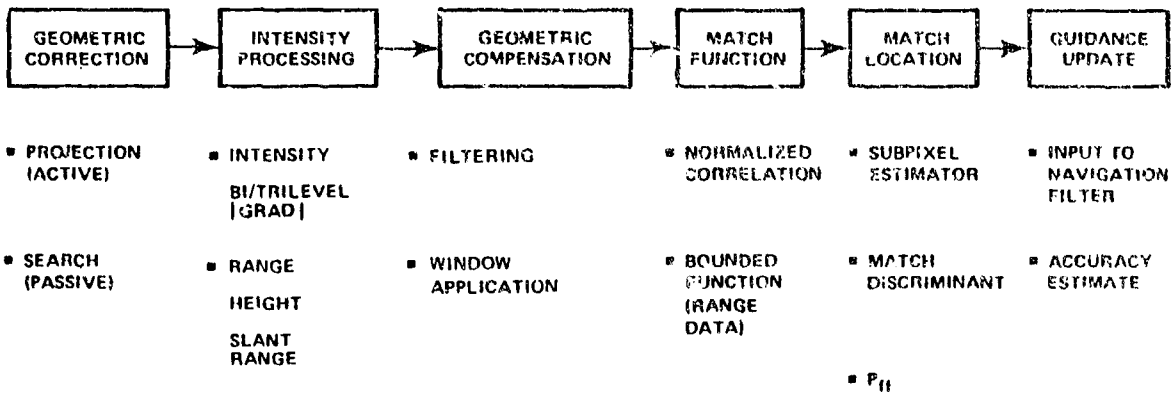


Figure 3 Correlation Algorithm Approach

Feature Matching Algorithm (Fig. 4) - Suggested geometric correction is the same as for correlation processors. Feature extraction provides compensation for contrast reversals and intensity mispredictions. Geometric compensation for residual distortions can be addressed by allowing tolerances between matching features.

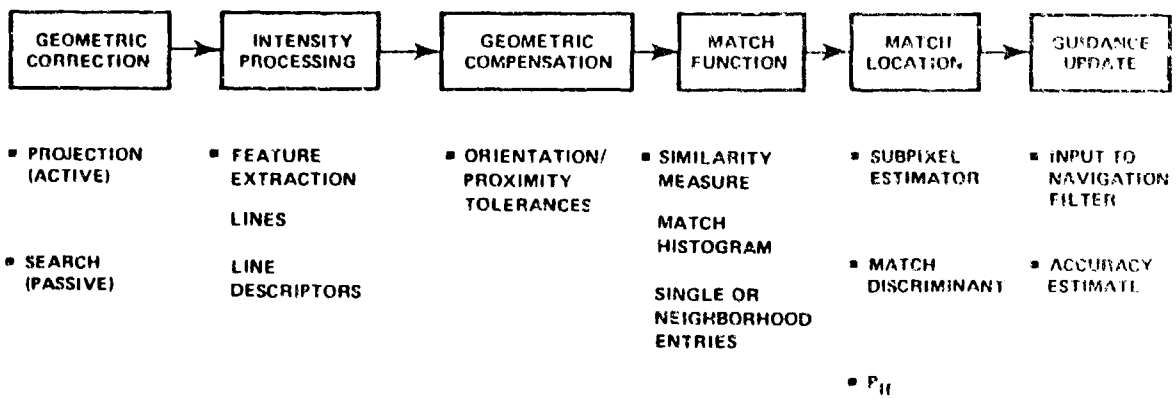


Figure 4 Feature Matching Algorithm Approach

Since matching is performed on extracted features, the features can be matched by using only the feature list tables (a match histogram approach), or by shifting one image relative to the other and computing a match value at each offset (similar to the correlation algorithm procedure). With the match histogram approach, the match value between two features is accumulated in a histogram at a position corresponding to the offset between the features. To accomodate tolerances for geometric distortion compensation, the match value can be entered at all points within a neighborhood

of the offset (which will tend to smooth the resulting match histogram). Both the correlation approach and match histogram approach are conceptually equivalent and differ only in the computational procedure. The particular selection will depend upon the match scheme. The additional functional requirements shown in Fig. 4 are the same as for the correlation processor, including subpixel estimation, pre-mission performance estimation, fix reliability/accuracy estimation and guidance update.

SUMMARY

Concepts have been presented which were important in the development of scene matching processors within the context of the ATH program. Specifically, operating conditions include day/night, adverse weather conditions with a fixed target objective using synthetically generated references. Each of the primary issues has been outlined and suggested processing approaches have been presented.

REFERENCES

1. Berry, J.E., Yoo, J.K., "Preprocessing for Advanced Image Matching Techniques", NAECON, May 17, 1979, 79CH 1449-8, p. 86-91.

2. "Address Modification Image Technology Program: Interim Final Report", Honeywell Electro-Optics Center, March 1979.

3. "Phase Correlation for Terminal Homing", Lockheed Missiles and Space Co., Inc., Report LMSC-D556449, February 1977.

4. "Pattern Matcher Development Study", Rockwell International, C76-1594.22/034A, February 1979.

5. "Optimal Spatial Filters: Final Report", Systems Control, Inc., 30 September 1978.

6. Tisdale, G.E., "A Digital Image Processor for Automatic Target Cueing, Navigation, and Change Detection", SPIE Conference, Reston, Virginia, April 1977.

7. "High Accuracy Cruise Missile Terminal Guidance System", Grumman Aerospace Corporation, 1 December 1978.

8. "Image Sensor Measurements Program", Hughes Aircraft Co., Hughes Research Laboratories, October 1977.

9. "Concept Validation of Depth-Aided Target Acquisition for the Cruise Missile", Northrop Research and Technology Center, NRT C78-42R, November 1978.

SESSION IIIB

IMAGE UNDERSTANDING AND AUTONOMOUS
ACQUISITION FOR MISSILE GUIDANCE

Paper No. IIIB-1, Presented at the Workshop on Imaging Trackers and Autonomous Acquisition Applications for Missile Guidance, 19-20 November 1979, Redstone Arsenal, Alabama.

AN AUTOMATIC IMAGE RECOGNITION SYSTEM FOR MILITARY IMAGE ANALYSIS

Durga P. Panda and Raj K. Aggarwal
Honeywell Systems and Research Center
2600 Ridgway Parkway
Minneapolis, Minnesota 55413

ABSTRACT

Honeywell has been involved in state-of-the-art image analysis research for target screening as well as guidance application under contracts from DARPA, AFAL, NV&EOL and other government agencies. Over the past two years Honeywell has developed a context dependent automatic image recognition system* for analyzing the imagery automatically and detecting tactical as well as strategic targets in the image. The main features of the image recognition system are sequential frame processing, symbolic image segmentation, syntactic recognition, recognition of multi-component objects and conflict removal. In this paper we describe various components of this context dependent automatic image recognition system and information flow between these components.

INTRODUCTION

A general block diagram of the automatic military image recognition system is shown in Figure 1. The image is first segmented and man made object (MMO) is detected in the segmented image by a statistical technique. The output of the MMO detector is processed by secondary screening target detector which further reduces false alarms based upon true size, temperature, etc., of the targets on the ground plane. Sequential frame analysis is used to improve the performance of the target detector. A syntactic recognition scheme uses knowledge of the component description of the targets in recognizing targets that are large enough to show component detail. For images that are too small to show any detail a statistical recognition scheme is used. Sequential frame analysis is employed to take advantage of frame to frame consistency in the imagery to improve the overall performance of the system. The small image statistical classifier and the large image syntactic classifier are combined by a configuration analysis scheme to recognize multiple component structures

*This research was conducted under a joint sponsorship of DARPA (Major Larry Druffel, Image Understanding Program Manager) and AFAL (Mr. Hank Lapp, Thermal Imaging Group).

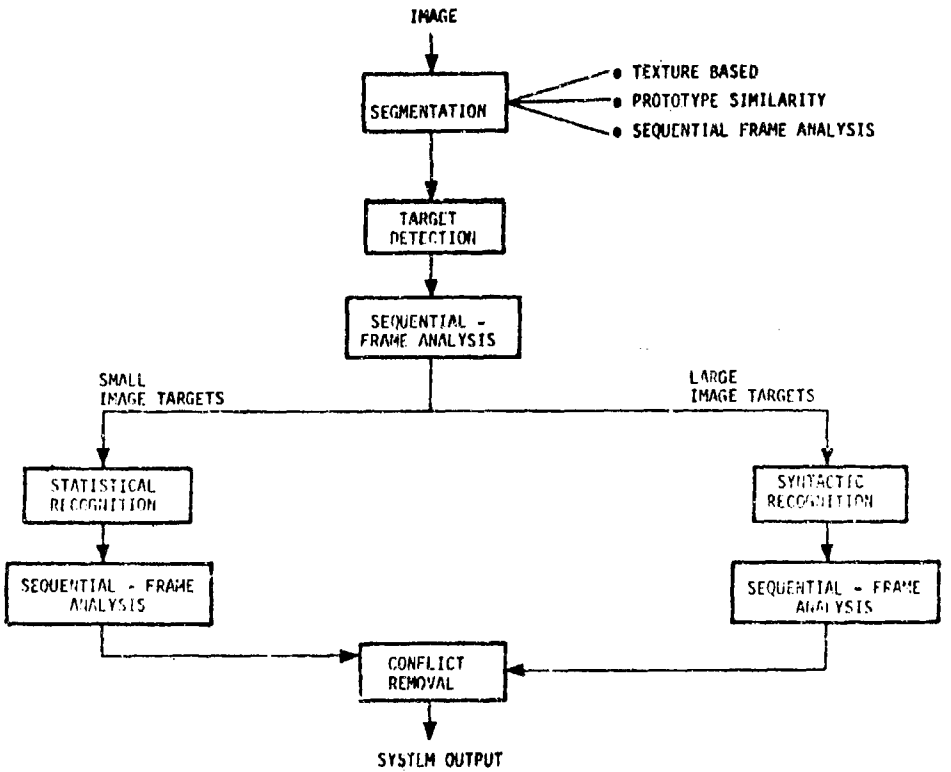


Figure 1. Context Dependent Automatic Military Image Recognition System.

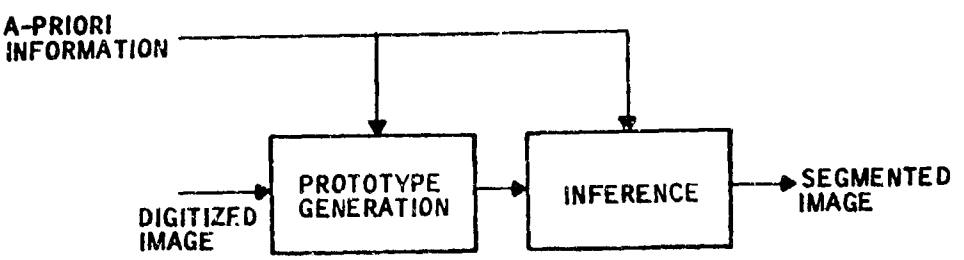


Figure 2. Block Diagram for Prototype Similarity Transformation

such as SAM sites, vehicle convoys, airport and to remove conflicts. The output of the conflict removal function is recognized targets that have tactical importance or are important based on mission analysis. In the following sections we describe the individual components of the system in detail.

SEGMENTATION

Image segmentation is performed by prototype similarity transformation technique [1] which is a method for transforming an image into a set of symbols, each of which represents the relationship of a local region to other parts of the image. A general block diagram of prototype similarity transformation is shown in Figure 2. Generating prototypes is equivalent to finding a maximal set of mutually dissimilar cells. A cell is a pixel or a collection of pixels, depending upon the required resolution in the segmented scene. The generated set of prototypes is used to label each cell in the image. A priori information about the scene is used to guide an inference process to give meaning to each cell in the symbolic image.

Segmentation of individual components of a target can also be done by using the prototype similarity transformation technique. This is done by interactive use of the technique at progressively higher cell resolution as shown in Figure 3.

In the segmentation technique, results of the segmentation of previous frame are used as the starting points of segmentation in the present frame. In prototype similarity transformation in the initial choice of prototypes is the same as the prototypes generated in the previous frame. The advantage of this is that the performance of the segmentation technique approaches the asymptotic value as time proceeds.

SECONDARY SCREENING IN TARGET DETECTION

Secondary screening is a target detection function which is based on the concept that if a segmented object is indeed a target then appropriate features values of the object transformed to the ground plane should match those of the actual target. This concept of matching true object features in ground plane is shown in Figure 4. Implementation of the secondary screener along with a conventional statistical classifier improves the target screener performance by using a priori knowledge about the true target parameters.

The output of segmentation is used to detect and recognize targets such as tanks and trucks. A preliminary screening of non man-made objects (MMO) is first performed on the segmented image by a linear classifier.

The detected objects are further screened based on the true size, temperature or other physical properties. Classification for secondary screening is performed using image features, sensor parameters and physical dimensions of all anticipated targets. The sensor parameters needed are

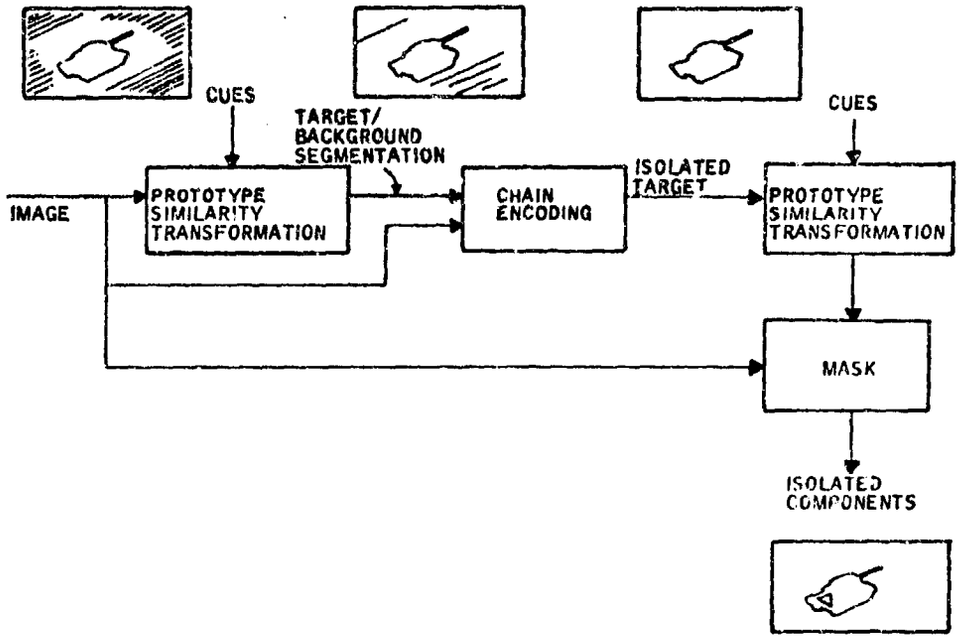


Figure 3. Component Extraction of Tactical Targets Through Iteractive Use of Prototype Similarity.

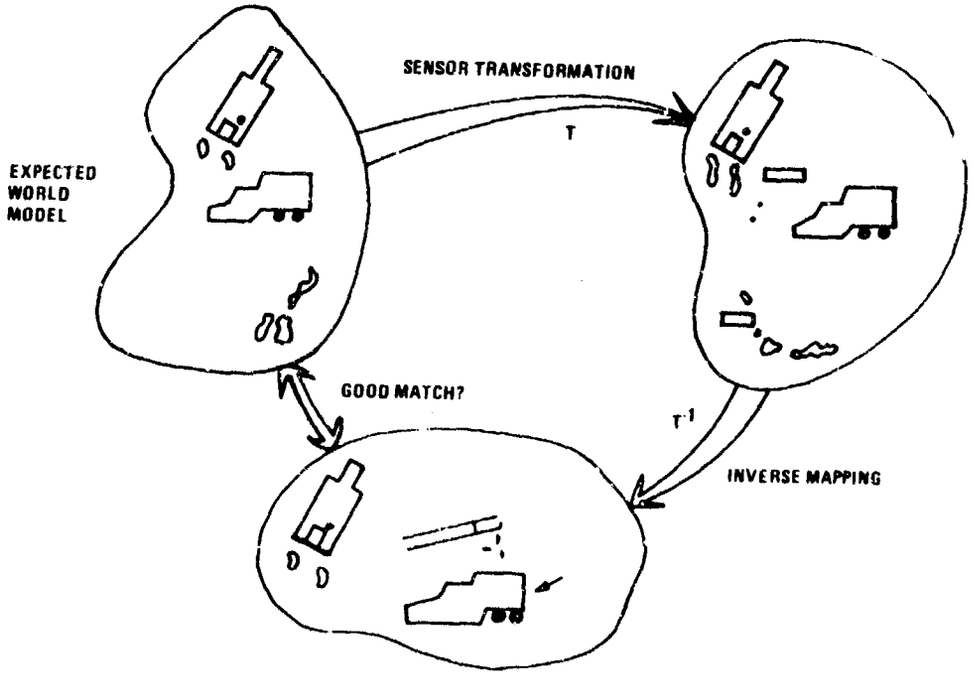


Figure 4. Inverse Mapping Concept in Secondary Screening.

the angular subtense of the Field of View (FOV), pixel dimensions of the FOV in the image plane, the angle of depression of the LOS and the altitude of the sensor location or carrying aircraft.

System noise in an image recognition system affects the performance of the system in two ways. Firstly, the target may fail to meet the segmentation criteria of the system, resulting in a missed target. Secondly, the feature values of the segmented objects may be erroneous, resulting in missed targets as well as false alarms. Improved false alarm and detection is achieved by accumulating information regarding the locations and the feature values of the objects from frame to frame.

In the sequential frame analysis we first determine an interframe sequence of extracted objects containing a given candidate target in the present frame. We then determine if the classifier result on the candidate target in the present frame is consistent in certain manner with the classifier results on other objects from the past frames in the sequence. An inconsistent classifier result is modified in some prespecified manner that yields better classification results. This method of "smoothing" the classifier result consists of three distinct steps, frame alignment, interframe object matching and decision smoothing.

The frame alignment technique estimates the relative translation, rotation and scale change between two successive frames. To estimate this frame-to-frame change, segmented image frames and an associated feature vector for each segmented object in the frame are computed first. A symbolic matching of segmented objects in the two frames is then performed to determine the correspondence between objects in the successive frames. The classifier decision made on a candidate target in the present frame is modified based on the decisions made on the same object in the immediate past frames using maximum likelihood estimate.

SYNTACTIC AND STATISTICAL TARGET RECOGNITION

At short ranges, when the target images are large enough to show detailed components linguistic recognition techniques are used to classify the detected targets into one of the various target types. When the target image is too small to show any structural detail, a Knn Classifier is used to classify the targets.

As it turns out, the number of features required for statistical pattern recognition is often very large, which makes the idea of describing complex patterns in terms of a (hierarchical) composition of simpler subpatterns very attractive. Also, the number of possible descriptions is very large in the case of tactical targets from relatively close range. In such a case it is impractical to regard each description as defining a class. Consequently, the requirement of recognition is better satisfied by a syntactic description of each class rather than by its classification.

The assumption in this syntactic approach to tactical target recognition are:

- Images of tactical targets are large enough to show structure.
- It is easier to recognize target components than the target.

The first assumption deals with the sensor-target range. If the range is too large to show any details inside the target, one would have to resort to statistical recognition techniques. But as the sensor-target range decreases and the target structure becomes discernable, syntactic recognition schemes become feasible. From our experience, if the target area is of the order of one-half to one percent of sensor FOV, syntactic recognition schemes are feasible. This translates to about a ten centimeter pixel resolution.

The second assumption deals with the relative ease of recognizing target and its components. If it is easier to recognize a target than its components, as would be the case when target image is only a few pixels, one would not employ syntactic recognition schemes. But in low quality images where the recognition based on target outline is not very reliable, a syntactic scheme can be successfully used to recognize targets provided the assumption on target image size holds. Even for good quality images, target orientations will result in different target outlines. Consequently, one will need several classifiers for each type of target. In principle, one set of syntactic rules can be generated to recognize the target from all aspect angles. Syntactic recognition schemes can also be successfully used for partially occluded targets where conceivable statistical recognition schemes would fail.

A syntactic target recognition technique has been successfully developed and demonstrated for FLIR images [3] by Honeywell. An example of syntactic target recognition is shown in Figure 5. The top row of the figure shows, from left to right, the input image, coarse segmentation, component extraction. The bottom row of the figure shows, from left to right, classification of the components and target recognition.

Small image statistical classifier and large image syntactic classifier are combined into a single adaptive target classifier system. The system is guided by a control module which is programmed to select one of a set of criterion functions in selecting appropriate classifier. Further detailed description of the system is given in AIRS program final report [4].

CONFLICT REMOVAL USING A NETWORK KNOWLEDGE MODEL

Conflict removal combines object information and relational context information for modifying classifier decisions that are inconsistent with our world knowledge. The process requires modeling and representing the world knowledge regarding objects in the scene and determining an optimal

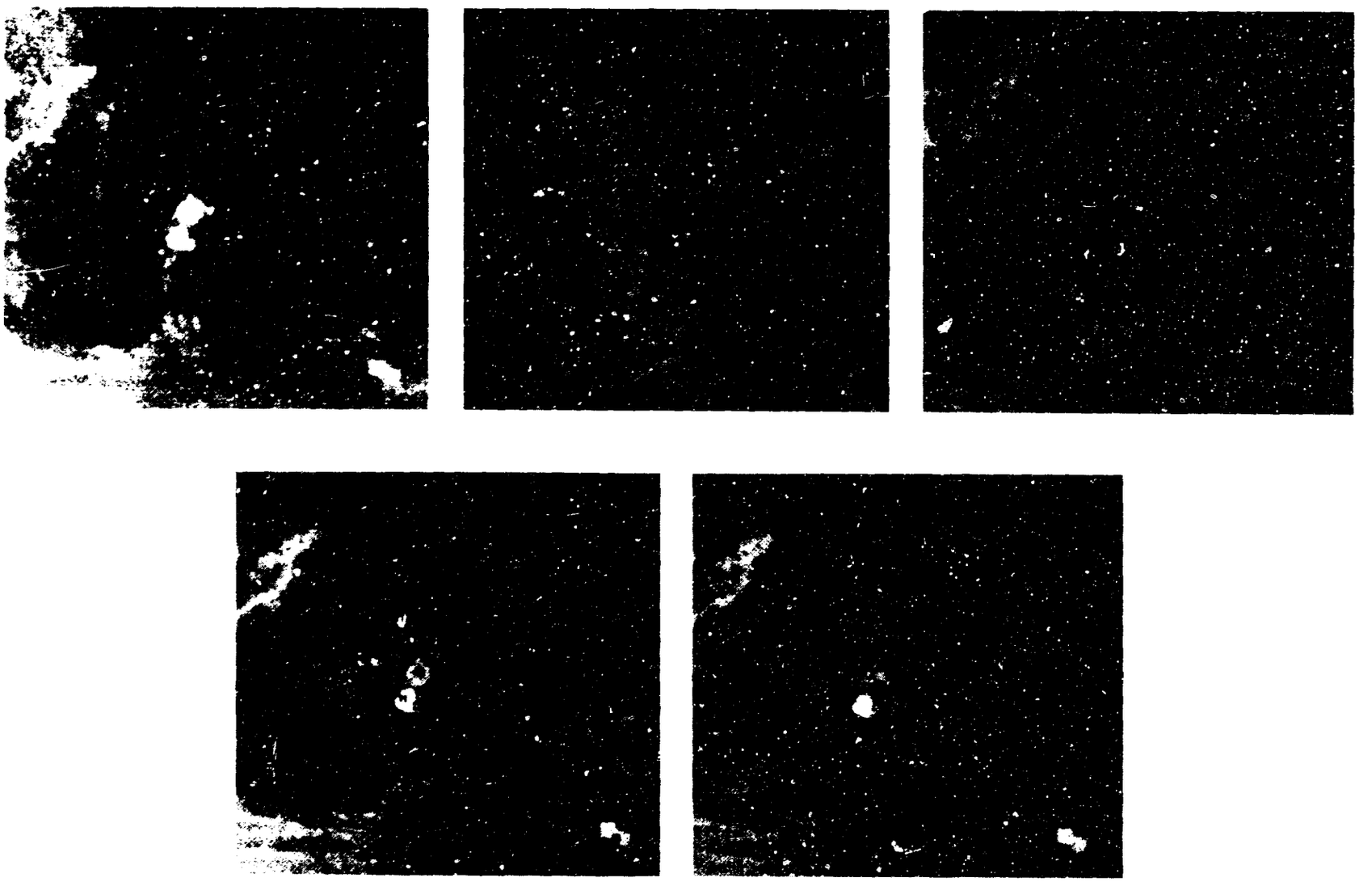


Figure 5. Syntactic Target Recognition.

way, called search strategy, of examining the scene using the knowledge model. The method can also be used for recognizing scene components containing multiple objects. Examples of such objects are airports, SAM sites, convoys, and bunkers. Various methods of modeling the knowledge and using the model to recognize mission oriented scenes exist in the literature [5]. The methods depend on the particular application of the system. We have combined appropriate concepts from various systems and developed a knowledge model and a search strategy for military tactical importance in imagery [6].

Conflict removal is performed by detecting inconsistent configurations in the scene. An example is a tank in the middle of a river. If the structural relationship between two recognized objects, one recognized as a tank and another recognized by the background classifier as a river, is such that the tank is located in the middle of the river then that particular configuration is flagged as inconsistent with the world knowledge network model. In such cases the target, the tank in our example, is reclassified to a "don't know" category. This conflict or inconsistency is removed by a sequential frame analysis, which is analogous to a human operator taking several looks at the scene of interest when he is not confident of his recognition result for the given scene.

Conflict removal can be effectively applied in reducing false alarm and using a priori scene or mission information in recognizing complex targets. Consider, for example, the mission of detecting and locating a truck convoy. A network model is used [4] for representing the essential relational structure of a convoy. A FLIR image frame with a vehicle convoy is shown in Figure 6a. The image is highly texturous and contains many "blobs" that have the general appearance of the desired target. Indeed, statistical detection of target results in a large number of candidate target objects as shown in Figure 6b. However, the relational structure of many of these candidate target objects is not compatible with the description of the convoy. Application of the relational constraint in the network model of convoy results in the targets shown in Figure 6c, leading to the final display of the result as in Figure 6d.

With the application of conflict removal, its output constitutes final system output as shown in Figure 1.

CONCLUSION

Automatic target screener technology has come a long way since the pioneering Augmented Target Screener Subsystem [7,8] (ATSS) of USAF. Honeywell has developed the technology to a point where more and more advanced a priori knowledge can be used in the target screener. Many of the general artificial intelligence techniques have been successfully adapted to solve the real world problem in target screener. This has greatly enhanced the target screener in capability as well as performance. In experimental analysis the target screener system has recognized small image and large image tanks, trucks and vehicle convoys in various conditions of contrast, clutter, aspect angle, occlusion and range-to-target.

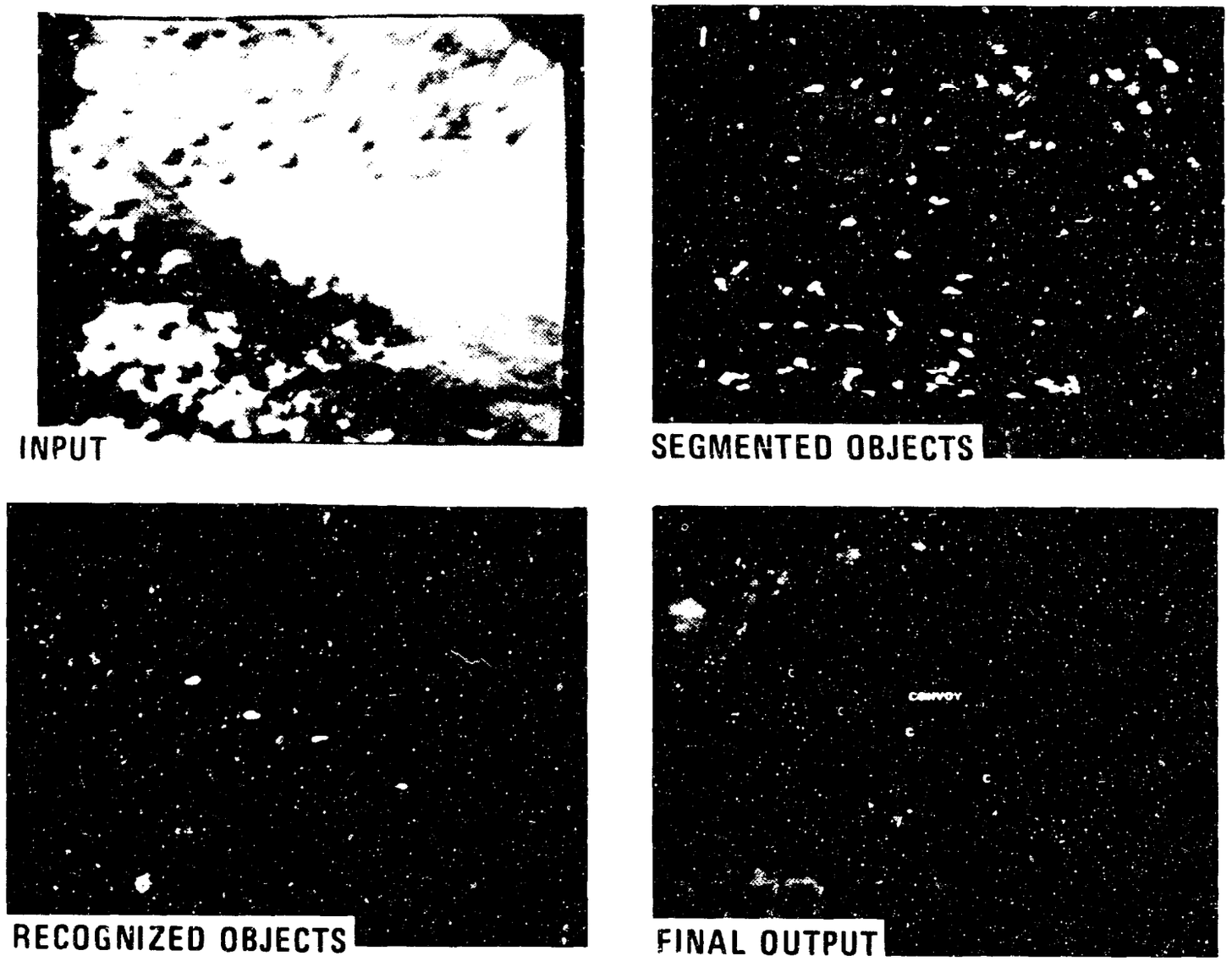


Figure 6. Convoy Recognition

ACKNOWLEDGMENTS

The authors wish to thank Dr. M. Geokezas, the Manager of Information Science at Honeywell for helpful discussions and encouragement. They also wish to thank Mrs. Susan Mattson for skillful typing in preparing this manuscript.

REFERENCES

1. R. K. Aggarwal, Image Segmentation Using Prototype Similarity, Proceedings of the IEEE Conference on Pattern Recognition and Image Processing, May 1979.

2. R. M. Haralick, et al., Textural Features for Image Classification, IEEE Transactions on System, Man and Cybernetics, Volume SMC-3, November 1973.

3. R. K. Aggarwal and T. M. Wittenburg, Syntactic Recognition of Tactical Targets, Proceedings of the DARPA Image Understanding Workshop, November 1978.

4. D. P. Panda, et al., Automatic Image Recognition System, Final Report, Honeywell, Inc., Systems and Research Center, in preparation.

5. F. Hayes-Roth, Representation of structured events and efficient procedure for their recognition, Pattern Recognition, Volume 8, 1976.

6. D. P. Panda, Configuration Analysis for Scene Recognition, Technical Memo, Honeywell, Inc., Systems and Research Center, March 1, 1979.

7. G. D. Swanlund, Augmented Target Screener Subsystem (ATSS), AFAL-TR-74-184, October 1976.

8. M. Geokezas and R. Jennewine, Target Screener/FLIR System, NAECON-77, 1977.

Paper No. IIIB-2, Presented at the Workshop on Imaging Trackers and Autonomous Acquisition Applications for Missile Guidance, 19-20 November 1979, Redstone Arsenal, Alabama.

TARGET DISCRIMINATION WITH MULTI-DIMENSIONAL SENSORS

Douglas N. DeFoe
CAI, a Division of Recon/Optical, Inc.
550 West Northwest Highway
Barrington, Illinois 60010

ABSTRACT

This paper examines one method of detecting, acquiring and windowing a target by using more than a single kind of information. Specifically, this method of target discrimination was developed for a Passive Tracker/Ranger System in order to detect aircraft. Inherent to the Tracker/Ranger is the production of range and motion information, as well as brightness of the scene. Fortunately, these three characteristics are those which are most likely to distinguish an aircraft from other objects, so the task of detection is easily implemented with such a system. Nevertheless, the method applies equally to the detection of other classes of targets whose distinguishing characteristics are less obvious to a given sensor system.

INTRODUCTION

The tasks of target detection, acquisition and windowing by an imaging sensor rest on the ability to discriminate the image of a target from the image of the background. Of course, the implication is that there exist some key characteristics (or, more properly, some combinations of characteristics) by which target objects are distinguishable from non-target objects. Table I is a list of several candidate characteristics which are both observable by imaging sensors and may help to discriminate one class of objects from other classes.

TABLE I. SOME DISTINGUISHING CHARACTERISTICS

- Brightness
- Color
- Surface Texture
- Edge Smoothness
- Symmetry
- Periodicity
- Motion
- Range
- Size
- Shape Composition
- External Spatial Relationships
- Internal Spatial Relationships

Note that various types of resolution (contrast, spectral, spatial, etc.) can play crucial roles in the definition of these terms. For any one characteristic we may wish to detect certain values within set bounds or, more loosely, values sufficiently different from a norm or mean. For example, if in an acquisition mode, the system may be set to detect differentness, but if in a reacquisition mode the system may be set to detect values within bounds of the last known value. Further, in a reacquisition mode new characteristics may become significant, such as position and track. For any specific case then, these terms must be carefully defined.

The particular set and relative importance of characteristics must take into account the particular class of targets to be detected and possibly the geometry and circumstances of observation. The point made here is that no one of the candidates nor any one combination is either sufficient or a necessary measure of "class"-ness. However, the probability that a given object belongs to a given class is at least as much a function of degree of concurrence as it is a function of magnitude of occurrence.

Based on this argument, in order for a sensor to more reliably discriminate certain types of objects, it is necessary that it detect or calculate a set of characteristics and examine their spatial coincidence. Both the implementation of generating raw maps of characteristics and the formal equation which combines them into a composite map are case specific; nevertheless, composite mapping is a powerful technique for a variety of tasks required by an autonomous weapon system.

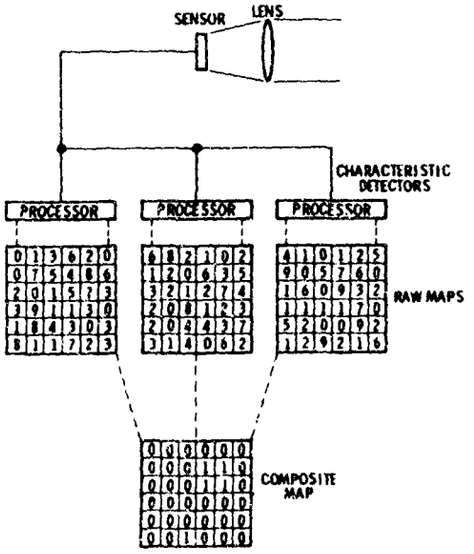


Figure 1. Composite Mapping

Several features of composite mapping are worth enumerating at this point. Detection, acquisition and windowing can be controlled by a single composite map equation, in that acquisition is no more than a decision to track a detected target and windowing is automatically

defined as the set of map locations at which a target is detected. With only slight modification and/or level gating, the same equation can be used in a reacquisition mode. Threat assessment and recognition programming can profitably use the same technique. Besides its applications, it is noted that a major strength of composite mapping is that multiple targets are processed in parallel. Furthermore, with proper formulation, the composite map values are more closely related to the confidence of "class"-ness than to any specific characteristic of the target. Finally, mapping provides excellent methods of discriminating against noise, in that both spatial coherence and temporal continuity can be imposed as conditions to detection.

In summary, composite mapping consists of the mapping of specific characteristics in a scene and combining these maps into a single map in a way which discriminates one class of objects from all others. It is a simple, powerful and valuable technique for many aspects of autonomous image analysis. The following describes a specific implementation of this technique as a method for detection of aircraft from an airborne platform.

BACKGROUND

Under Air Force Contract F33615-78-C-1562 CAI was tasked with conducting the preliminary engineering design of an Advanced E-O Tracker/Ranger System. The system is to be used on tactical fighter aircraft and linked to the gun director computer. One of the subtasks was to define an approach to long range autonomous detection and acquisition. Naturally, the two related tasks of windowing and reacquisition are implied for the proper operation of a tracking system. As previously pointed out, each of these subtasks can be thought of as a form of target discrimination. Prior to detailing a discrimination method, however, it is appropriate to describe the entire system and define some of its relevant capabilities.

CAI's approach to tracking and ranging utilizes area correlation as the calculation process. For ranging, two lenses and sensors are mounted with parallel optical axes. Because the parallax between two views of an object produces a misregistration inversely proportional to the range, the range of an object can be calculated by cross-correlating the two images in the pair of CCD image planes. Similarly, tracking is accomplished by the cross correlation of two images displaced in time rather than space.

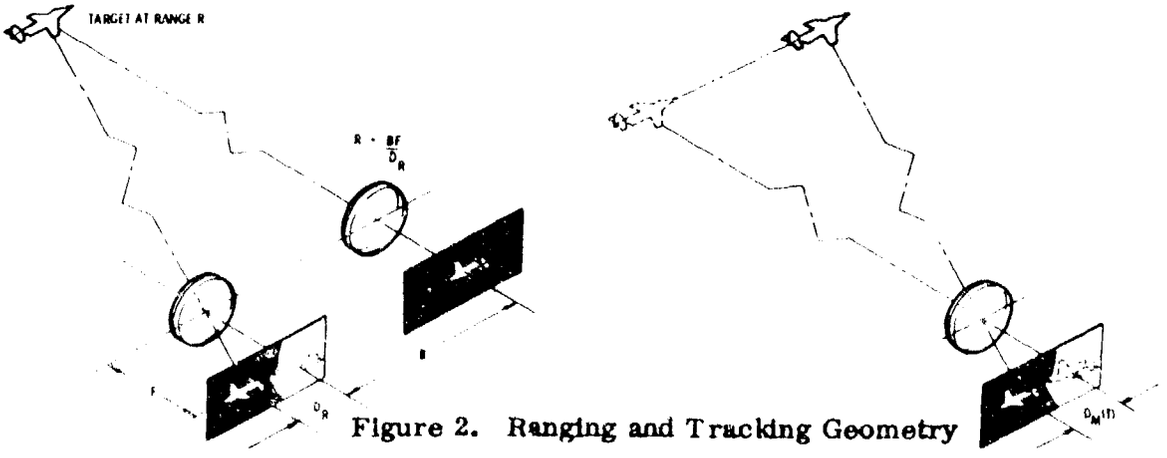


Figure 2. Ranging and Tracking Geometry

CAI's correlation algorithm has several important features which reflect on the system size, speed of calculation and autonomous operation. First is the high accuracy which measures image misregistration to small fractions of a pixel. As a result, precise ranging and tracking functions can be packaged in a small volume. Demonstration hardware has shown an rms error in image shift calculation at .02 pixel for a 10:1 signal-to-noise ratio. With filtering to reduce the noise, the error drops to a fraction of this value.

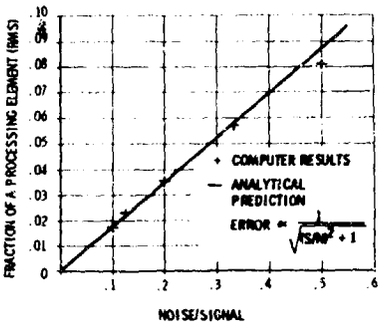


Figure 3. Correlator Accuracy

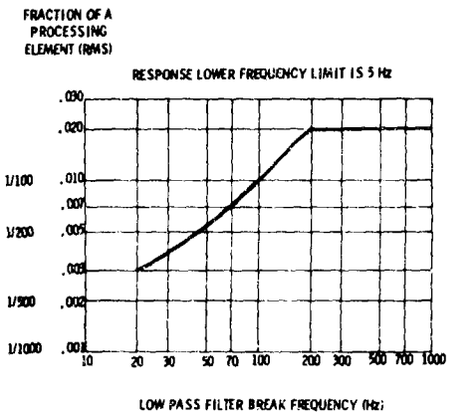


Figure 4. Correlator Demonstrator Accuracy (Noise Filtered)

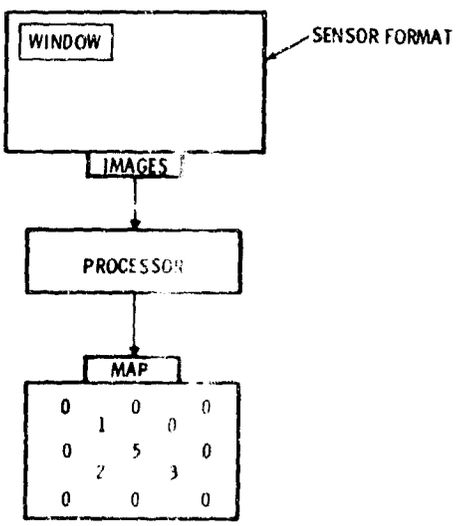


Figure 5. Information Flow

The second relevant feature is that the correlation is performed on relatively small subfields or windows. This means that the image in the field-of-view can be processed as a map of several smaller fields. Naturally, this ability to segment the image into small and overlapped processing areas is of crucial importance for using the correlation output in composite mapping.

The third important feature is that CAI's algorithm lends itself to pipeline processing. That is to say that the formula for the calculation of misregistration can be implemented in a form which builds up the result as the image on the sensor is read out. After the last pixel of the correlation subfield has been read only a few arithmetic operations are required to produce the result. The importance of this is that correlation is performed at the imaging rate and very high speed operation is possible.

Physically, the system (see figure 6) consists of an essentially spherical 8-inch head which houses the optics, image planes and a third axis torquer and encoder. Two lenses are used primarily for ranging, and a third lens with a larger field-of-view is used solely for tracking. Behind this head is a cylinder which houses the electronics, the two primary axis torquers and other necessary subsystems. For the detection mode of operation, only the ranging portion of the system would be used because of the better resolution afforded by these lenses. Range, motion and brightness maps would all be at the same scale, coming as they do from the same portion of the system.

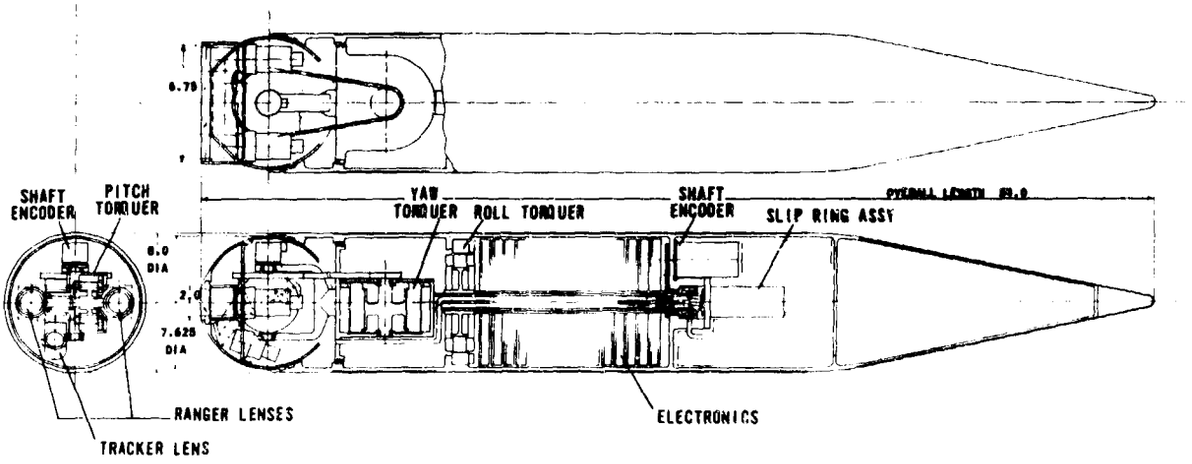


Figure 6. Airborne E-O Tracker Layout

The last step required to set the stage for composite mapping is the generation of high resolution, high "contrast" maps of the appropriate characteristics. Each of the features discussed above makes it possible to generate such maps of range and motion, but the realization is accomplished by a second kind of pipeline operation - the pipeline correlation of the set of overlapping windows in the field-of-view. This step, which is easily implemented, yields a signal similar to the video signal, but whose amplitude is not the brightness but the image shift. This signal is produced at the same rate and with essentially the same resolution as the video signal. While the correlation maps are derived from subfields containing several pixels and it would seem that the resolution is reduced, in fact the correlation result is the shift of the larger portion of the image, rather than a blurred value. Hence, it is reasonable to consider the resolution as nearly equivalent to a pixel-by-pixel resolution.

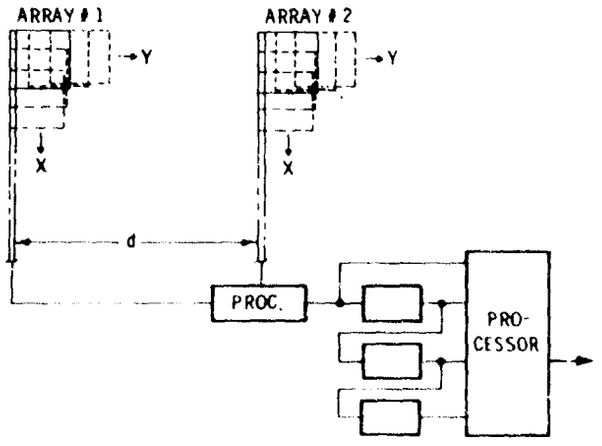


Figure 7. Pipeline Subfield Processing

The Tracker/Ranger System is, then, a compact imaging sensor which for each field-of-view can produce three data maps of brightness, range and motion. That these three characteristics can be considered key characteristics for discriminating aircraft from background is obvious. There remains only the need to formalize an equation which will combine them in a composite map.

THE COMPOSITE MAP EQUATION

For the application of detecting airborne targets, several considerations must be included in the composite map equation.

1. Brightness is not in itself reliable information. In the absence of range or motion information indicating the probable presence of a target, brightness variation should be ignored. However, where range or motion information indicates the presence of a target, unusual brightness should be considered corroborating information.

2. Against a fairly close background it is uncertain whether range or motion of a target will be a distinctive characteristic. However, against a nearly infinitely distant background target range is assured to be a distinctive characteristic. Hence, for a mean range greater than some threshold it is appropriate to weight the equation in favor of range information.

3. Because the system is mounted in an airborne platform, motion information is relative rather than absolute. Furthermore, if there is a target in the field-of-view it is not known a priori whether an exceptional or unusual value for motion would pertain to a target or background. For a target covering a small portion of the field, target motion would be unusual, while for a target covering a large portion of the field the background motion would be the unusual value. This ambiguity can be resolved if one assumes that targets always are nearer in range than background. For any composite map location, then, the sign of the amplitude should be solely determined by the range map value for that location.

4. Motion can be reduced from a vector to a scalar because for this application we are interested only in detecting a distinctive vector. The reduction of order is accomplished by subtracting the mean vector and taking the magnitude of the resultant for each map location.

5. Again, because we are interested only in distinctive values for detection and acquisition, we should normalize each map by subtracting the mean and dividing by the standard deviation prior to the generation of a composite map.

Taking these considerations into account, the equation for amplitude of the scalar map at location (i, j) is given by:

$$C_{ij} = \left[\left| D_{M',\,ij} \right| \frac{D'_{R,\,ij}}{\left| D'_{R,\,ij} \right|} + D'_{R,\,ij} \right] \times \left[\left| B_{ij} \right| + 1 \right]$$

where:

C_{ij}	is the composite map value
$D_{M,\,ij}$	is the normalized scalar motion map value
$D'_{R,\,ij}$	is the normalized range map value allowing for the weighting discussed above
B_{ij}	is the normalized grey level

The manner in which the range map is weighted is by offsetting the mean value to a value midway between the actual mean and the value for the maximum desired detection range. The effect is to make it much less probable that composite map values for background will be positive, and only slightly less probable that values for targets will be positive. Because the motion and brightness components are absolute valued, a stronger distinction between target and background is forced.

By this equation a composite map is generated. A threshold is set as a tradeoff between false alarm rate and probability of detection. A further constraint on the detection logic is imposed, such that three adjacent map locations must have amplitudes greater than threshold for the system to indicate the presence of a target. Excellent results for this method have been predicted on statistical computer runs.

RESULTS

In the following results, the threshold level has been set to keep the false alarm rate at one per hour. Note that this figure does not reflect using temporal continuity as a constraint, so a false alarm will have only minimal effect on the system and will disappear on the next generated map.

Figures 8 and 9 show the predicted probability of detection for a head-on view of a MIG 23 at ranges of 18,000 ft and 24,000 ft against an infinitely distant background. Figure 8 shows the effect of motion, assuming that the target brightness is statistically indistinguishable from the background and figure 9 shows the effect of brightness variation, assuming no detectable motion. Based solely on range information, this worst-case view of a target shows 21% chance of detection at 24,000 feet and 73% chance of detection at 18,000 ft. Only a small amount of motion or brightness variation or a slightly better cross-sectional coverage brings the probability of detection to more than satisfactory levels.

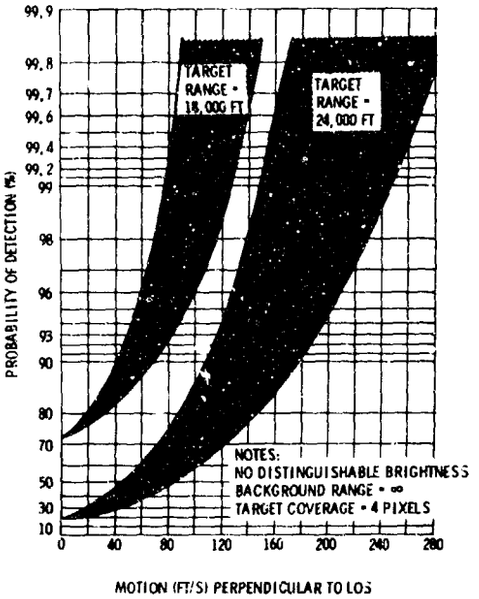


Figure 8. Probability of Detection vs. Motion

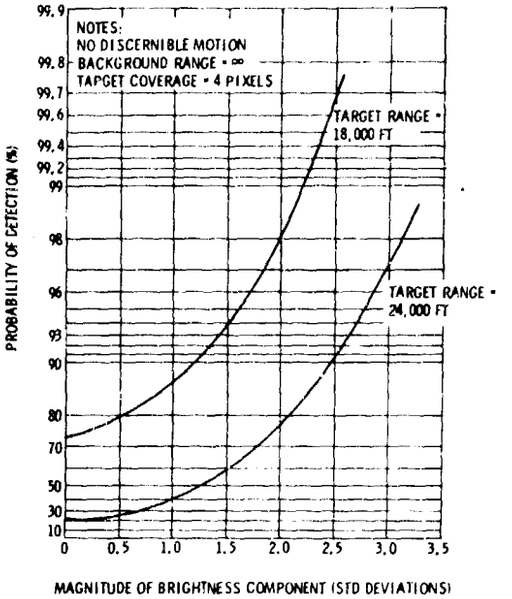


Figure 9. Probability of Detection vs. Brightness

In the case of indistinguishable range, as in looking down at a low-flying target, we note that the expected cross-sectional coverage is significantly larger and the motion is better defined, being referenced against a well structured background. Figure 10 shows the probability of detection for a target and background at 24, 000 and 25, 000-foot ranges, respectively, as a function of velocity (perpendicular to the line-of-sight) and cross-sectional image area. Again, brightness is assumed to be indistinguishable in this graph.

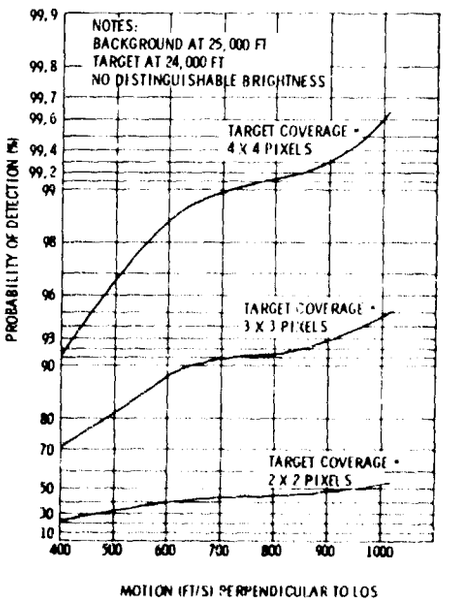


Figure 10. Probability of Detection vs. Coverage and Motion

As these examples illustrate, the detection method gives very good results for even the worst cases. We expect to be able to detect at long ranges airborne targets in any geometry with few failures. Equally important, we expect a false alarm rate which is quite satisfactory for either man-in-the-loop or autonomous systems.

CONCLUSIONS

As demonstrated by the Advanced E-O Tracker/Ranger System designed by CAI, composite mapping is a technique which offers the ability to detect and acquire targets of a particular class with great precision, as well as provide an automatic windowing function. Since this technique is essentially a form of image processing which, loosely speaking, makes objects of a particular class "bright," multiple targeting is a built-in feature. Further, it has potential as a method of target recognition. Carrying the analogy of brightening targets along, some characteristics which indicate target type within a class can be included to give the map "color." Threat assessment might require a different set of characteristics, but is just as easily implemented. It is not difficult to perform a variety of analyses once the hardware for mapping various characteristics exists in a sensor.

Both motion detection and range detection are realizable procedures with today's technology. Together they provide the basis for the autonomous targeting of aircraft by passive imaging sensors.

Paper No. IIIB-3, Presented at the Workshop on Imaging Trackers and Autonomous Acquisition Applications for Missile Guidance, 19-20 November 1979, Redstone Arsenal, Alabama.

MAN-MADE OBJECT DETECTION

H.C. Schau
Martin Marietta Aerospace
Orlando, Florida 32855

Abstract: A series of algorithms is presented which detect and localize man-made objects in images. The technique presented has low memory requirement, is easily implemented, and makes use of any a priori information in a natural manner. Results are shown for both hot and cold targets in 8 to 14 μ FLIR images. A discussion is included concerning the extension of this technique to aid the target classification problem.

1.0 INTRODUCTION

The areas of image processing and image pattern recognition have seen meteoric growth in the last several years, particularly in their applications to fire control systems and autonomous acquisition devices. This rapid growth has been precipitated by a new generation of solid state sensors and a host of powerful microprocessors available in militarized configurations. The microprocessor revolution has stimulated the already active area of digital signal processing and eased hardware constraints on the implementation of numerical algorithms developed in the research laboratory. Whereas in the past, target detection techniques were limited by the availability of hardware, the current techniques which are envisioned to be primarily under software control exist under a new set of constraints such as memory, number of multiples (speed), and the ease by which a priori information may be employed to aid the decision process.

As might be expected, the great activity in the area of autonomous acquisition has brought about a myriad of techniques for target detection and identification.[6,8] This is desirable since applications are usually specific in their requirements so that only a few of the many techniques can even be considered for implementation. In this paper we present a technique for localizing man-made objects (MMO) and performing a first order classification on the detected objects. The overall technique is presented as a series of individual numerical algorithms, and each is discussed separately in the next section. The desired output is the position and extent of possible man-made objects which have properties, such as size or shape,

which fall within preset bounds. The techniques will be demonstrated with FLIR (forward-looking infrared) scenes in the 8 to 14 μ atmospheric window.

It was initially expected that the line-to-line ac coupling of a FLIR sensor would require some form of quasi-dc restoration prior to any attempt at target localization; however, results have shown that in general this is not necessary. A dc restored scene (the data shown in this paper is originally dc, a FLIR simulation routine ac couples each line while giving several percent gain and bias offset to account for LED and amplifier nonuniformity) results in fewer false target regions that must be considered and thrown out. Results between raw FLIR data and their dc restored counterpart are not appreciably different. Although only 8 to 14 μ data are shown here, it is expected that with minor adjustments the technique will work for images in any wavelength band.

Figure 1 shows a flow chart of the individual algorithms which make up the overall technique. As will be discussed in more detail later, the

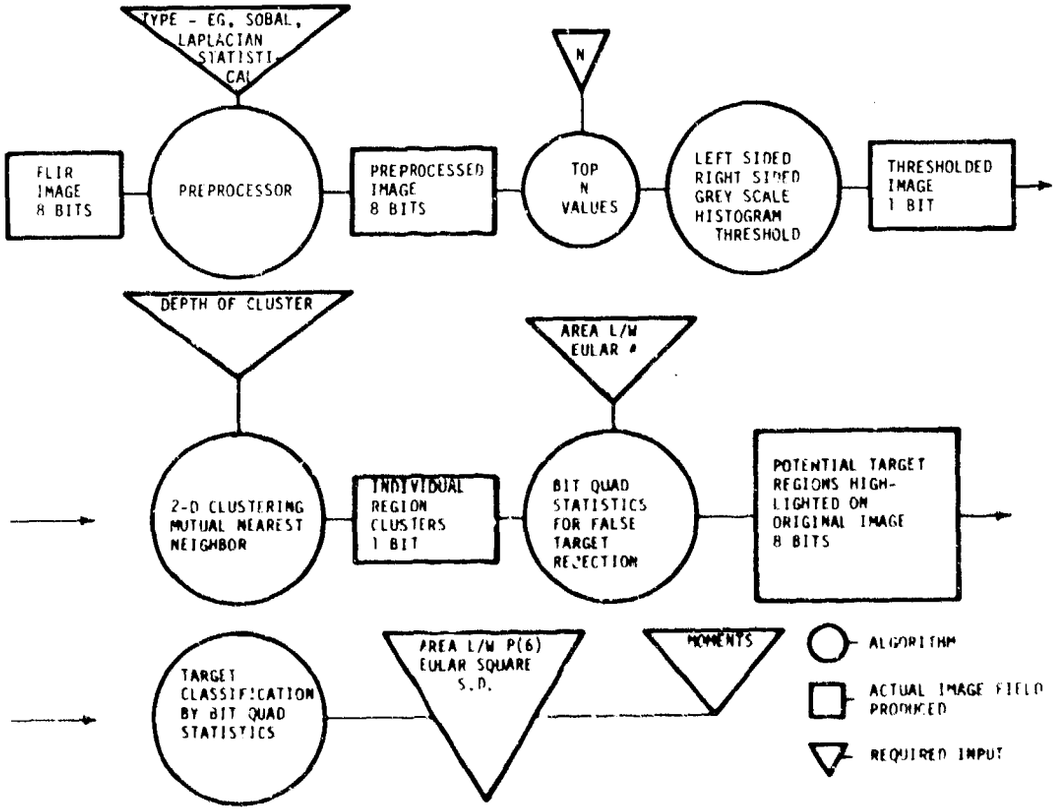


Figure 1. Man-Made Object Detection Sequence

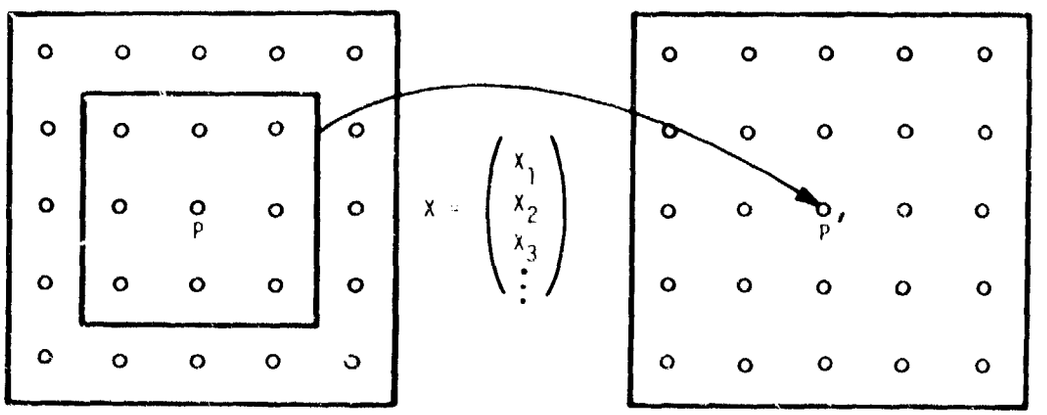
advantages of the overall technique are its simplicity, low memory requirement, and ease in application of a priori information. Disadvantages are the requirement of a preprocessor such as an edge extractor, and the use of global rather than local information. This can cause problems in scenes with two or more closely (within several pixels) lying targets, since several close targets could be accidently considered as one. No problems have yet been encountered in this area.

The basic philosophy of the MMO detection technique under consideration is that either there is a gradient at the object-background boundary (regardless of whether the target is hotter or colder than its surroundings) or the target has more internal structure (with higher spatial correlation) than natural clutter. In reality both cases are accepted.

2.0 ALGORITHMS

2.1 Preprocessor - Threshold

The first two sections shown in Figure 1 are the preprocessing and threshold algorithms. The preprocessor is a neighborhood modification processor (NMP) shown in Figure 2. The choice of sobal, laplacian, etc., depends somewhat on the application. We have not found any one type to be a clearly superior preprocessor in our work (we employ the modulus of any filter in this work, so that all preprocessed results are positive). Since most workers are familiar with results of preprocessors such as those considered here, we will include preprocessing results in a later section.



X_i = INPUT INTENSITY AT POSITION i OF NEIGHBORHOOD OF P

F(X) = OUTPUT INTENSITY AT POINT P

Figure 2. Neighborhood Modification Preprocessor

The next block (Figure 1) is the threshold algorithm. This algorithm works on pixel pairs where the first value is the pixel grey level at a particular location while the second value is edge or laplacian value at the same location. A specified number of the numerically largest processed values are then retained while all other pixels are thrown out ("N" in Figure 1). Typically 1.5 to 3.0 percent of the pixels are saved. From the remaining pixels, the grey level histograms are formed as shown in Figure 3. Notice that this is quite different than the total grey level histograms as illustrated. The thinned histogram is then used to

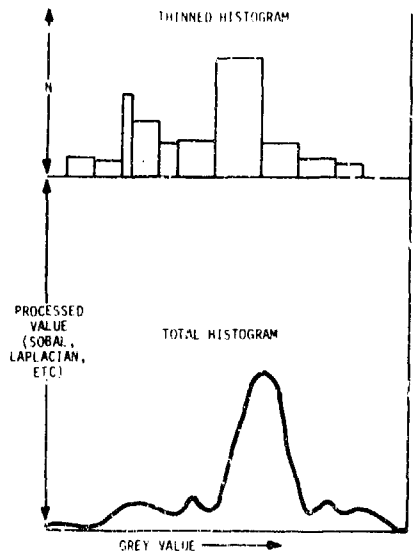


Figure 3. Histogram Thinning Process

produce two binary (one bit) thresholded images by a right seeking and left seeking algorithm. The basic assumption is that the left seeking threshold is looking for hot targets. The search is started on the right side of the histogram (hot side), and the threshold is defined as the first valley after the first peak. All pixels to the right of this threshold are set black (binary 1) while the rest of the pixels are set white (binary 0), including pixels thrown away which are always set to white. If the search extends for more than half of the total number of pixels contained in the thinned histogram, pixels to the left of the threshold are set black while these to the right are set white. The right seeking alrorithm works similarly from the left and presumes to find cold targets. Figure 4 illustrates the two algorithms. Due to the relatively few number of pixels (200 to 1000) to be distributed among 256 histogram "bins," some smoothing of the thinned histograms is required to define peaks and valleys. Although smoothed estimates of the thinned histograms may be produced, this has been found to be unnecessary. If a peak is defined as at least two successively decreasing pulses and a valley as at least two successively increasing pulses, the noisy nature of the thinned histogram does not appear to effect results. This allows effective operation of the algorithm while not requiring further numerical

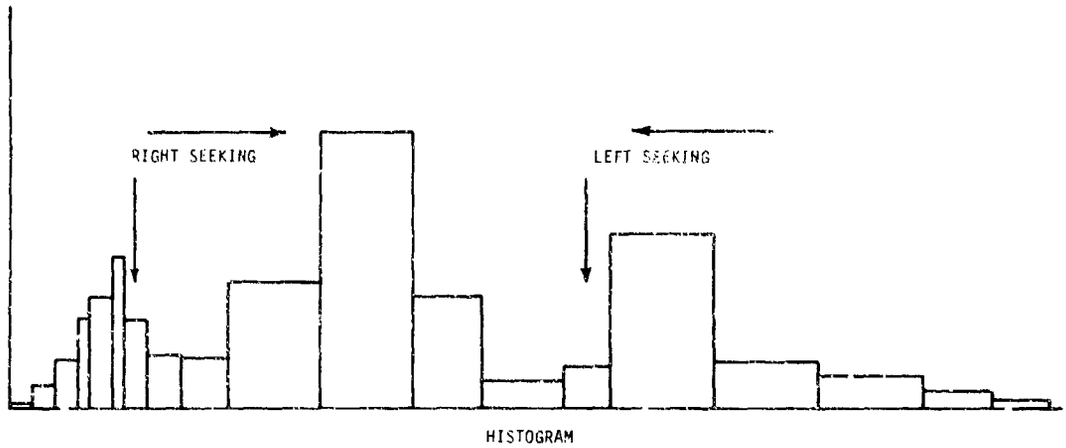


Figure 4. Histogram Mode Clustering Algorithm for Locating Thresholds Which Find Hot Targets (Left Seeking) and Cold Targets (Right Seeking)

processing to produce smoothed estimates. Employing global information in this form has the advantage of requiring only very small amounts of memory while retaining simplicity. Histogram manipulations may be performed easily on a microprocessor. The resulting one bit images contain at most 1.5 to 3 percent of the number of data points of the original image. As an example, consider a 512x512, 8 bit image; resulting binary images would require typically 10^3 bits of storage as compared with $2x10^6$ bits in the original image.

The next set of figures illustrate the threshold technique. Figure 5 is a FLIR image showing a burning hulk (bright object left of center), armored personnel carrier (APC) (left of burning hulk), tank obscured by a tree (center), and two tanks to the right (one only partially visible in the field of view). The scene is 128x128 pixels. Figure 6 shows the thresholded scenes for the sobal (left seeking in 6A, right seeking in 6B) and laplacian (left seeking in 6C, right seeking in 6D) operators. Notice that the sobal finds boundary points while the laplacian finds interior points as expected. Figure 7 shows the thresholded images for the third central moment operator. Figure 7A and 7B illustrates 7x7 window (left seeking in 7A, right seeking in 7B), while 7C and 7D present a 3x3 window size (left seeking in 7C, right seeking in 7D). It can be seen that the third central moment acts much like both sobal and laplacian. Although it will not be shown, the second central moment (variance) has been found to be very useful also. It can be observed that in all cases target points are turned black while only a few background points are chosen. The threshold algorithm is somewhat bothered in this data set by a line of data drop-out along the left hand margin. This data tends to use up the available data for the thinned histogram (only a fixed number of points are employed), since it is a region of high gradient. In any event the result of the algorithm can be seen.

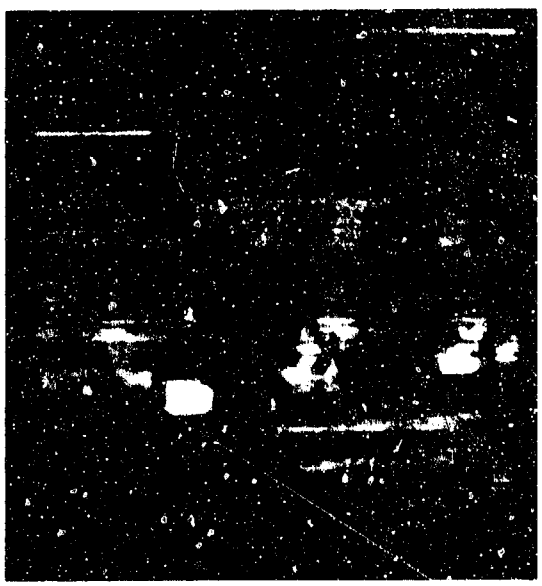
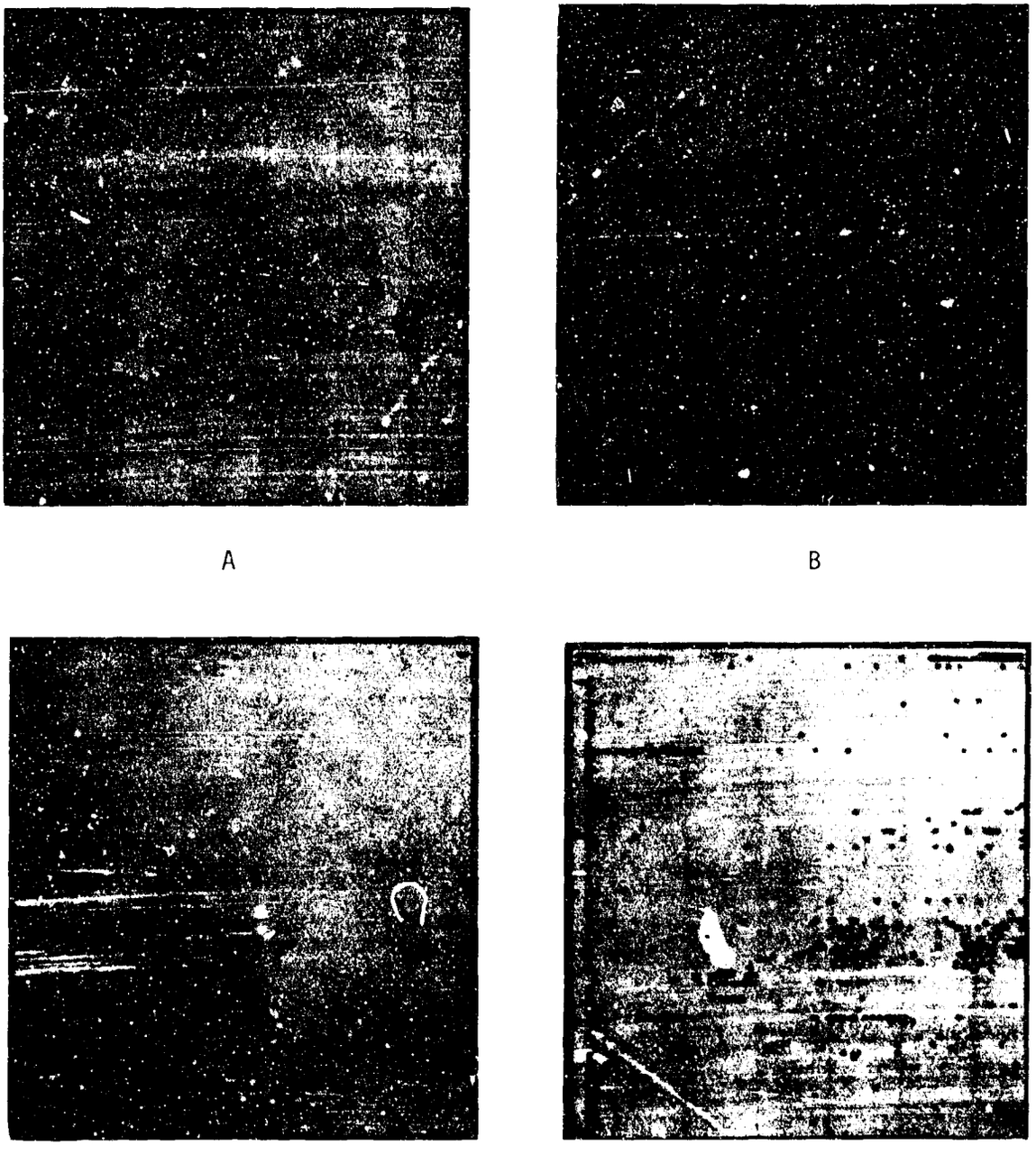


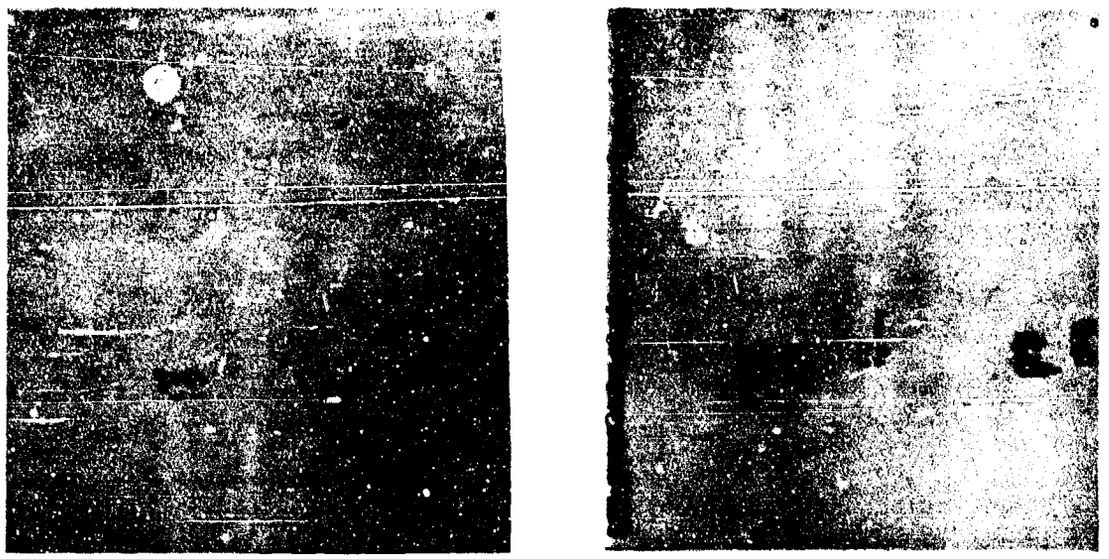
Figure 5. 8 to 14 μ FLIR Scene Containing Tanks, APC, and Burning Hulk



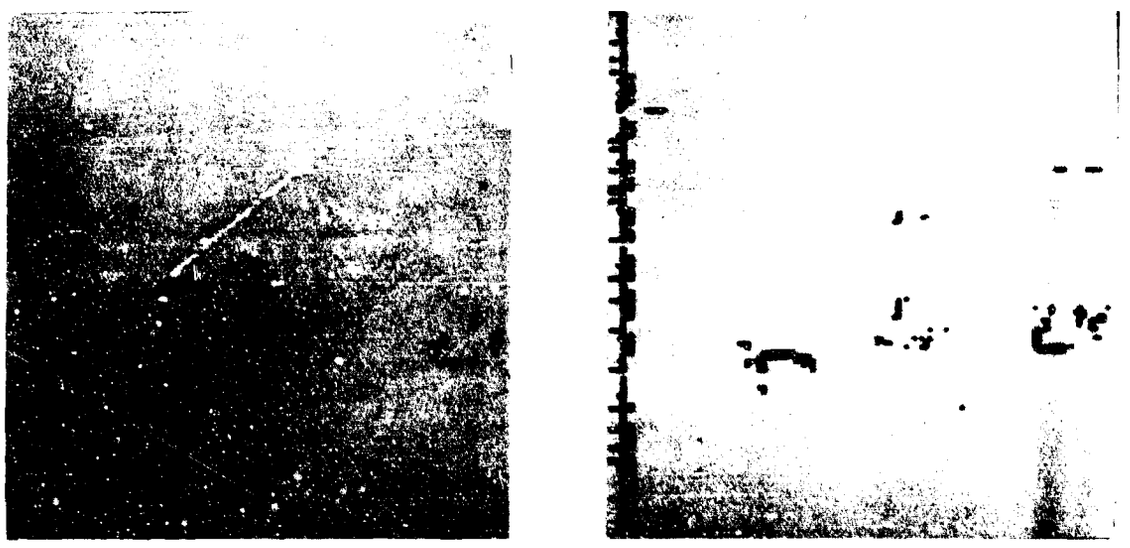
A B

C D

Figure 6. Thresholded Scene from Figure 5. Upper Scenes Display Sobal (bA - left seeking, 6B - right seeking); Lower Scences Show Laplacian Modules (6C - left seeking, 6D - right seeking)



A B



C D

Figure 7. Thresholded Scene from Figure 5. Upper Scenes Display 7x7 Third Central Moment (6A - left seeking, 6B - right seeking); Lower Scenes Show 3x3 Third Central Moment (6C - left seeking, 6D - right seeking)

2.2 Spatial Clustering[10]

The binary image produced by the thresholding algorithm is then segmented on the basis of pixel clusters. This is performed through a clustering algorithm. There are a great variety of clustering algorithms available; we have chosen one known as an agglomerative mutual neighborhood clustering algorithm. Figure 8 gives a brief explanation of the algorithm while Figure 9 shows an example of how results vary as a function of depth of clustering. The depth of clustering may be chosen without any prior knowledge and will not appreciably affect results (we have chosen a depth of clustering of 5). When knowledge of target size or correlation is known, the depth of clustering may be changed to enable the algorithm to work slightly more efficiently. An example might be the prior knowledge that one is trying to locate a bunker rather than a tank. In any regard the results are not critically sensitive to the depth of clustering chosen.

GIVEN AN ARRAY OF LABELED POINTS $l_i(X_i\ Y_i)$ MUTUAL NEIGHBOR VALUE BETWEEN LABELED POINT l_i AND l_j IS DEFINED AS

$mnv(l_i,l_j) = M+N$ WHERE

l_i IS THE M^{th} NEAREST EUCLIDEAN NEIGHBOR of l_j

l_j IS THE Nth NEAREST EUCLIDEAN NEIGHBOR OF l_i

THE TIGHTNESS OF CLUSTERS MAY BE EXTERNALLY CONTROLLED BY THE DEPTH OF CLUSTERING

Figure 8. Region Segmentation Using Agglomerative Mutual Nearest Neighborhood Clustering

2.3 Bit Quad Statistic

After the binary image has been clustered into several groups of pixels, each group is considered as a possible target location. A natural statistic for binary images are the number of bit quads as shown in Figure 10, of which there are 6 types. Counting the number of each type in each pixel cluster is simple and may be done in a parallel or pipeline manner. By simply counting the number of bit quads, global properties of the region under consideration may be estimated as shown in Figure 11. In the examples to be shown, the area, length-to-width ratio, and Eular number are set with rather wide bounds to reject regions that do not fall within our definition of a target.

Figure 12 shows the next frame of FLIR data following that shown in Figure 5. Figure 13 shows in order from the top the results of a sobal operator and the left and right seeking threshold algorithms. The binary

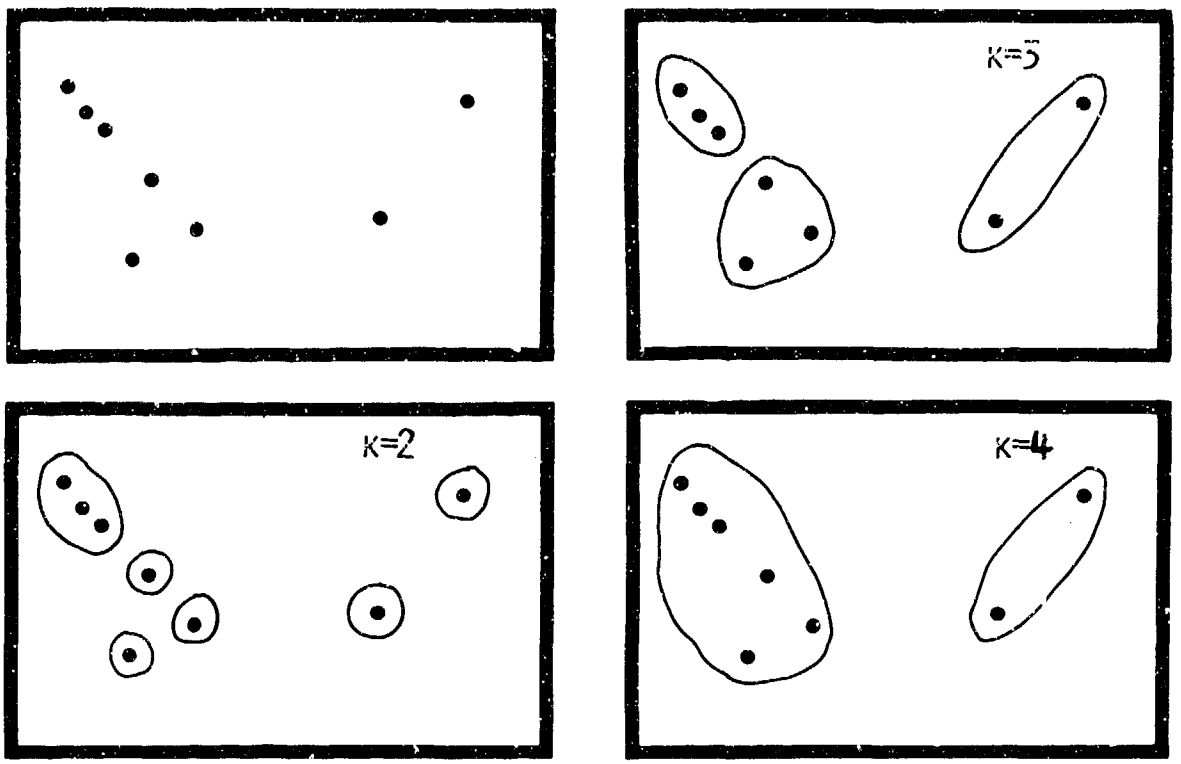


Figure 9. Clustering Example for Several Values of Depth of Cluster, K

BASIC DESCRIPTOR OF A BINARY IMAGE IS THE BIT QUAD. EXAMPLE:

$$Q_0 = \begin{array}{|c|c|}\hline 0 & 0 \\ \hline 0 & 0 \\ \hline \end{array}$$

$$Q_1 = \begin{array}{|c|c|}\hline 1 & 0 \\ \hline 0 & 0 \\ \hline \end{array} \quad \begin{array}{|c|c|}\hline 0 & 1 \\ \hline 0 & 0 \\ \hline \end{array} \quad \begin{array}{|c|c|}\hline 0 & 0 \\ \hline 0 & 1 \\ \hline \end{array} \quad \begin{array}{|c|c|}\hline 0 & 0 \\ \hline 1 & 0 \\ \hline \end{array}$$

$$Q_2 = \begin{array}{|c|c|}\hline 1 & 1 \\ \hline 0 & 0 \\ \hline \end{array} \quad \begin{array}{|c|c|}\hline 0 & 1 \\ \hline 0 & 1 \\ \hline \end{array} \quad \begin{array}{|c|c|}\hline 0 & 0 \\ \hline 1 & 1 \\ \hline \end{array} \quad \begin{array}{|c|c|}\hline 1 & 0 \\ \hline 1 & 0 \\ \hline \end{array}$$

$$Q_3 = \begin{array}{|c|c|}\hline 1 & 1 \\ \hline 0 & 1 \\ \hline \end{array} \quad \begin{array}{|c|c|}\hline 0 & 1 \\ \hline 1 & 1 \\ \hline \end{array} \quad \begin{array}{|c|c|}\hline 1 & 0 \\ \hline 1 & 1 \\ \hline \end{array} \quad \begin{array}{|c|c|}\hline 1 & 1 \\ \hline 1 & 0 \\ \hline \end{array}$$

$$Q_4 = \begin{array}{|c|c|}\hline 1 & 1 \\ \hline 1 & 1 \\ \hline \end{array}$$

$$Q_D = \begin{array}{|c|c|}\hline 1 & 0 \\ \hline 0 & 1 \\ \hline \end{array} \quad \begin{array}{|c|c|}\hline 0 & 1 \\ \hline 1 & 0 \\ \hline \end{array}$$

Figure 10. Shape Descriptors Using Binary Images

SHAPE AND TOPOLOGICAL ATTRIBUTES MAY BE GENERATED BY THE NUMBER OF BIT QUADS CONTAINED IN AN IMAGE, FOR EXAMPLE:

AREA

$$A=\frac{1}{4}n(Q_1)+\frac{1}{2}n(Q_2)+\frac{7}{8}n(Q_3)+n(Q_4)+\frac{3}{4}n(Q_D)$$

PERIMETER

$$P=n(Q_2)+\frac{1}{\sqrt{2}}[n(Q_1)+n(Q_3)+2n(Q_D)]$$

EULAR NUMBER

$$E=\frac{1}{4}[n(Q_1)-n(Q_3)-2n(Q_D)]$$

Figure 11. Global Properties Computed From Local Bit Quad Counts

Figure 12. Next Frame of FLIR Data after Figure 5

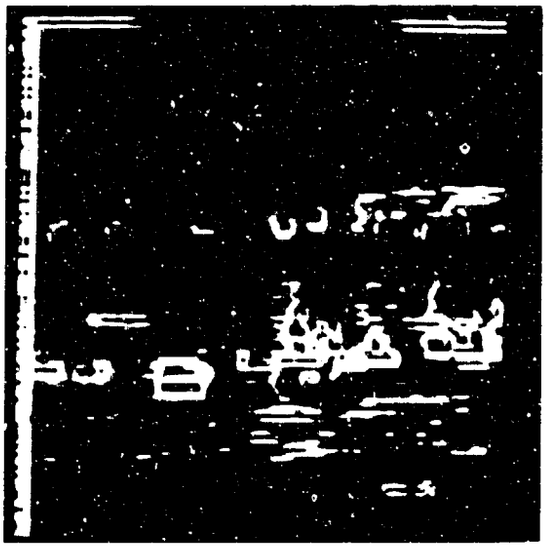


Figure 13. (Top to bottom, counterclockwise) Sobal output of Figure 12, left seeking thresholded image, right seeking thresholded image.

images were clustered and shape parameters computed. The five regions which passed are shown in Figure 14; four additional regions were rejected. Figure 15 shows the original scene reduced in intensity with the five "passed" regions highlighted. It can be seen all are targets. Figures 16 through 18 show a similar sequence for the third central moment operator where only four regions are found. This is not too surprising since the rejection of false targets was trained with the sobal results. In both cases no targets were found for the right seeking threshold.

3.0 COLD TARGETS

Perhaps the more difficult problem in detection of man-made objects is finding cold targets. Figure 19 shows a FLIR image of two burning hulks, (bright objects upper left) with a road running diagonally just below them. Several cold tanks are located above the road directly across from the hulks just to the right of the center of the image. Two muzzle flashes are seen. Figure 20 shows the highlighted results of the complete algorithms with the right seeking threshold. Notice that all operators found some part of the tank group. Notice also the data dropouts on the left margin has again caused several false targets. Figure 20E shows the third central moment operator with the left seeking threshold which has found the hulk. Other operators found no targets with the left seeking threshold. The hulk was also found in the 7x7 third moment - left seeking algorithm (upper hulk in 20D) and the 3x3 3rd moment - left seeking (20F). It can be observed that the technique does a credible job of finding the MMO in the scene. Hulks are generally not found because we have set the limits on area and Eular number to reject regions of unstructured hot pixels which occur in high density.

4.0 IDENTIFICATION

The algorithm set described herein is not intended to identify MMO, but does yield additional information that may aid in classification. Consider the distribution of bit quads from Figure 14 for five targets.

N(1)	N(2)	N(3)	N(4)	N(D)	Target - Side View
24	8	0	0	0	Tank - Deck
11	8	3	3	0	Tank
10	12	8	3	0	APC
12	42	12	0	0	Burning Hulk
11	4	5	0	0	APC

A glance at this limited set indicates there may be some classification information contained in the bit quad statistics. In any regard, it is important to train the rejection routine (and classification routine if this proves feasible) with the particular preprocessor (e.g., sobal, laplacian, etc.). This can be seen by the average bit quad distribution for several preprocessors compared for the same target group.

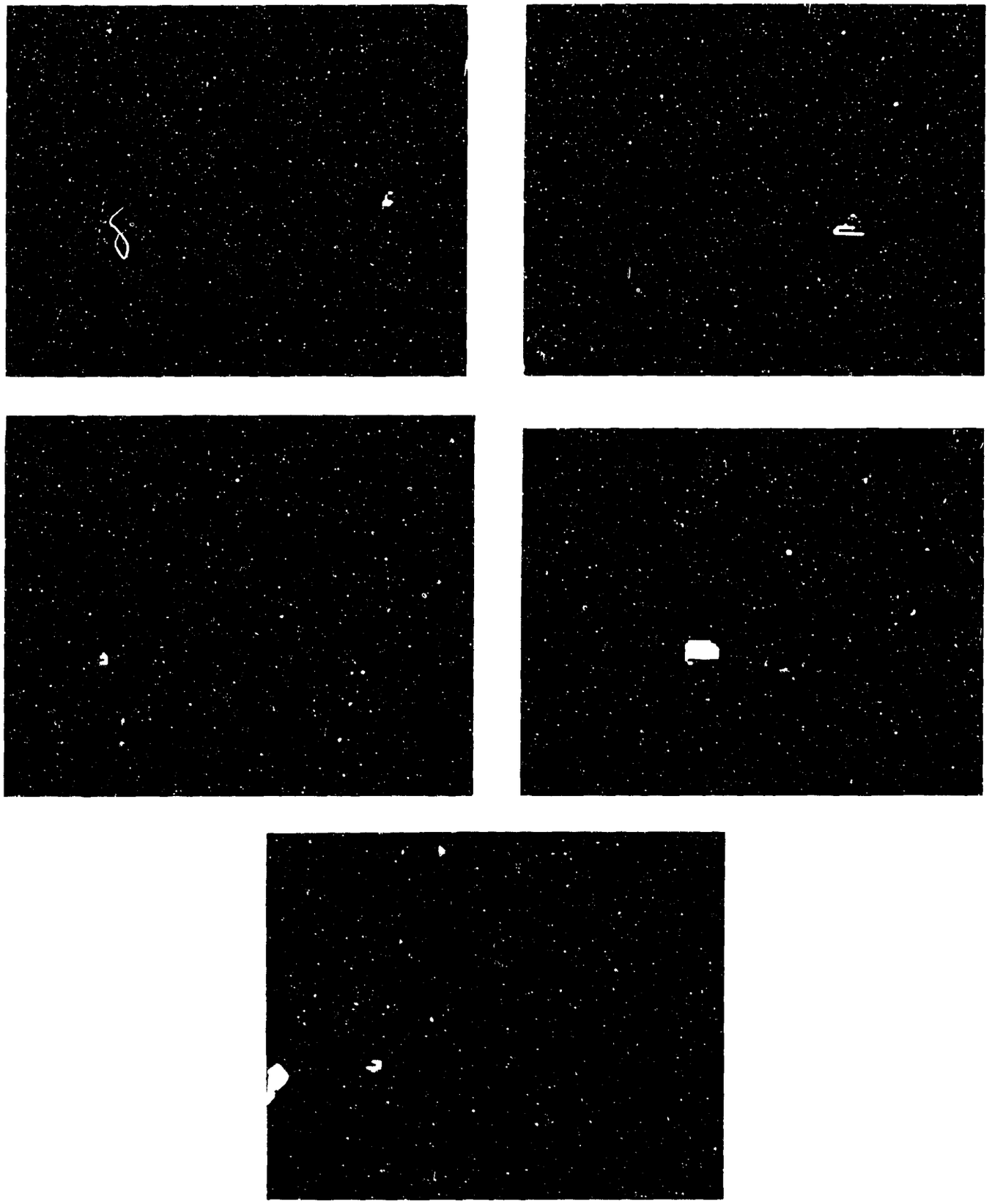


Figure 14. Five Potential Target Regions Segmented Out of Figure 13 (Left Seeking)

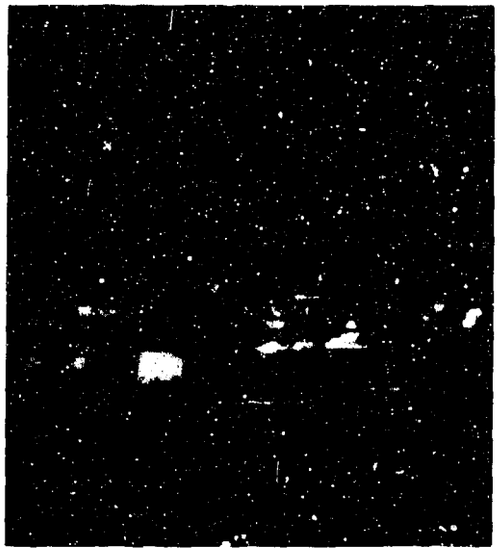


Figure 15. Highlighted Target Regions of Figure 12 from Sobal Operator

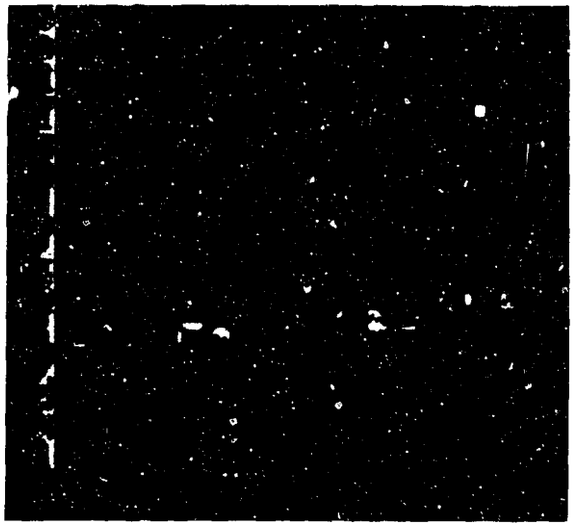


Figure 16. (Top to bottom, counterclockwise) 3 x 3 third central moment, left seeking thresholded image, right seeking thresholded image

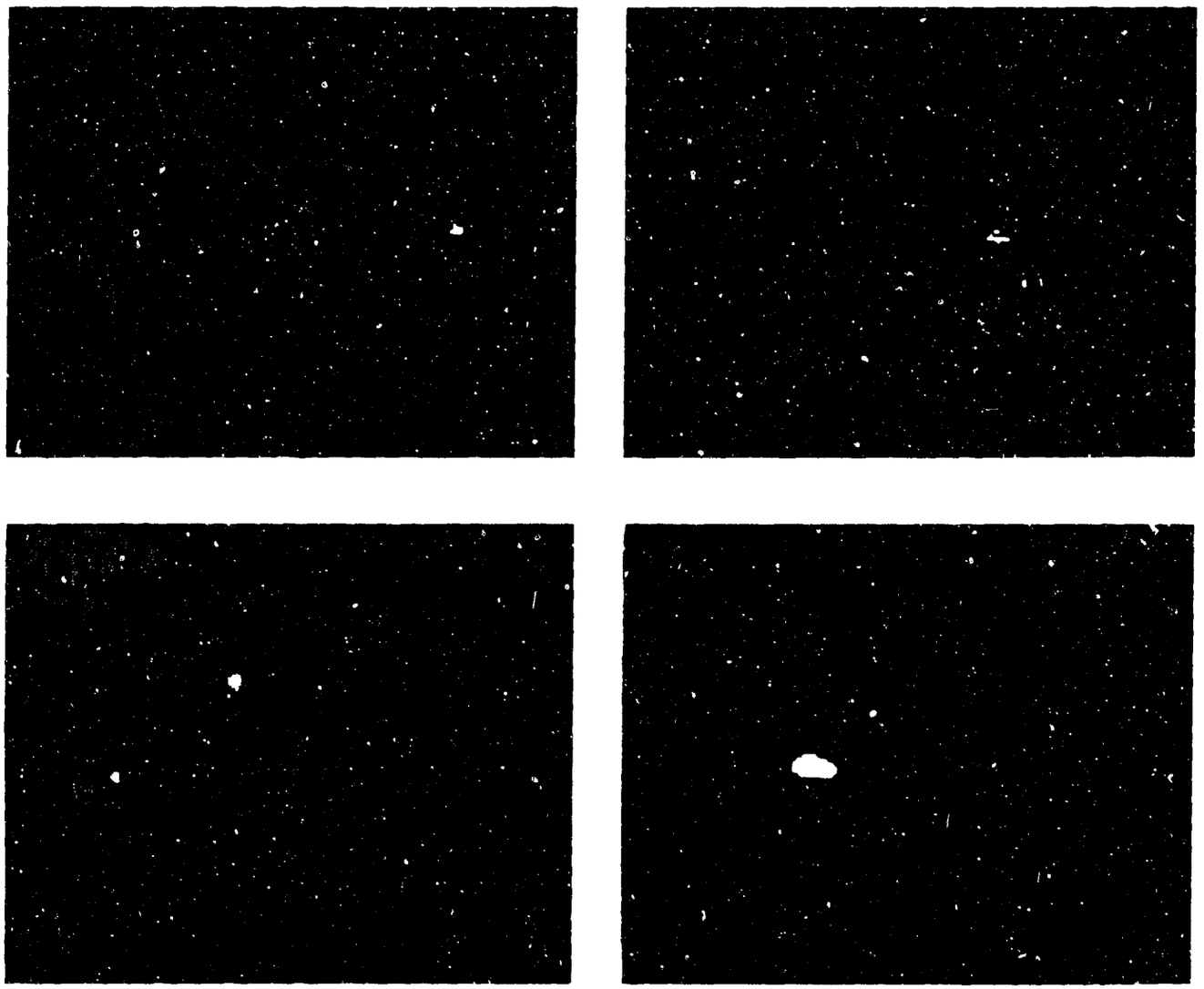


Figure 17. Four Potential Target Regions Segmented Out of Figure 16 (Left Seeking)

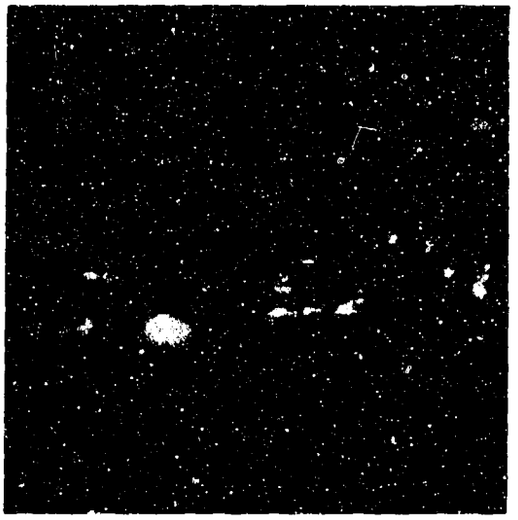


Figure 18. Highlighted Regions of Figure 12 from 3x3 Third Central Moment Operator

Figure 19. 8 to 14 μ FLIR Image Showing Two Burning Hulks and Several Cold Tanks

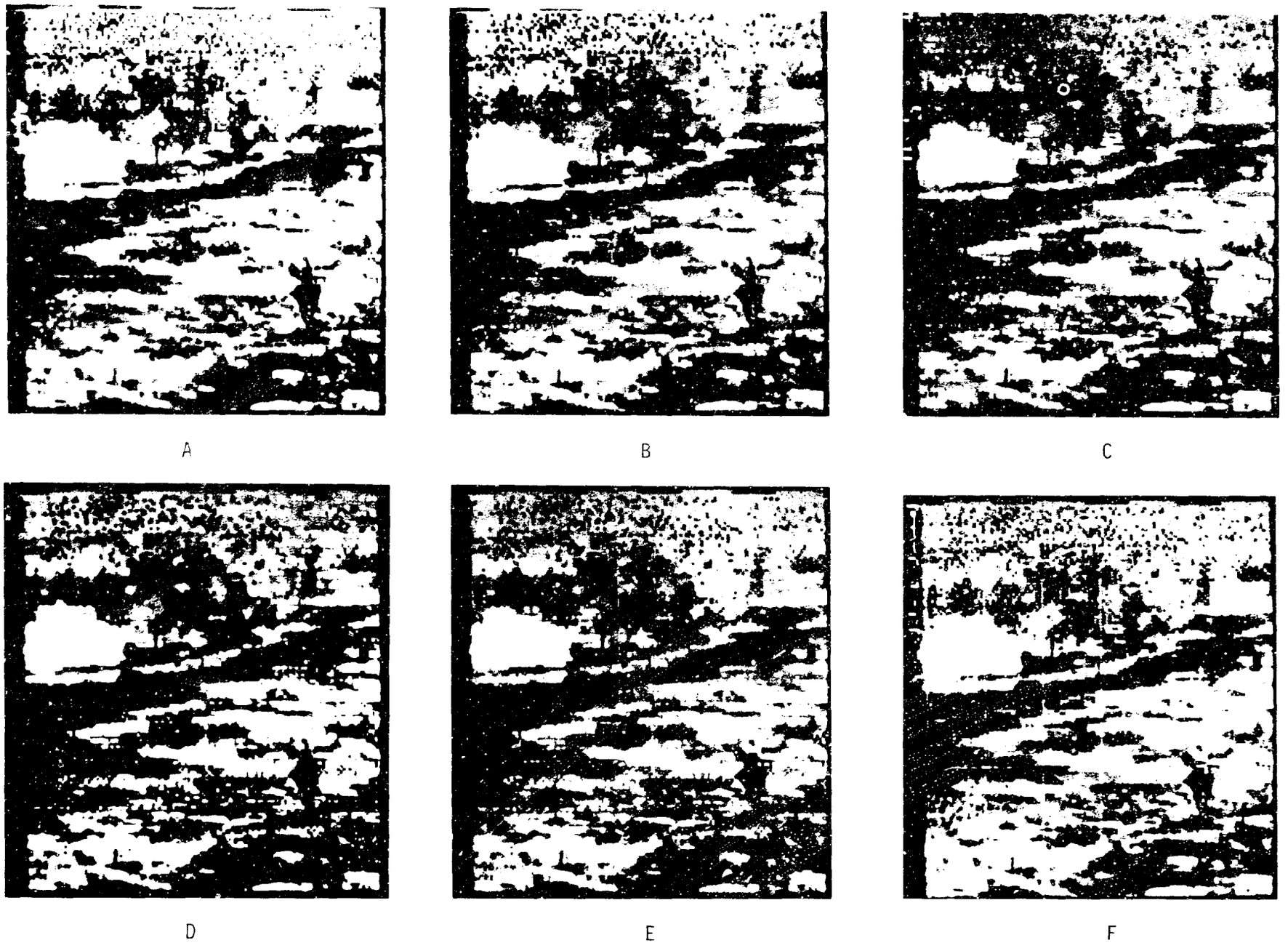


Figure 20. Highlighted potential target regions of Figure 19. They are: A) Sobal, right seeking; B) Laplacian #1, right seeking; C) Laplacian #2, right seeking; D) 7 x 7 Third central moment, left seeking; E) 3 x 3 Third central moment, left seeking; F) 3 x 3 Third central moment, right seeking

N(1)	N(2)	N(3)	N(4)	N(D)	Type
11.25	13.0	5.0	2.0	0.5	Sobal
19.25	2.25	0.25	0.25	0.75	Laplacian No. 1
12.0	5.0	0.75	0	0.25	Laplacian No. 2
15.25	8.0	1.25	0.25	0	Third Central Moment

$$\text{Laplacian No. 1} = \begin{pmatrix} 1 & -2 & 1 \\ -2 & 4 & -2 \\ 1 & -2 & 1 \end{pmatrix}$$

$$\text{Laplacian No. 2} = \begin{pmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{pmatrix}$$

Results are as expected; edge extractors such as the sobal provide boundaries which have high N(2) counts where laplacians enhance isolated interior points (high N(1) values). The third central moment operator achieves results between these two.

5.0 CONCLUSION

As stated in the introduction, there are a variety of numerical techniques for localizing and classifying potential targets for fire control and autonomous acquisition applications. This is necessary since each application has specific constraints and will permit consideration of only a few possible solutions. In this paper we have presented a technique for detecting and classifying military targets in unrestored FLIR imagery. The algorithms presented require a minimum of memory and are capable of fast implementation. Prior information is employed in a natural manner to enhance performance; however, no penalty is paid for instances where this information does not exist. By making the algorithm not independent on any one piece of specific information, results are generally consistent for many applications and conditions.

Results are encouraging on data sets containing both hot and cold targets in the presence of false targets. It is expected that this technique will lend itself to the solution of many autonomous acquisition problems when a final algorithm is added which considers each potential target region and performs an identification on the basis of a set of extracted features.[4,5] Work toward this end is currently in progress.

6.0 REFERENCES

1. Pratt, W.K., Digital Image Processing, Wiley Interscience, New York (1978).

2. Schau, H.C., "Statistical Filter for Image Feature Extraction," submitted to Pattern Recognition.

3. Gray, S.B., "Local Properties of Binary Images in Two Dimensions," IEEE Transactions on Computers, C-20, 551 (1971).

4. Hu, M., "Visual Pattern Recognition by Moment Invariants," SRE Trans Inf. IT-8, 179 (1962).

5. Dudani, S., Breeding, K.J., McGhee, R.B., "Aircraft Identification by Moment Invariants," IEEE Transactions on Computers, C-26, 39 (1977).

6. Tisdale, G., "A Digital Image Processor for Automatic Target Cueing, Navigation, and Change Detection," SPIE Proc., 101, 112 (1977).

7. Panda, D.P., Rosenfeld, A., "Image Segmentation by Pixel Classification in (Grey Level, Edge Value) Space," IEEE Transactions on Computers, C-27, 875 (1978).

8. Geokezas, M., Jennewine, R., "Target Screener/FLIR System," SPIE Proc., 101, 84 (1977).

9. Schacter, B.J., Davies, L.S., Rosenfeld, A., "Some Experiments in Image Segmentation by Clustering of Local Feature Values," Pattern Recognition, 11, 19 (1978).

10. Gowda, K.C., Krishna, G., "Agglomerative Clustering Using the Concept of Mutual Nearest Neighborhood", Pattern Recognition, 10, 105 (1978).

Paper No. IIIB-4, Presented at the Workshop on Imaging Trackers
for Autonomous Acquisition Applications for Missile Guidance,
19-20 November 1979, Redstone Arsenal, Alabama.

The Classification of Regions in a Difference Picture for Motion Analysis

Ramesh Jain
Department of Computer Science
Wayne State University
Detroit, Michigan 48202
U.S.A.

Abstract

Analysis of a frame sequence for the recognition and tracking of moving objects is becoming one of the active areas of computer vision. Difference pictures have been used for the motion analysis and the segmentation of a dynamic scene. This paper is concerned with the problem of classifying regions of a difference picture. We modify the method of Jain and Nagel for the classification of the regions in a difference picture resulting in a more robust approach. A novel method is proposed for the identification of the regions due to the occlusion of two or more moving objects. The results of several experiments are presented.

Introduction

Computer analysis of motion is attracting increasing attention of researchers [6,7]. Change detection is an important part of motion analysis. In some systems each frame of a sequence describing the dynamic scene is segmented and then matching techniques are used for the change detection in the frames [5]. In other systems low level methods are used for change detection [1-4]. This paper is concerned with the latter approach.

Motion of the objects results in the transformation in the frames of a sequence describing the dynamic scene. A binary difference picture can be prepared to represent changes in the frames due to transformations resulting from motion, by comparing intensities at the corresponding pixels of two contiguous frames of the sequence. The regions of connected 1 entries in the difference picture due to the covering of the background by a moving object image, the uncovering of the background by a moving object, or both the covering and the uncovering of the background are called regions of type O, B, or X, respectively [2, 3, 4]. It has been shown that the knowledge of the type of a region gives important information for motion analysis and for segmentation of scenes into stationary and nonstationary scene components. For determining the type of regions Jain and Nagel [2] proposed a method based on the computation of a ratio called CURREF. This ratio has been used by Jain et al. [4] for the extraction of images of the moving objects from an image sequence.

In this paper it is shown that the ratio CURREF may give incorrect classification in some situations for the X type regions. Also, the method of Jain and Nagel gives wrong classification in case of the occlusion of two or more moving objects. We modify the method of classification by slightly changing the definition of the ratio.

The modified ratio removes the limitations of the classification proposed by Jain and Nagel.

Occlusion of moving objects poses problems. We propose a method for the detection of regions in a difference picture due to occlusion of two or more objects. The difference picture region due to the occlusion of one or more moving objects by other moving objects are termed OC type regions. It is shown that the OC type regions can be indentified in the case of the running occlusion also.

A region in a difference picture is defined in [1-4] as a set of 4-connected points. It is observed that the regions formed by a set of 8-connected points are more robust in motion analysis. We present some examples supporting this fact. The results of the modified method for classification of the regions in the O, B, X and OC regions are presented.

Definitions

A frame is a two dimensional array of size M X N. All images, unless otherwise stated in this paper, have the same size. Consider an image A overlayed on another image B. The position of a segment S of the image A in the image B would mean the corresponding pixels in the image B which represent the segment in the image A. For brevity, when there is no ambiguity, we say "some pixels of segment S in B" in place of "some pixels from those pixels in B which correspond to the position of the segment S in A".

A Difference Picture (DP) is a binary picture generated by comparing two frames. The DP is generated by placing a '1' in those pixel positions for which the corresponding pixels in the two frames being compared have an appreciable difference in their grey level characteristics. The difference picture is usually prepared for two frames of the same dynamic scene taken at contiguous time instants. These frames will be called the previous and the current frames of the frame pair.

It should be mentioned here that in [1-4] original TV frames were condensed and then the difference picture was prepared using comparison based on the second order statistics. In this paper we present results of our experiments with computer generated frames of size 50 X 50. For comparing two frames we use gray levels of the corresponding pixels; if the gray levels of the corresponding pixels in the frames under comparison differ by more than 10 then the pixels are considered to be different.

A DP region is a set of 4-connected nonzero DP pixels containing at least 10 elements. A pixel is considered to be an edge point if the value of the Sobel operator at that point is above a given threshold.

A previous frame edge picture is a binary picture having a 1 entry in those pixel positions which are edge points in the difference picture and in the previous frame. Similarly a current frame edge picture has 1 entries in those pixel positions which are edge points in the difference picture and the current frame.

For the classification of the regions of a difference picture the ratio CURREF was defined as:

$$\text{CURREF} = \text{CC}/\text{CP}$$

Where CC(CP) is the number of points which are both extreme points of the DP region and edge points in the current (previous) frame. The extreme points of a region are the leftmost and rightmost 1 for a row and topmost and bottommost 1 for column. It has been shown [2] that for O,B, and X type regions the value of this ratio is greater than 1, less than 1, and about 1, respectively.

Displacement of a Straight Line

Let us consider a very simple situation; namely, the displacement of a straight line segment in a frame sequence. One can easily verify that a displacement of the line usually results in two lines in the difference picture. One line is at the previous frame position of the line segment and the other is at the current frame position. The lengths of the lines in the difference picture will be same as those of original lines. There is an exception to this fact, however. When the line is displaced parallel to itself, there will be still two line segments in the difference picture, but their lengths will be equal to the displacement. One line will be part of the line segment in the previous frame and the other line will be part of the line segment in the current frame. Note that if the line is displaced by more than its length along the direction then the lengths of the fragments will be equal to the length of the segment. A very important fact is that if a line is displaced in the direction of its orientation and the displacement is less than the length of the line, then DP has only fragments of the line in its current and previous positions.

Motion of a Homogeneous Segment

An image segment may be displaced, rotated, or changed in the size or shape due to the motion of the object resulting in the segment. Let us consider simple displacement. Due to the displacement, no new edges will be generated in the frame and there will be no change in the lengths of the edges.

The entries in the difference pictures may be obtained by marking all those points which are segment points in one frame and are not segment points in the other frame. A very interesting and useful fact is that all the regions in the difference picture are bounded by edges at those pixels which are edges either in the previous or in the current frame. Note, however, that for those points which are edge points in the previous as well as the current frame, there will be no edge point in the DP. This happens only when an edge is displaced in the direction of its orientation. A direct consequence of this fact is that in many cases an object may result in two regions in DP, one at the front end and the other at the rear end. This occurs when in the segment corresponding to the object there are at least two different edge segments parallel to the direction of the motion (see [4]). The region at the front end is due to the covering of the background by the image segment. The region at the rear is due to the uncovering of the background. Note that the region at the front will be bound on all but one sides by the edges in the current frame but on one side by edges in the previous frame. Note that for the object under consideration the extreme points of the region and the edge points are same.

When the motion is in the direction such that no two edge segments are parallel to the direction of the motion, the regions at the front and rear ends may not be clearly separated. Depending on the shape of the segment, they may be either 4-connected or 8-connected. The 4-connectivity has been used for defining a region [1-4], but it seems that 4-connectivity is very sensitive to shape and distance moved. This is illustrated in Fig. 1. Note that if the displacement of the object is such that the image in Fig. 1 is displaced by 2 pixels each to the south and the east then there are two 4-connected regions in the DP; if the displacement in these directions is 4 pixels each then there is one 4-connected region; and if the displacement in these directions is 6 pixels each then there are two 4-connected regions in the DP. In all these cases the DP has only one 8-connected region. This example illustrated that 8-connected regions are more consistent for motion analysis.

Note that when there is only one region due to the motion of the object the extreme points of the region no longer cover all the edge points. Many edge points may be lost i.e. may not be considered in the computation of the CURREF. Some of the lost edge points may be previous edge points and some may be current edge points. If the shape of the object is such that equal numbers of both edge points are lost then the result may not be affected much. The result would still classify the region an X type. In many situations (see Fig. 1e), however, this may not be true. If more previous frame or more current frame edge points are lost then the classification based on extreme points of a region may give the wrong classification. The ratio of current edge points to the previous edge points would classify the regions correctly. Thus we modify the ratio CURREF to be

$$CURREF = CEP/PEP$$

Where CEP(PEP) is the number of current (previous) frame edge points for the region.

Two Segments in a Frame

As the next step in the understanding of DP, let us consider a frame sequence containing two homogeneous segments. If the motion of the objects corresponding to the segment is such that the segments are in two different parts of the frames then each segment may be considered independently and all the facts about the DP discussed in preceding sections will hold.

As the first type of interaction between the segments consider Fig. 2. In the previous frame the objects A and B are such that the segments are separated by the background component. In the current frame, however, the object A is occluded by the object B resulting in the image segment shown in Fig. 5b. In the DP the regions at the occlusion end of the objects will merge to form a single region. In Fig. 2c both the regions at the occlusion end should have been type O, but the resulting region due to the merger will not be necessarily type O. The type of the region at the occlusion end is governed by several interacting factors, such as: shape of the objects, displacement of segments, distance between the segments in the previous frame, amount of occlusion. Depending on these factors, some current and/or previous frame edges be lost. The ratio, and hence the type of the region depends on these factors. A region at the occlusion end may have ratio CURREF anywhere between very small and very large, classifying a region in any category.

Observe, however, that the leading edges of both objects in the direction of motion in the previous frame will be part of the occluding region and these edges were disjoint in the previous frame. Thus for the occluding region in DP there will be at least two disjoint previous frame edge fragments. (An exception to this is when a bigger segment completely occludes a smaller segment.) This will result in two separate current frame fragments also. In the case of a single segment there are single current frame and previous frame edge fragments. Thus the presence of two or more fragments in a region indicates occlusion. It should be mentioned here that it is possible that one or more image segments may be displaced such that there is only one region due to the segment in the DP and this region is merged with the region due to the other object; in this case also the above observation is valid.

Running Occlusion

By running occlusion we mean that though there is no occlusion in the previous or the current frame, an object has occupied the position in the current frame which was occupied by other object in the previous frame. This is shown in Fig. 3. The regions at the rear end of A and the front end of B are not affected, but the regions at the front

end of A and the rear end of B merge to form a single region in the DP. In the merger some edges are lost. The lost edges are some previous frame edge fragments of B and some current frame edge fragments of A. This tends to make the class of the region random. Fortunately the observation made in case of occlusion is valid in case of running occlusion also. There will be two or more edge feagments of previous edges and of current edges. The feature which could help us in distinguishing running occlusion from the occlusion is the fact that the regions at the other ends of the objects, if any, are similar type in case of occlusion but are different type in case of running occlusion.

Classification

Based on the preceding discussion, a region of the DP may be classified using the following approach:

Find the DP and the current and previous frame edge pictures.

For a DP region find the number of disjoint current frame and previous frame edge fragments for the region.

If the number of current frame edge fragments or previous frame edge fragments is more than 1 then the region is OC type.

If the number of fragments is 1 each then compute the ratio CURREF. If the CURREF is more than $1+\alpha$ then the region is O type; if the CURREF is less than $1-\beta$ then the region is B type, and if the CURREF is between $1-\beta$ to $1+\alpha$ then the region is X type. In this paper we set values of α and β to 0.1.

Results

Figs. 1 through 3 show several frame pairs and their DPs. The classification approac described in the preceding section has been applied to the regions of these difference pictures and several other frame pairs. Fig. 1 shows a frame pair in which an object results in one DP region. The CURREF for the region in Fig. 1e is 1.016 and hence it is classified as a X type region. The CURREF using the old method is 0.828 and hence misclassifies the region as B type region.

In Fig. 2 the occlusion of the object results in the DP having three regions. The region Q has more than 1 edge fragment in previous and current frame edge pictures and hence is correctly classified as OC type. Regions P and R are classified as B type. This gives us the correct information that the DP is due to the occlusion. In the DP of Fig. 3 the region Q is classified as OC type and the region P and R are classified as B and O type, respectively. This information tells us that there is running occlusion in the fram

Conclusion

This paper presents a better method for the classification of regions in the difference picture. It is shown that the new definition reduces the possibility of misclassification of a region. 4-connectivity definition gives regions which are too sensitive to noise and coincidences. The 8-connectivity definition is more robust for the classification of regions.

A method is proposed for the recognition of regions due to the occlusion and the running occlusion. Our experiments with several computer generated sequences show that the methods proposed are robust. This demonstrates that even complex processes like occlusion can be analysed using only low level processing.

References

1. Jain, R., Militzer, D. and Nagel, H.-H., Separating nonstationary from stationary scene components in a sequence of real world TV images., Proceedings of IJCAI-77, pp 612-618, August 1977.

2. Jain, R. and Nagel, H.-H., On a motion analysis process for image sequences from real world scenes., Technical Report, Fachberreich Informatik, Universitaet Hamburg, April 1978.

3. Jain, R. and Nagel, H.-H., On the analysis of accumulative difference pictures from image sequences of real world scenes., IEEE Trans. On PAMI, Vol. 1, pp 208-214, April 1979.

4. Jain, R., Martin, W., and Aggarwal, J.K., Segmentation through the detection of changes due to motion., Computer Graphics and Image Processing, Vol. 11, pp 13-34, 1979.

5. Martin, W.M. and Aggarwal, J.K., Computer analysis of dynamic scenes containing curvilinear figures. Pattern Recognition, Vol. 11, No. 3, pp 169-178, 1979.

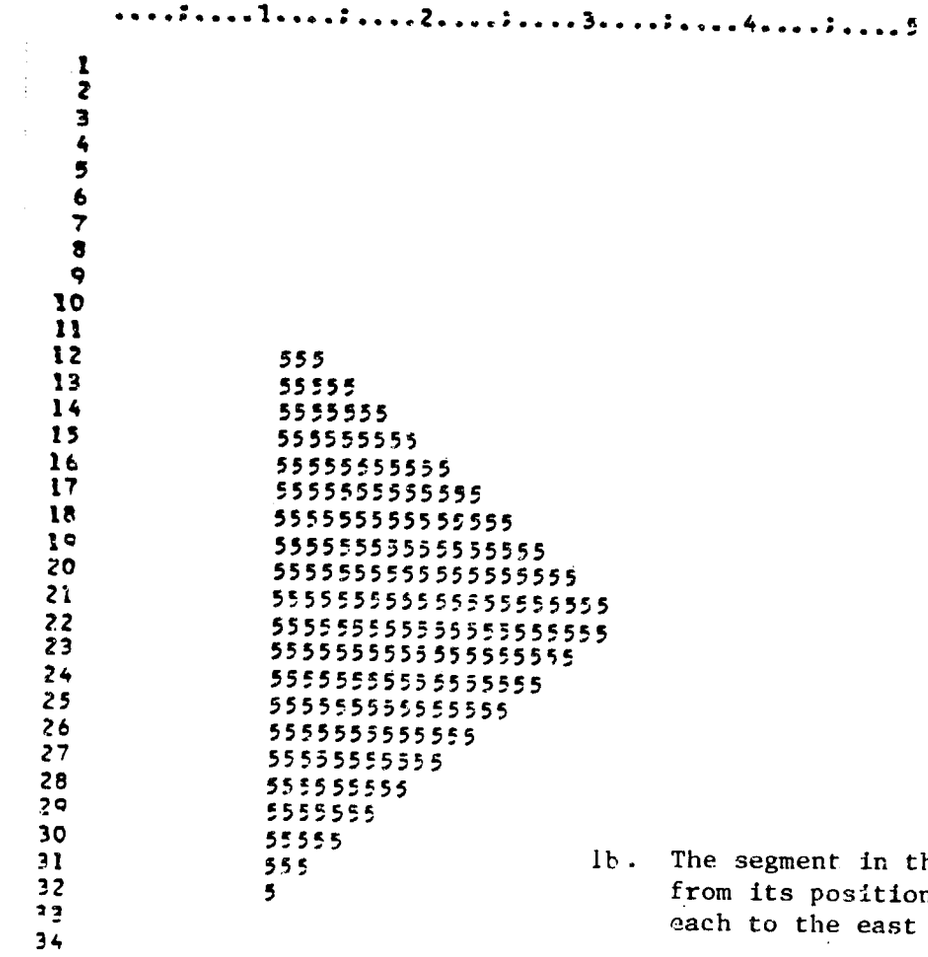
6. Nagel, H.-H., Analysis techniques for image sequences., IJCPR-78, Kyoto/ Japan, pp. Nov. 1978.

7. Workshop on Computer Analysis of Time Varying Imagery, Philadelphia, April 5-6, 1979.

: PREVIOUS FRAME IS

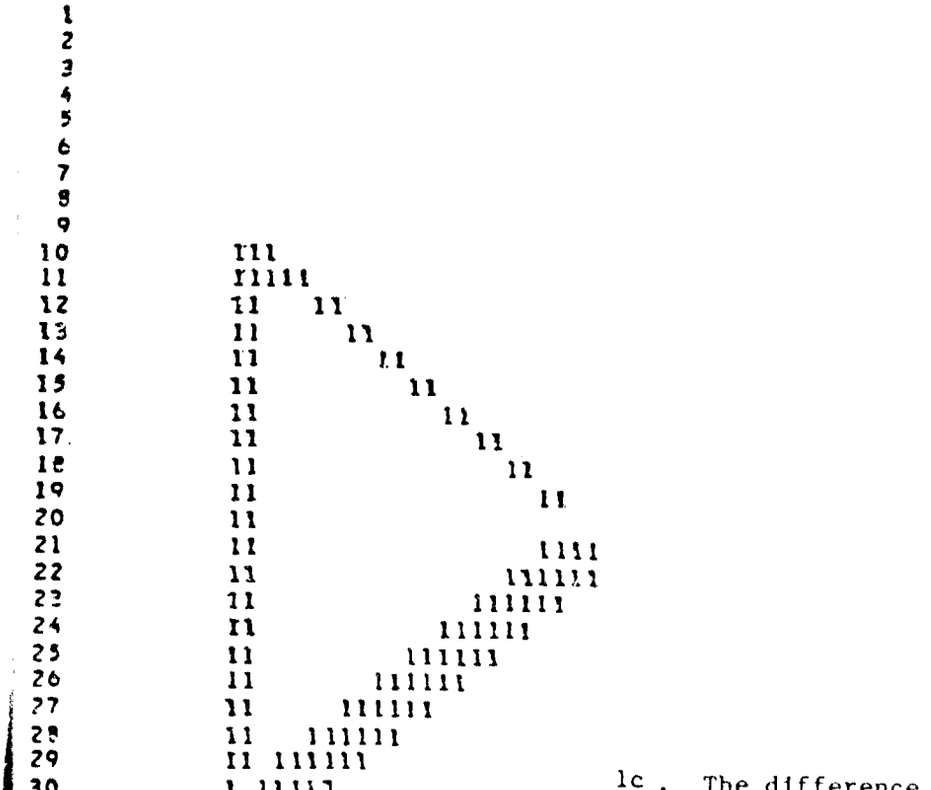
```
....:....1....:....2....:....3....:....4....:....5

 1
 2
 3
 4
 5
 6
 7
 8
 9
10          555
11          55555
12          5555555
13          555555555
14          55555555555
15          5555555555555
16          555555555555555
17          5555555555555555
18          55555555555555555
19          555555555555555555555
20          5555555555555555555555
21          555555555555555555555
22          55555555555555555
23          5555555555555555
24          555555555555
25          55555555555
26          555555555
27          5555555
28          55555
29          555                    1a. An image segment in the previous frame.
30          5
31
```

HE CURRENT FRAME IS

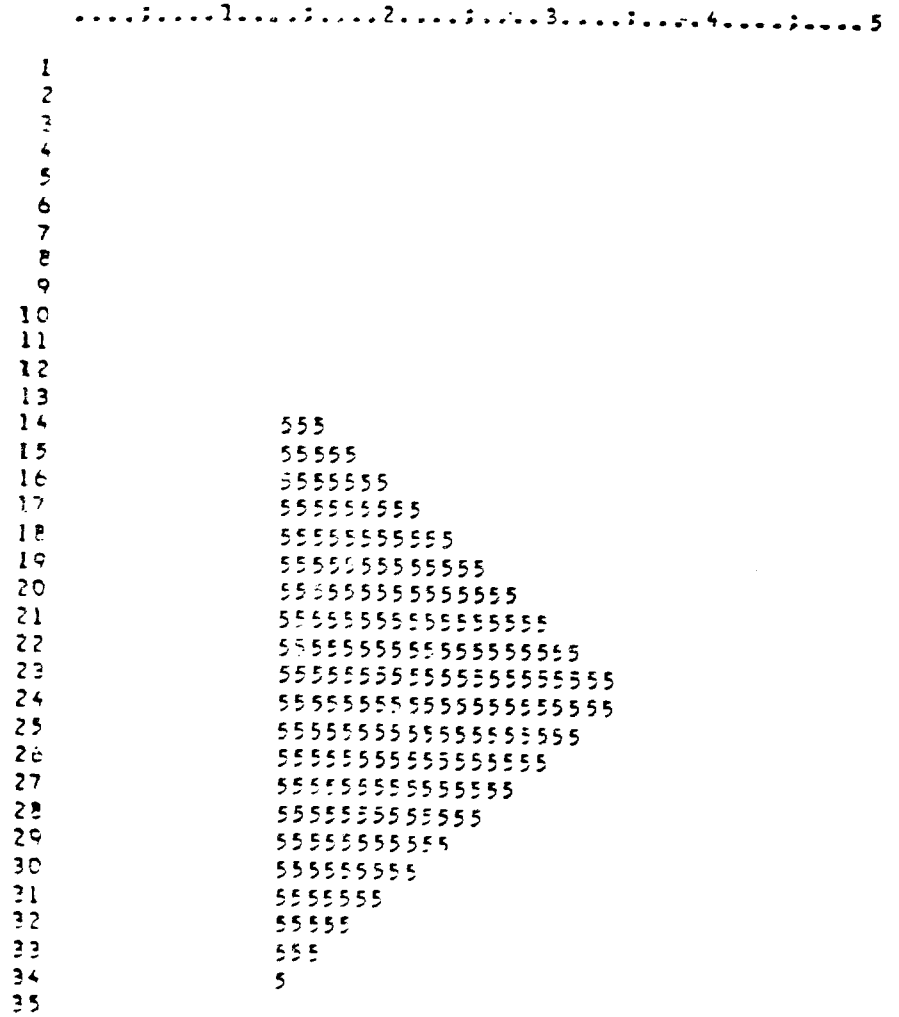


1b. The segment in the current frame has been displaced from its position of the previous frame by 2 pixels each to the east and the south.

IE DIFFERENCE PICTURE

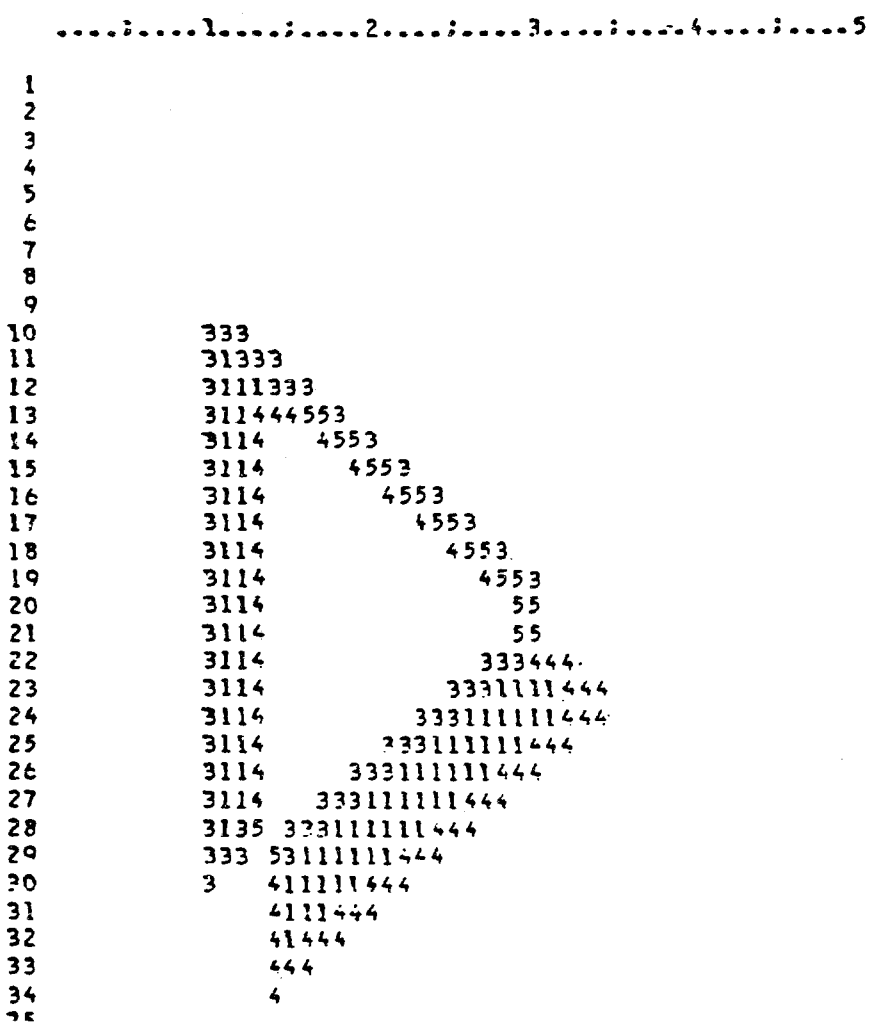


1c. The difference picture for 1a and 1b.

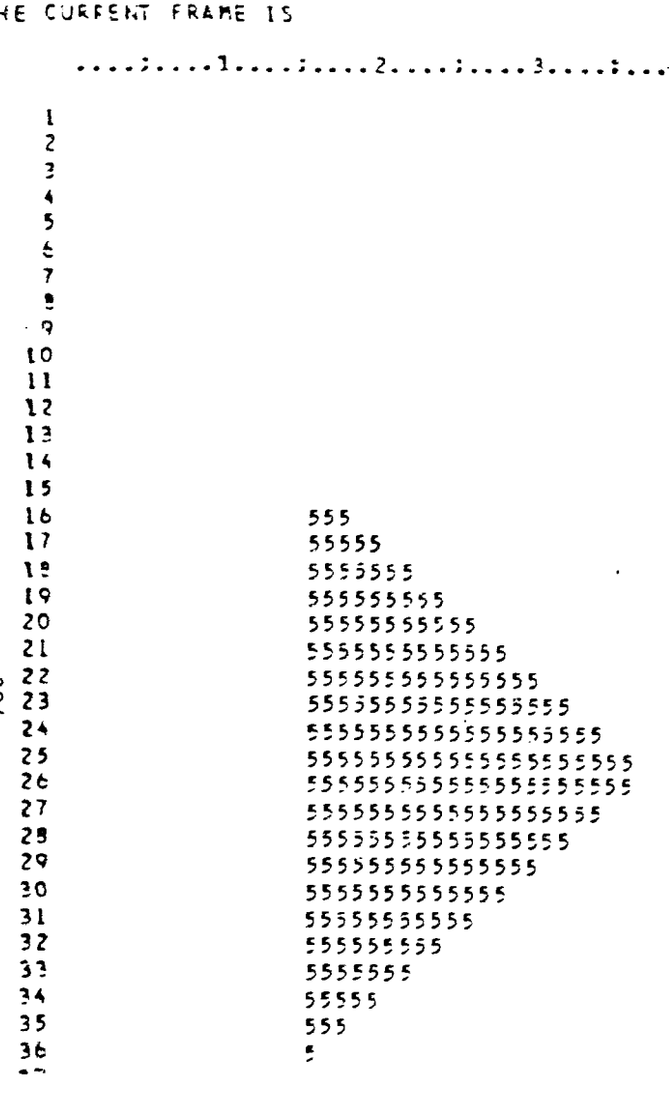


1d. The segment in this current frame has been displaced from its position of the previous frame by 4 pixels each to the east and the south.

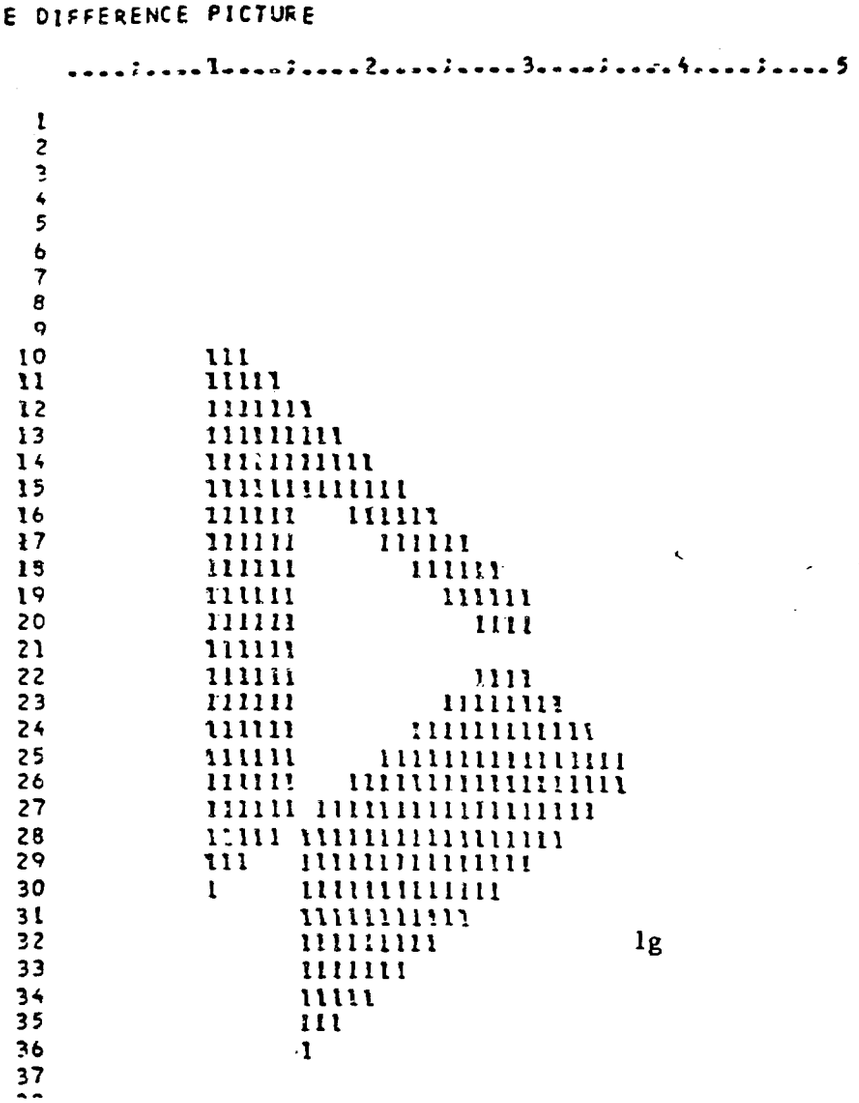
THE FINAL EDGE PICTURE IS



1e. The labelled difference picture for 1a and 1d. 1,2,3,4, and 5 represent difference entry, an edge point in the difference picture but not in previous and current frames, edge point in the difference and the previous frame, edge point in the pifference picture and the current frame, and edge point in the difference picture, the previous frame respectively.

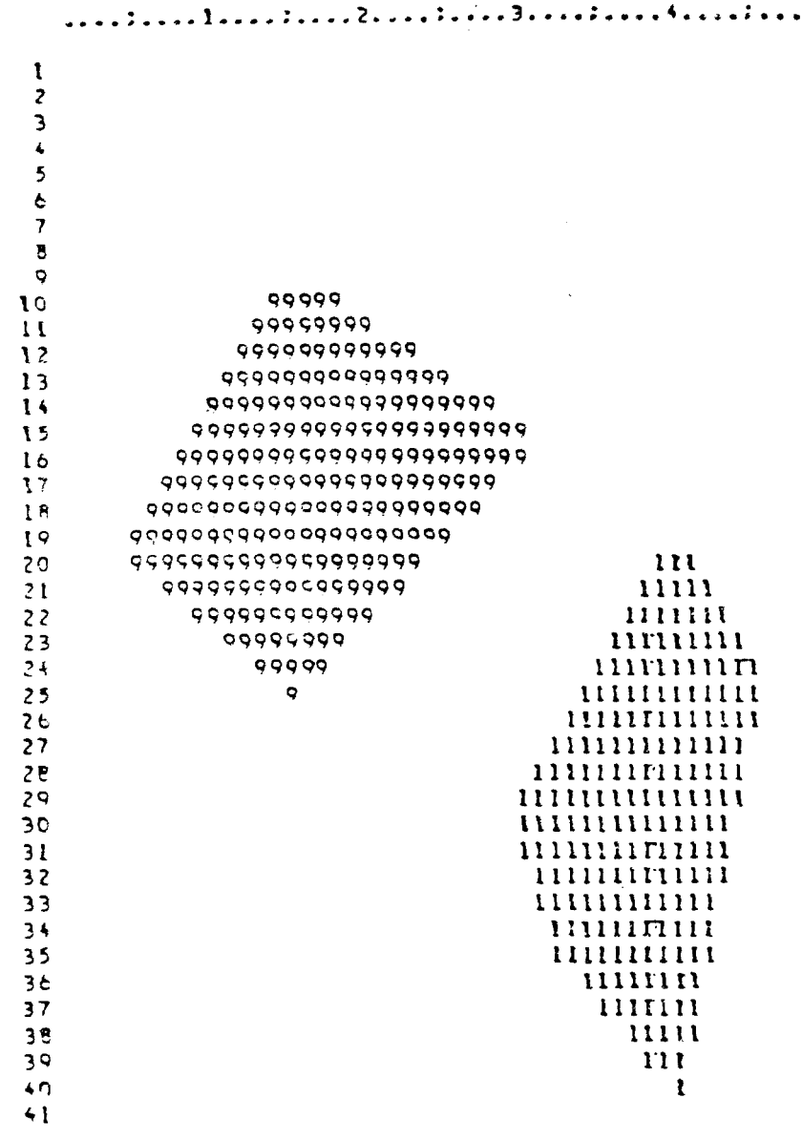


1f The image segment in this current frame has been displaced from its previous frame position by 6 pixels each to the east and the south.



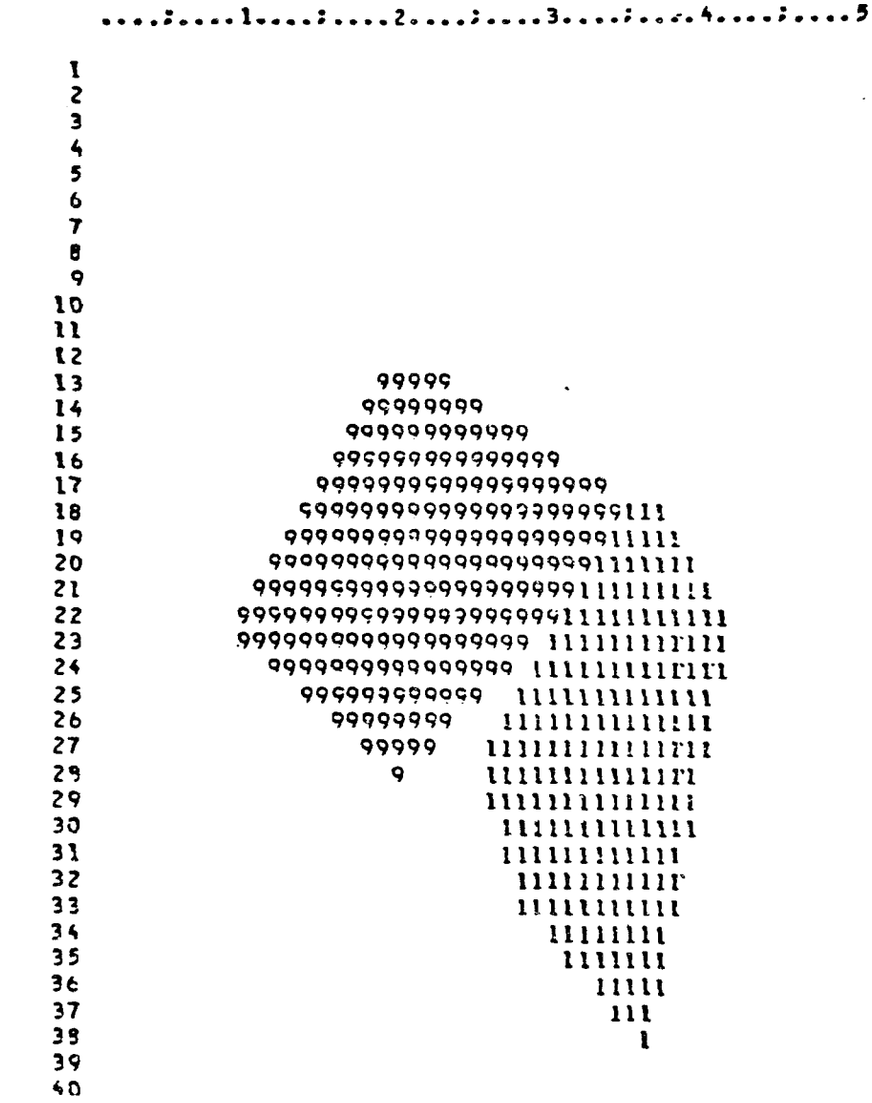
lg. The difference picture for 1a and 1f.

THE PREVIOUS FRAME IS



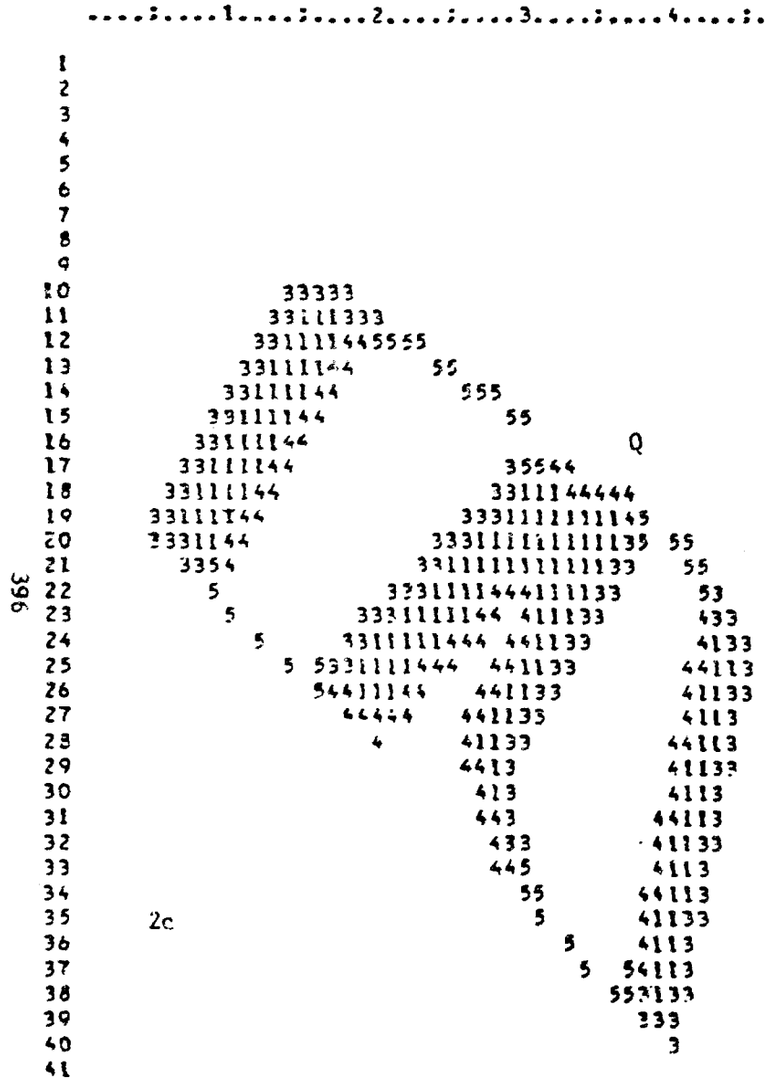
2a. The previous frame contains two image segments having gray levels 89 (labelled 9) and 41 (labelled 1).

THE CURRENT FRAME IS



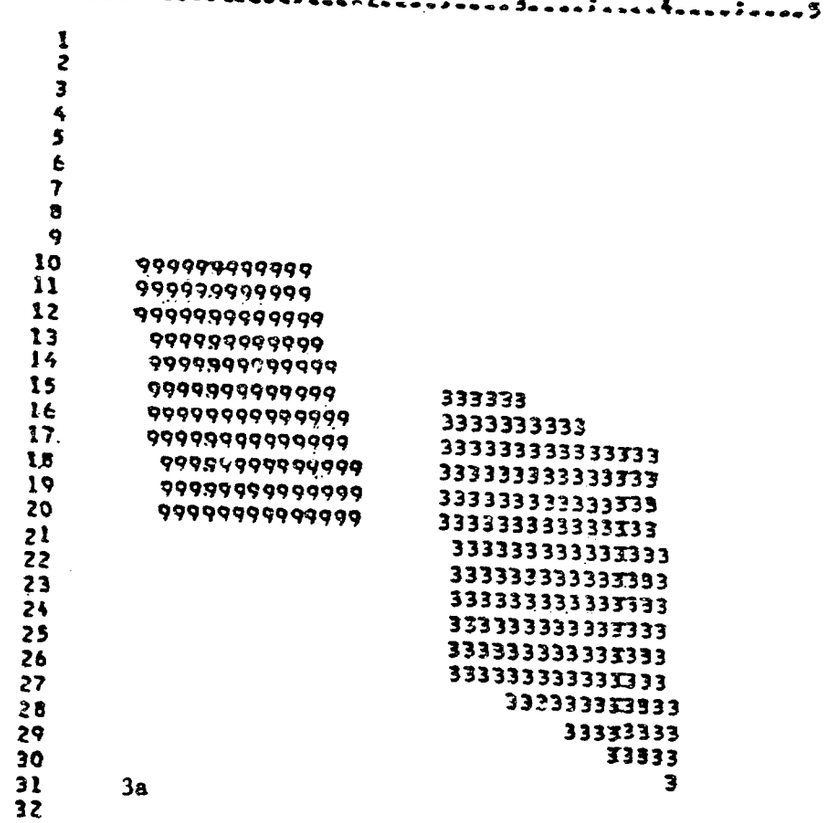
2b. The current frame showing displaced objects. Note that in the current frame there is occlusion.

THE FINAL EDGE PICTURE IS

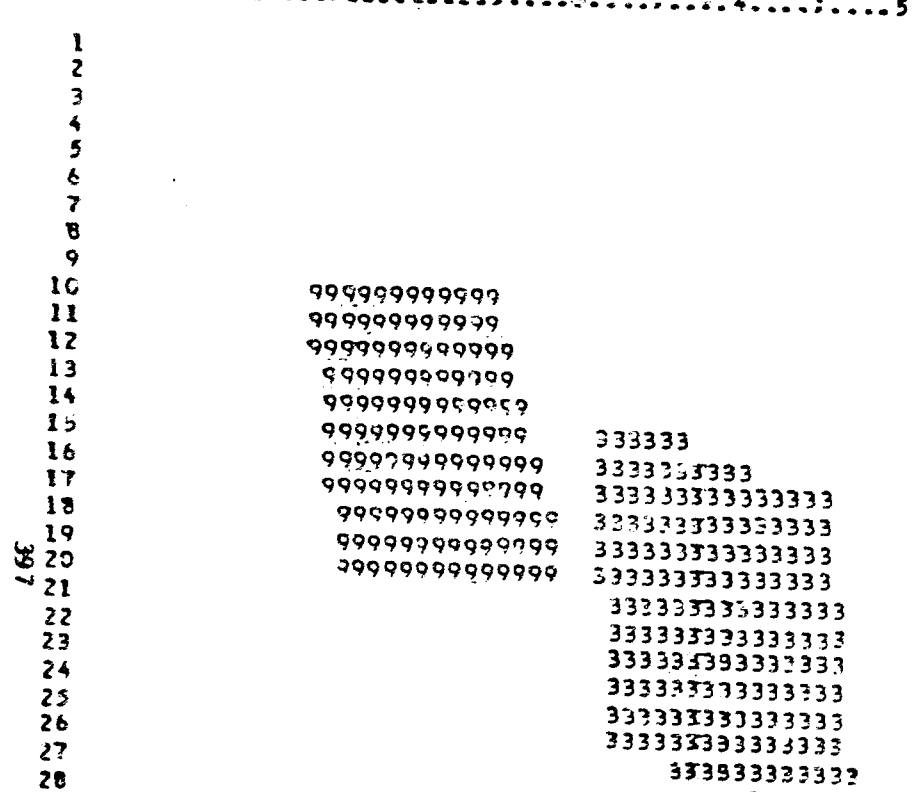


2c. The labelled difference picture for 2a and 2b. Note that there are 2 previous frame and two current frame fragements in the region Q.

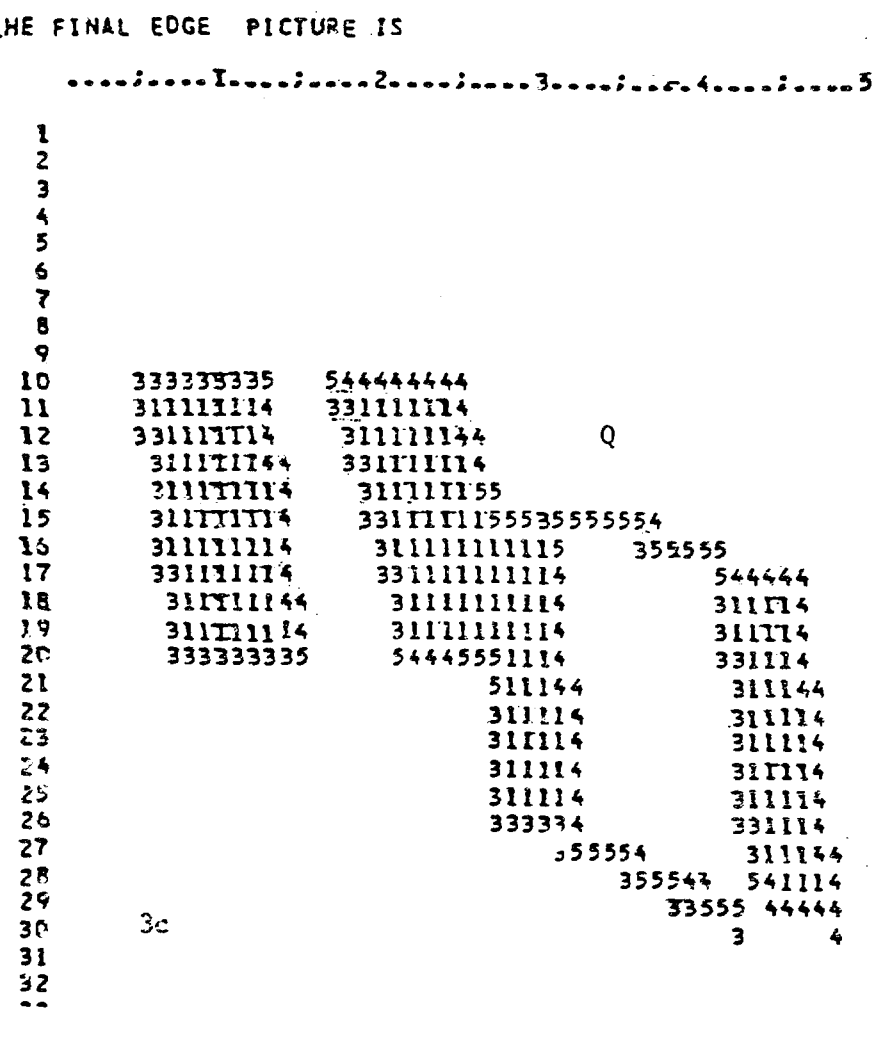
THE PREVIOUS FRAME IS



3a. The previous frame containing two image segments.



3b. The current frame shows the running occlusion.



3c. The labelled difference picture for 3a and 3b. Note several current frame and several previous frame edge fragments in the region Q.

Paper No. IIIB-5, Presented at the Workshop on Imaging Trackers and Autonomous Acquisition Applications for Missile Guidance, 19-20 November 1979, Redstone Arsenal, Alabama.

COMBINED ARMS FOR IMAGE UNDERSTANDING

Dr. John F. Lemmer

Pattern Analysis and Recognition Corporation
228 Liberty Plaza
Rome, New York 13440

ABSTRACT

The many approaches of Pattern Recognition and Artificial Intelligence to the image understanding problem are both complementary and overlapping. Various approaches are compared and contrasted. A Combined Approach for Research Methodologies and Systems (Combined ARMS) is proposed.

INTRODUCTION

The question, "Is image understanding Pattern Recognition (PR) or Artificial Intelligence (AI)?" continues to evoke emotion [4] It is the thesis of this paper that if image understanding is to offer real solutions to real problems, it must, as a minimum, use both the PR and AI approaches. In this paper we argue further that, even now, the major differences between some PR and some AI approaches lie largely in the point of view of the experimentor and in the experimental environment. We feel that researchers can profit from both points of view and that experimental environments fully supporting both approaches could accelerate the growth of image understanding. We also provide some concrete suggestions as to how the two points of view can be operationally merged.

To begin, we will show examples of Pattern Recognition (both decision theoretic and syntactic) and Artificial Intelligence applied to the same problem. The first pass through the examples will illustrate use of each technique in solving that part of an example problem to which it is best adapted. The second pass through the examples will show each technique extended to solve more of the overall problem. On this pass, it will be clear that each technique begins to borrow from the others. Finally we will suggest a method of combining the techniques which will hopefully preserve the best of each.

IMAGE UNDERSTANDING TECHNIQUES

A high level recognition problem is illustrated in Figure 1. Given an image as shown symbolically in the figure, the objective is to classify the cross hatched object as a dam. Decision Theoretic Pattern Recognition (DTPR) techniques seem especially well adapted to segmenting the image into regions of "land", "water" and "concrete." Syntactic Pattern Recognition (SynPR) and AI techniques seem best able to conclude, given the segmentation, that the concrete object is a dam. Why this is so will now be illustrated.

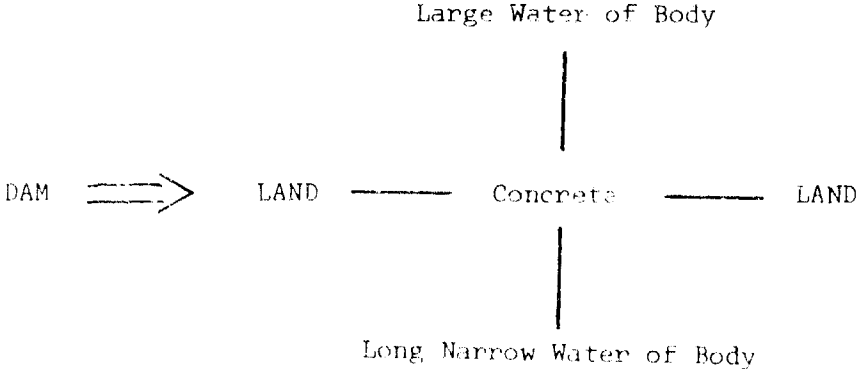
Decision Theoretic Pattern Recognition (DTPR)

At one level DTPR maps picture points (pixels) in image space into points in measurement space so that, hopefully, image points which ought to receive the same classification will cluster together when mapped into this space. This mapping and clustering is illustrated in Figure 2. In the figure it is assumed that two measurements have been made on each pixel: gray level and some texture feature termed "roughness."[3] These measurements will be sufficient to separate "water" pixels from "concrete" pixels if and only if such pixels map into effectively disjoint regions of measurement space. If the measurements achieve separation, then an image pixel can be classified according to the region into which it maps in measurement space. Thus, DTPR "classifiers" are nothing more than procedures for describing and determining the measurement space region into which image pixels fall. The major thrust of practical DTPR is not to produce clever classifiers, as many suppose, but to find appropriate measurements. If measurements do not cause like pixels to cluster, no classifier can do a good job. It is in the selection of measurements that problem domain knowledge is most often incorporated into DTPR.

Once pixels have been classified, regions can be found in image space in which neighboring pixels have received like classification. It is common, in DTPR, to then make measurements on these regions in order to attempt to find a higher level classification for the region.

Syntactic Pattern Recognition (SynPR)

Syntactic Pattern Recognition applied to image understanding attempts to express the (spatial) relationship among primitive entities in the form of a grammar.[5] For example, a grammar capable of classifying an object as a dam might include a production of the following form:



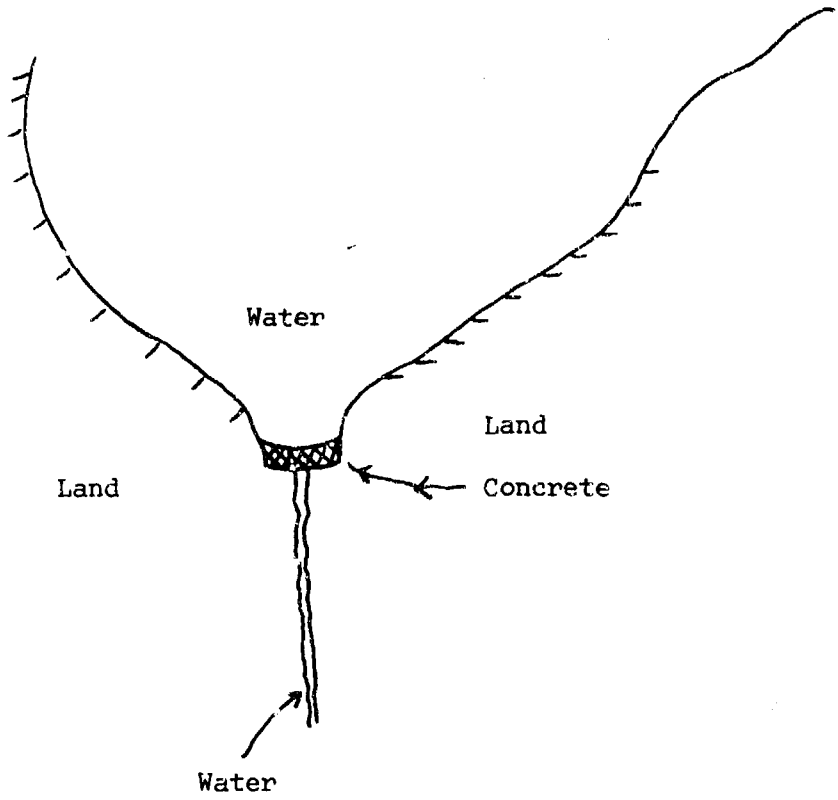


Figure 1 Example Problem

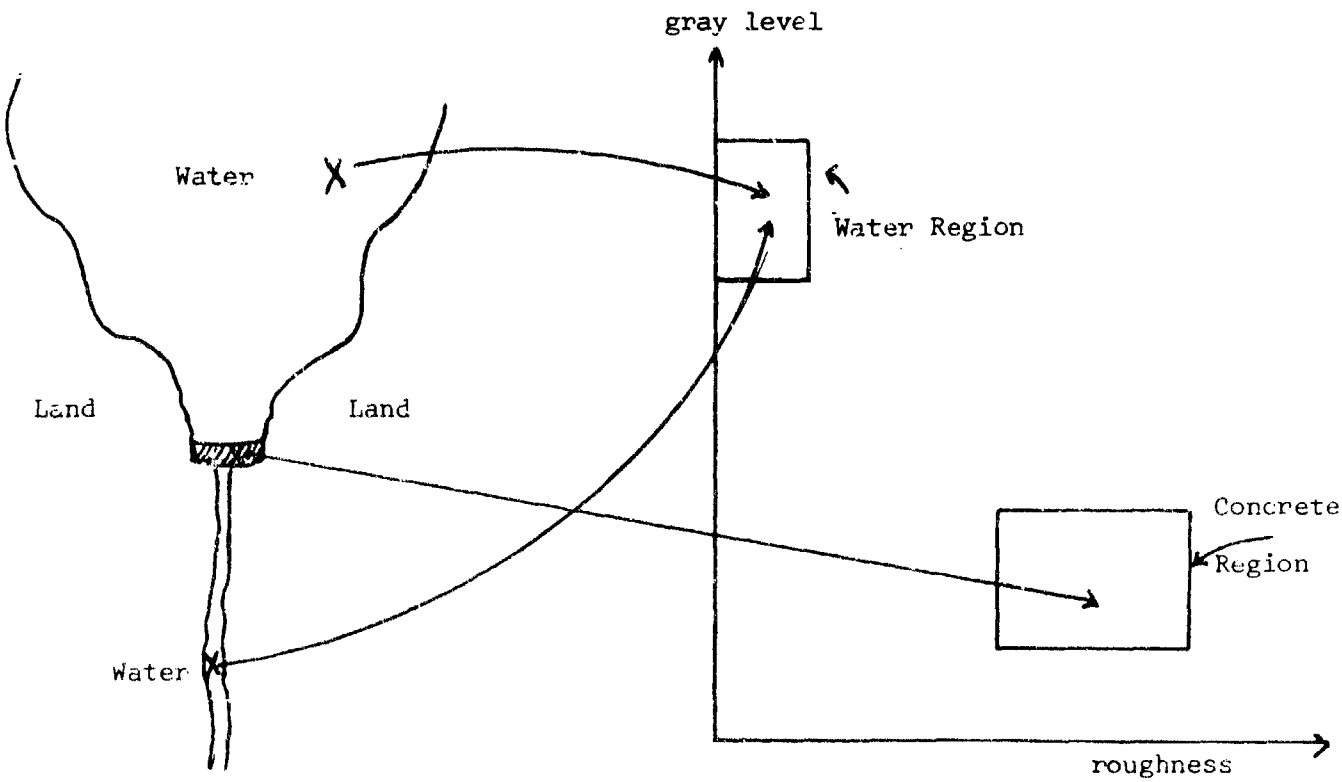


Figure 2 Mapping to Measurement Space

This production would explicitly represent the spatial relationship of contextual information implying that the concrete object was dam.

To classify the concrete object, the terminals (i.e. water, land) would be parsed. Note that it is not necessary that all terminals be recognized before parsing begins. This production coupled with parsing strategy could be utilized so that recognition of concrete and a large water body would, for example, trigger a search for a long narrow water body. It is possible that the grammar would also contain a production in which "DAM" could be further parsed as a "hydroelectric plant."

Artificial Intelligence

An AI "expert" system [2] might attempt to capture a human expert's knowledge in a form suitable for machine use. For example, an expert photo interpreter might say "a dam is usually a large concrete object with a large body of water on one side and stream or river on the other, etc." This knowledge could be represented by productions as shown in Figure 3. The nodes represent events. The directed edges imply directions of inference (and often, the opposite direction of cause effect). The weights, w_i, (probabilities in recent work [2]) represent the strength of the inferences which can be made from various events, Such expert knowledge is already available for some photointerpretation techniques, and available in a form almost ready for inclusion in an expert system. For example see [6].

Like the syntactic approach, knowledge in the production can be utilized to trigger searches for other events. Like the SynPR approach, the AI approach could have productions of <dam> and <quay> leading to some common events. Unlike the SynPR approach, no geometric relationship is implicit in the structure of the productions themselves. However, such information can, if desired, be included in the definition of the AI events.

SOLUTION OF THE "ENTIRE PROBLEM"

It has not yet been discussed how any approach solves the entire problem of going from pixels to "DAM." How does SynPR obtain its terminals? How does AI recognize events? How does DTPR conclude that the concrete object is a dam? Restated, no approach as described so far goes from pixels to high level symbolic representation.

Syntactic Pattern Recognition and Artificial Intelligence

A simplistic answer is obvious: DTPR is used as a terminal or event recognizer for SynPR or AI. Indeed if one looks at current work, especially in AI. one usually finds decision theoretic procedures implicitly present and buried deep in the system. These procedures, however, are generally handled in an ad hoc manner, hard to modify or isolate from the rest of the system.[1,8]

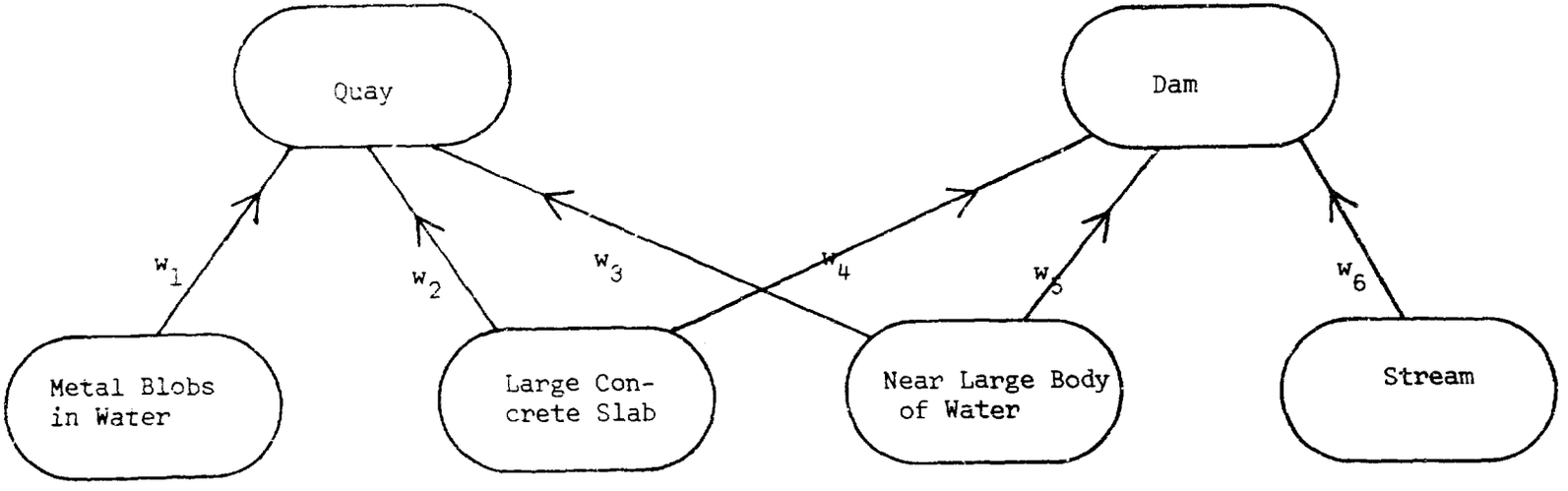


Figure 3 AI Approach

- Larger Feature Vector
  ‹GL, Roughness, L, W, WL1, WL2›
- Hierarchial Decision (Control of Measurement Process)

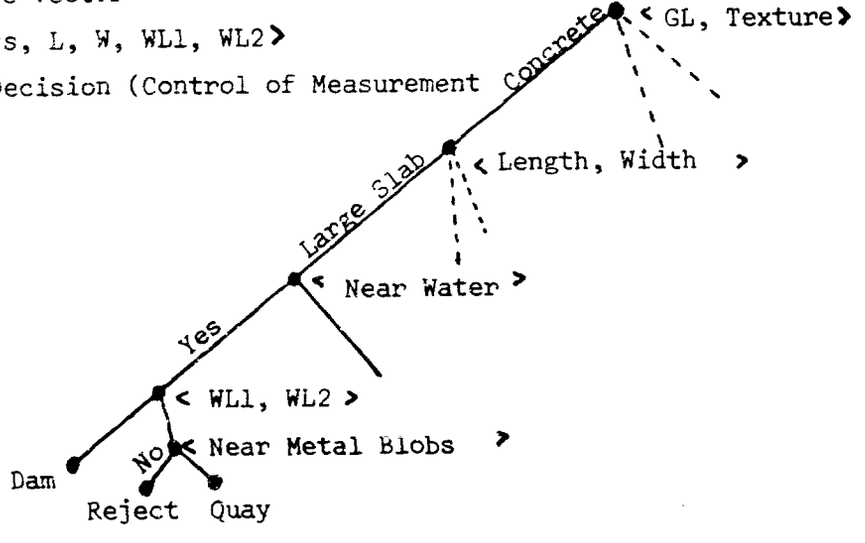


Figure 4 DTPR Extended

While such an approach to detecting primitive events might be adequate in an experimental environment, it is likely in a real environment that the primitive recognition must receive a great deal of attention. Proper selection of measurements and proper analysis of measurement variation will be key aspects of the ability to adequately recognize terminal or primitive events. It is at the measurement level that both sensor effects and distortions external to the sensor will have their greatest effects. It is unlikely that preprocessing (such as producing albedo images) will be able to account for all such effects.

Experience in practical application of DTPR shows that measurements selected according to some model of the sensor and environment almost always perform differently than expected and cause problems which effect the utility of various measurements. There is no substitute for the analysis of large amounts of real data. Indeed it is likely that there will be a feedback effect. SynPR terminals and AI events will dictate initial measurements, but the actual performance of the measurements on real images will undoubtedly at times suggest different terminals and events. Thus, one may conjecture that significant image understanding progress on real problems will require an experimental environment which provides tools for all approaches and minimizes the isolation of experimentors. We will return to this ideal later. Indeed, the environment should support a single person conducting all types of experiments.

Decision Theoretic Pattern Recognition

We have yet to answer how DTPR might classify the concrete object of our example as a dam. In answering the question we will uncover a surprising similarity between DTPR and AI expert systems.

In order to classify the concrete object as a dam DTPR would probably expand the dimensionality of the measurement space. For example, in addition to gray level (GL) and roughness, DTPR might expand to region analysis and include the length (L) and width (W) of the slab and the length of the two adjacent bodies of water, (WL1 & WL2), as shown in Figure 4.

Assume that the above measurements are adequate to have dams cluster in measurement space. Notice that the measurements selected imply a sequential order for making the measurements (L and W cannot be computed until pixels have been formed into a region). In practice, such sequential measurement extraction is greatly expanded upon so that a hierarchial decision tree is produced such as shown in Figure 4. In the decision tree approach, certain measurements are made only if certain results are obtained from previous measurements. Thus, DTPR, like SynPR and AI, has a control strategy. Indeed, there are many methods for optimizing the structure of the decision tree to control error rate, measurement cost and other properties. [7]

Operationally speaking it becomes difficult to distinguish DTPR using a decision tree from an expert AI system incorporating a control strategy. Both the DTPR approach and the AI approach will cause a sequence of measurements to be made until a classification is achieved. However, in the AI approach, the sequence of measurements is usually thought of as determined dynamically during classification while in DTPR the sequence is essentially determined before classification begins. Intuitively, in AI, the results of the measurements made so far, together with the data in the knowledge base itself are processed to determine what measurement should be made next. In DTPR, the next measurements to be made can be "looked up" in the decision tree.

Combined Image Understanding Systems

It may be argued that expert AI systems with a control strategy are more flexible, adaptable, and understandable than DTPR decision trees. It may be argued that DTPR with decision trees is more efficient than AI. A discussion of this point will lead to an idea of how a combined system might possess the best of both approaches.

Expert systems are more adaptable since they can, in general, compute the next measurement to be made given any sequence and value of previous measurements. Thus, as a new problem is encountered, a good sequence of measurements can be determined dynamically. Also, it is possible to alter the measurement sequence by introducing hypotheses about the classification. In terms of image understanding applied to map updating, the hypothesis might be formed based on old maps. In terms of image understanding applied to autonomous target acquisition, hypotheses might be formed from data acquired before launch from sensors not located in the autonomous acquisition system itself.

Presumably, however, if the same classification problem is to be solved repeatedly using the same type of data, the control portion of the expert system would request nearly identical sequences of measurements. It is unlikely that the dynamic control process would optimize the sequence of measurements to the same degree that could be done "off line" by techniques applied to DTPR decision trees. Thus it would make sense to have repetitive problems (and those of autonomous acquisition or of mapping are likely to be repetitive) initially analyzed by an expert system and optimized and studied for efficiency by a DTPR system. A structure for such a combined system is shown in Figure 5.

The expert input shown in Figure 5 may come from a "problem domain" expert who knows little or nothing of computer procedures for classification. For example, in the autonomous acquisition problem, he might be an expert photo interpreter. The measurement expert input comes from someone who has broad knowledge concerning the raw data from which classification is to begin. The feedback loop is present since it may not be feasible to recognize some of the primitive events specified by the problem domain expert. The problem statement indicated in the figure can be considered as some hypothesis which will recur frequently.

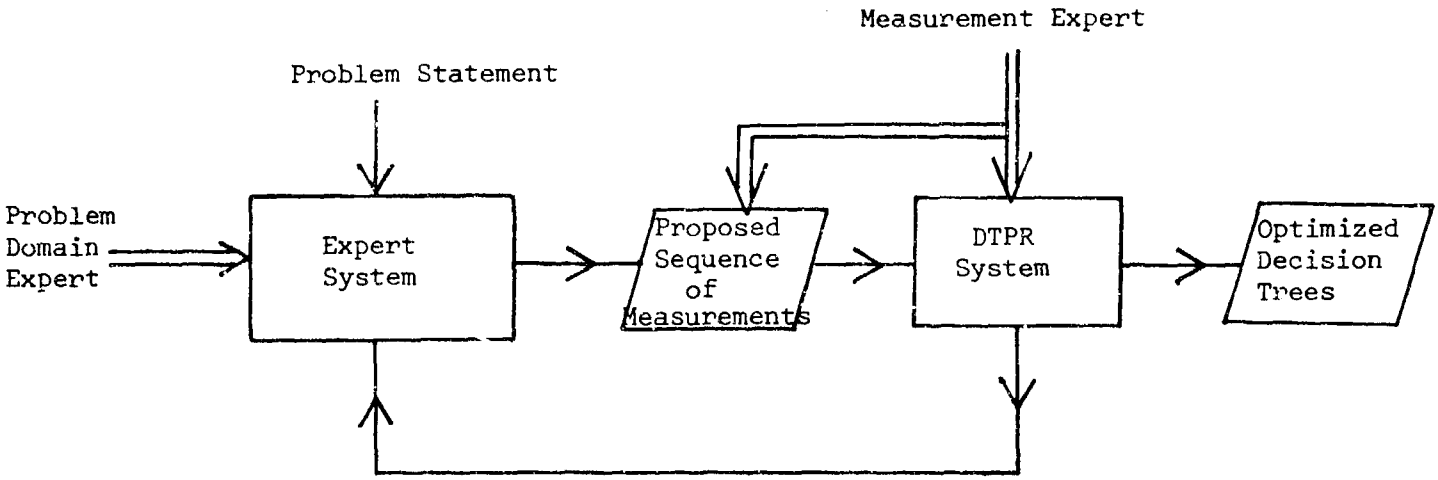


Figure 5: Combined System

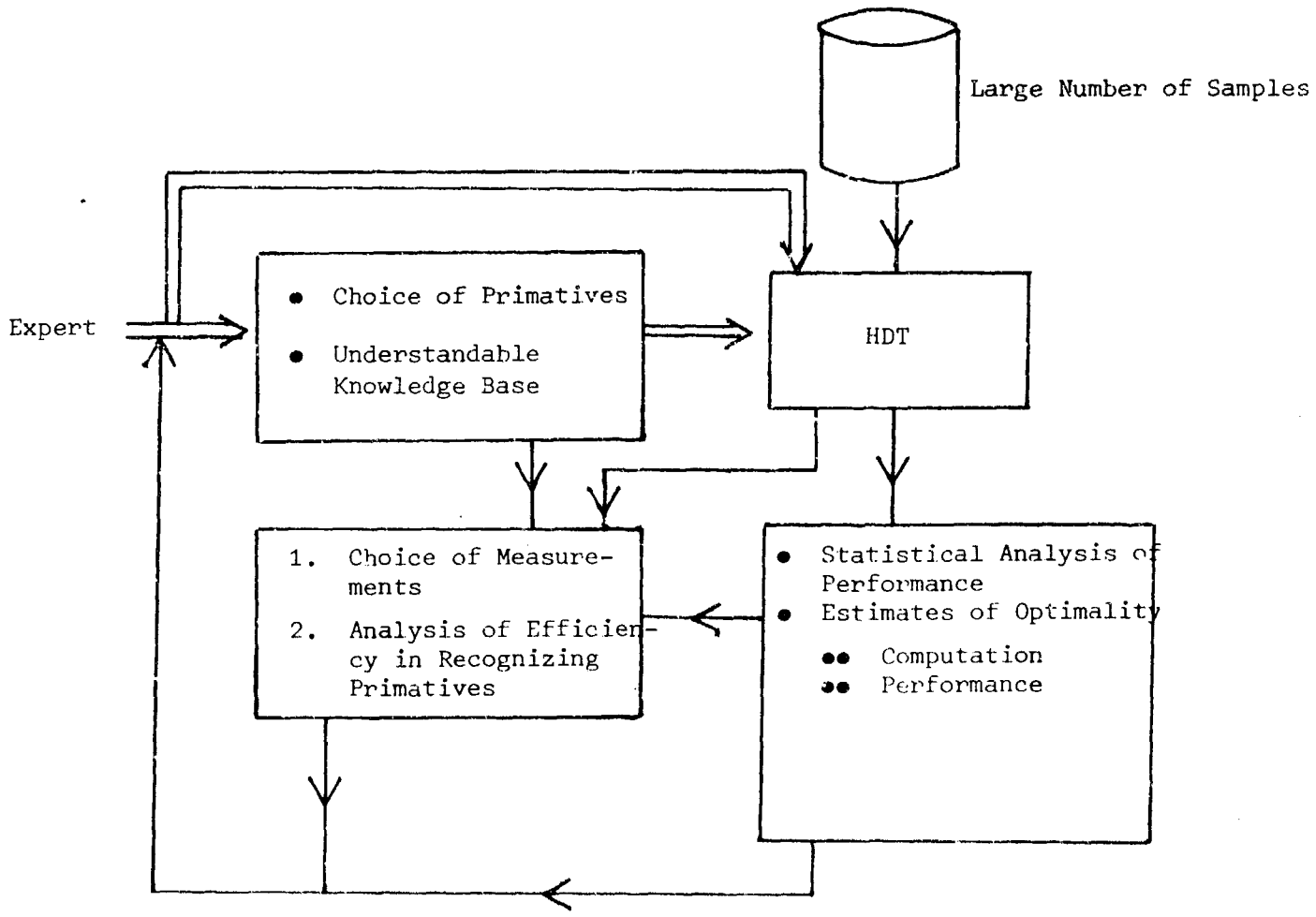


Figure 6: Combining Points of View and Environments

SUMMARY

The above discussions have focused on overall similarities of various approaches to image understanding without glossing over the differences. Indeed, ways of capitalizing on the differences have been suggested. Table 1 summarizes the similarities of the various approaches. Table 2 lists differences in three major areas, use of expert knowledge, use of image context, and methods for controlling the classification process.

The environments in which PR and AI experimentation are carried out are often quite different. Table 3 attempts to highlight a number of these differences. We feel that it is quite possible that performance on real image understanding problems could be enhanced by combining the two environments. Such a combination might result in an experimental flow as shown in Figure 6. Figure 6 represents the same flow as Figure 5 but highlights different aspects.

We feel that the major approaches to image understanding are both complementary and overlapping. We feel the time is at hand for practical image understanding to be implemented from a synergistic combination of techniques. We feel that the exploration of such synergism should take place in a common environment conducive to all points of view, utilizing the strong points and compensating for the weaknesses of each approach. The environment should be one in which no technique is presumed, a priori, to be superior to another. Finally, it should be an environment which encourages research in how real images can be practically understood, using humans for some tasks, if required. It should not be a system for investigating how ideal images ought to be classified.

ACKNOWLEDGEMENTS

My thanks to Jim Cambier and Mike Zoracki for their comments and suggestions on the original drafts, and to Nancy DeGeorgio for preparing the drafts and manuscript.

All potentially use expert knowledge from problem domain but

- differ in "other" knowledge experts must possess to develop the knowledge
- differ in "explanatory" capability

All can use context but differ in "naturalness" of structuring

All offer a control scheme but differ in efficiency, optimality, and adaptability

All rely on basic image measurements but

- differ in how measurements are selected
- how well performance is analyzed

Table 1 Similarities

	DTPR	SYNPR	AI
Use of Knowledge	• Choice of Measuremets • Structuring Decision Tree	Structure of Grammar	Inherent in Production System
Context	• Elements in Feature Vector or • Decision Criterion at Node in Decision Tree	Inherent in Structure of Grammar	Spatial Relations Between Segments can be included in events in production system
Control	• Implicit in Decision Tree: • Optimized "Off Line"	Inherent in Parsing Strategy	Dynamic Choice of Next Action Supports Dynamic Hypothesis Formation and Explanations

Table 2 Differences

PR

- Extensive analysis of measurement quality and variability
- Testing on large numbers of cases
- Statistical analysis of performance
- Vary measurements to improve results
- Parameters learned from samples

AI

- Extensive interaction with "Pure Expert"
- Analysis of performanc includes understandability
- Vary knowledge base to improve performance
- Parameters estimated by experts

Table 3 Different Environments

Paper No. IIIB-6, Presented at the Workshop on Imaging Trackers and Autonomous Acquisition Applications for Missile Guidance, 19-20 November 1979, Redstone Arsenal, Alabama.

PROTOTYPE AUTOMATIC TARGET SCREENER* GOALS AND IMPLEMENTATION

By

D. E. Soland, D. V. Serreyn, R. C. Fitch, T. G. Kopet
G. O. Prom, R. C. Reitan, M. O. Schroeder

HONEYWELL, INC.
Systems and Research Center
2600 Ridgway Parkway
Minneapolis, Minnesota 55413

ABSTRACT

The Prototype Automatic Target Screener (PATS) is being developed at Honeywell under contract with the Army Night Vision and Electro-Optical Lab. The system consists of hardware for image enhancement to improve the imagery displayed to the operator of a FLIR and hardware and software for real time detection, recognition and cueing for selected tactical targets.

The PATS system will operate with standard 525 and 875 line TV formats. Decisions on target classification and location are updated every 1/10 second. The resultant decision is displayed by means of symbology overlays on the video display to the operator.

The hardware consists of twenty-two 6" x 9" boards featuring charge coupled devices to perform the high speed functions for image segmentation. It incorporates a bit-slice microprogrammable digital processor for classification as well as a bit plane structure frame memory. The hardware fits into a box slightly larger than an ATR box and dissipates approximately 200 watts. Each board is somewhat modular in function and boards of the same function could be easily substituted.

*The work leading to this paper was supported in part by the U.S. Army Night Vision and Electro-Optical Laboratory Contract DAAK70-77-C-0248.

INTRODUCTION

Under contract with the Army Night Vision and Electro-Optical Lab, Honeywell has simulated, designed, and is in the process of fabricating and testing a Prototype Automatic Target Screener (PATS). This program started in late 1977 and will culminate with ground and flight testing in early 1980. The system is designed to interface with a Common Module FLIR. The PATS system will automatically detect and recognize targets and cue the FLIR operator. This paper discusses the goals of the PATS program and then the actual implementation of the target screener. Additional information about the simulation were presented at the April 1979 SPIE Conference[1].

Image Enhancement Goals

Three image enhancement functions are to be provided as part of PATS. These are: (1) adaptive constrast enhancement, (2) DC restore for AC coupled detector systems, and (3) automatic global gain and bias controls. The performance goals are such that: (1) local area control should not exceed 1 percent of the total scene (~2,500 pixels) and (2) the MRT degradation should be less than 10 percent. The synthetic DC restore will restore to the displayed image a proportion of the DC or background component of the scene and eliminate the streaking or overshoot effects commonly associated with AC coupling. This will be accomplished such that the normalized mean square error on two specified test patterns will be less than 20 percent. The two test patterns are alternating horizontal black and white bars and a black and white diagonal target.

Target Screener Goals

The target screener is designed to operate with any RS-343 standard 875 line video as well as RS-170 standard 525 line video. Both operate at a 60 Hz field rate. The specific FLIR system for testing will be a Lohtads Common Module FLIR. For image enhancement, the system must process every frame but the target screening function is required to process 10 frames/second or every third frame. For each processed frame a minimum of ten objects must be processed.

The target screener shall be capable of classifying extracted candidate targets into one of five classes. These five target classes are:

- 2½ ton truck
- tank
- armored personnel carrier

- track mounted radar-controlled anti-aircraft
- track mounted anti-aircraft missile launcher

The target classes can easily be changed through software. The recognition capability goal is at ranges where the probability of human detection is 90 percent; the probability of recognition by the screener shall be 80 percent. The target screener is to work from the 90 percent detection range to 1/3 that range. Originally, range measurements were excluded, but the PATS system is being modified to be tested with and without measured range. In Lohtads, range is measured by a laser.

The average false alarm rate shall not exceed one per frame processed normally. There is also a priority search mode where only the last two target classes are the targets to be cued. The objective with this mode will be lower false alarm rate. Specifically, the goal is one false alarm per 200 frames.

PATS System Design

The PATS system has several functions which it must perform on video data. These functions are shown in Figure 1. The first thing that is performed is image enhancement. This function is primarily for the displayed imagery but also may aid the target screening function. The rest of the functions shown in Figure 1 relate to the target screening function. Image segmentation must first be done to outline regions or objects of interest. Once the objects have been segmented, certain features must be measured which are used for initial recognition or classification within the frame. All objects in a frame are classified as to clutter or type of target.

The object classification for each frame is accumulated over a series of frames. When confidence is high enough that the decision is a specific target, a symbol indicating the classification is displayed. The sequential frame classification is called interframe analysis and is used to reduce the false alarm rate.

In the ensuing paragraphs of this paper we will discuss the implementation of these functions. Figure 2 shows the hardware configuration modules for each function.

The first module is the sync and timing generation. This module consists of two boards--one for sync separation and video switching and one for system timing. The sync separation and video switching is shown in Figure 3. The sync separator extracts the composite sync from the video. From the composite sync signal, basic sync signals such as vertical reset, field indica-

tor and horizontal sync are derived. Since the video is AC coupled, the video data must be black clamped.

Another function performed by this board is the multiplexing of video signals. Video to the monitor can either be raw video, enhanced video, analog test signals, or digital test signals. Similarly, the video to the target screening functions may be either enhanced video or raw video.

The second board is the system timing generator shown in Figure 4. This board produces sync signals and clocks that are synchronized with the incoming video. Two clocks are produced--a 455 clock and a 512 clock. The 455 clock is used for the analog CCD shift registers in PATS while the 512 clocks are used for the digital hardware. The signals generated by the board are commonly available in single chip form for 525 line commerical television but not for 875 line. For this reason, the function had to be built from MSI chips. The line rate must be manually set to agree with the FLIR configuration. For the 875 line format, we have 512 or 455 samples per 32 microseconds whereas with the 525 line rate we have 512 or 455 samples per 53 microseconds. The number of samples per line is considered sufficient for the current system requirements but can be increased if necessary.

The image enhancement function shown in Figure 2 consists of two boards which perform synthetic DC restoration, global gain and bias control and adaptive constrast enhancement. These are implemented with charge coupled devices and a microprocessor among other standard MSI and LSI parts. The particular functions and implementations are discussed in the references[2] and will not be repeated here.

The feature extraction function shown in Figure 2 consists of autothreshold hardware, interval and first level feature hardware. The autothreshold hardware (Figure 5) consists of two analog processing boards which do intensity thresholding and edge derivation. Data is compared to a calculated adaptive threshold and a digital output is produced.

One of the functions performed by the autothreshold hardware is the generation of "hot" and "cold" signals. "Hot" data consists of those values above the background by a specific amount whereas "cold" is data below the background by a specified amount. The background filter is a two-dimensional low-pass recursive filter which operates at video rates. The threshold is computed from the video after the background estimate is subtracted. The threshold is based upon the variance of the video. The threshold value is multiplied by a predetermined constant to provide the video comparison. Exceedance of the threshold produces a logical true signal.

The second function performed by the autothreshold boards is the generation of edges for objects in the scene. The basic edge computation is a two-dimensional horizontal component. The edge threshold is based upon the scan line average of the analog edge. After the threshold computation and edge comparison, a logical edge signal is produced indicating the presence of an edge at that particular location in the video.

Digital signals from these two boards (edge, hot, and cold) is input to the interval boards shown in Figure 6. The PATS interval circuits include the implementation in bipolar TTL logic of a number of functions.

- Generation of an interval based upon previous, present, and next scan line edge, hot and cold signals.
- Validation of an interval as meeting certain practical constraints.
- Storage and generation of key interval-related data.
- Making interval data available to the processor memory and informing it that valid data is ready.

Interval generation is based upon the presence of a hot or cold signal in coincidence with an edge. Without the presence of an edge the data is invalid. Line delays are provided by digital shift registers and some of the logic is implemented in PROMS. As a result of this hardware the following first level feature data are stored in latches:

- Line number or Y position
- Number of intervals for each line
- Starting X position for each interval
- Width of each interval
- Background estimate at the start of each interval
- Sum of the intensities within each interval
- The bright count within each interval
- Indication of when the edge associated with the interval was located (start or end or both)
- Indication of interval as being either "hot" or "cold"

At the end of each line, the number of intervals and the Y position are transferred. At the end of each valid interval, data associated with that interval is transferred.

The next function in Figure 2 is the data memory. This memory stores up to ∿2,500 sets of interval data. This is a static RAM memory and data is transferred to it via DMA control. Data from the interval board are dumped into a FIFO shift register. A maximum of 21 intervals per line can be stored. This data is later transferred into a second FIFO and then finally into the data memory.

The data memory is part of the subsystem called CPU1. The CPU1 system architecture is shown in Figure 7. The basic processing unit consists of a high speed microprogrammed data processing register and arithmetic logic unit (4-2903), a 16 x 16 high speed multiplier, a microprogram sequencer and an addressing register and arithmetic logic unit (4-2901). The microprogram is stored in high speed RAMs during checkout and debug stage and will be transferred to PROMs for testing.

The CPU1 is interfaced to the various memories via the data and address bus and also to an external computer for debug and checkout. The second computer, A DFC LSI 11/2, is not used during the actual operational mode of the PATS hardware. It is, however, used as part of the training of the hardware.

Both CPU1 and CPU2 have access to the symbol generation hardware shown in Figure 8. Only the CPU2 connection is shown. Symbols are generated by writing vectors into a graphics bit plane. The data is read out of the bit plane at video rates. This is accomplished by using a parallel to serial converter on the output and addressing only every eight pixels. The displayed video is replaced by the symbol. The symbol size and shape is programmable.

Once a target is detected the necessary data for symbol generation are X, Y position within the frame, target classification and target size. As the target moves, the displayed symbol is erased and a new symbol is generated. As the target gets larger, so does the symbol.

In the lower portion of Figure 2 is shown an A/D conversion block. There are two A/D converters in the PATS converter unit shown in Figure 9. One A/D determines the digital value of the background estimate at the beginning of each interval. The second A/D is used for digitizing the entire video frame. Also included on the board are provisions for testing the A/D and testing the frame store memory. Both converters are 8 bit high speed TRW A/D converter chips.

An additional item included as part of the A/D converter unit is a 12 bit summer. At the start of an interval, the summer is cleared, and data is then summed over the entire width of the interval. This gives the sum of intensities from which the average intensity can be calculated. Only the eight most significant bits of the sum are transferred to CPU1 for processing.

The digital data coming from the second A/D converter are transferred to the Memory #2 (frame store) at video rates. This memory is made up of memory bit planes as shown in Figure 10. Each bit from the A/D converter goes to a designated bit plane. The bits are shifted into a serial shift register and on every eighth pixel, data is transferred into the actual memory chips. Eight 16K x 1 dynamic RAMS with access times of 375 nsec are used for the memory. This data can be randomly accessed by CPU1 for calculations necessary to do the recognition and classification of targets.

Software

Much of the processing for detection and recognition of targets is done in CPU1. In Figure 11, the software functions are shown. The software sequence is:

- Bin matching
- Median filter
- Object feature generation for clutter removal
- Clutter recognition
- Recognition features
- Classification
- Interframe analysis
- Symbol generation

All processing in CPU1 is completed in 0.1 second.

Bin matching associates the one-dimensional intervals characterized by the interval features into sets of intervals which determine two-dimensional objects. That is, CPU1 reads interval data from the memory, reorders them and then writes them back. The matches of intervals on a scan line by scan line basis is primarily determined by the location of each interval within its scan line.

Once a bin is complete, i.e., no more intervals match, CPU1 will smooth the boundaries of each bin using a one-dimensional median filter of width five. The values which are input to the filters are the endpoints of the intervals making up each bin. A separate filtering operation is done on the left and right hand edges of each object.

Object features are computed by CPU1 on the median filtered bins in a hierarchial fashion. That is, less expensive features are computed initially to do preliminary clutter screening and more expensive features are computed on the unrejected objects in order to do additional clutter screening and object recognition.

The classification algorithm is the k-nearest neighbor. The recognition classifier puts each active object bin into one of five target categories using moment features for that object and stores that classification together with the object size and location. The data is then processed by the interframe analysis.

The interframe analysis associates objects between frames. Using a Baye's decision, the object is classified and a symbol is generated. The symbol generated is directly related to the classification derived.

The second CPU can do interframe analysis and symbol generation. This allows one to check out the object matching. CPU1 will be doing this in the hardware to be delivered. CPU2 is still needed for training and diagnostics. Figure 12 shows the functions that CPU2 can provide. It has the capability of dumping data to or reading data from the two memories, and is used to gather test data.

PATS Physical Characteristics

The PATS hardware presently consists of twenty-two 6" x 9" boards that fit into a chassis that is slightly larger than an ATR box. Much of the space is used for spacing between cards because of sockets used in the hardware build. The system draws about 200 watts of DC power from the power supplies. The power supplies provided as part of the hardware will operate with either 60 Hz or 400 Hz, 115 volts AC line power.

SUMMARY

The PATS hardware is designed to reduce operator workload, and provide real time multiple recognition. It does not tire like human operator and hence will operate consistently and reduce response time.

With appropriate modification, the system can be used for target acquisition, weapon delivery and missile guidance.

REFERENCES

1. Soland, D. E., and P. M. Narendra, "Prototype Automatic Target Screener", Proceedings, SPIE, Volume 178, April, 1979.

2. Narendra, P. M., and R. C. Fitch, "Real Time Adaptive Contrast Enhancement for Imaging Sensors," Proceedings, SPIE, Volume 137, May, 1978.

The authors wish to thank Dr. P. Narendra, Mr. R. Larson, and Dr. D. Panda for initial algorithm simulation studies; Mr. H. Huber and Mr. S. Elvine for the building the hardware; and Ms. K. Huber for typing this manuscript.

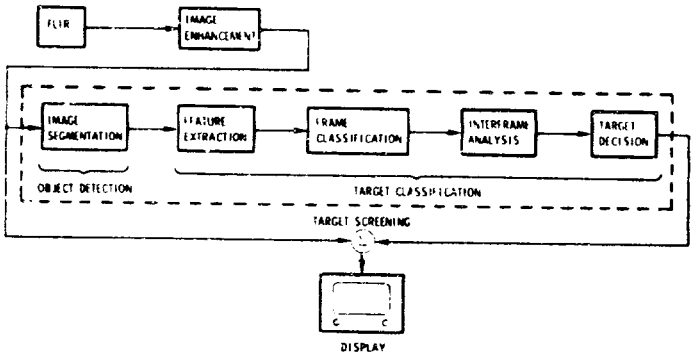


Figure 1. PATS Target Screening Function

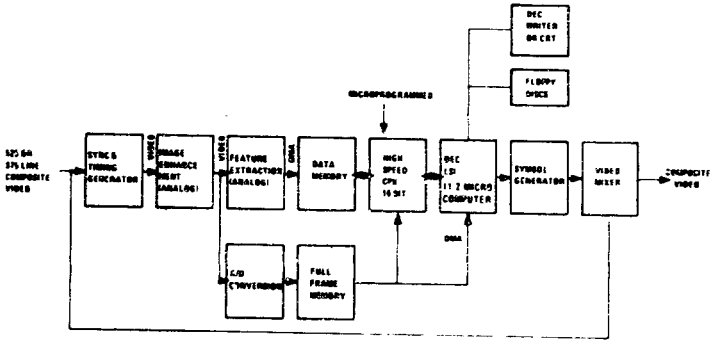


Figure 2. PATS Hardware

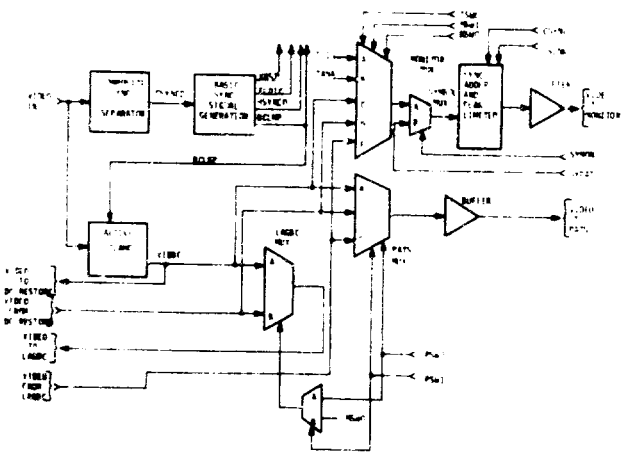


Figure 3. Sync Separation and Switching

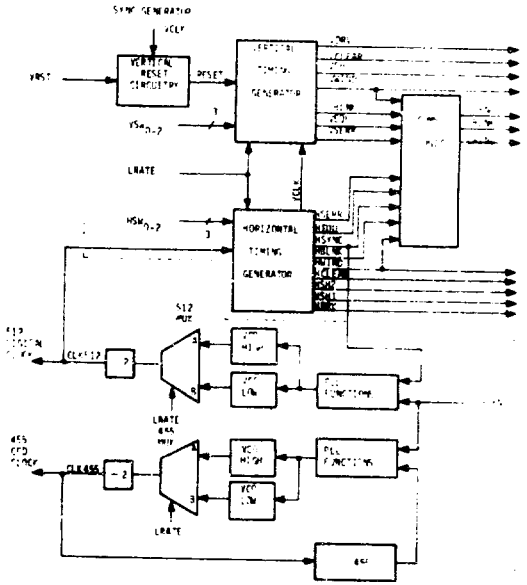


Figure 4. System Timing Generator

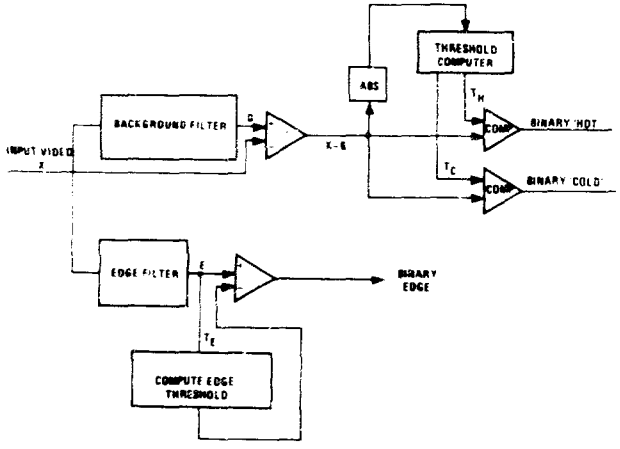


Figure 5. Autothreshold Hardware

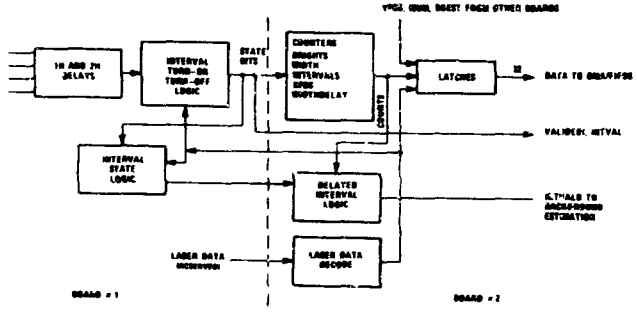


Figure 6. Interval Hardware

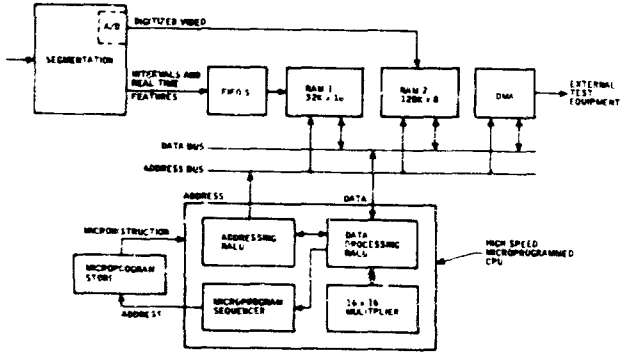


Figure 7. CPU1 System Architecture

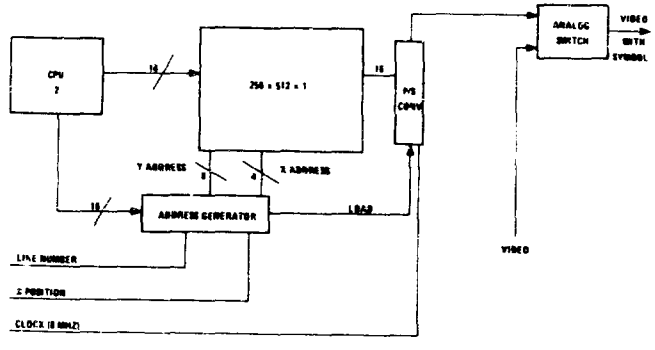


Figure 8. Symbol Generation

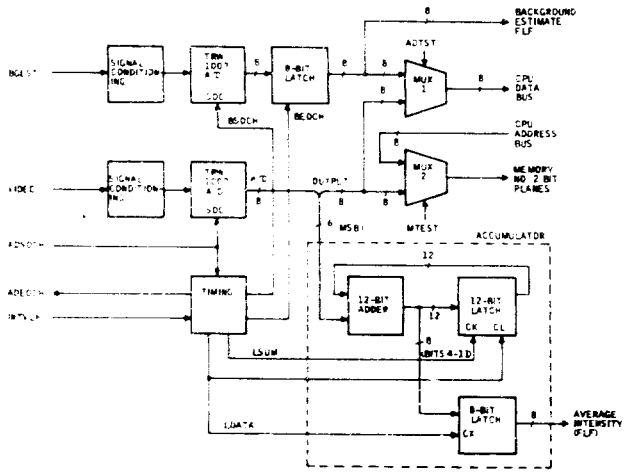


Figure 9. Analog-To-Digital Converter Unit

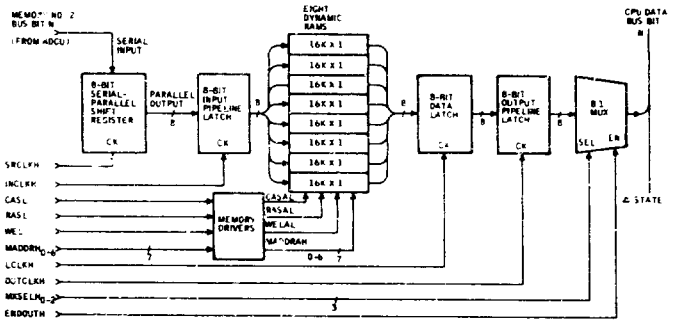


Figure 10. Memory Bit Plane

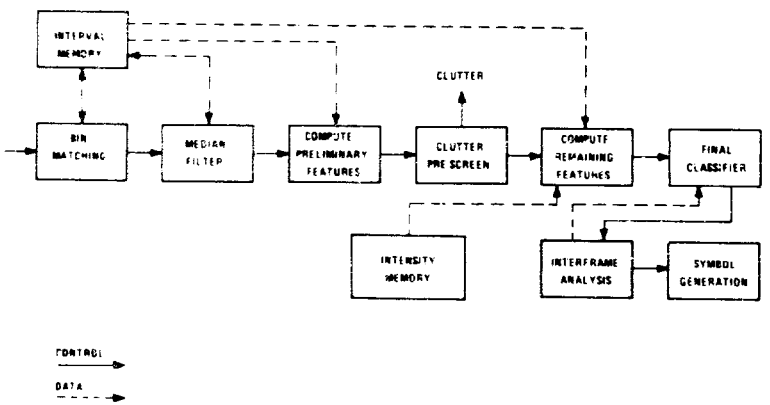


Figure 11. Software Functions

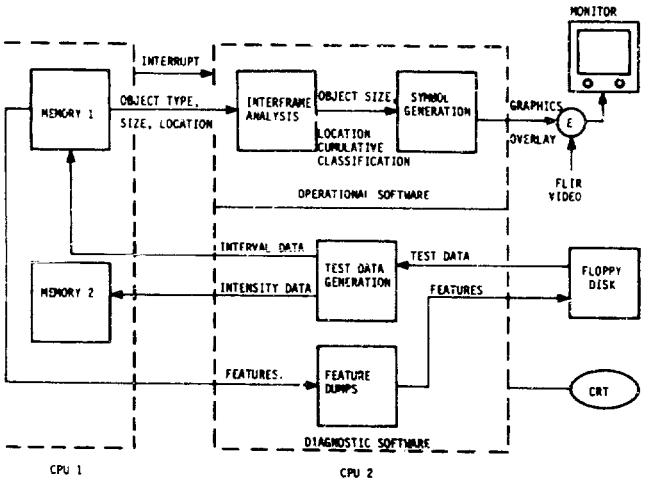


Figure 12. CPU2 Software Functions

SESSION IV

PROPRIETARY OR CLASSIFIED IMAGE
PROCESSING ALGORITHMS FOR MISSILE GUIDANCE

AND

PAPERS SUBMITTED FOR PRESENTATION

Paper No. IVB-4, Presented at the Workshop on Imaging Trackers and Autonomous Acquisition Applications for Missile Guidance, 19-20 November 1979, Redstone Arsenal, Alabama.

Multi-Color Infrared Detectors †

J. D. Jensen, A. C. Bouley, S. Foti and G. M. Black
Naval Surface Weapons Center
White Oak, Silver Spring, Maryland 20910

ABSTRACT

The ability of the eye to distinguish different colors is clearly an important source of information gathering. It has long been recognized that infrared systems would profit greatly by such an ability in areas such as target acquisition and identification, clutter and decoy rejection. No technology, however, has as yet emerged which could offer this capability at an affordable cost to most weapon systems. The present in-house program has been successful in demonstrating a new technical approach to multi-color infrared systems which is simple, straightforward and inexpensive. The new technology is based upon multilayers of thin epitaxial film detectors sandwiched in small dots, each dot being capable of detecting three or more infrared colors.

I. Introduction

This paper is a brief overview of the infrared detector research and development efforts conducted over the past few years at NSWC.[1-5] Thus far the program has covered basic research on the materials and development of generic demonstration devices. The program is rapidly approaching the point where the technology should be transferred to industry which hopefully could make devices available to the weapons community within the next few years.

The nature of the present program is the development of a technology based on single crystal epitaxial films of II-IV-VI compounds, the main products of this technology being photovoltaic narrow band self filtering infrared detectors and photovoltaic multi-color infrared detectors. The element involved from column II of the periodic table is cadmium; from column IV, lead and tin; and sulfur, selenium and tellurium from column VI. These elements form stable compounds over wide ranges of alloy composition, with the corresponding continuous range photosensitive cut-off wavelengths from 2 to 14 microns.

II. Variable Band Self Filtering Detectors

Variable band self filtering infrared detectors (VBSFID) are detectors having sharp cut-on and cut-off wavelengths which can be independently controlled and continuously varied between 2 and 14 microns by adjusting the alloy compositions of two epitaxial layers. The desirability of this

capability, of course, stems from the fact that targets, non-targets and the atmosphere all contain a significant amount of spectral structure. The ability to tailor a detector to fit the spectral signature of certain objects (in a cost effective manner) clearly represents a desirable feature of infrared systems.

The simplest form of the VBSFID is shown in Fig. 1. The infrared transparent barium fluoride (BaF_2) serves as the single crystal substrate upon which a single crystal filter layer is grown on one side, and single crystal detector layer is grown on the other. The filter layer will absorb radiation at wavelengths shorter than a cut-off wavelength determined by the alloy composition. The detector layer will absorb (and therefore detect) radiation at wavelengths shorter than a somewhat longer wavelength cut-off. The net result is a well defined spectral band of photosensitivity. It should be pointed out that the filter layer need not be on the opposite side of the substrate from the detector layer. As will be shown in Section III, the detector layer can be grown directly on the filter layer. The detector itself is a photovoltaic Schottky barrier device formed by lead or indium non-ohmic contacts vacuum deposited on the p-type semiconductor film. Gold is used for a common ohmic contact. The photosensitive region is the area under the non-ohmic contact.

The materials involved in the films are Cd, Pb, Sn, S, Se and Te. The Cd-Pb-Sn-S-Se system is shown in Fig. 2. The plot shows the continuous variation of filter or detector cut-off wavelengths at three temperatures. Alloys of PbSnTe are also part of the IV-VI family which cover part of the cut-off range shown here. Other than the binary compound PbTe, however, these alloys have not been included in this project. The materials are grown in a bell jar evaporator, containing a source, shutter and substrate as shown schematically in Fig. 3. The particular II-IV-VI alloy is pre-synthesized in polycrystalline form by the reaction of the elements in an evacuated quartz ampoule held at elevated temperatures. The material is granulated and placed in the quartz source as shown. An extra source of the chalcogenide, S, Se or Te is heated separately so as to mix with the molecular beam. This assures p-type conductivity in the 10^{17} cm^{-3} region. The temperatures shown in Fig. 3 are adjusted somewhat for the particular alloy being grown. Films are grown at a rate of approximately two microns per hour.

Typical characteristics of the two "components" of the sandwich device are shown in Figs. 4 and 5. Figure 4 shows that a single layer 5-15 microns thick can be a rather effective spectral filter. Figure 5 is the I-V characteristic of a short wavelength detector, $PbS_{0.8}Se_{0.2}$ at 77 K.

The spectral responses of a variety of detectors are shown in Figs. 6-13. These data represent a variety of alloys, broad band, narrow band and unfiltered devices, at room temperature and below, at wave-

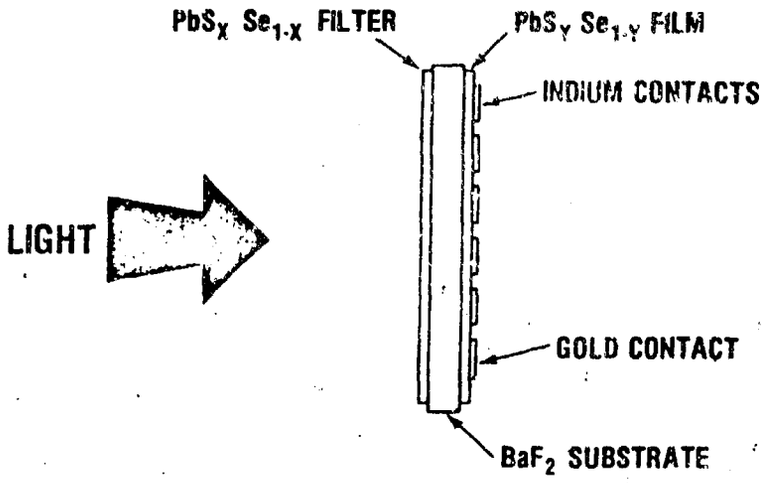


FIG. 1 CONFIGURATION FOR A VARIABLE BAND SELF FILTERING DETECTOR.

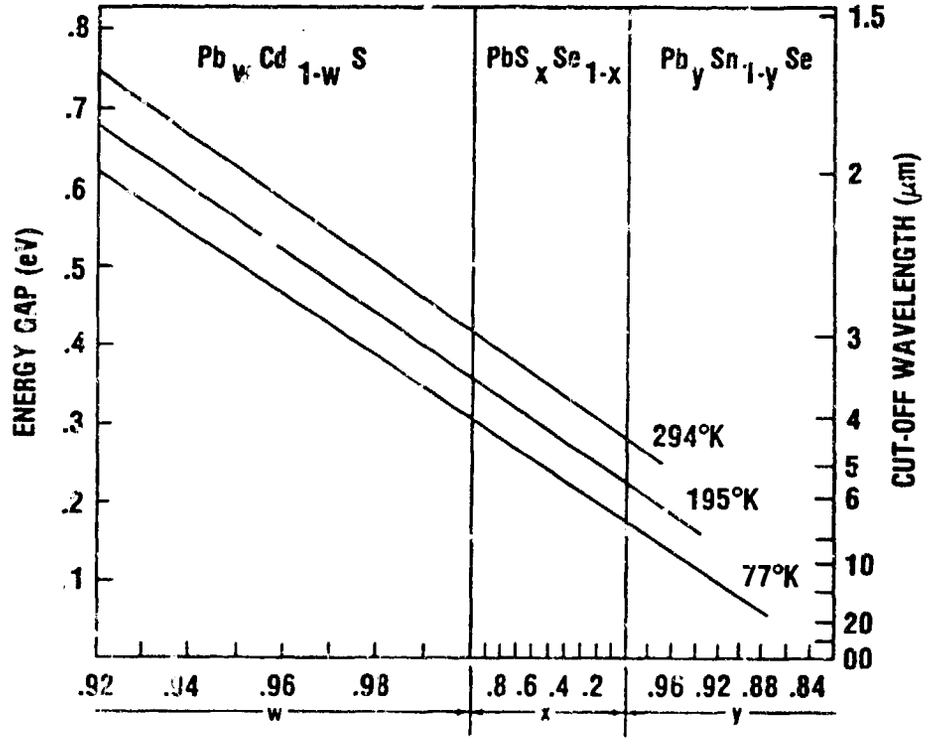


FIG. 2 ENERGY GAPS AND CUT-OFF WAVELENGTHS OF II-IV-VI ALLOY SEMICONDUCTORS.

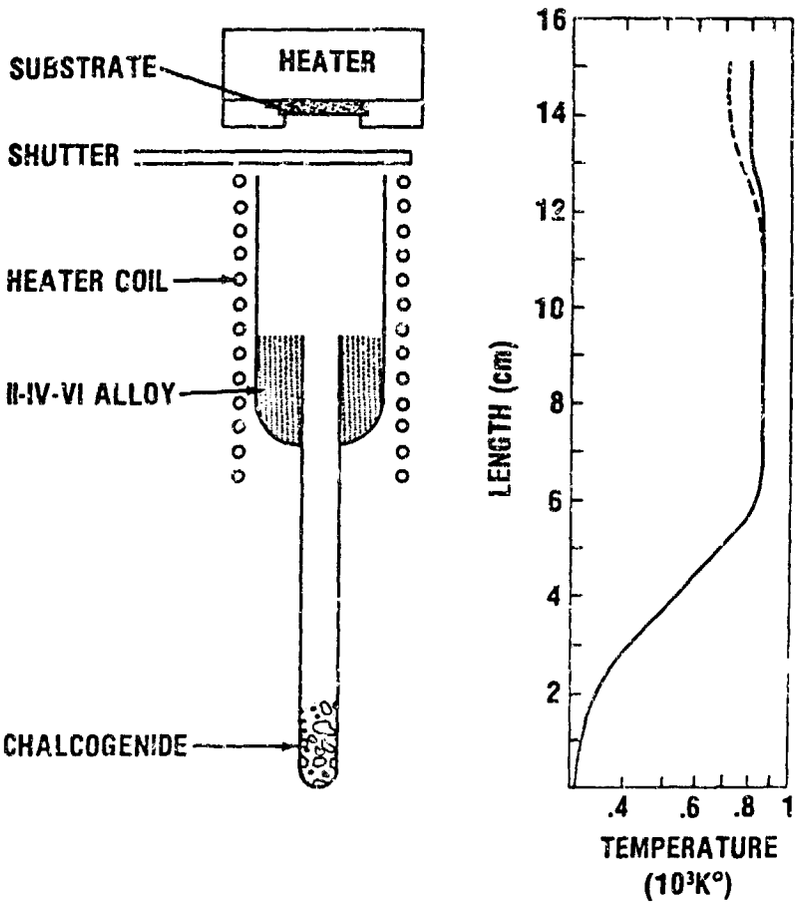


FIG. 3 SCHEMATIC OF FILM GROWTH APPARATUS.

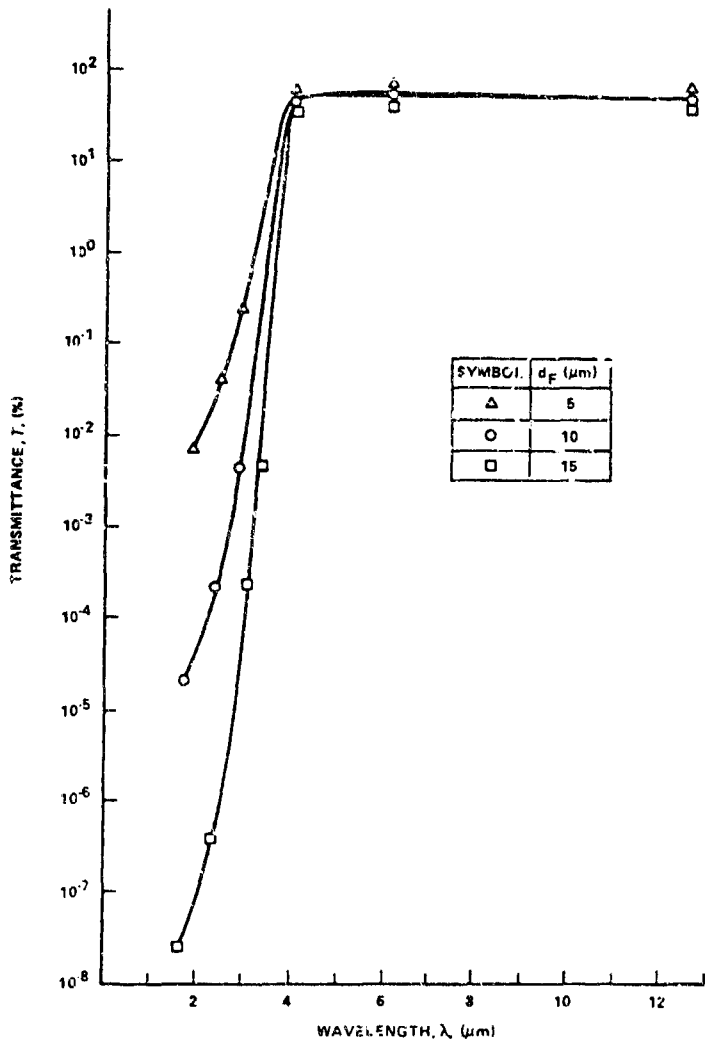


FIG. 4 TRANSMITTANCE, T, FOR VARIOUS PbS FILTER THICKNESSES, d_F.

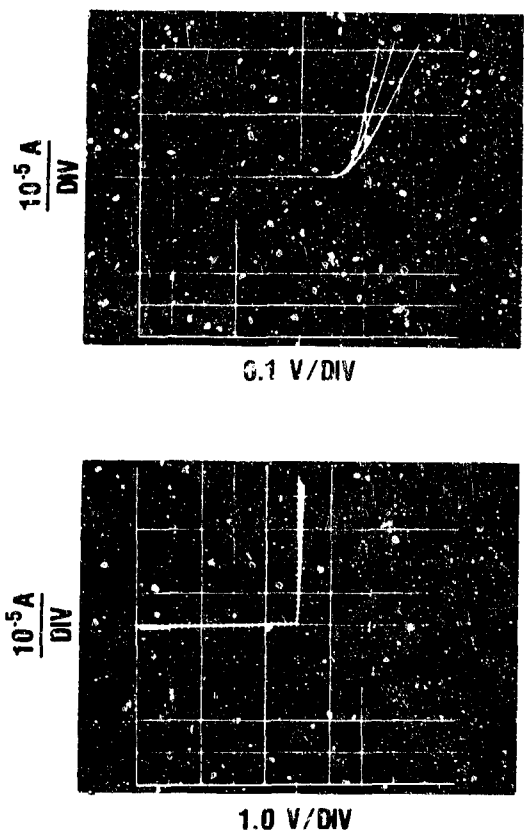


FIG. 5 I-V CHARACTERISTICS OF $PbS_{0.8}Se_{0.2}$ SCHOTTKY BARRIER DETECTORS AT 77 K

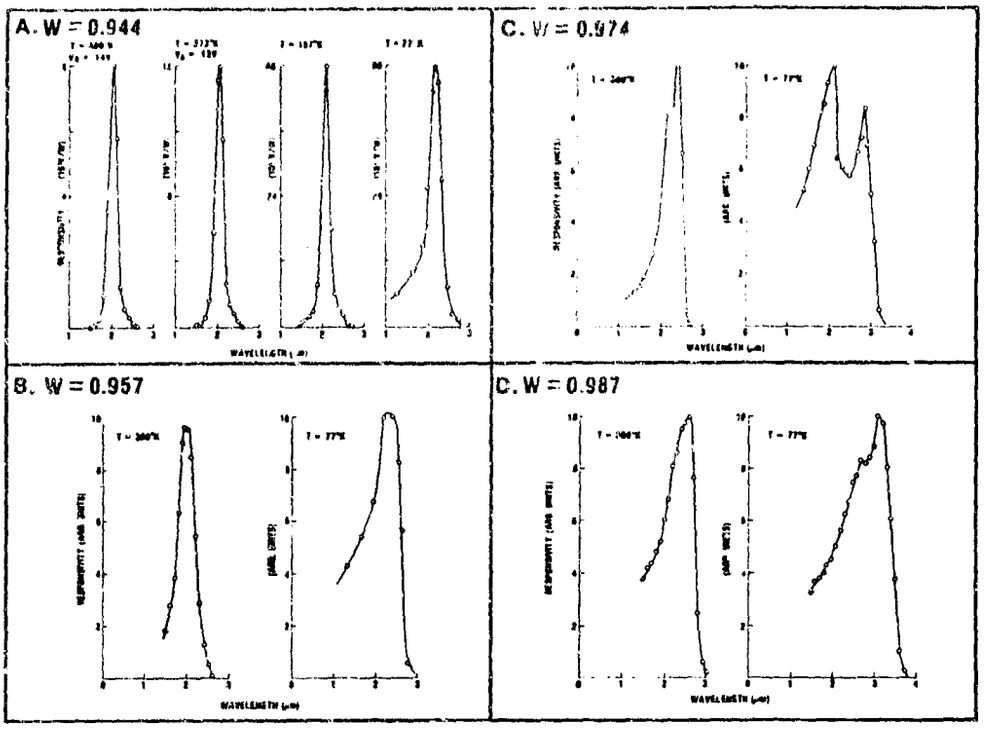


FIG. 6 RESPONSIVITY OF $Pb_WCd_{1-W}S$ SCHOTTKY BARRIERS.

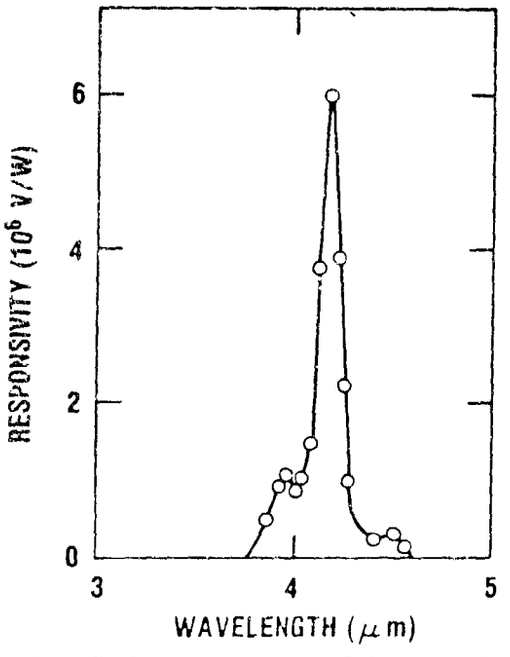


FIG. 7 RESPONSIVITY OF A $PbS_{.85}Se_{.15}$/PbS DETECTOR AT 77 K. PbS IS THE FILTER LAYER.

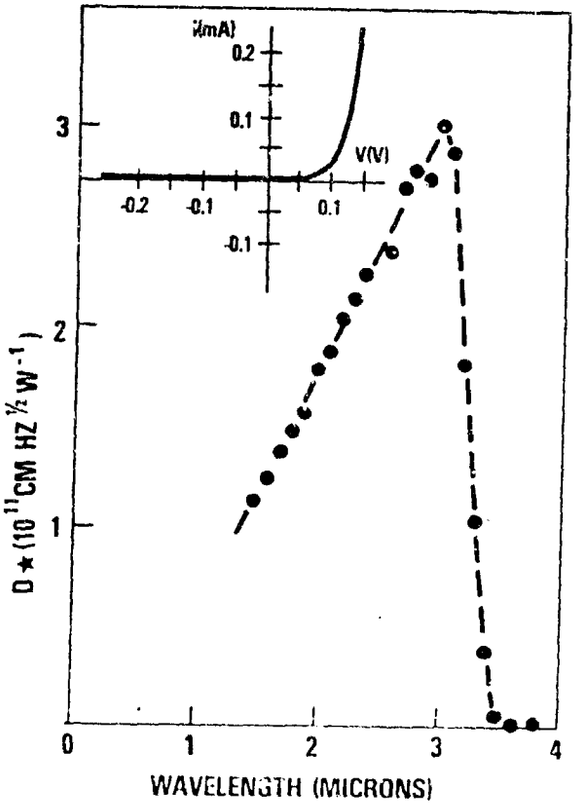


FIG. 8 SPECTRAL RESPONSE OF AN UNFILTERED, UNBIASED PbS/LEAD SCHOTTKY BARRIER DETECTOR AT 197 K WITH 20°FOV. I-V CHARACTERISTIC ALSO AT 197 K.

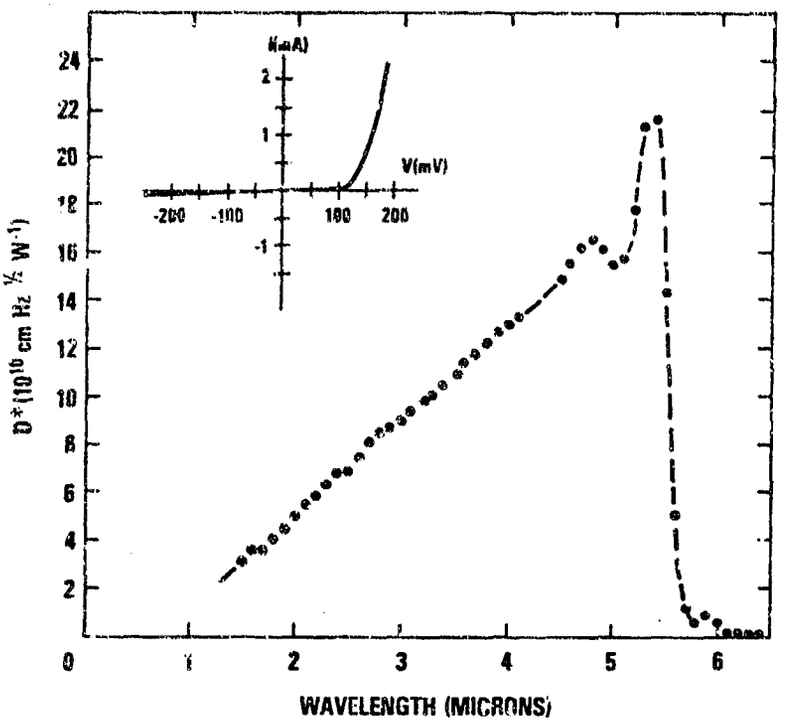


FIG. 9 SPECTRAL RESPONSE OF AN UNFILTERED, UNBIASED LARGE AREA ($3x10^{-2}cm^2$) PBTE/LEAD SCHOTTKY BARRIER DETECTOR AT 80 K WITH 20°FOV. I-V CHARACTERISTIC ALSO AT 80 K.

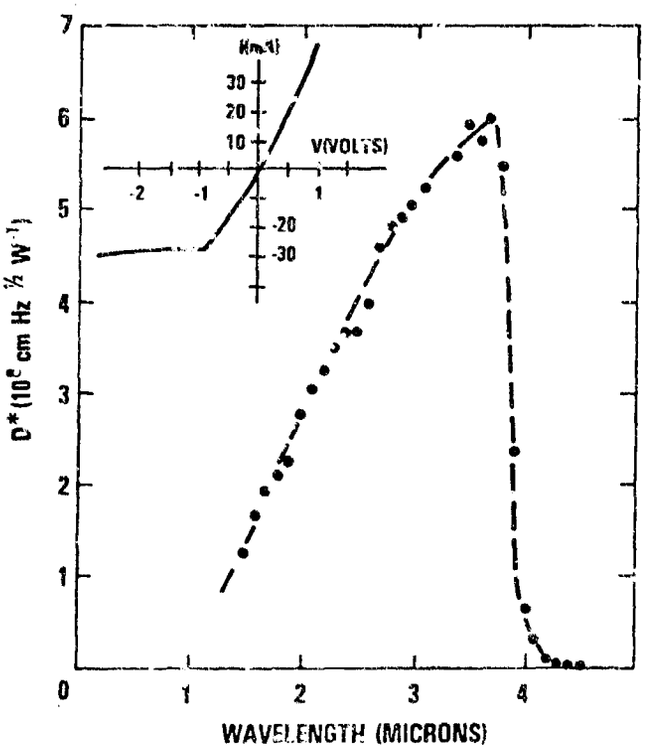


FIG. 10 ROOM TEMPERATURE RESPONSE OF PBTE DETECTOR SHOWN IN FIG. 5. RESPONSE TIME AT 300°K IS 3 μSEC.

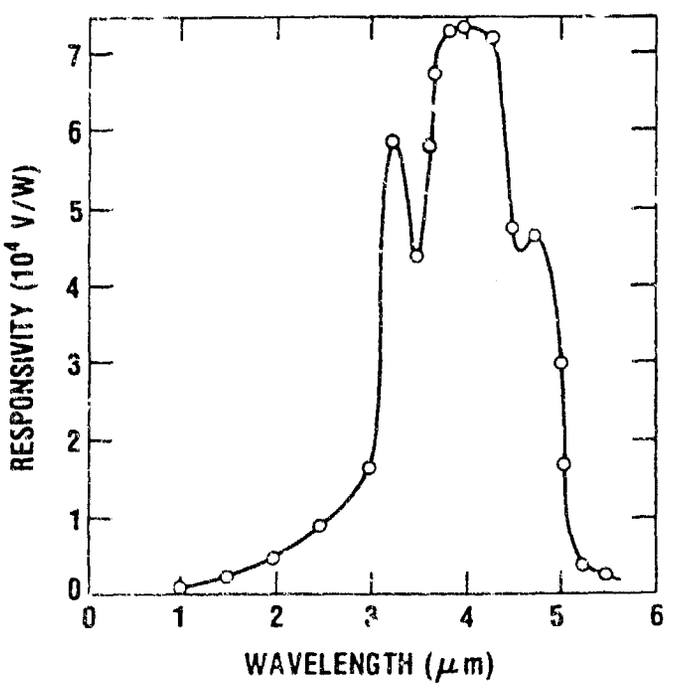


FIG. 11 RESPONSIVITY OF $PbS_{.50}Se_{.50}/Pb_{.98}Cd_{.02}S$ DETECTOR AT 77°K.

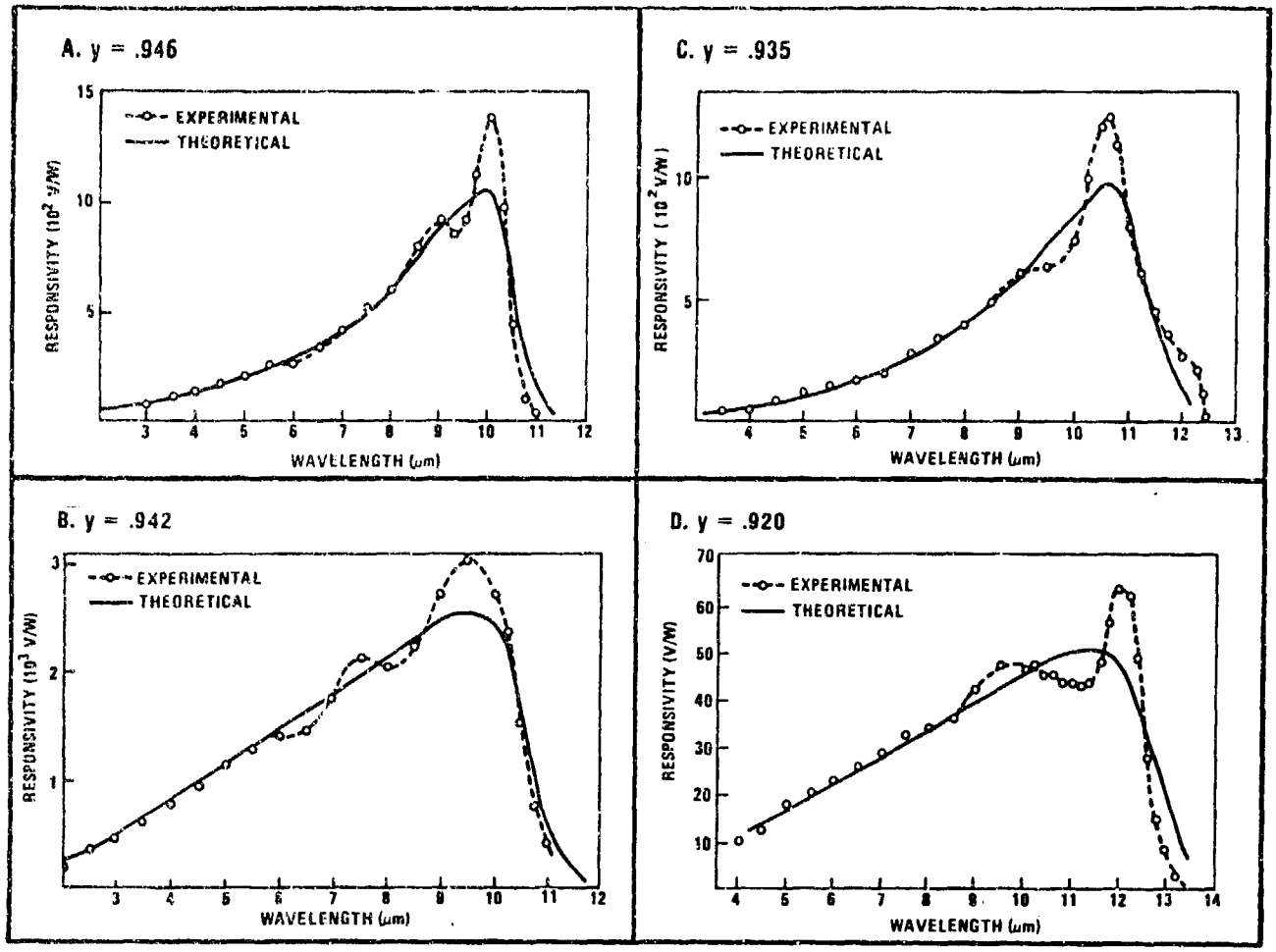


FIG. 12 RESPONSIVITY OF $Pb_ySn_{1-y}Se$ SCHOTTKY BARRIERS AT 77°K.

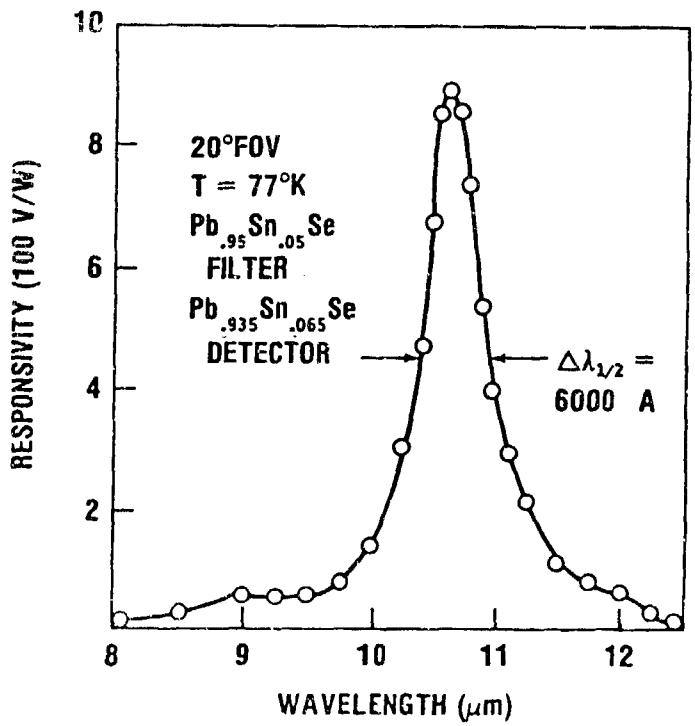


FIG. 13 RESPONSIVITY OF $Pb_{.93}Sn_{.07}Se/Pb_{.95}Sn_{.05}Se$ DETECTOR AT 77°K.

lengths ranging from 2 to 13 microns. The collection of data demonstrates the versatility of this II-IV-VI epitaxial film technology to meet a variety of military infrared detector needs, in a manner which should prove to be very cost effective. Many of the response curves show some degree of oscillatory behavior. These are interference fringes associated with the fact that the film thicknesses are comparable to the wavelengths involved. The unfiltered detector shown in Fig. 9, for instance, clearly displays a rapidly damped oscillation near the cut-off edge. Interference effects in the filter layer and detector layer can be used together to enhance performance, such as in the narrow band response shown in Fig. 7. Here the thicknesses of the films were adjusted such that a fringe maximum in the transmission of the filter layer coincides with a fringe maximum in the photoresponse of the detector layer. The sensitivity in terms of D* relative to the familiar spectral response chart for commercial detectors is shown in Fig. 14. It is, of course, not valid to compare the response of our detectors with others on this chart. This is a 180° room temperature field-of-view chart with the commercial detectors approaching the theoretical limit. Our detectors exceed this limit because the filter layer, being an integral part of the detector unit, is cooled and serves not only as a spectral filter, but also as a cold shield. In addition, the FOV of our detectors is limited to about 20° by a cold metal shield in our test set up. The plot does serve, however, to show the sensitivity of the devices relative to this familiar standard.

III. Multi-Color Detectors

Another specific area that the II-IV-VI single crystal film technology can address, is that of multi-color infrared detectors. The general configuration of such a device is shown in Fig. 15. It is clearly a very logical extension of the variable band self filtering detectors, discussed in the last section. In the multi-color configuration, the detector layer for one color is the filter layer for another, except for the longest wavelength cut-off layer which only detects. The geometry is intended to be such that a resolution element consists of two, three or more color detectors with the detector of a specific color being displaced from the others within the resolution element. The elements can be made very small, limited by the particular design and the fact that n leads must come from each n-color element. Each layer has its own Schottky contact which can be the basis of an independent electrical channel for the appropriate signal processing of the various colors. It should be noted that because of the back-side illumination through the transparent BaF_2 substrate, the incident light is not obstructed by the leads and contacts. The band diagram for the $BaF_2/PbS/PbS_{0.5}Se_{0.5}/Pb$ system is shown in Fig. 16. The manner in which the bands are bent at the interfaces is such as to contain the photoexcited minority carrier (electrons) in the material in which it was excited, and not such as to trap it at the interface, enable it to recombine at the interface or inject it into the next semiconductor layer. The spectral responses of several multi-color detectors are shown in Figs. 17-20.

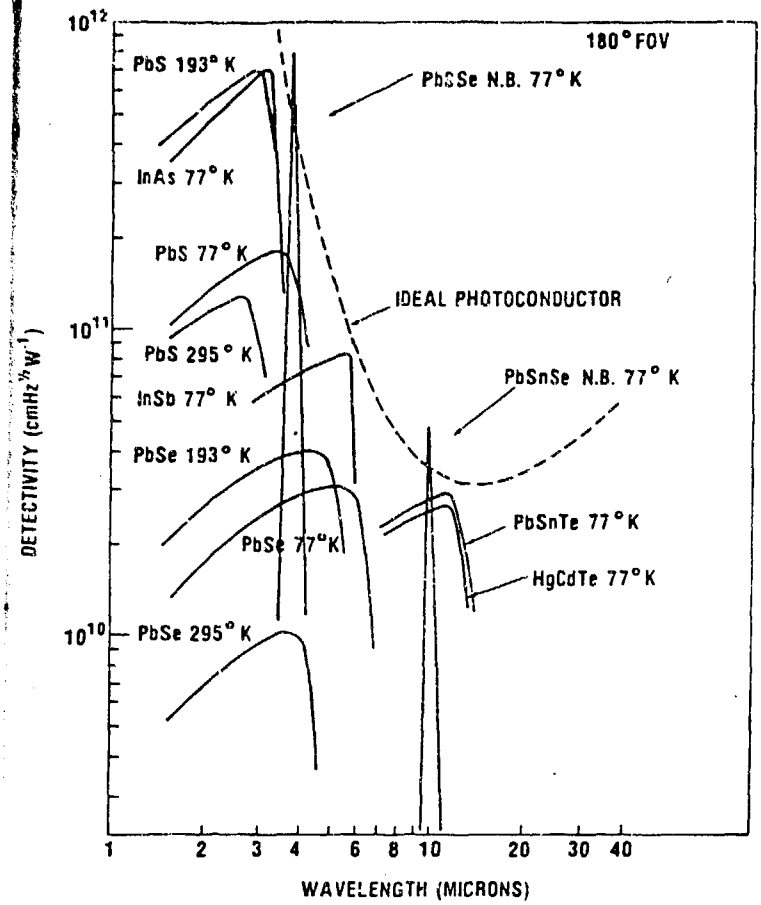


FIG. 14 PLOT SHOWING D* OF NARROW BAND COLD SHIELDED DETECTORS RELATIVE TO STANDARD CHART OF COMMERCIAL DETECTORS.

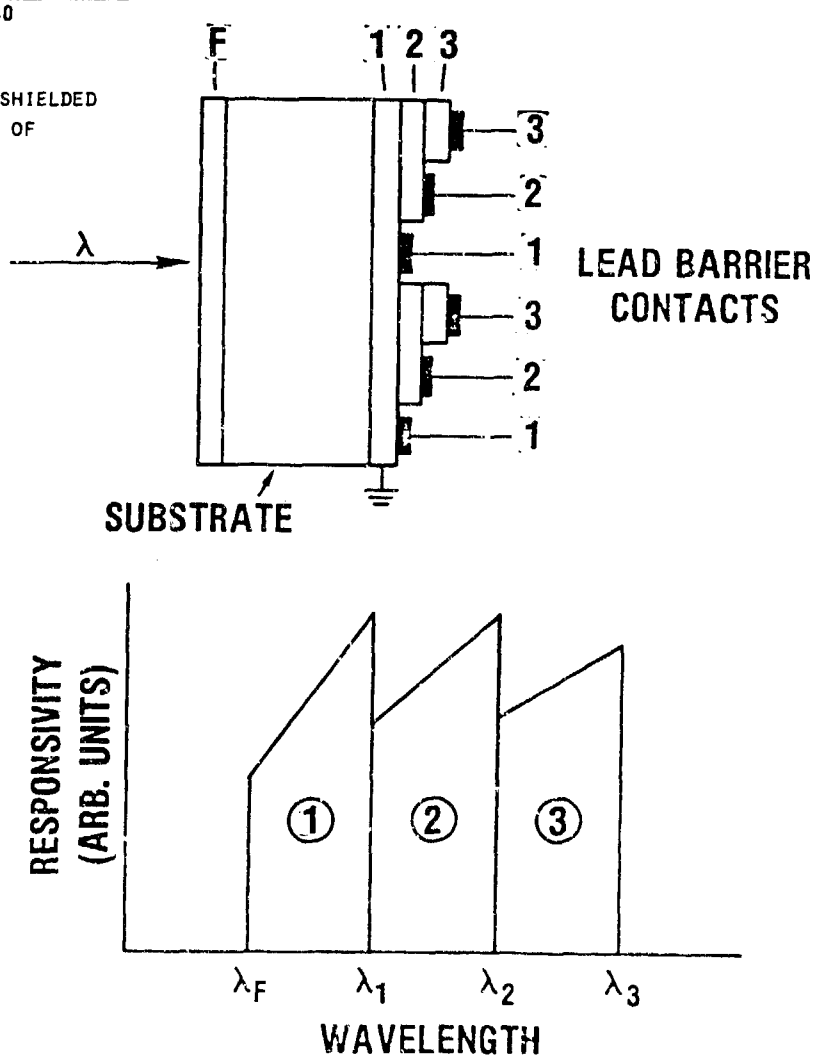


FIG. 15 MULTI-COLOR ARRAY CONFIGURATION.

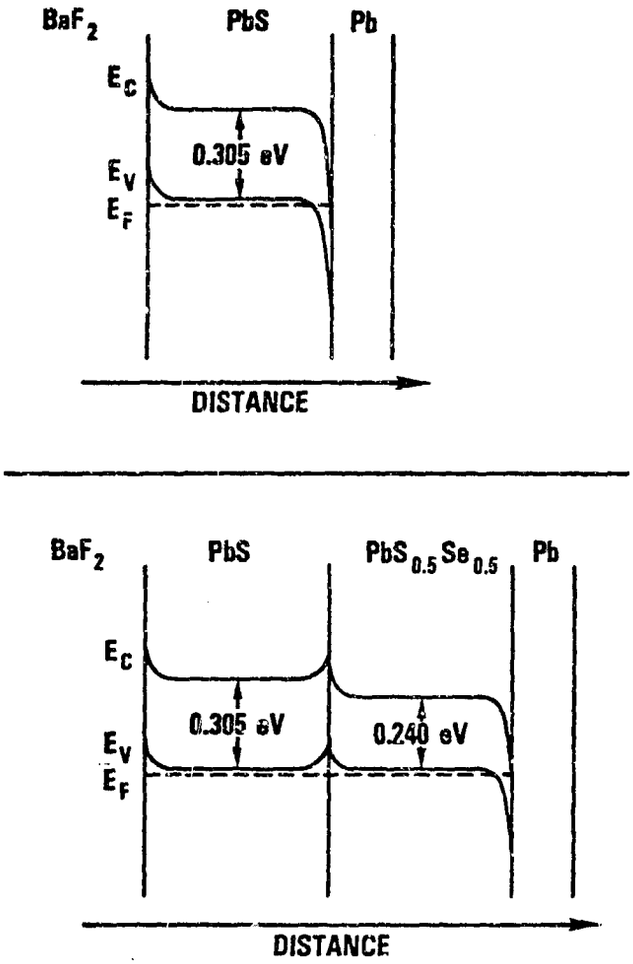


FIG. 16 ENERGY BAND DIAGRAMS OF TWO-COLOR DETECTOR.

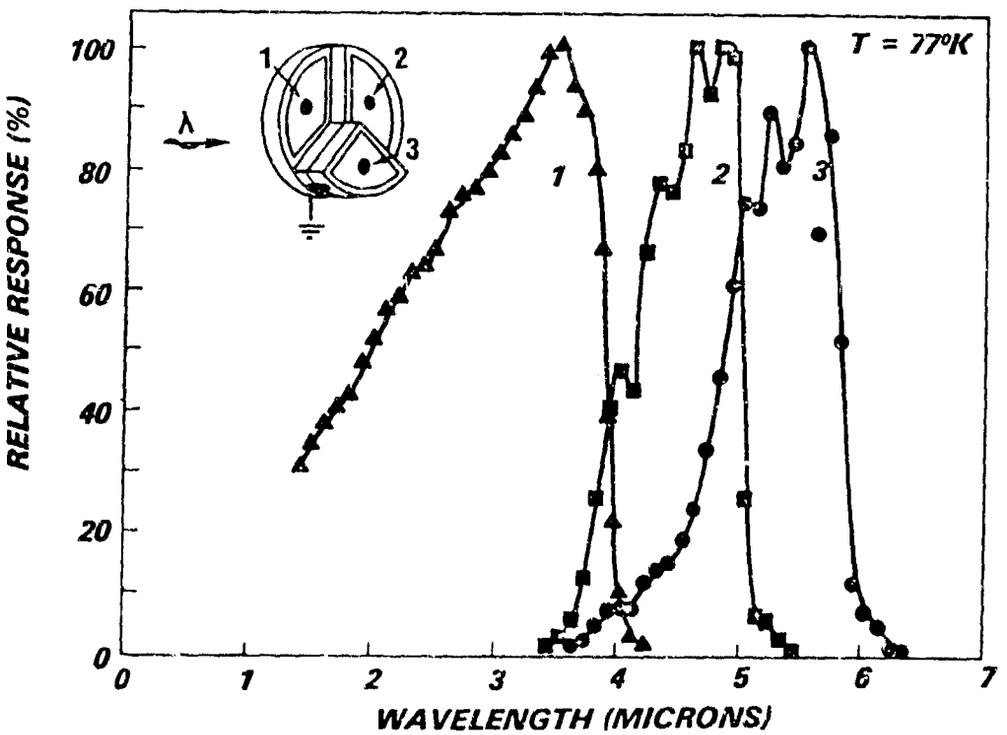


FIG. 17 RESPONSIVITY OF PbS_xSe_{1-x} TRI-COLOR DETECTOR.

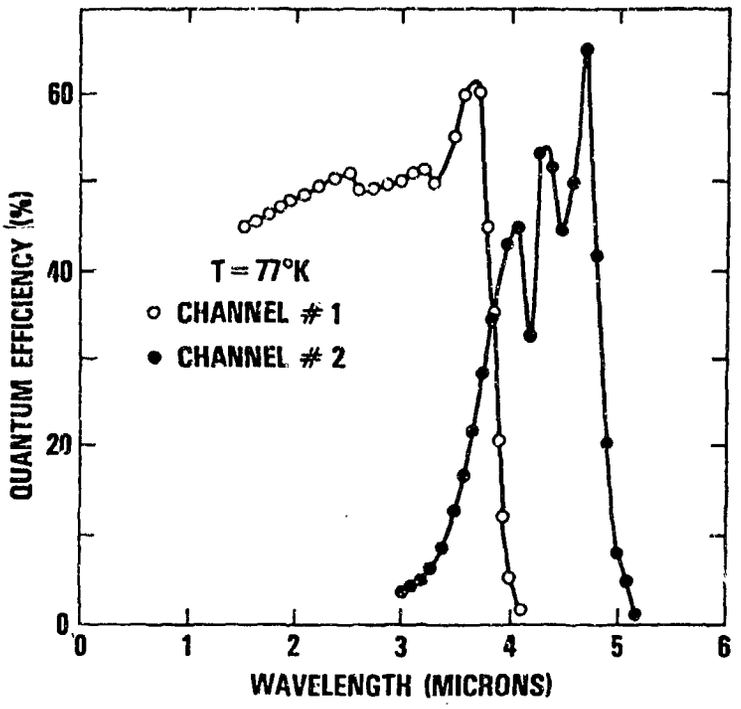


FIG. 18 QUANTUM EFFICIENCY OF TWO-COLOR PbS_xSe_{1-x} DETECTOR.

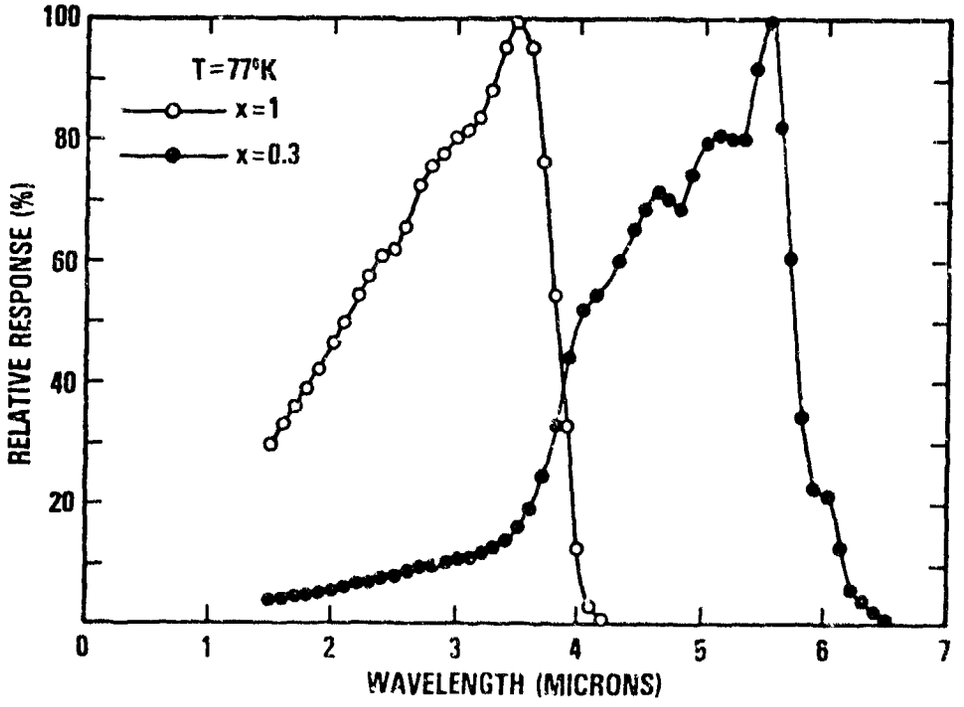


FIG. 19 RELATIVE RESPONSE OF TWO-COLOR PbS_xSe_{1-x} DETECTOR.

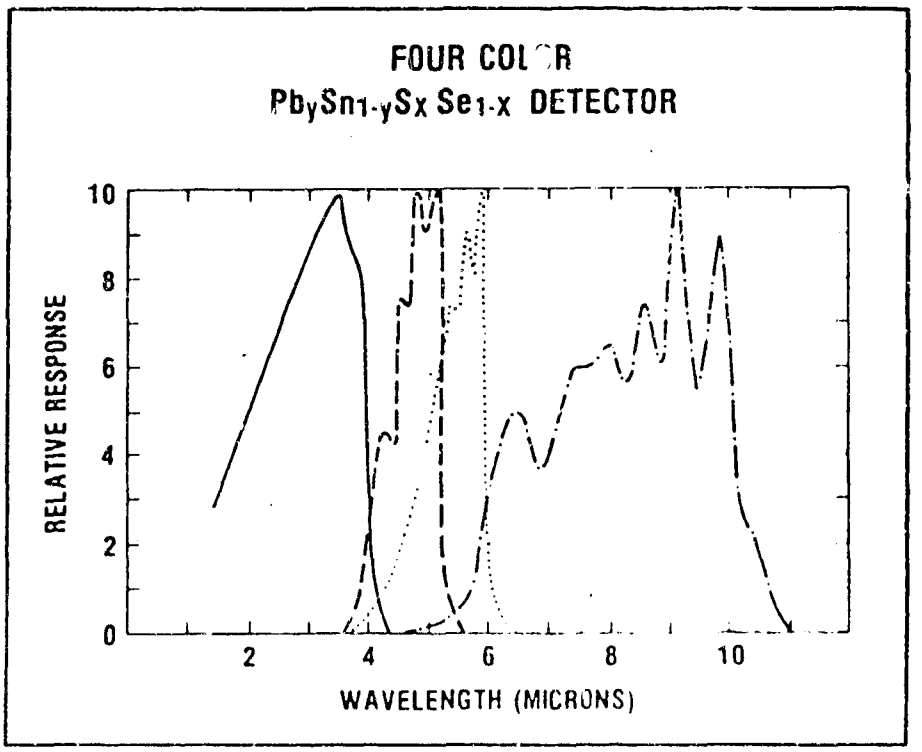


FIG. 20 RELATIVE RESPONSE OF A FOUR-COLOR $PB_YSN_{1-Y}S_XSE_{1-X}$ DETECTOR.

IV. Summary

The data presented here is meant to demonstrate the potential of the II-IV-VI single crystal film technology. The inherent advantages of the technology and the devices produced therefrom are summarized as follows:

- Versatility...the entire infrared region from 2-14 microns covered by one family of alloys.
- Variable Band...cut-on and cut-off wavelengths independently controlled.
- Cold Shielded...cut-on filter layers serve as cold shields.
- Multi-color...n-color cold shielded detectors made up of (n+1) layers of different alloy compo ition.
- Low Power Dissipation...photovoltaic devices operated at zero bias.
- Back-Side Configured...illuminated through the substrate so that optically active area not obscured by leads.
- Thin Film...inherently simple film structures that should be low cost in high volume production.

The future development of the technology is probably best directed toward the non-imaging, single element or small array applications, such as in guidance, laser detection, fuzing or remote sensing.

References

1. "Surface charge transport in PbS_xSe_{1-x} and $Pb_{1-y}Sn_ySe$ epitaxial films" J. D. Jensen and R. B. Schoolar, J. Vac. Sci. Technol. 13, 920 (1976).
2. "Composition-tuned PbS_xSe_{1-x} Schottky-barrier infrared detectors" R. B. Schoolar, J. D. Jensen, and G. M. Black, Appl. Phys. Lett. 31, 620 (1977).
3. "Narrowband detection at long wavelengths with epitaxial $Pb_ySn_{1-y}Se$ films" R. B. Schoolar and J. D. Jensen, Appl. Phys. Lett. 31, 536 (1977).
4. "Composition Tuned II-IV-VI Alloy Semiconductor Infrared Detectors" R. B. Schoolar, J. D. Jensen, G. M. Black, and D. L. Demske, Proceedings of the IRIS Detector Meeting, March 1977.
5. "Multispectral PbS_xSe_{1-x} and $Pb_ySn_{1-y}Se$ Photovoltaic Infrared Detectors" R. B. Schoolar, J. D. Jensen, G. M. Black, S. Foti, and A. C. Bouley, to be submitted to Infrared Physics (1979).

* Supported by the Naval Air Systems Command, Naval Research Laboratory and NSWC R&D Funds.

Unclassified paper submitted for Workshop on Imaging Trackers and Autonomous Acquisition Applications for Missile Guidance, 19-20 November 1979, Redstone Arsenal, Alabama.

A REAL-TIME DIGITAL IMAGE SIMULATION FACILITY WITH APPLICATIONS FOR EVALUATION OF IMAGE BASED MISSILE GUIDANCE SYSTEMS

Stephen C. Noble
Ampex Corporation
401 Broadway, Redwood City, CA 94063

ABSTRACT

A real-time digital image simulation facility is described that is suitable for generating image sequences for the test and evaluation of imaging trackers and autonomous acquisition applications for missile guidance. The facility can generate synthetic images from computer programs or record video image data in real-time, process it, and then display it in real-time.

The system will have the capability of recording and playing back in real-time component color video data in 525 line format or high resolution monochrome video data in 875 line format. The maximum planned simulation rate is 30 megapixels per second for 8-bit pixels.

INTRODUCTION

This paper describes a real-time image simulation facility under development at Ampex which utilizes the recently introduced Ampex Parallel Transfer Drive (PTD) technology. For this facility the drive technology is being extended from 9 parallel tracks to 18 parallel tracks to increase the simulation rates available to a maximum of 30 megapixels per second. This development is internally funded by Ampex with the principal application being high quality color television simulation.

Simulations for a wide variety of other applications are readily achieved with this system because of the ability to record and playback data at arbitrary data rates from DC to 30 megapixels per second. The simulation of image sequences for the test and evaluation of imaging trackers and autonomous acquisition systems for missile guidance is one such application.

DIGITAL IMAGE SIMULATION FACILITY DESCRIPTION

The digital image simulation facility is an integrated digital system with digital color video storage capability combined with a dedicated high performance digital processor (DEC PDP 11T55). The block diagram for the system is shown in Fig. 1. The system will allow for processing either component or composite video signals. In addition, the system will accept a variety of other signals.

Hardware Elements of the Facility

The hardware elements of the facility consist of a high quality video source, video A/D converters, input and output data processors, a cylinder buffer system, a parallel transfer disk (PTD/18) controller with PTD 9318 digital image disk storage units, a set of video D/A converters, a high quality output display and a PDP 11T55 processor with peripherals.

A/D and D/A Converters

The A/D converters and the D/A converters will be 9-bit Ampex units for PAL or SECAM composite signals. For component color signals (e.g. RBG, YIQ or YUV) a set of three TRW 8-bit A/D and D/A converters are designed into the system. For high resolution monochrome signals a single A/D converter is required for image data input.

Input and Output Data Processors

The disk interface write unit will convert the image source data for recording onto the disk drives. The disk interface read unit reconverts for output to display or to system under test. Initially, it is planned to have a word rate (18-bit words) of 10.7 MHz. Extension to 14 MHz is planned.

Output Display

For color simulations the output display is a high resolution monitor with 9 MHz bandwidth video channels and a color picture tube with four times as many color dots as a standard color tube. The monitor is configured to operate on RBG or NTSC composite signals. Plug in demodulators are also available for PAL, PAL-M, and SECAM composite signals. For direct output to a system under test, no display is required. Additional displays will be required for high resolution monochrome simulations.

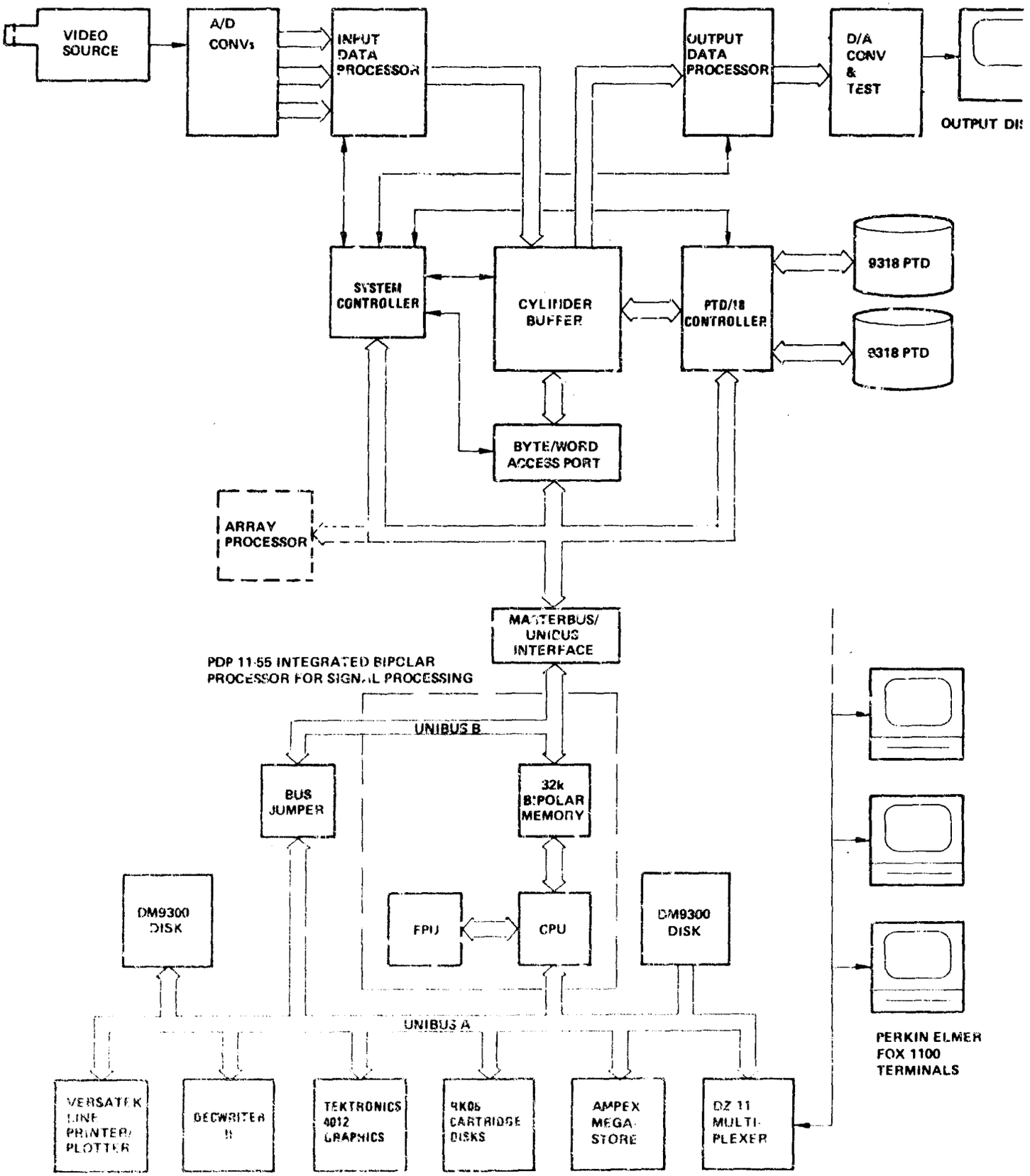


Figure 1 Digital Image Simulation Facility

Digital Image Storage

The digital image storage system is composed of two DM9300 300 megabyte Ampex disk drives modified to provide up to 18 parallel data channels for recording image data continuously for up to 30 seconds. The modifications include read/write amplifiers, timebase correctors, signal processing circuits and servo control circuits from the Electronic Still Store (ESS) system design. In addition, a special system control unit and a computer interface unit is being designed and built to control the transfer of digital video data between the digital television system, the disk storage system and the digital processor.

Cylinder Buffer

The cylinder buffer is a 1 megabyte dynamic RAM memory configured to allow data transfer of digitized video sampled at rates up to 18 MHz (16-bit words). It is provided with a high speed video input interface, a high speed video output interface, a bi-directional PTD/9318 interface and a PDP 11/55 interface which will allow both single byte/word transfers and DMA transfers. Paralleling of data (16 bits to 128 bits) is used to gain bandwidth. The memory is implemented with 16k 375 nsec cycle time, 200 nsec access time dynamic RAMS. Error detection and correction is included in the design.

The cylinder buffer allows continuous data recording from the video input to the PTD. Additionally, interactive signal processing can be sustained by the PDP11/55 while refreshing the output display. The cylinder buffer simplifies the disk accesses as all read or write operations from the disk are of a complete cylinder (1 revolution X 18 surfaces). A system controller controls contention problems and grants cylinder buffer accesses.

Digital Processor

The high performance digital processor is a Digital Equipment Corporation (DEC) PDP 11T55. This integrated bipolar processor is ideally suited for signal processing. Through the use of very high speed bipolar memory integrated circuits (IC's), the central processing unit (CPU) can move between the bipolar memory had the CPU in 300 ns. This is three times faster than the speed available with computers using core memory tied to the computer bus. In addition, a high speed floating point unit (FPU) is provided for high speed, high precision multiplication and addition. The processor is complemented with 96k words of Ampex core

memory, an Ampex Megastore (a core memory system that emulates a fixed head disk with zero latency), 16 video terminals for software development, a Versatek printer/plotter for hardcopy, a Tektronix graphics terminal and Ampex 300 Megabyte disk drives. A 9600 baud communication link to the AVSD PDP-11/45 graphics system is planned for sharing of resources. Figures 2 and 3 are pictures of the Digital Video Simulation Facility. Figure 2 is the processor and video equipment room. Figure 3 is the evaluation room (temporarily used for software development on the computer terminals).

Facility Operation

The simulation facility will record the video to be processed (component, composite or monochrome) in a 30 second continuous period. The signal processing experiments to be performed on the system will be carried out by the processor. Typically, experiments will require from one to ten hours of processing time for a 30 second simulation. For applications where computer intensive simulations are used repeatedly, the addition of an array processor will be required.

Facility Performance

The digital video simulation facility will provide the capability to store for processing video signals at bit rates up to 240 MHz. This will provide a variety of word sizes and word rates for various signals. The following are examples:

4-bit word radar signals	60 MHz
8-bit word composite video	30 MHz
16-bit word high resolution or component video	15 MHz

Additional data sources include Sonar, Wide Dynamic Range IR, Low Light Level TV, 875 line high resolution monochrome video and the output of an Optical Processor.

Software Elements of the Facility

A variety of signal processing software is being developed as required. The following list gives the most likely candidates:

Bandwidth compression
Image enhancement

Figure 2 Processor and Video Equipment Room

Figure 3 Evaluation Room

Image restoration
Spectral analysis
Digital filtering
Information extraction
Hardware simulation
Optical system analysis

Tables 1 and 2 describe in detail some of the signal processing experiments possible with this system.

This digital signal processing capability provides a powerful complement to our optical signal analysis capability. In particular, it provides analysis, simulation and comparative tools which can be used to detemine the best combination of optical and digital technologies to solve a specific processing problem.

Software Simulation Tools

The software for this system is a natural extension of the power and flexibility of the UNIX Time-Sharing System developed at Bell Laboratories for the DEC PDP-11 computer. A versatile set of modular signal processing programs has been developed which communicate via inter-process I/O channels called pipes. [1] This structure allows each module to be a small program that efficiently performs an elementary signal processing function.

Each module reads a signal data stream from its standard input file, processes it, and writes the resulting signal data stream on its standard output file. Several such modules may be cascaded by connecting them with inter-process pipes. A pipe connects the standard output of one module to the standard input of the next. The connection behaves like a normal disk file as far as each module is concerned, but is implemented with a FIFO buffering mechanism. This allows the module processes to execute concurrently, yet communicate efficiently with one another. This structure also eliminates the need to store intermediate data files, but does not preclude it.

Most modules are easy to design, write, and debug, since they are typically less than two pages long, and are written in the language C. [2] C provides a rich selection of operations and data types and the ability to impose useful structure on both control flow and data.

Signal data streams are a sequence of segments. Each signal segment begins with a header which describes the signal data comprising the remainder of the segment. The header information includes:

1. An illegal numeric quantity as a consistency check,
2. The number of samples in the segment,
3. The number of sample elements per sample (each sample is a multi-channel vector),
4. The number of samples in a row (for 2-D signals),
5. The data type of the sample elements (e.g., integer, floating, etc.),
6. The number of bytes used by a sample element.

The actual signal data is a sequence of binary sample elements, using the natural machine representation for each element type. Figure 4 illustrates a sequence of signal segments. The header structure is simple to maintain; each module examines the incoming header and outputs a header suitably modified to account for the processing to be performed.

The UNIX shell (command line interpreter) allows the user to execute a signal processing module merely by typing its name. Some modules require arguments, and these are typed after the name, all separated by spaces. Normally, the standard input of a program is attached to the user's keyboard, and the standard output is attached to the display. The Shell allows these to be redirected to files by typing a " < " or " < ", followed by the file name. It also provides a way to pipe the standard output of one program to the standard input of the next with the symbol " | ". As an example, consider the following simulation used in bandwidth compression simulations:

The Shell Command Line

```
Sht 4 4 < image.S | Scode ht44.q | Sdecode ht44.q | Siht 4 4 > new image.S
```

runs the 4 programs "Sht", "Scode", "Sdecode", and "Siht" simultaneously, each deriving its input from the output of the program on its left. The left-most program obtains its input from the file "image.S", and the right-most program is writing onto the file "new image.S". The "Sht" module performs Hadamard transforms on subpictures; the arguments "4" and "4" specify the number of rows and columns in the subpictures. The "Scode" module selects and quantizes the Hadamard coefficients using the scheme described by the file "ht44.q". The "Sdecode" module performs the inverse of "scode", and "Siht" does the inverse Hadamard transform.

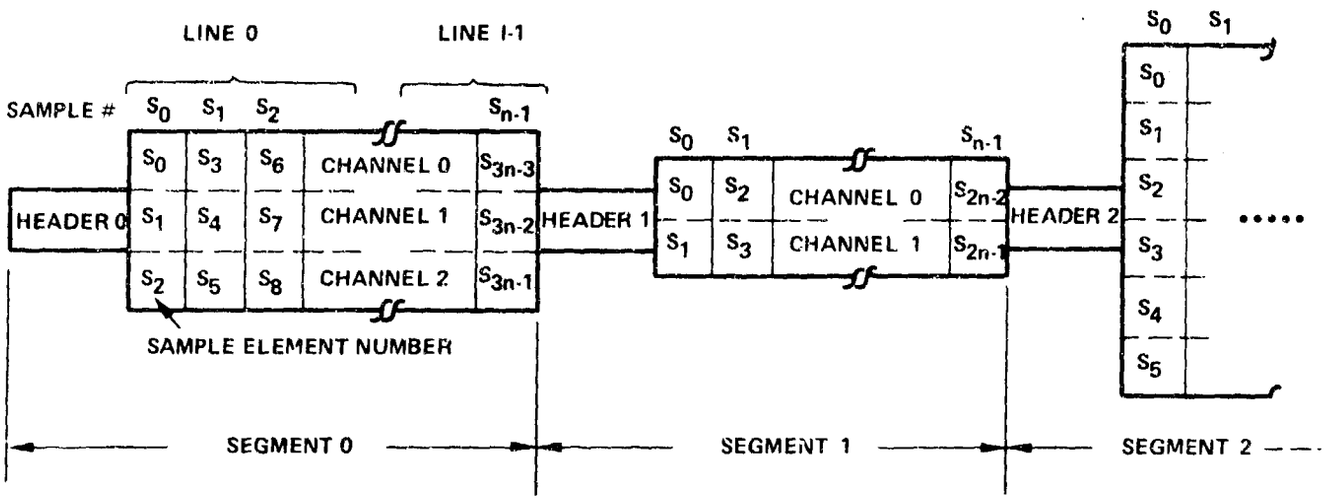


Figure 4 A Sequence of Signal Segments

If desired, one could insert the module "Sbsc" between "Scode" and "Sdecode" to simulate a binary symmetric channel with a specified error rate. Similary, by replacing " > new image.S.. with " | Stv", the compressed image could be viewed on the television monitor. The module string can be broken at any point for debugging or plotting purposes. For example, when designing a coding scheme, it would be nice to know the probability density function of each of the coefficients. For a given image, we can generate an approximation with the histogram and plotting modules: Sht 1 8 < image.S | Shist-1024 1023 | Splt | mp yields a plot, Fig. 5, which is a histogram of 8 coefficients resulting from Hadamard transforms on 1 by 8 subpictures. Intermediate signals can be saved by inserting the "tee" program which writes its standard input to its standard output, but also writes a copy to a specified file.

A partial list of existing signal processing modules follows:

Shead:	Put a signal header on a raw data file
Sunlace:	Separates an interlaced frame into 2 fields
Slace:	Interlaces 2 fields into a frame
Sht:	Computes Hadamard transform on arbitrary subpictures
Siht:	Computes inverse Hadamard transforms
Sct:	Computes Cosine transforms on arbitrary subpictures
Sict:	Computes inverse Cosine transforms
Sdpcmcod:	Performs DPCM encoding with specified linear predictor and quantizer
Sdpcmdec:	Performs DPCM decoding with specified linear predictor
Scode:	Performs specified quantization scheme for each channel

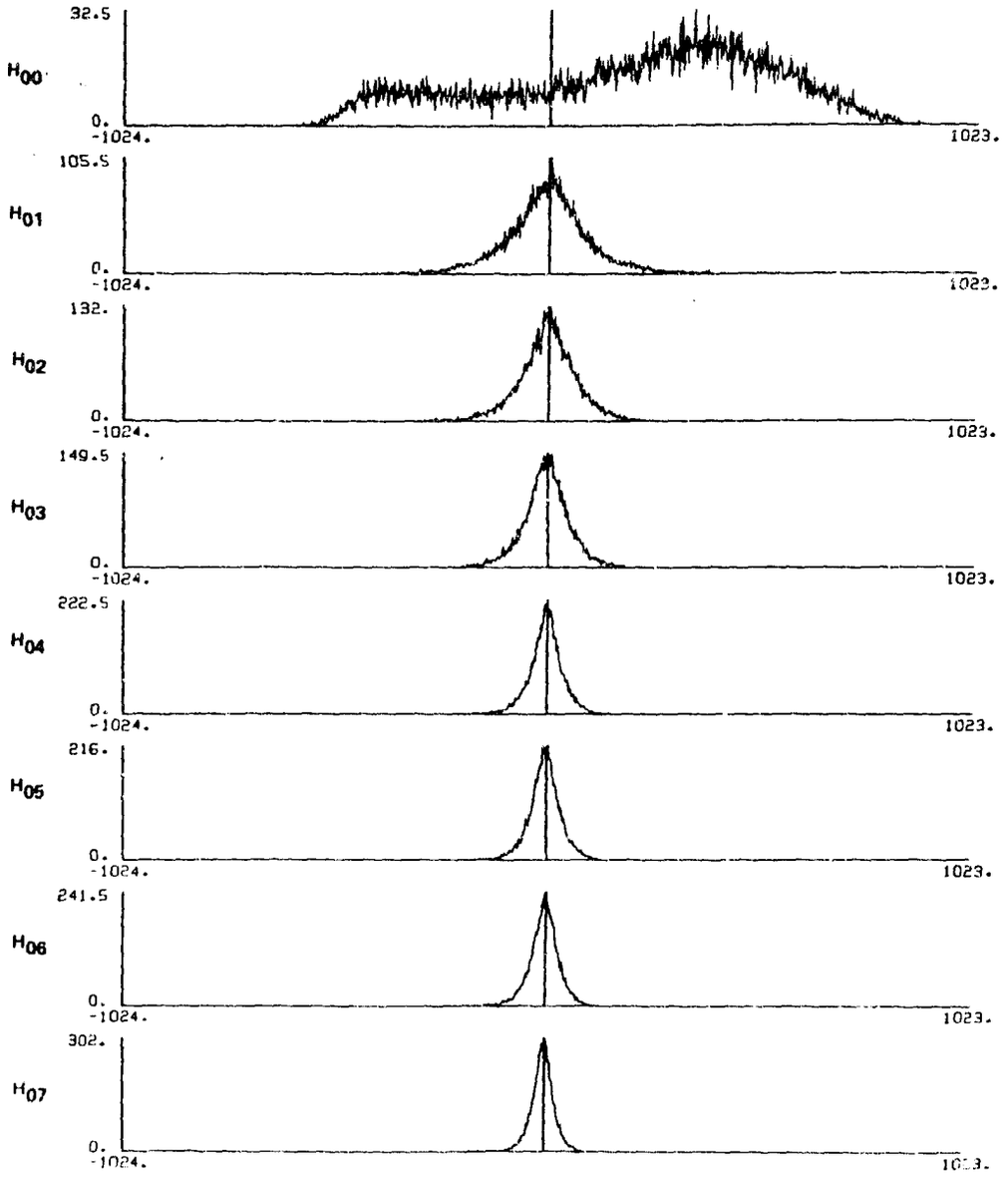


Figure 5 Histogram of Hadamard Transform Coefficients for 1 x 8 Subpictures

Sdecode:	Inverse of Scode
Sbsc:	Simulates binary symmetric channel with given error rate
Satos:	Convert ascii data to signal segment
Sft:	Computes Fourier transforms
Shalftone:	Display image on dot matrix plotter
Stv:	Display image on TV monitor
Smse:	Computes NMSE between two images
Shist:	Computes estimate of probability density function (histogram)
Splt:	Make vector plot with aces
Sdump:	Print signal header and data
Sift:	Computes Inverse Fourier transforms
S2dft:	Computes large 2-D Fourier transforms
Sconv:	Convolve two sequences
Slaplace:	Laplacian edge enhancement
Sgamma:	Alter gamma
Sinterp:	Zooms a decimate image with 2-D interpolation and filtering
Strans:	Transpose large arrays
Sconv:	Convert between signal data types
Slog:	Takes logarithm of signal data
Sscale:	Scales signal data
Smix:	Mixes signals streams

Table 1

Digital Image Processing Experiments I

(Issues of General Interest to Broadcasting)

Analysis of PCM Sampling Schemes

Determine the characteristics of the following sampling schemes in the various composite standards: 3 times the color subcarrier (3 fsc). 4 fsc. 3 fsc with phase alternate line coding (pale), and 3 fsc + δ such that an even number of samples are taken per scan line. Each of these schemes has advantages and disadvantages relating to sample rates and digital signal processing.

Sample Rate Conversion

Investigate the feasibility and the problems associated with the conversion from one sampling rate to another such as 3 fsc to 4 fsc. Problems such as the preservation of resolution and whether the rate conversion should be all digital or involve a D/A and an A/D conversion will be investigated.

Standards Conversion

Methods to implement cost effective standards conversion will be studied. In particular, interest will be directed toward techniques that will permit substantial commonality of future braodcast products between the different standards - - PAL-M, NTSC, PAL and SECAM.

Image Magnification and Manipulation

Picture Element interpolation schemes will be studied to find ways to generate digital zoom techniques

Table 1 (Continued)

Noise Reduction and Error Masking

Various techniques such as conditional line or picture element replacement will be studied to improve picture quality. These simple techniques will be compared to the performance of more complex digital processing techniques.

Digital Color Decoding

Digital decoding of composite color signals such as NTSC to YIQ or PAL to YUV interactions between this type of decoding process and the method of PCM sampling will also be studied. Particular emphasis will be placed on the feasibility of repeated digital recordings of video in the composite form.

Table 2

Digital Image Processing

(Issues of Interest Outside Broadcasting)

Image Enhancement and Restoration

A number of the more complex techniques such as inverse filtering, wiener filtering, bandwidth extrapolation and maximum entropy will be investigated. Emphasis will be placed on improved optical processor performance by enhancing the processor output. New device developments such as CCDs may make these signal processing techniques applicable to broadcast television.

Bandwidth Compression

The more promising adaptive 2 and 3 dimensional techniques using Hadamard transform, cosine transform, and DPCM coding will be investigated. Compressions of 4 to 1 (22 Megabits/second) have been achieved on broadcast quality color TV.

Wide Dynamic Range Image Processing

Processing of wide dynamic range infrared (IR) image data is of interest. Direct A/D conversion and recording of IR sensor outputs with dynamic ranges up to 13 bits (85 dB S/N) at 5 MHz bandwidth is possible. Bandwidth compression, image enhancement and pattern recognition techniques applied to this IR data is of interest

Information Extraction and Signal Identification

Automatic analysis of image data in both the image spatial domain and the transform domain will be considered. The evaluation of algorithms for automatic image tracking and autonomous acquisition for missile guidance is one application.

SUMMARY AND APPLICATIONS TO MISSILE GUIDANCE SIMULATIONS

A system capable of real-time simulations of image sequences typical to those encountered in imaging trackers and autonomous acquisition applications has been described.

Two distinct applications are the recording of real-time sensor data for the evaluation of algorithms and the synthesis of image data for the evaluation of specific hardware.

Recording of real-time sensor data permits extensive evaluation of the sensor itself and the testing of various algorithms for image enhancement and analysis.

Synthesis of real-time image data permits evaluation of hardware systems that use image data as input. This permits the generation of image sequences that would be otherwise hard to obtain. An example would be the manipulation of image data to represent the image obtained from a sensor on a vehicle that was in a terrain following mode of operation. The alternatives of complex terrain following simulators and actual flight tests are cost prohibitive in many situations.

1. D. M. Ritchie & K. Thompson, "The UNIX Time-Sharing System", C.A.C.M., Vol. 17, No. 7, July 1974, pp 365-375.

2. D. M. Ritchie, "C Reference Manual", in Documents for Use With the UNIX Time-Sharing System, Bell Telephone Laboratories, Sixth Edition, 1975.

Unclassified paper submitted for Workshop on Imaging Trackers and Autonomous Acquisition Applications for Missile Guidance, 19-20 November 1979, Redstone Arsenal, Alabama.

IMAGE PROCESSING USING VICAR

T. A. Nagy and J. D. Childs

Systems and Applied Sciences Corporation

ABSTRACT

Image processing techniques are extensively used by astronomers (typically on two-dimensional spatial data) in order to perform the functions of: image enhancement, edge detection, feature extraction, image filtering, image segmentation and pattern classification. Results of data processed by the VICAR (Video Image Communications and Retrieval) image-processing system are presented. The VICAR concept was initially developed by the Jet Propulsion Laboratory (JPL) in order to process images from lunar and Martian space probes. This concept is based on the system control of all I/O functions and a particular calculation or operation performed in one application program. Serial operation of one or more application programs will result in the desired process performed on the data. The utility of the VICAR image-processing system is demonstrated through a series of images including astronomical objects (visual and infrared) as well as patterns. The VICAR system can be utilized on either a main-frame computer (e.g. IBM S/360) or on a mini-computer (mini-VICAR developed for a DEC PDP 11/45). In addition, the results of a conversion study of VICAR to a DEC VAX 11/780 computer is presented.

I. VICAR - A DESCRIPTION

VICAR is a general purpose digital image-processing system that was developed at the Jet Propulsion Laboratory in 1966. Its primary applications were the processing of lunar and Martian space probe data, although it has been used for more general astronomical applications in the intervening years.

VICAR consists of a systems portion and a set of application routines. The image-processing functions are performed by the applications routines which are written in FORTRAN, assembly language, or some combination of the two languages. The systems routines, which are basically written in assembly language, control the execution of the application routines, perform the image data management and handle I/O. The user specifies the operations to be performed on the image data by definition of job specifications in the VICAR control language.

The VICAR systems routines comprise three sets of software:

VTRAN – An ancillary program that translates process descriptions in VICAR control language into a form recognizable by the VICAR system. The output of VTRAN is a set of Job Control Language

VTRAN (cont.) - and a task queue for a VICAR run. The task queue includes references to applications programs in their intended order of execution and pointers to necessary data sets and parameter blocks.

VMAST - The resident part of the VICAR system. VMAST consists of the executive for VICAR along with the general system I/O routines used by the applications programs. VMAST loads the transient supervisor VMJC between tasks.

VMJC - A transient task that prepares the computer environment for VICAR applications programs. VMJC interprets the job specification (output of VTRAN), sets up control blocks, opens data files, copies image labels, builds task parameter tables, etc. VMJC also overlays itself with the next applications program to be executed.

There is a set of I/O routines called VMIO that is resident in VMAST. These routines perform such functions as assigning logical device numbers, opening, closing, reading from and writing to data sets, loading tasks or data into core, obtaining task parameters and terminating a task either normally or abnormally. These routines were developed to save core space and to provide for the most efficient transfer of large quanities of image data. There exists also a general-purpose VICAR subroutine set which includes routines to perform data conversion, to check I/O operations and to perform magnetic tape utility functions.

The application programs perform the actual image manipulation and are transient routines that are called in by VMJC through VMAST. These programs employ the VMIO and VICAR general-purpose subroutines to perform their functions. There currently exists a massive set of image-processing applications routines that have been tested and utilized for several years. Included are programs that can perform image generation, grey scale transformations, algebraic operations, logical operations, image measurement, annotation, display, geometric operations (rotation, magnification), image combination, projection, correlation, filtering, and Fourier transform computation.

The body of applications program is expandable. The VICAR user can easily code and incoroprate new routines into the system. VICAR provides a particularly useful environment for testing new image processing algorithms. The modularity of the VICAR system permits any combination of applications programs to be employed.

VICAR facilitates data set management by means of a standarized data set labelling scheme. Each data set may have attached to it a set of labels consisting of the following parts:

system label - essential information such as the image size in lines and samples. Always present.

history label - provides processing history of the image. The history label is appended optionally with the execution of many of the applications routines.

user label - an optional title inserted by the user to identify the image.

Versions of VICAR have been developed for both batch and interactive environments. In the batch version the user can set up a job scheme, have it translated by VTRAN and submit the JCL and task queue output to the VICAR program. Line printer listings and photowrite displays can be generated to see the results of a particular run. With the interactive system, the user can issue a command at a terminal which will initiate the execution of a task on a previously-defined dataset. The interactive environment will also provide for intermediate display and processing verification during a job scheme, in addition to the kinds of output available with a batch system.

The user's interaction with VICAR is through the job control language. A typical job scheme or terminal session would include the following steps:

1. Allocation of data sets. Permanent image data sets are usually stored on magnetic tape, one file per image. Disk data sets may also be allocated for temporary image storage and access during execution of a VICAR scheme.

2. Specification of applications routines. Image processing applications routines are entered in the order in which they are to be executed. Included are the name of the routine, the symbolic names for the necessary input and output data sets (allocated in step 1), the output image size, and any relevant keyword options or parameter values required for the application function desired.

3. Labelling of data sets (optional). User labels may be added or replaced on image data sets at any point during the VICAR run.

The VICAR language also provides for setting up DO-loops to facilitate repetitive operations. A set of VICAR control statements called a procedure may also be built and given a reference name. Such a procedure may then be invoked from within a job scheme by name with arguments if necessary.

An excellent overview of VICAR and its uses may be found in Reference 1. More specific information about the VICAR system may be found in References 2-4. A feasibility study was made to the effort required to convert mini-VICAR from the DEC PDP 11/45 to a DEC VAX 11/780. (Reference 5).

II. DIGITATION OF THE DATA

Computer manipulation of data and subsequent image processing of "picture" data is possible only if the data exist in digital form. A discussion of the various methods to accomplish this goal is given by Castleman (Reference 1).

The character (pattern) data present here represent a portion of a data page captured on 35 mm microfilm. A portion of one of the frames in turn was input to an EMR (model 6584) photoelectric Optical Data Digiter (ODD). The test data frame was made on the ODD which was set up for a production operation and so the optics were not permitted to be modified to allow digitization of an entire 35 mm frame. The sensor is an EMR model 575 Image Dissector with a + 15% uniformity over a 90% area. It is possible to generate

an image of up to 4096 x 4096 (picture elements) pixels (test image was done at 512 x 512) with a static addressing accuracy of 3% and a repeatability of 0.1%. The ODD transfers the input image via the optics to the sensor which is an intensity function detector. The information is then processed by an intensity function encoder to provide a 256-level, binary-intensity-digital output (grey level).

The digital version of Comet Kohoutek was provided by Dr. D. Klinglesmith (NASA/GSFC) who generated the digital image from a photographic date on a Photometric Data Systems (PDS) microdensitometer, model 1010A. The data was scanned at the full scan rate (S=255) with a 50 μm spot size. The resultant digital image was 1808 by 1981 pixels.

Both of the above digital images had a VICAR compatible header added to the beginning of the data so that the images could then be processed by VICAR applications programs.

III. IMAGE PROCESSING

There are a vast number of techniques and procedures that are employed in the image processing and analysis of astronomical data. This discussion will limit itself to three rather general areas: image enhancement, edge detection and pattern recognition.

Image Enhancement

In the process of creating a digitized image there are several factors which can corrupt the desired image. The optics of the telescope and digitizing systems may geometrically distort the image. Film emulsion may be nonuniform across the plate, grain size and mottling may generate an overall noise level, the emulsion sensitivity to the incoming radiation may be very nonlinear. The sensitivity of the digitizer may contribute undesired effects upon the final image.

There are several routines in the VICAR applications library which can be used to correct for geometric distortions. GEOM and LGEOM are used to perform spatial transformations. One technique is to have a grid of fiducial marks (reseaux) superposed upon the imcoming image. The digitized image can then be registered to a standard, undistorted grid and then the reseaux can be "averaged out" of the image.

Sensitivity corrections on the grey scale can be performed by point operations on each pixel (picture element). If a simple sensitivity transfer function suffices to make the overall correction, the routine STRETCH may be employed. STRETCH performs grey scale transformations using either a function or a table. Position-dependent sensitivity corrections can be calibrated by taking a series of flat field images taken at known exposure times. A photometric correction file is generated and further exposures can then be photometrically corrected using such routines as FICOR and MICOR.

Noise removal can be performed in several ways. Periodic noise may be generated by imperfect data transmission. The Fourier transform routines FFT1 and FFT2 are used to identify and help eliminate periodic signals present

in 1- and 2-dimensional images. Random noise can be supressed by means of applying highpass or lowpass filters.

Figure 1 is an image of the Comet Kohoutek. The image can be considered to be made up of the slowly-varying broad features of the comet superposed over a star field consisting of randomly-located sharply-spiked objects. Depending upon the application, it may be desirable to emphasize either of these two features of the image.

The resulting Kohoutek digital image split up the spatial picture into two regions as shown in Figure 1. Region A contains the head and nearby tail features of the comet. Region B is basically a star field.

The Fourier transforms of regions A and B are shown in Figures 2 and 3. Both transforms exhibit high frequency components away from their centers. Such high frequency contributions are due mainly to the presence of the sharp-spiked stars in the raw image. A predominant feature in the transform of the comet region in Figure 2 is the strength of the low frequency components located about the center of the figure. The presence of the comet in region A contributes these low frequency features. Either the comet or the stars may be supressed in the image by multiplying the transform by an appropriate function and performing an inverse Fourier transform to generate the enhanced image.

Edge Detection

Edge enhancement is a process of emphasizing the grey level gradient at the borders of objects in an image. Several techniques may be employed: subtracting a blurred image from itself and scaling the difference, filtering the image with an appropriate impulse filter, or taking the derivative of the image.

Figure 4a shows the image of several characters which were generated at a constant grey scale intensity. A second image was generated by shifting the original image one sample to the right and one line downward. This second image was subtracted from the original image and an absolute value was taken of the difference. The resulting image is shown as Figure 4b, where the edge of the characters can be seen to stand out. This "45° derivative" technique is most useful for distinguishing regions of changing grey scale gradient from regions of constant grey level.

The characters of Figure 4a were deliberately "blurred" in Figure 4c by applying a box-shaped lowpass filter to the image (using the routine BOXFILT). This blurred image was used to show the edge enhancement technique that employs an impulse filter. The impulse filter for edge enhancement is basically a positively-peaked function surrounded by negative sidelobes. Figure 4d is the result of applying such a filter to figure 4c. An undesirable feature of this technique is a "ringing" effect or overshoot.

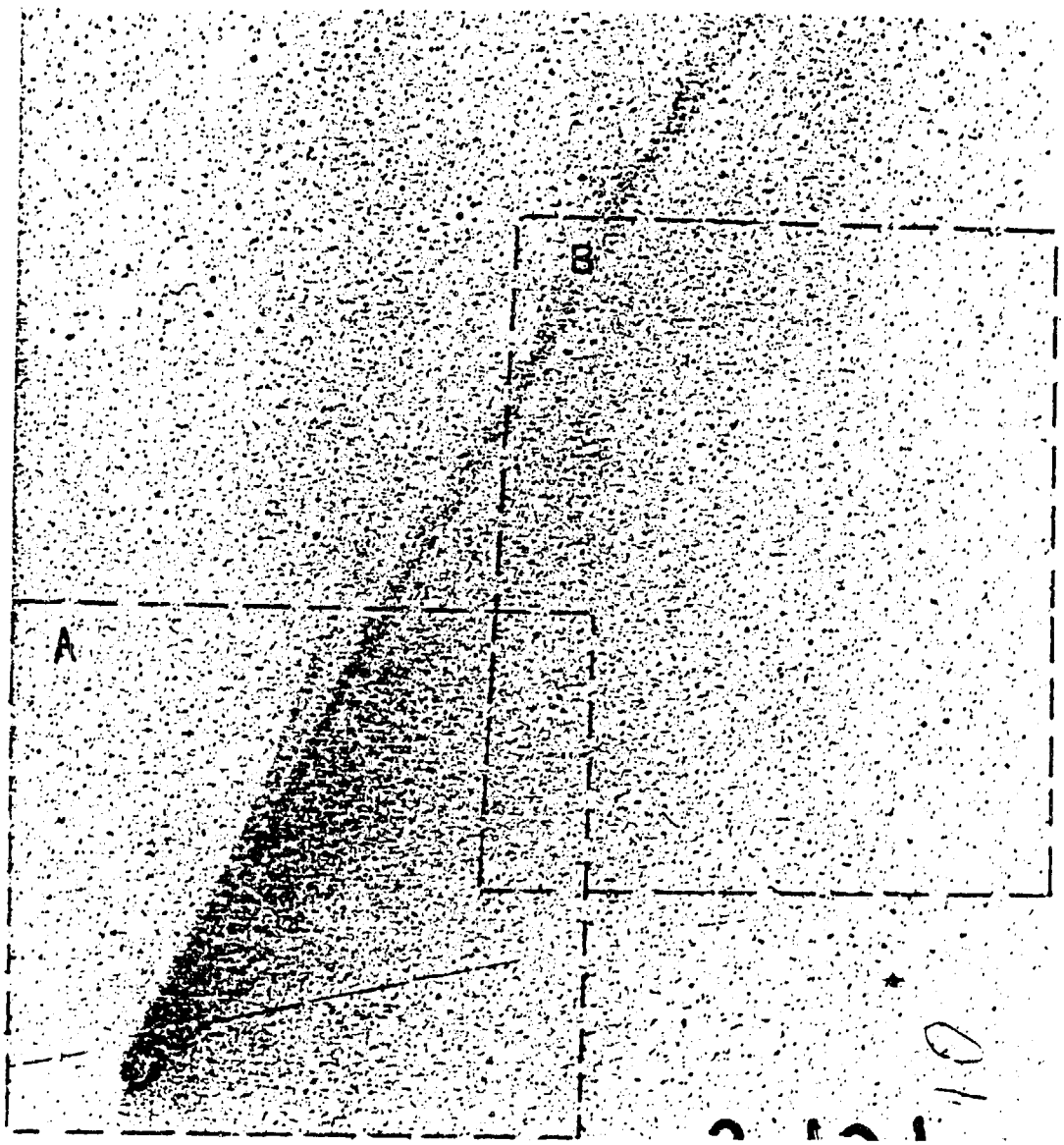


Figure 1. Photograph (spatial image) of comet Kohoutek.

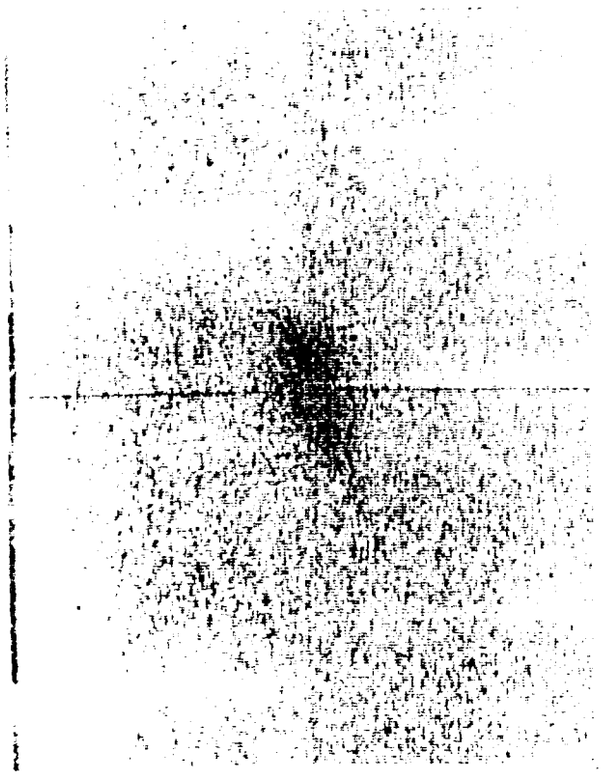


Figure 2. Fourier transform of comet.

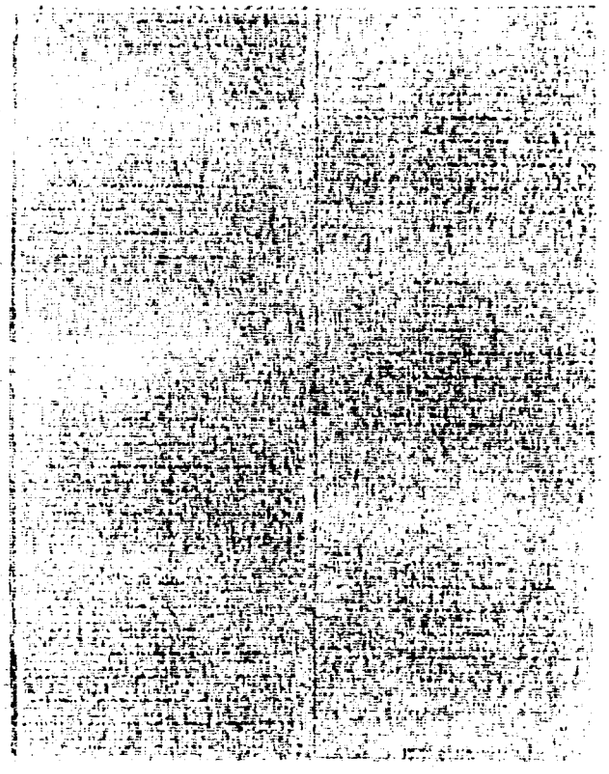


Figure 3. Fourier transform of star field.

Figure 4a. Original Image.

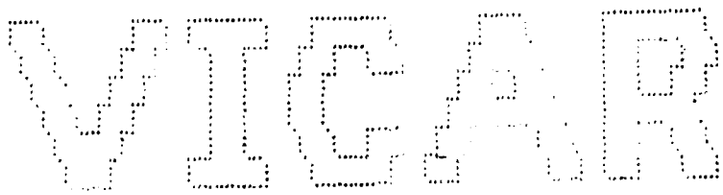


Figure 4b. 45°-derivative technique.

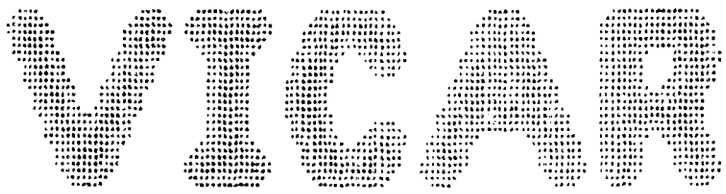


Figure 4c. Blurred Image.

Figure 4d. Impulse filter technique.

Figure 4. Examples of edge detection.

Pattern Recognition

The general goal of pattern recognition in remote sensing applications is to extract certain information from an image concerning the objects within that image. The steps involved are object isolation, feature extraction, and object classification. For example, a photograph of a star field can be analyzed to determine the location and magnitude of the individual stars, here, the "features" are the position and intensity of the stars. As another example, a program was developed at JPL to isolate galaxies within a star field image. These algorithms are described in two JPL publications (References 6,7).

Figure 5 is a digitized image of a page from an astronomical catalogue. What is desired is to set up an automatic procedure for extracting the character information from the image. In that way, the entire catalogue can be scanned and digitally processed to produce a catalogue data base for a computer.

We recognize that such an image as figure 5 contains only a small closed set of possible characters, namely the numbers zero through nine. A possible approach for doing the analysis is:

1. Register the image. If the decimal points are used as fiducial marks, the image may be rotated and geometrically stretched onto a standard grid. The approximate location of the characters can then be inferred.

2. Position each character. For each character position, a photocenter calculation can be made to identify the precise location of the center of the character.

3. Identify the character. Two techniques have been used for this purpose. One technique is to perform photocenters on the top and bottom halves and the left and right hand sides of each character to obtain position information on the different parts of each character. This process was found to uniquely identify each character in three steps (Reference 8). The second technique used was to compare each of the characters with a registered standard character set. The mean value of the image difference of the unknown character and each of the characters in the standard set are calculated; the minimum mean value difference will yield the correct identification.

Figure 6 shows a reordered image of the characters of figure 5 (generated using the VICAR routine INSECT). The top row of characters was designated the standard set and was subtracted from each of the subsequent rows. The ratio of the mean value to the standard deviation of the image differences was extracted using the routine BOXSTATS and is shown in Table 1. In all cases except for the character "three" there is a striking identification to the actual character. The "three-five" ambiguity could further be resolved with one more additional test, such as a photocenter calculation on some portion of the character.

Figure 5. Image of page from a star catalogue.

Figure 6. Image of reordered characters.

Table 1 - Table of Mean/(Standard Deviation) of Difference Of Standard Character Set From Other Characters.

Standard Character Set

	0	1	2	3	4	5	6	7	8	9
0	16	8	9	9	8	8	10	9	8	11
1	9	19	14	12	9	12	7	11	10	8
2	10	12	17	10	8	9	8	10	9	7
3	8	10	9	15	7	10	7	9	8	8
4	10	13	13	12	11	12	7	8	9	9
5	9	11	10	17	8	16	7	10	8	8
6	11	8	9	8	6	7	0*	9	9	8
7	9	9	9	11	6	9	7	15	8	7
8	9	10	10	9	7	8	9	9	22	8
9	11	8	9	10	8	11	8	8	8	15

Note: There was only one character "six" in the image.

IV CONCLUSIONS

VICAR is a versatile general-purpose image processing system. It has had years of proven experience at many sites in performing remote sensing applications. VICAR provides a general framework for image work; in addition to the large set of general and specialized applications routines existing, new algorithms and routines can be developed and easily implemented in a high level language (FORTRAN) into the VICAR system. There are batch as well as interactive versions of VICAR. The interactive version, mini-VICAR, has been implemented on a PDP-11/45.

Figures 2 through 6 in this paper are the result of line printer output on standard 1200 lpm printers. The techniques demonstrated here yield definitive results for identification of patterns, noise removal and edge detections with very standard software, computers, peripherials and commercially available digitizers.

REFERENCES

1. K. R. Castleman, Digital Image Processing, Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1979.

2. J. B. Seidman, VICAR Image Processing System Guide to System Use, Publication 77-37, Jet Propulsion Laboratory, Pasadena, California, May, 1977.

3. "VICAR-Version 3", JPL Digital Image Processing Manual, Section 1, Jet Propulsion Laboratory Internal Document 68-369, Pasadena, California, August 30, 1968.

4. Image Processing Laboratory User's Documentation of Applications Programs, Jet Propulsion Laboratory Internal Document No. 900-670, Pasadena, California, October 11, 1974.

5. J. D. Childs, R. H. Cornett, A. A. Henden, and T. A. Nagy, Final Report-VICAR Conversion Study, Systems and Applied Sciences Corporation, Riverdale, Maryland, September, 1979.

6. J. J. Lorre and D. J. Lynn, Application of Digital Image Processing Techniques to Astronomical Imagery 1977, Jet Propulsion Publication 78-17, Pasadena, California, April 15, 1978.

7. J. J. Lorre, Application of Digital Image Processing Techniques to Astronomical Imagery 1978, Jet Propulsion Laboratory Publication 78-91, Pasadena, California, November 1, 1978.

8. T. A. Nagy and J. M. Mead, "Digitation of Astronomical Catalogues," Bulletin of the American Astronomical Society, 9, 598, 1978.

Unclassified paper submitted for Workshop on Imaging Trackers and Autonomous Acquisition Applications for Missile Guidance, 19-20 November 1979, Redstone Arsenal, Alabama.

TARGET DETECTION USING HYBRID DIGITAL-ANALOG CORRELATION TECHNIQUES

M. Wohlers and J. Mendelsohn
Research Department
Grumman Aerospace Corporation
Bethpage, New York 11714

ABSTRACT

Hybrid system concepts are discussed that utilize two-dimensional digital image processing together with analog optical matched filtering to provide generalized correlation operations of interest for target detection in cluttered backgrounds. The systems offer the advantages of providing many nonlinear image processing and enhancement operations that can best be accomplished in a digital fashion — together with a matched filtering or correlation operation that is most efficientily accomplished in an analog fashion using Fourier optical techniques. An example of the application of these ideas to the location of a building will be described.

INTRODUCTION

The field of digital image processing, particularly image enhancement, provides many examples of techniques such as histogram equalization, edge detection, thresholding, etc. that are highly nonlinear operations and that can be achieved very efficiently using special purpose hardware. On the other hand, the matched filtering or correlation operation between a scene and a template whose aperture is commensurate in size to that of the scene's, imposes a severe digital computational burden that is still well beyond the state-of-the-art of small volume digital hardware implementation. Fortunately, this is the area where analog Fourier Optical techniques excel and so one is naturally led to consider merging these two technologies for the task of target identification in cluttered scenes.

Although the basic idea of merging digital and analog image processing technology is attractive, it is not at all clear how we should explore the wide variety of combinations of image processing operations that can be achieved. This paper presents an example of a preliminary study that was done using computer simulation, that attempted to explore some of these operations that appear to be most interesting for the specific task of target location.

DIGITAL PREPROCESSING TECHNIQUES

An investigation was made of the use of digital enhancement techniques to preprocess images before analog matched filtering or correlation techniques are used to determine target location in the real or sensed scene. The reasons for the use of digital enhancement are two fold; first is the fact that the sensed scenes, in general, have poorer contrast than the model scenes from which the matched filters are to be made, and secondly, the model scenes do not contain all the target detail present in the actual images so that the effective "noise level" due to the lack of this detail can be large in an analog correlation processor. Both of these factors are demonstrated by the sensed scene and the corresponding model scene for the building target shown in Fig. 1; note the poor scene contrast and the additional roof detail in the actual scene (as well as the other objects in the scene such as automobiles in the parking lot that appear above the building).

The digital enhancement techniques that were investigated were selected with a view toward their final implementation in hardware. Thus, the level of processing was kept to a minimum. Specifically the images, represented by the mxn pixel array, F(i,j), was operated on by a translation invariant filter whose point spread function is represented by the qxr pixel array H(i,j), to yield the enhanced image Q(i,j) (again an mxn pixel array) given by the convolution

$$Q(i,j) = \sum_{n_1=1}^{q} \sum_{n_2=1}^{r} H(n_1,n_2)\, F(i-n_1+2,\ j-n_2+2)$$

The filters selected for investigation involved the smallest arrays, H(i,j), that yield interesting results namely, 3 x 3 arrays. The following six arrays were utilized in the study — the first three correspond to high pass filters and the last three to Laplacian filters

$$H_1 = \begin{bmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{bmatrix} \qquad H_2 = \begin{bmatrix} -1 & -1 & -1 \\ -1 & 9 & -1 \\ -1 & -1 & -1 \end{bmatrix} \qquad H_3 = \begin{bmatrix} 1 & -2 & 1 \\ -2 & 5 & -2 \\ 1 & -2 & 1 \end{bmatrix}$$

$$H_4 = \begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix} \qquad H_5 = \begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix} \qquad H_6 = \begin{bmatrix} 1 & -2 & 1 \\ -2 & 4 & -2 \\ 1 & -2 & 1 \end{bmatrix}$$

The images to be processed were also subjected to an initial nonlinear intensity stretching before the filtering operation was performed. The variation of intensity on the raw digital data was 0 to 255 and this variation was mapped into 100 to 137 on the model and 100 to 160 on the sensed scene by assigning all measured intensities less than or equal to 100 to 100 and all intensities greater than or equal to either 137 or 160 to 137 or 160 respectively; the intervening intensity values were linearly scaled between these values. This preprocessing was selected by first displaying the results of various stretching operations to a human who then decided that the selected levels yielded the best contrast in the resulting image.

The images processed by the nonlinear stretching were first correlated directly without any additional filtering — the results indicated that this would not be acceptable since the resulting correlation matrix had a maximum whose position was not related to the relative position of the target in the scene.

The filtering operation was then applied to both the sensed scene and the model image. The magnitude or intensity obtained with the first Laplacian filter (H_4) is shown in Fig. 2. Note that the resulting model image was thinned by

eliminating all pixels except the ones shown in Fig. 2b. The magnitude of the resulting images (Figs. 2a and 2b) were then correlated with each other and the resulting correlation matrix had a maximum value in the 25th row and 13th column which corresponds (to within 1 pixel accuracy) to the relative position differential of the target between the model target coordinates and the sensed scene coordinates. Figure 2c shows a scan through a portion of the 25th row of the correlation matrix showing the position of the peak and its relative sharpness (for reference purposes the scene was 100 pixels wide). Figure 3 shows the results of the same operation applied to the Laplacian-filtered sensed scene as various amounts of thinning are done to the Laplacian-filtered model. Note that the correlation peak is sharpened as more thinning is done — this is because the sensed scene did not contain the row of detail on the roof which was in the model and which then become additional noise in the correlation operation.

ANALOG OPTICAL CORRELATION

Once the preprocessing operation has been selected, the model scene or scenes can be processed and the results stored. The real time implementation of the concept as a target locator then requires that a system be configured that will accept a "live" scene as sensed by appropriate devices, e.g., visual, infrared, or microwave, and the preprocessing operation done to the scene before the final correlation with the previously stored model scenes. We envision a system in which a digital image preprocessing module will provide the necessary scene conditioning or preprocessing. The two-dimensional image emerging from this module will be impressed upon an optical beam that will then be processed in an analog fashion through an optical matched filter system shown schematically in Fig. 4. This system used fixed holographic optical elements that allow (Fig.4b) for the parallel processing of the incoming scene with many different matched filters. Thus one can simultaneously achieve the correlation of the preprocessed live scene with many different modeled scenes — this allows for the identification of the target in the scene as well as the accommodations for different target orientation or scale. References 1 through 3 discuss some of the ramification of the analog optical matched filtering technology. A further note is worth commenting upon since it will impact the necessary optical components and that is the fact that a coherent optical matched filter is not required if the scene preprocessing and model preprocessing results are first converted to intensity images, i.e., magnitudes taken, before the correlation operation (see Ref. 4). The results described in Figs. 2 and 3 of this paper employed "intensity" images in the correlation operation and thus would be suitable for an incoherent matched filter implementation.

SUMMARY

This paper presented the results of very preliminary studies that demonstrate that simple digital image enhancement techniques could be applied to sensed imagery so that subsequent correlation (that can be achieved very efficiently with analog optical processors) yield target location even though initial sensed imagery has poor contrast and the available models of the targets contain only partial detail. The resulting hybrid digital/analog systems offer the potential of the high processing rates achievable with analog optical systems together with the nonlinear image processing operations that can be implemented most efficiently with digital processing.

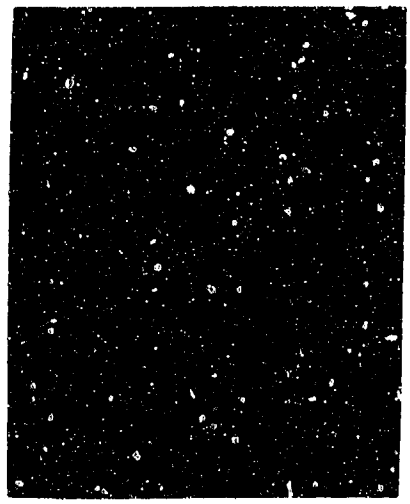
REFERENCES

1. Grumet, A., U.S. Patent #3,779,492, "Automatic Target Recognition System," December 1973.

2. Leib, K., et al., "Aerial Reconnaissance Film Screening Using Optical Matched-Filter Image-Correlator Technology," Applied Optics, Vol. 17, p 2892, September 1978.

3. Mendelsohn, J., et al., "Digital Analysis of the Effects of Terrain Clutter on the Performance of Matched Filters for Target Identification and Location," SPIE, Vol. 186, 1979.

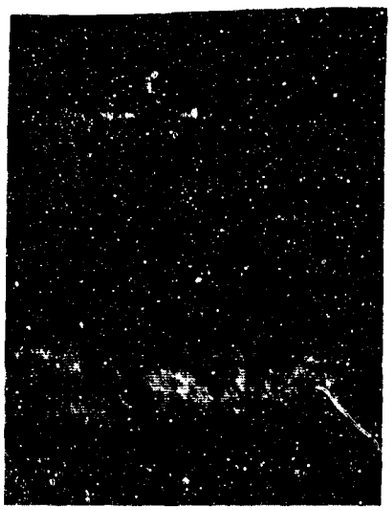
4. Wohlers, M., "Optical Matched Filtering of Incoherent Images," Applied Optics, Vol. 18, November 1979.



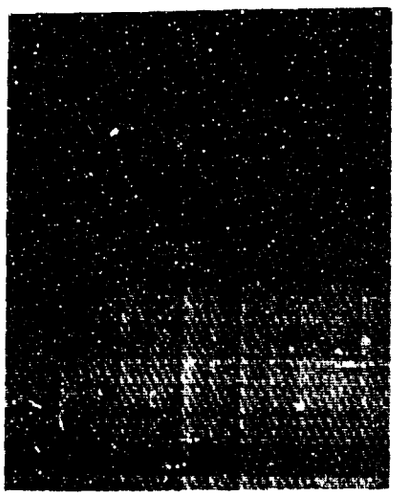
a) Sensed

b) Model

Fig. 1 Sersed and Model Scene to be Correlated - Original Scenes Before Preprocessing

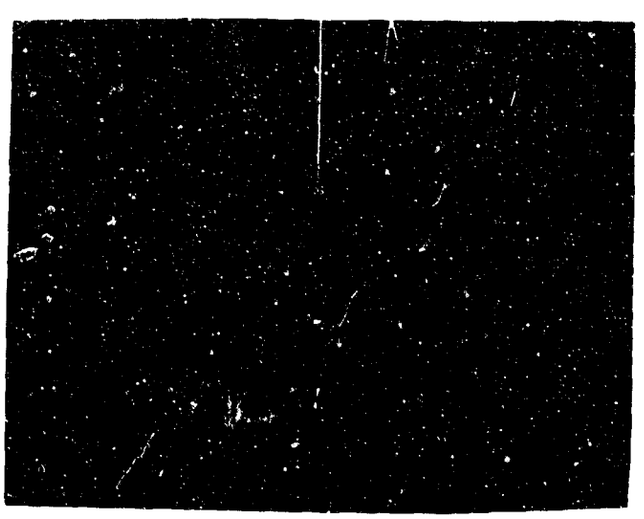


a) Filtered Scene



b) Filtered Model (Thinned)

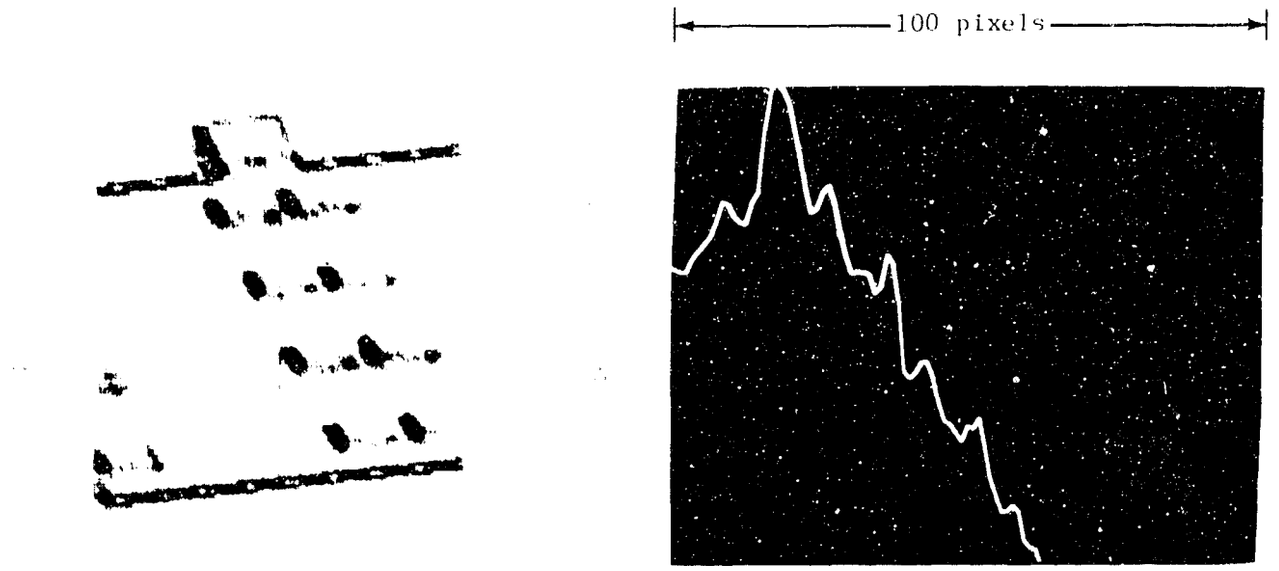
13 pixels

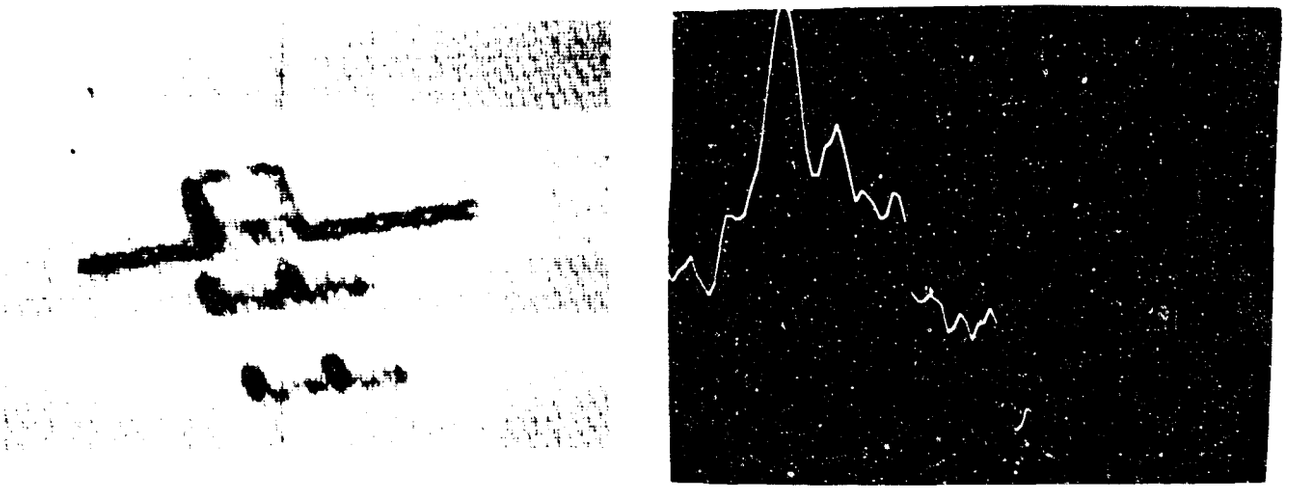


100 Pixels

c) Scan Through 25th Row of Correlation Function of a) and b)

Fig. 2 Correlation Between Preprocessed Scenes





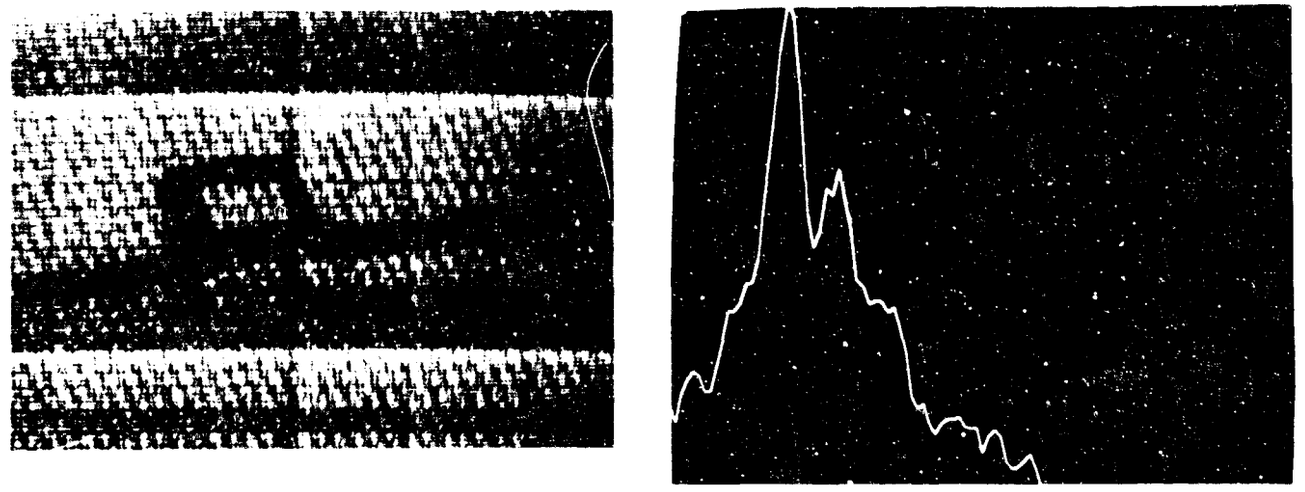


Fig. 3 Model Thinning Effects in Correlation with Scene

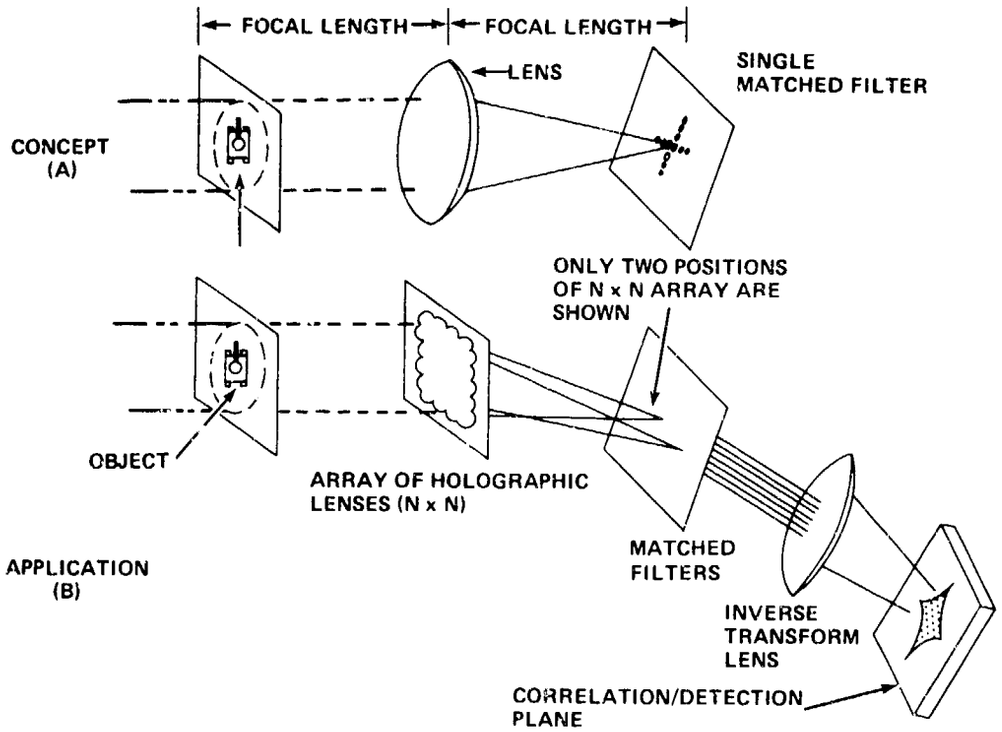


Fig. 4 Coherent Optical Systems to Achieve Matched Filter or Correlation Processing

Unclassified paper submitted for Workshop on Imaging Trackers and Autonomous Acquisition Applications for Missile Guidance, 19-20 November 1979, Redstone Arsenal, Alabama.

PERCEIVED IMAGE QUALITY-TV IMAGE TRACKING SYSTEMS

G. Sandhu
General Research Corporation, Huntsville, AL 38505

E. V. LaBudde
Burroughs Corporation, West Lake, CA 91361

ABSTRACT

Results of the subjective tests designed to determine the importance of random noise interference in various portions of the video band are reported. Broad, narrow and mixed bands of white noise distribution effects throughout the video band are investigated. Good agreement results between the experimental data and predicted performance from computer simulation models. The analysis indicates that noise equalization, by preemphasizing and deemphasizing a certain portion of the video, results in a considerable improvement in resolution and gray scale of the image.

1. INTRODUCTION

Recent advances in solid state imaging devices and solid state analog memory and correlation devices have made it feasible to develop a practical fire-and-forget terminal seeker employing image correlation and television trackers. The basic performance limitation, however, has been the perceived limiting resolution and gray scale of the target image in a cluttered environment. In this study, we seek to answer as precisely as possible the following questions: What is the relative importance of random noise in various parts of the video spectrum? How does the human visual mechanism resolve the image in the presence of noise? Is it possible to model the eye as a system block with reasonable accuracy in arriving at an overall assessment of interfering effects? How effectively does noise equalization improve system performance? Preliminary answers to these questions are reported here. They are "preliminary" because they depend to a large extent on the test equipment used, the viewing conditions, and observer judgements.

Broad, narrow and mixed bands of white noise spectrum are added to the output of an imaging sensor viewing the standard resolution chart. The limiting resolution and gray scale of the resultant image are observed by two independent observers without the "a priori" knowledge of the simulated conditions. The results are recorded as a function of signal to noise (S/N) ratios. A theoretical model of predicting the limiting resolution is developed and its performance is compared with the experimental data. Good agreement results between experiment data and the predicted performance from computer simulation models. To minimize noise interference effects, noise

equalization techniques are applied. A certain portion of frequencies in the video signal is boosted (preemphasized) before the video is processed and the same band is suppressed (deemphasized) at the receiving end. Considerable improvement in both limiting resolution and gray scale results over all possible ranges of S/N ratios.

2. HUMAN VISUAL MECHANISM

The conventional means of specifying the S/N ratio of an image resolving system is to relate the peak signal to rms noise. This figure of merit does not relate the limitations imposed by the noise to the increased Modulation Transfer Function (MTF) at spatial frequencies less than the limiting spatial frequency. It is desired to evaluate the broadband implications more rigorously, when the observer's ability to recognize information is impaired. The primary parameters [1-4] relating to this impairment are, signal MTF, target contrast, eye Contrast Threshold Function (CTF) and noise power spectrum. The spatial domain interrelation between these parameters is shown in Figure 1. The eye contrast threshold response at 60 centimeters viewing distance corresponds to the acquisition mode. Once the target is acquired, the observer moves closer (20 centimeter viewing distance) for identification.

As seen from Figure 1, the limiting resolution, f_ℓ, is 620 TV lines (signal MTF = eye CTF) in the acquisition mode and 660 TV lines in the identification mode. The increased identification limiting resolution is due to the fact that the viewer has zeroed in on a particular object and is not interested in the spatial frequencies which clutter the object. The perceived limiting resolution in the presence of noise can be derived by subtracting the square-root of the sum of the noise modulation squared and the eye contrast threshold squared or the value of f_ℓ satisfying the following relation

$$\int_0^{f_\ell} NPS(f)df + CTF^2(f_\ell) = A_t \cdot MTF^2(f_\ell).$$

where A_t is the apparent target contrast. For a typical case of 0dB (1 volt rms white noise over 5 MHz or 400 TV lines) S/N ratio, the perceived limiting resolution in the acquisition mode decreases to 180 TV lines. We will varify this from the experimental data in the following section.

3. EXPERIMENTAL RESULTS

The test set used to simulate noise conditions is illustrated in Figure 2. In a test, two observers view an image of a resolution chart and gray scale with no knowledge of simulated noise conditions. All noise is frequency-limited with a response of 18 dB/octave up to 3 MHz and a response of 12 dB/octave from 3 to 5 MHz. A standard EIA resolution chart is viewed on a monitor at a distance of about 60 centimeters from the observer under normal lighting conditions. The detailed test set description is described in [4].

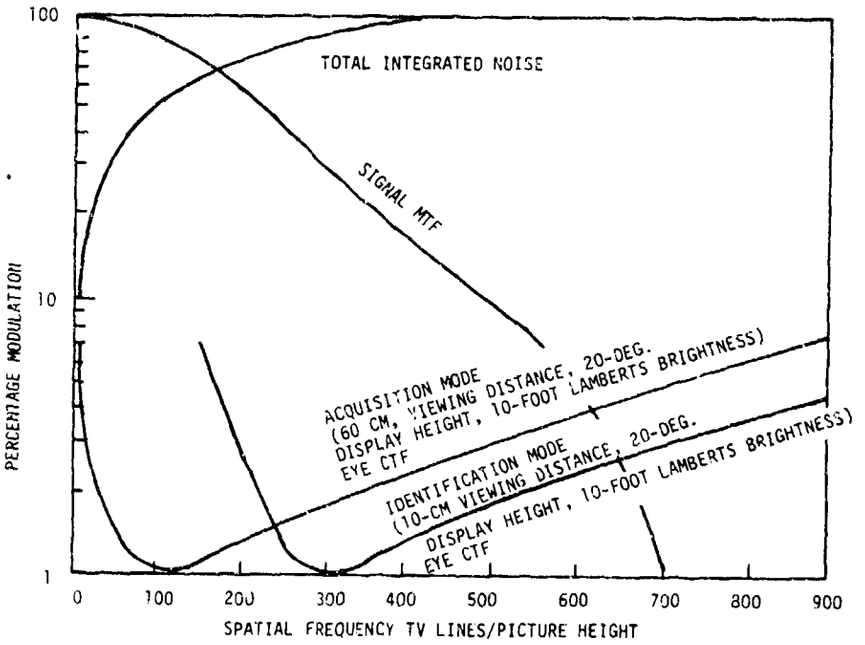


Figure 1. Human Visual Spatial Domain Interrelation Parameters

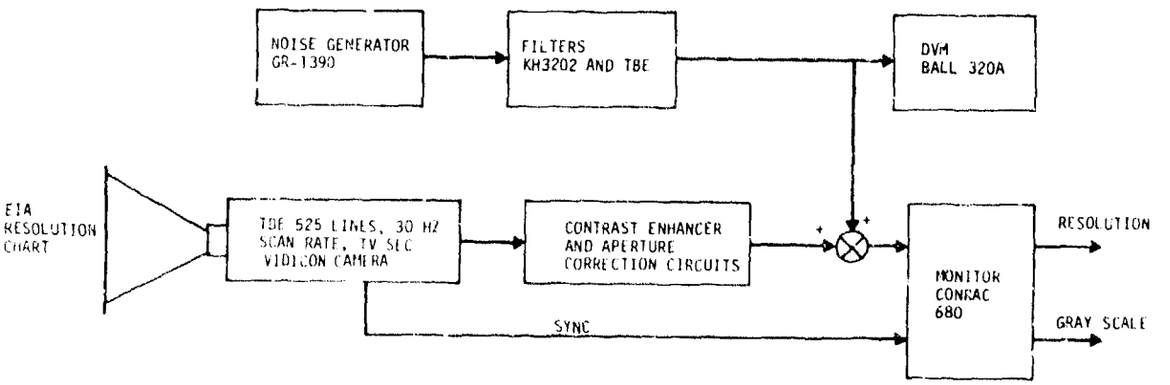


Figure 2. Test Set Discription

3.1 WIDE BAND GAUSSIAN WHITE NOISE

White gaussian noise is added to the video signal before displaying the image of the resolution chart on the monitor. The corresponding resolution and gray scale is observed for a specified S/N ratio. The results are shown in Figure 3. It is seen that observed horizontal resolution is very close to the theoretical model prediction over a wide range of S/N ratio. Both horizontal and vertical resolution appear to fall off linearly (20 TV lines/dB) for S/N ratio below 10 dB. The gray scale is also a linear function of S/N ratio, but over a large range (0-20 dB).

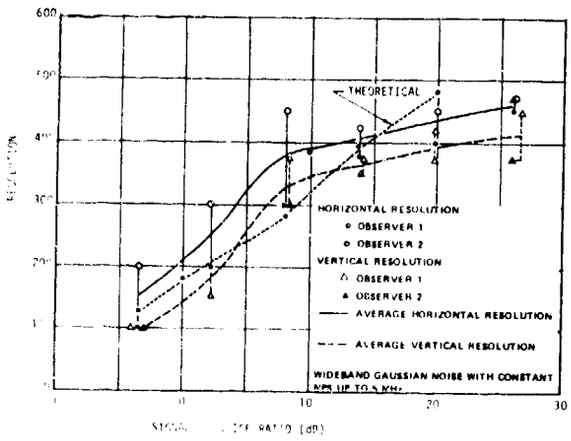


Figure 3a. Wide Band Gaussian White Noise Effects (Continued)

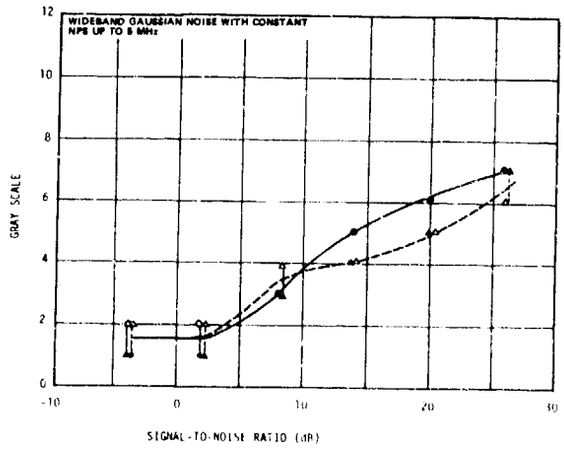


Figure 3b. Concluded

3.2 BAND-LIMITED LOW-FREQUENCY NOISE

The perceived image quality resulting from band-limited, low-frequency noise simulations is shown in Figure 4 for a constant noise power spectrum and a constant noise rms value. It is interesting to observe in Figure 4(a) that vertical resolution falls rapidly up to a cutoff frequency of 10 KHz. The same phenomenon also appears, although less dramatically, for higher values of S/N, ration [4]. It appears that low-frequency noise, particularly less than 10 KHz, is extremely objectionable. The rapid decline of horizontal resolution at 1 to 5 MHz can be expected since most of the white Gaussian noise lies in this region for constant rms input noise. The gray scale appears to fall uniformly with upper cutoff frequency in all the cases as shown in Figures 4(b) and 4(d).

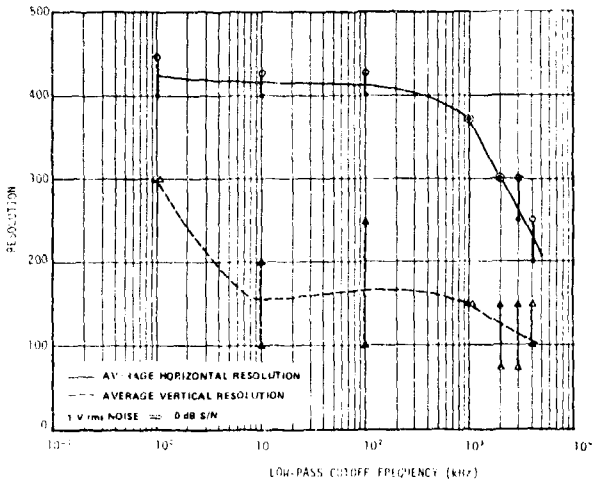


Figure 4a. Band Limited Low Frequency Noise Effects

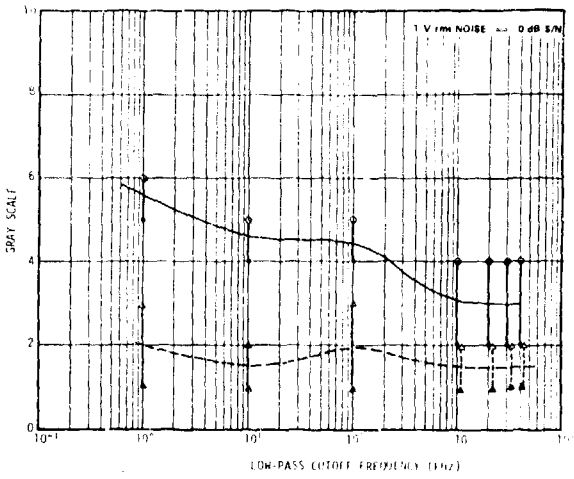


Figure 4b. Continued

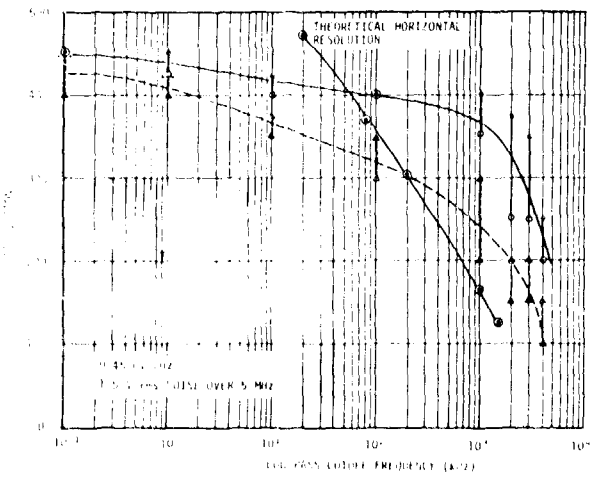


Figure 4c. Continued

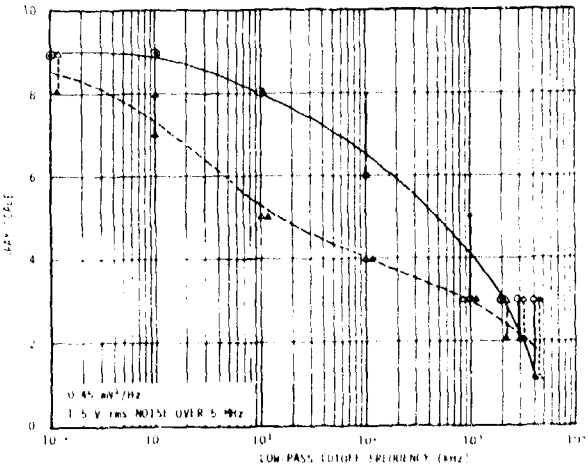


Figure 4d. Concluded

3.3 BAND-LIMITED HIGH-FREQUENCY NOISE

White Caussian noise from a noise generator is passed through a high-pass filter whose cutoff frequency is varied from 0.1 to 4 MHz. For each reading, the noise source output is adjusted to provide a constant rms noise input to th video signal. The results are shown in Figure 5(a) and 5(b). Figure 5(c and 5(d) show the results of constant noise power spectrum.

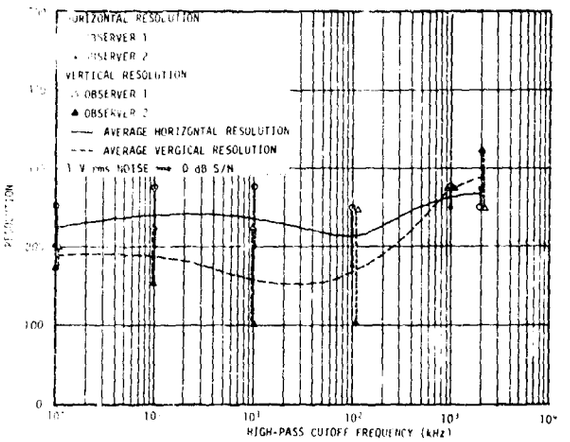


Figure 5a. Band Limited High Frequency Noise

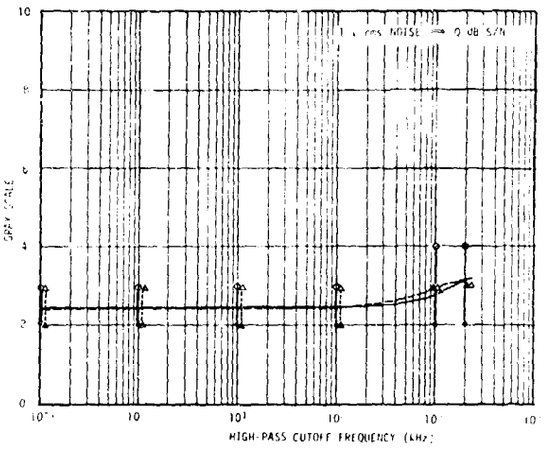


Figure 5b. Continued

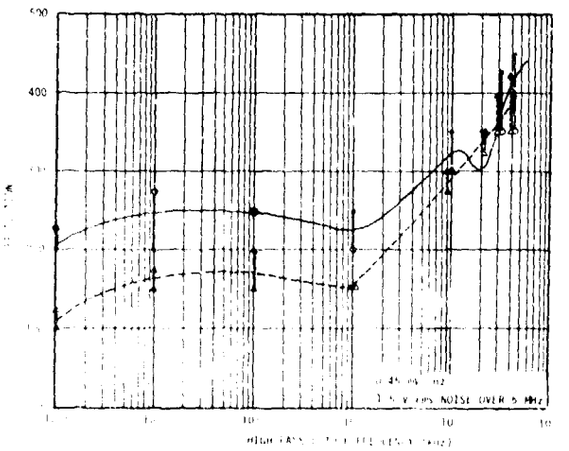


Figure 5c. Continued

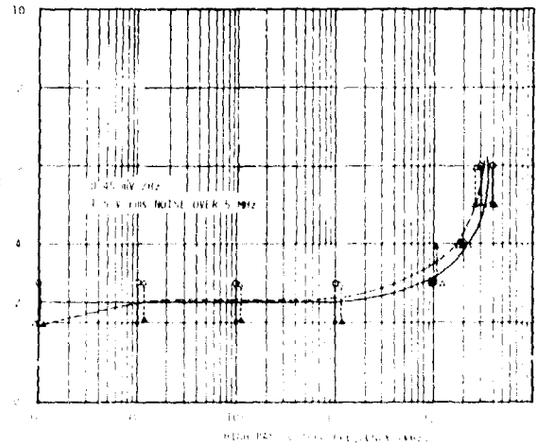


Figure 5d. Concluded

The noise effects seem very severe in the frequency range of 0.1 to 1 MHz. Figure 5(c) shows that a 0.45 V^2 change in noise power between 0.1 and 1 MHz result in an improvement in horizontal resolution of 100 TV lines (225 to 325), while a 1.5 V^2 change in noise power between 1 and 4 MHz results in similar improvement in resolution of 100 TV lines (325 to 425). This indicates that a decrease of almost one-third in noise power in the video band of 0.1 to 1 MHz results in the same improvement in resolution as in the 1- to 5 MHz videoband. As a result, preemphasis and deemphasis of video signal for noise equalization must spread over the frequency range of 0.1 to 1 MHz to have an appreciable effect.

3.4 BANDPASS WHITE GAUSSIAN NOISE

The white Gaussian noise generator output is passed through a bandpass filter before being added to the video signal. Filter center frequency was varies from 0.1 to 4 MHz for a constant Q, and each time the noise source output is adjusted to get constant rms value at the output of the filter. The filter rolloff is 18 dB/octave in the frequency range of 0 to 2 MHz and 12 dB/octave from 2 to 5 MHz. The measured filters response is reported in [4]. Figure 6 illustrates the results of band limited noise interference. Both resolution and gray scale appear very sensitive to noise power around 100 KHz. The horizontal resolution has a maxima and vertical resolution has a minima when the noise power is spread from the low-frequency range to the high-frequency range. Both maxima and minima occur at approximately 100 KHz. Their sharpness depends on the amount of noise power contained per cycle.

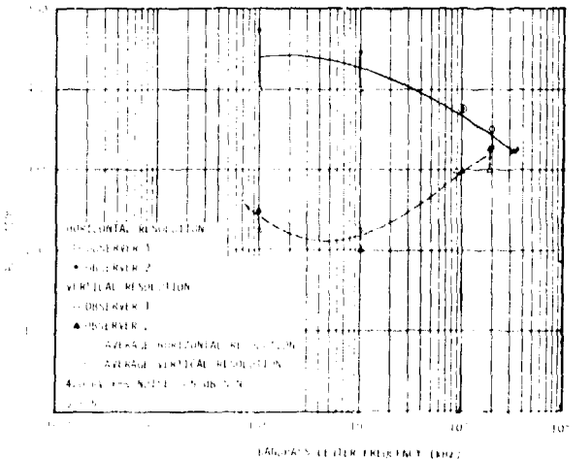


Figure 6a. Bandpass Gaussian White Noise Effects (Continued)

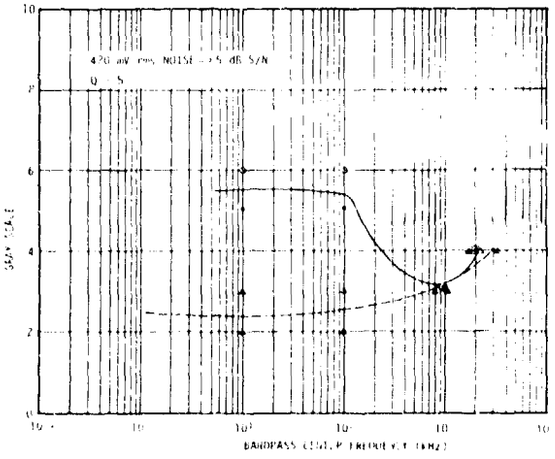


Figure 6b. Concluded

3.5 BAND-REJECT WHITE GAUSSIAN NOISE

For the band-reject noise simulation, output of the noise source is passed through the band-reject filter before being added to video. The center frequency is varied from 0.1 to 5 MHz for a constant Q and a constant rms noise output. The observed data for resolution and gray scale as a function of band-reject center frequency is shown in Figure 7. The vertical gray scale degrades in the frequency range of 0.1 to 1 MHz (a result opposite to that of the bandpass noise simulation).

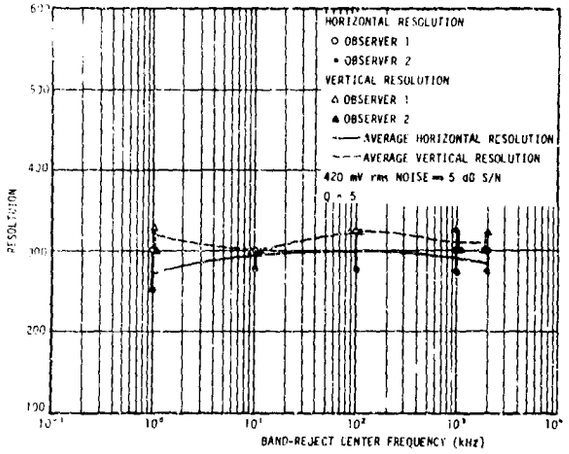


Figure 7a. Band Rejects White Gaussian White Noise Effects (Continued)

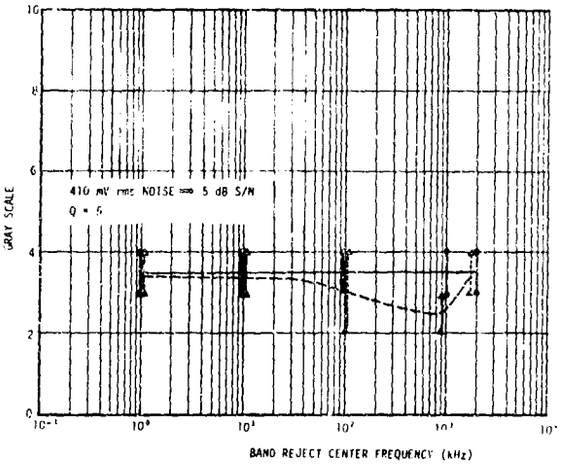


Figure 7b. Concluded

4. NOISE EQUALIZATION

In almost all the observations made during noise simulation, noise effects appear relatively more severe in the frequency range of 0.1 to 1 MHz in the video spectrum. This indicates that if a certain band of frequencies (near 0.1 to 1 MHz) in the video signal is boosted before the video is processed (preemphasized) and the band is suppressed at the receiving end (deemphasized), a considerable system improvement can result. The process is called noise equalization and is frequently used in FM transmission.

Preemphasis networks $\left(TFP = K\left[\frac{s+w_c/K}{s+w_c}\right]^{n(K,G_c)}\right)$ must have the frequency response shown in Figure 8.

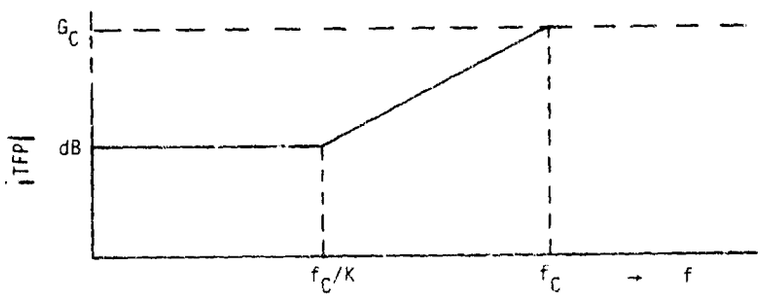


Figure 8. Preamphasis Network Frequency Response

Deemphasis is the reverse of preamplification, having a transfer function of the form:

$$TFD = \left[\frac{s+w_c}{K(s+w_c)}\right]^{n(K,G_c)}$$

Figure 9 illustrates the system performance observed with two sets of equalization parameters, (K=4, f_C = 1 MHz) and (K=7, f_C = 5 MHz) for the simpliest form (n=1) of noise equalization.

Considerable improvement in resolution and gray scale results, particularly for frequencies below 1 MHz. The improvement is uniform only in the case of horizontal gray scale as shown in Figure 9(b). It seems that equalization parameters (f_C, K) are a function of noise power spectrum and higher order form of noise equalization may result in a further improvement.

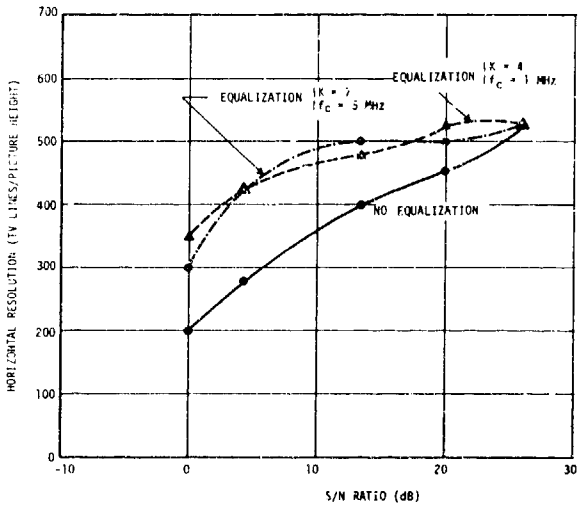


Figure 9a. Noise Equilization Effects (Continued)

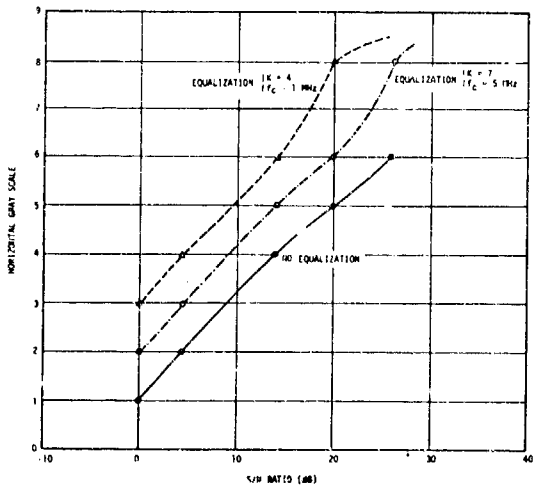


Figure 9b. Continued

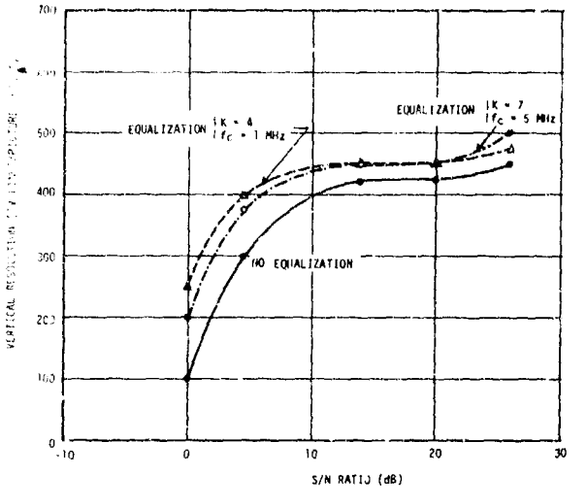


Figure 9c. Continued

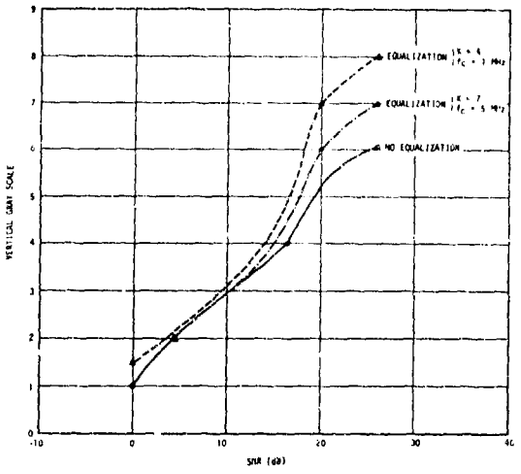


Figure 9d. Concluded

5. CONCLUDING REMARKS

From the large data base observed, it appears that the noise interference in the 0.1 to 1 Mhz video spectrum severly limits the preceived image quality. Noise equalization by preemphasizing and deemphasizing the video over this range results in a significant improvement. The eye image resolving models reasonably correlate with the experimental data.

As an application to image tracking system, the perceived horizontal and vertical limiting resolution can be transformed to minimum area of the resolution element that can be resolved. One can then establish the probability of detection, identification and recognition of the targets under various system specifications and operating conditions. Detailed digital computer simulation have been developed by treating the electro-optical system as an information system. Signal MTF, target contrast and noise power spectrum is convolved with target characteristics, atmospheric conditions, platform characteristics, TV camera optical and electrical characteristics, video display characteristics, and observer capabilities in arriving at the smallest area of resolution element that can be resolved under various operating conditions. Optimum performance is realized by maximumization S/N ratio in the video band where the noise effects are severe. For example, one can boost the signal MTF by aperture correction techniques and reduce the pre-amplifier noise in the desired frequency band by proper selection of its parameters. These techniques have been applied [4] with favorable results.

ACKNOWLEDGMENT

The authors wish to thank E. W. George and J. V. Beard for acting as independent observers for a large amount of visual resolution and gray scale information recorded. Additional thanks go to Steve Kosovac of GRC who reviewed the manuscript and made valuable suggestions.

REFERENCES

1. Rose, A., "The Sensitivity Performance of the Human Eye on an Absolute Scale," J.O.S.A., Vol. 38, No. 2, February 1948.

2. Steedman, W., and Backer, C., "Target Size and Visual Recognition," HUMAN FACTORS, Vol. 2, August 1960.

3. Johnson, J., "Analytical Description of Night Vision Devices," PROCEEDINGS OF THE SEMINAR ON DIRECT-VIEWING ELECTRO-OPTICAL AIDS TO NIGHT VISION, Biberman, L., Editor, Institute for Defense Analyses Study S254, October 1966.

4. G. S. Sandhu, E. V. LaBudde, "Random Noise Interferences and Possible Solutions to its Effect on Television Systems," Teledyne Brown Engr., EEINT2003, May 1976.

5. Electro-Optics Handbook, RCA Technical Series EOM-II, 1974.

ATTENDEES LIST

WORKSHOP ON IMAGING TRACKERS AND AUTONOMOUS ACQUISITION APPLICATIONS FOR MISSILE GUIDANCE

Adams, Stanley
Jaycor Company
205 South Whiting, Suite 500
Alexandria, Virginia 22304

Aggarwal, Raj K.
Honeywell, Inc.
2600 Ridgway Parkway, N.E.
Minneapolis, Minnesota 55413

Ahuja, Narendra
University of Illinois
Coordinated Science Lab
Urbana-Champaign, Illinois 61801

Akerman, Alex III
Martin-Marietta Aerospace
P.O. Box 5837, MS 362
Orlando, Florida 32855

Albanes, Willy
Computer Sciences Corp.
6022 Technology Drive
Huntsville, Alabama 35805

Alexander, P. Martin
U.S. Army Missile Command
ATTN: DRSMI-TEL (R&D)
Redstone Arsenal, Alabama 35898

Armstrong, Wayne T.
Ford Aerospace & Communications Corp.
Aeronutronic Division
P.O. Box A, Ford Road
Newport Beach, California 92663

Ankeney, Lawrence A.
Air Force Armament Laboratory/DLMA
Eglin AFB, Florida 32542

Atkinson, R. J.
General Electric Company
P.O. Box 1585
Huntsville, Alabama 35807

Babineaux, T. L.
Optical Coating Laboratory
2789 Giffen Avenue
P.O. Box 1599
Santa Rosa, California 95402

Bain, George S.
Vought Corporation
P.O. Box 225907, Unit 2-54330
Dallas, Texas 57265

Bair, Max E.
Environmental Research Institute of Michigan
P.O. Box 8618
Ann Arbor, Michigan 48107

Baird, A. M.
Analytics
2500 Maryland Road
Willow Grove, Pennsylvania 19090

Baker, L. Ray
Lockheed Missiles & Space Co., Inc.
P.O. Box 1103, West Station
Huntsville, Alabama 35807

Baker, Roland E.
Naval Weapons Center
China Lake, California 93555

Balopole, Harvey
Fairchild Camera & Instrument Co.
300 Robins Lane
Syosset, New York 11791

Banks, Donald S.
Raytheon Co., Missile Systems Div.
Hartwell Road
Bedford, Massachusetts 01730

Barksdale, William J.
UDS
4900 Bradford Drive
Huntsville, Alabama 35805

Barnes, Carlton
M & S Computing, Inc.
P.O. Box 5183
Huntsville, Alabama 35805

Barone, Frank R.
Naval Research Laboratory
Washington, D.C. 20375

Barron, Roger L.
Adaptronics, Inc.
1750 Old Meadow Road
McLean, Virginia 22102

Bauman, Ltc. Edward J.
U.S. Air Force Academy, DFACS
US Air Force Academy, Colorado 80919

Bauman, J. L.
U.S. Army Missile Command
ATTN: DRSMI-RGC
Redstone Arsenal, Alabama 35898

Baxter, Denmer D.
Martin-Marietta Aerospace
P.O. Box 5837
Orlando, Florida 32855

Bayston, Tom E.
Martin-Marietta Aerospace
P.O. Box 5837, MS 362
Orlando, Florida 32855

Beckerleg, Richard A.
Raytheon Co., Missile Systems Div.
P.O. Box 740
Lowell, Massachusetts 01853

Belsher, John F. III
Rockwell International
3370 Miraloma Avenue
Anaheim, California 92803

Benning, Carl J. Jr.
Texas Instruments
P.O. Box 222013, MS 3407
Dallas, Texas 75222

Berkowitz, Kenith
Sperry Microwave
13133 34th Street
Clearwater, Florida 33518

Bevan, Edward J.
Naval Weapons Center
China Lake, California 93555

Bevin, Jim
Naval Weapons Center
Code 3912
China Lake, California 93555

Bills, Alfred E.
Analytic Sciences Corp.
907 Santa Rosa Blvd.
Ft. Walton Beach, Florida 32548

Billingsley, James E.
Vought Corporation
P.O. Box 226144
Dallas, Texas 75266

Binford, Dr. T.
Stanford University
Stanford, California 94305

Binns, Richard F.
Northrop Corp. EMD
500 E. Orangethorpe
Anaheim, California 92801

Black, G. M.
Naval Surface Weapons Center
White Oak Laboratory
Silver Spring, Maryland 20910

Blair, William P.
Aerospace Corporation
Bldg. 115, MS 2246
2350 E. El Segundo Blvd.
El Segundo, California 90245

Boggs, John E.
CAI, Division of Recon/Optical, Inc.
550 W. Northwest Highway
Barrington, Illinois 60010

Boland, Joseph S. III
Auburn University
Electrical Engineering Dept.
Auburn, Alabama 36830

Boman, David J.
Night Vision & Electro-Optics Lab
ATTN: DELNV-AC
Ft. Belvoir, Virginia 22060

Bonley, A. C.
Naval Surface Weapons Center
White Oak Laboratory
Silver Spring, Maryland 20910

Boone, Bradley G.
Johns Hopkins University
Applied Physics Laboratory
Johns Hopkins Road
Laurel, Maryland 20810

Borrell, Lt. Ronald
U.S. Army Missile Command
ATTN: HEL Laboratory
Redstone Arsenal, Alabama 35898

Boudreaux, E. J.
Naval Surface Weapons Center
White Oak Laboratory, Code F46
Silver Spring, Maryland 20910

Bouley, Alan C.
Naval Surface Weapons Center
White Oak Laboratory
Silver Spring, Maryland 20910

Boyd, W.
Texas Instruments, Inc.
P.O. Box 222013, MS 3402
Dallas, Texas 75222

Brandenberger, John R.
Raytheon Co., Missile Systems Div.
Hartwell Road
Bedford, Massachusetts 01730

Breedlove, James R. Jr.
Los Alamos Scientific Laboratory
M-8/MS 263
Los Alamos, New Mexico 87545

Briglia, Frank A.
U.S. Army Missile Command
ATTN: DRSMI-REI
Redstone Arsenal, Alabama 35898

Brown, Jim L.
M & S Computing, Inc.
P.O. Box 5183
Huntsville, Alabama 35805

Browne, Bernard H.
Sperry Microwave
13133 34th Street
Clearwater, Florida 33518

Buck, Carl
M & S Computing, Inc.
P.O. Box 5183
Huntsville, Alabama 35805

Bush, Maj. Carlos
Defense Mapping Agency, Aerospace Center
St. Louis Air Force Station
St. Louis, Missouri 63118

Buynak, George R.
Goodyear Aerospace Corporation
1210 Massillon Road
Akron, Ohio 44315

Cain, Z. W.
Scientific Systems, Inc.
186 Alewife Brook Parkway
Cambridge, Massachusetts 02138

Calhoun, Sue
Honeywell, Inc.
2600 Ridgway Parkway, N.E.
Minneapolis, Minnesota 55413

Callas, Louis
General Electric Company
901 Broad Street
Utica, New York 13502

Carlos, John Don
Jaycor Company
205 South Whiting, Suite 500
Alexandria, Virginia 22304

Carnaghie, John A.
ASD, U.S. Air Force
ASD/AERE
Wright-Patterson AFB, Ohio 45433

Carrison, Craig
Honeywell, Inc.
2600 Ridgway Parkway, N.E.
Minneapolis, Minnesota 55413

Carruth, Robert T.
Systems Planning Corporation
1500 Wilson Blvd.
Arlington, Virginia 22209

Casey, H. L.
McDonnell Douglas Astronautics/STL
Box 516
St. Louis, Missouri 63166

Castiglia, Noie F.
Litton Amecon
5115 Calvert Road
College Park, Maryland 20740

Chaffie, Gerald M.
RCA Corporation, Bldg. 10-8-3
Delaware and Cooper Streets
Camden, New Jersey 08102

Chammas, Albert B.
Northrop Corp. EMD
500 E. Orangethorpe
Anaheim, California 92801

Chestnut, Walter G.
SRI International
333 Ravenswood Avenue
Menlo Park, California 94025

Childs, J. D.
Systems & Applied Sciences Corp.
6811 Kenilworth Avenue
Riverdale, Maryland 20840

Chiles, Ross M.
Northrop Corporation
500 E. Orangethorpe
Anaheim, California 92801

Christensen, Dr. Charles
U.S. Army Missile Command
Redstone Arsenal, Alabama 35898

Church, Dr. Charles H.
Assistant Director for Technology
Directorate of Army R, D and A
Department of the Army
Washington, D.C. 20301

Clarke, Richard J.
Raytheon Co., Missile Systems Div.
Hartwell Road
Bedford, Massachusetts 01730

Cleveland, William C.
Ford Aerospace & Communications Corp.
Aeronutronic Division
P.O. Box A, Ford Road
Newport Beach, California 92663

Close, Donald
Hughes Research Laboratories
3011 Malibu Canyon Road
Malibu, California 90285

Cobb, Thomas W.
Martin-Marietta Aerospace
P.O. Box 5837
Orlando, Florida 32855

Cockrell, Robert
Honeywell, Inc.
2600 Ridgway Parkway, N.E.
Minneapolis, Minnesota 55413

Cohen, Murray, MS C16
Sperry Gyroscope
Great Neck, New York 11020

Colin, E. W.
Teledyne Brown Engineering
Cummings Research Park
Huntsville, Alabama 35807

Collins, Robert H. III
Southern Research Institute
2000 Ninth Avenue, South
Birmingham, Alabama 35205

Cook, Francis J.
Adaptronics, Inc.
1750 Old Meadow Road
McLean, Virginia 22102

Cooley, Gerald L.
Westinghouse Electric Corp.
80 Holiday Office Center
Huntsville, Alabama 35801

Corday, R. E.
Computer Sciences Corporation
6022 Technology Drive
Huntsville, Alabama 35805

Crowell, Merton H.
System Planning Corp.
1500 Wilson Blvd.
Arlington, Virginia 22209

Cureton, William H.
Quantic Industries
999 Comercial Street
San Carlos, California 94070

Daniel, Jim
Boeing Aerospace
P.O. Box 1470
Huntsville, Alabama 35807

Darva, Ruda
Raytheon Co., Missile Systems Div.
Hartwell Road
Bedford, Massachusetts 01730

Dasarathy, Belur V.
M & S Computing, Inc.
P.O. Box 5183
Huntsville, Alabama 35805

Daugherty, Chris
Raytheon Co., Missile System Div.
Hartwell Road
Bedford, Massachusetts 01730

Davenport, Otis A.
Office of Missile Electronic Warfare
White Sands Missile Range,
New Mexico 88002

Davis, Allen
U.S. Army Missile Command
ATTN: DRSMI-RDD
Redstone Arsenal, Alabama 35898

Davis, Larry S.
University of Texas
Computer Sciences Department
Austin, Texas 78712

Dawson, Jerry O.
U.S. Army Missile Command
ATTN: DRSMI-DS(R&D)
Redstone Arsenal, Alabama 35898

Deal, Bruce
Northrop Corporation
500 E. Orangethorpe
Anaheim, California 92801

DeFoe, Douglas
CAI, Division of Recon/Optical, Inc.
550 W. Northwest Highway
Barrington, Illinois 60010

DeGrandpre, Jean-Louis
General Research Corporation
307 Wynn Drive
Huntsville, Alabama 35805

Delp, Edward
MIT, Lincoln Laboratory
P.O. Box 73
Lexington, Massachusetts 02173

Dembsey, Donald A.
Raytheon Co., Missile Systems Div.
Hartwell Road, M4-40
Bedford, Massachusetts 01730

Demus, D. A.
Texas Tech. University
Department of Mathematics
Lubbock, Texas 79409

Dessau, Harold R.
Martin-Marietta Aerospace
P.O. Box 5837, MP 171
Orlando, Florida 32855

Dickson, Richard
U.S. Army Missile Command
ATTN: DRSMI-RDD
Redstone Arsenal, Alabama 35898

Dorsett, Michael J.
U.S. Army Missile Command
ATTN: DRSMI-DS(R&D)
Redstone Arsenal, Alabama 35898

Druffel, LTC. Larry
Defense Advanced Research Project Agency
1400 Wilson Blvd.
Arlington, Virginia 22209

Ducote, Jere D.
U.S. Army Missile Command
ATTN: DRSMI-REO
Redstone Arsenal, Alabama 35898

Duren, Richard E.
Texas Instruments, Inc.
P.O. Box 222013, MS 3407
Dallas, Texas 75222

Durgin, Frank
Raytheon Co., Missile Systems Div.
Hartwell Road
Bedford, Massachusetts 01730

Edwards, Jerry C.
Teledyne Brown Engineering
Cummings Research Park
Huntsville, Alabama 35807

Edwards, Dr. Thomas R.
NASA-Marshall Space Flight Center
ES-64, Space Sciences Laboratory
Marshall Space Flight Center
Huntsville, Alabama 35812

Effinger, William, Code 3912
Naval Weapons Center
China Lake, California 93555

Ellis, Aaron
U.S. Army Missile Command
ATTN: DRSMI-DS(R&D)
Redstone Arsenal, Alabama 35898

Ellis, Tome W.
Texas Instruments, Inc.
P.O. Box 222013
Dallas, Texas 75222

Elston, R. A.
Ford Aerospace & Communications Corp.
P.O. Box A, Ford Road
Newport Beach, California 92663

Ely, Raymond L.
Johns Hopkins University
Applied Physics Laboratory
Johns Hopkins Road
Laurel, Maryland 20795

Emmons, Dr. George A.
U.S. Army Missile Command
ATTN: DRSMI-TEL, Bldg. 5400
Redstone Arsenal, Alabama 35898

Everhart, W. D.
Honeywell, Inc.
2600 Ridgway Parkway, N.E.
Minneapolis, Minnesota 55413

Eves, William D.
Litton Amecon
5115 Calvert Road
College Park, Maryland 20740

Fagan, James J.
U.S. Army Missile Command
ATTN: DRSMI-R
Redstone Arsenal, Alabama 35898

Faintich, Dr. M.
Defense Mapping Agency Aerospace Center/ST
St. Louis Air Force Station
St. Louis, Missouri 63125

Fairchild, David D.
Honeywell, Inc.
2600 Ridgway Parkway, N.E.
Minneapolis, Minnesota 55413

Farmer, Ray H.
U.S. Army Missile Command
ATTN: DRSMI-REO
Redstone Arsenal, Alabama 35898

Fay, Theodore D.
Teledyne Brown Engineering
Cummings Research Park
Huntsville, Alabama 35807

Ferguson, M. B.
Teledyne Brown Engineering
Cummings Research Park
Huntsville, Alabama 35807

Fetterly, D. R.
General Dynamics/Pomona Division
P.O. Box 2507, MZ 44-44
Pomona, California 91766

Fischler, Martin A.
SRI International
333 Ravenswood Avenue
Menlo Park, California 90265

Fitch, R. C.
Honeywell, Inc.
2600 Ridgway Parkway, N.E.
Minneapolis, Minnesota 55413

Foreman, Jesse W. Jr.
Teledyne Brown Engineering
Cummings Research Park
Huntsville, Alabama 35807

Foti, S.
Naval Surface Weapons Center
White Oak Laboratory
Silver Spring, Maryland 20910

Fowler, James M.
General Research Corporation
307 Wynn Drive
Huntsville, Alabama 35805

Fox, Jim
Bendix Aerospace Systems
1919 Green Road
Ann Arbor, Michigan 48107

Freitag, W. E., Code 39012
Naval Weapons Center
China Lake, California 93555

French, John
U.S. Army Missile Command
ATTN: DRSMI-TES
Redstone Arsenal, Alabama 35898

Friday, Ed
Air Force Weapons Laboratory
ATTN: DLMI, Bldg. 13
Eglin AFB, Florida 32542

Friedman, Daniel
System Planning Corporation
1500 Wilson Blvd.
Arlington, Virginia 22305

Froelich, Albert J.
Teledyne Brown Engineering
Cummings Research Park
Huntsville, Alabama 35807

Fruend, Bernard H.
General Dynamics/Convair Division
P.O. Box 80847, MZ 33-6110
San Diego, California 92138

Fry. Howard Y.
Computer Sciences Corporation
6022 Technology Drive
Huntsville, Alabama 35805

Galstan, Gerald N.
Science Applications, Inc.
2361 Jefferson Davis Highway
Arlington, Virginia 22202

Gandolfo, David A.
RCA, Advanced Technology Laboratory
Bldg. 10-8
Camden, New Jersey 08102

Garza, Rudy
Raytheon Co., Missile Systems Div.
Hartwell Road, M 3-14
Bedford, Massachusetts 01730

Gebhart, Wilford W.
General Research Corporation
307 Wynn Drive
Huntsville, Alabama 35805

Geis, Norman
Raytheon Co., Missile System Div.
Hartwell Road
Bedford, Massachusetts 01730

Gerhart, Grant R.
U.S. Army Tank R&D Command
ATTN: DRDTA-ZSA
Warren, Michigan 48090

Gerson, Gordon
Hughes Aircraft Company
Canoga Park, California 91304

Gibson, Faison P.
Lockheed Missiles & Space Co.
P.O. Box 1103, West Station
Huntsville, Alabama 35807

Giles, Mike
Office of Missile Electronic Warfare
ATTN: STEWS-ID-T
White Sands Missile Range, New Mexico 88002

Goehrig, George A. II
Night Vision & Electro-Optics Lab
Advanced Concepts Division
Ft. Belvoir, Virginia 22060

Goldman, R. B.
Analytics
2500 Maryland Road
Willow Grove, Pennsylvania 19090

Goodell, John B.
Westinghouse Electric Corp.
80 Holiday Office Center
Huntsville, Alabama 35801

Graf, Edward R.
Auburn University
Electrical Engineering Dept.
Auburn, Alabama 36830

Gray, David L.
U.S. Army Missile Command
Redstone Arsenal, Alabama 35898

Greene, Ronald W.
Sandia Laboratories
Div. 5631, Box 5800
Albuquerque, New Mexico 87185

Greer, C. E., MS 5/B168
Hughes Aircraft Company
Centinela & Teale Streets
Culver City, California 90230

Griffin, Robert W.
Kollsman Instrument Company
Daniel Webster Highway
Merrimack, New Hampshire 03054

Griffith, W. A.
U.S. Army Missile Command
ATTN: G&C Directorate
Redstone Arsenal, Alabama 35898

Grimes, Victor S. Jr.
U.S. Army Missile Command
ATTN: DRSMI-RDG, Bldg. 5400
Redstone Arsenal, Alabama 35898

Grommes, R. J.
Honeywell, Inc.
2600 Ridgway Parkway, N.E.
Minneapolis, Minnesota 55413

Grove, Ralph C.
Northrop Research & Tech. Center
One Research Park
Palos Verdes Peninsula, California 90274

Haaland, Sterling, Code 39501
Naval Weapons Center
China Lake, California 93555

Haas, William H.
Rockwell International
Satellite Systems Division
12214 Lakewood Blvd.
Downey, California 90241

Hackney, E. H.
Honeywell Electro-Optical Center
2 Forbes Road
Lexington, Massachusetts 02173

Haimes, Robert
Aerodyne Research
Crosby Drive
Bedford, Massachusetts 01730

Hake, Richard
SRI International
333 Ravenswood Avenue
Menlo Park, California 90265

Halberstam, M.
Raytheon Co., Kuraf Afterman
1578 Rt. 23
Wayne, New Jersey 07470

Halijak, Charles A.
University of Alabama, Huntsville
Electrical Engineering Department
P.O. Box 1247
Huntsville, Alabama 35807

Hall, Ernest L.
University of Tennessee
Tullahoma, Tennessee 37388

Hale, Trevor G.
General Dynamics/Convair Division
P.O. Box 80847
San Diego, California 92138

Hamer, James R.
Northrop Corp., Electro-Mechanical Div.
500 E. Orangethorpe
Anaheim, California 92801

Hamilton, Edward L.
AFAL/RWT-3
Wright-Patterson AFB, Ohio 45433

Hanley, Walter R.
Sperry Microwave
13133 34th Street
Clearwater, Florida 33518

Haralick, R. M.
Virginia Polytech Institute
& State University
Electrical Engineering Dept.
Blacksburg, Virginia 24061

Harding, Warren P.
Raytheon Company
Hartwell Road
Bedford, Massachusetts 01730

Hardy, John
Technology Service Corp.
2811 Wilshire Blvd.
Santa Monica, California 90403

Hart, Lawson L.
McDonnell Douglas Astronautics Co.
P.O. Box 516
St. Louis, Missouri 63166

Hartman, Richard
U.S. Army Missile Command
ATTN: DRSMI-RR
Redstone Arsenal, Alabama 35898

Hassin, Dr. Guido
Lockheed Missiles & Space Co., Inc.
P.O. Box 1103, West Station
Huntsville, Alabama 35807

Hayes, Charles
Lockheed Missiles & Space Co., Inc.
P.O. Box 1103, West Station
Huntsville, Alabama 35807

Hayes, Frank III
U.S. Army Missile Command
ATTN: DRSMI-REI, Bldg. 5400
Redstone Arsenal, Alabama 35898

Heaney, Albert A.
Sperry Microwave
13133 34th Street
Clearwater, Florida 33518

Helm, David P.
U.S. Army Signal Warfare Lab, DELSW-EA
Vinton Hill Farms Station
Warrenton, Virginia 22186

Helton, Monte K.
U.S. Army Missile Command
ATTN: DRSMI-REO
Redstone Arsenal, Alabama 35898

Henderson, Robert L.
General Dynamics/Convair Division
P.O. Box 80847, MZ 33-6110
San Diego, California 92138

Hester, Charles F.
U.S. Army Missile Command
Redstone Arsenal, Alabama 35898

Hicks, Bob
U.S. Army Missile Command
Redstone Arsenal, Alabama 35898

Hiller, Edwin R.
Raytheon Co., Missile Systems Div.
Hartwell Road, MS 46
Bedford, Massachusetts 01730

Hix, Rudolf G.
General Dynamics/Pomona Div.
P.O. Box 2507
Pomona, California 91766

Hobeli, Donald A.
Ford Aerospace & Communications Corp.
P.O. Box A, Ford Road
Newport Beach, California 92663

Hogan, J. J.
Goodyear Aerospace Corporation
1210 Massillon Road
Akron, Ohio 44315

Holben, Richard D.
Ford Aerospace & Communications Corp.
P.O. Box A, Ford Road
Newport Beach, California 92663

Holder, Don
U.S. Army Missile Command
ATTN: DRSMI-RDD
Redstone Arsenal, Alabama 35898

Hollister, Floyd
Texas Instruments, Inc.
P.O. Box 222013
Dallas, Texas 75222

Holter, Dale
U.S. Army Missile Command
ATTN: DRSMI-REI
Redstone Arsenal, Alabama 35898

Hopkins, Gilbert G.
Texas Instruments, Inc.
P.O. Box 222013
Dallas, Texas 75222

Horn, Dr. B.
MIT, Lincoln Laboratories
P.O. Box 73
Lexington, Massachusetts 02173

Hornsley, Thomas
General Research Corporation
307 Wynn Drive
Huntsville, Alabama 35805

Howes, Bradford S.
Raytheon Co., Missile Systems Div.
Hartwell Road
Bedford, Massachusetts 01730

Huang, Thomas S.
Purdue University
School of Electrical Engineering
West Lafayette, Indiana 47907

Hudson, Jerry
Air Force Armament Laboratory
Eglin AFB, Florida 32542

Huff, Dr. Jess B.
U.S. Army Missile Command
ATTN: DRSMI-RG
Redstone Arsenal, Alabama 35898

Hwang, J. J.
Nichols Research Corporation
7910 S. Memorial Parkway, Suite F
Huntsville, Alabama 35802

Iler, T. W.
Goodyear Aerospace Company
Dept. 475C-2
1210 Massillon Road
Akron, Ohio 44315

Ionescu, Tudor
Intermetrics, Inc.
701 Concord Avenue
Cambridge, Massachusetts 02138

Iuigo, Rafael
University of Virginia
Thornton Hall
Charlottesville, Virginia 22901

Jain, Ramesh
Wayne State University
Computer Science Department
Detroit, Michigan 48202

James, David V.
Ampex Corporation
401 Broadway
Redwood City, California 94063

Jelalian, Albert V.
Raytheon Company
430 Boston Post Road
Wayland, Massachusetts 01778

Jenkins, Billy Z.
U.S. Army Missile Command
ATTN: DRSMI-TDK(R&D), Bldg. 5400
Redstone Arsenal, Alabama 35898

Jensen, James D.
Naval Surface Weapons Center
White Oak Laboratory
Silver Spring, Maryland 20910

Jestice, Aaron L.
Sperry Microwave
13133 34th Street
Clearwater, Florida 33518

Johnson, C. Bruce
ITT Electro-Optical Products Div.
3700 E. Pontiac Street
Ft. Wayne, Indiana 46607

Jones, Billy E.
U.S. Army Missile Command
ATTN: DRSMI-ESNM, Bldg. 5400
Redstone Arsenal, Alabama 35898

Jones, Daniel J.
General Research Corporation
307 Wynn Drive
Huntsville, Alabama 35805

Jones, Edward C., DELNV-AC
Night Vision & Electro-Optics Lab
Ft. Belvoir, Virginia 22060

Jones, N.
Texas Instruments, Inc.
P.O. Box 22013, MS 3402
Dallas, Texas 75222

Jones, Richard S.
U.S. Army Missile Command
ATTN: DRSMI-ESNM, Bldg. 5400
Redstone Arsenal, Alabama 35898

Joseph, J. D.
Honeywell, Inc.
2600 Ridgway Parkway, N.E.
Minneapolis, Minnesota 55413

Justus, Jimmie J.
Rockwell International
3370 Miraloma Ave.
Anaheim, California 92806

Kattos, Nicholas C.
U.S. Army Missile Command
ATTN: DRSMI-YLM
Redstone Arsenal, Alabama 35898

Keat, John W.
U.S. Army Missile Command
Redstone Arsenal, Alabama 35898

Keene, John W. Jr.
Sanders Associates, Inc.
95 Canal Street
Nashua, New Hampshire 03060

Kinfey, Lloyd
Joint Cruise Missile Project Office
Mail Code JCM-M121
Washington, D.C. 20360

King, Richard W. Jr.
Raytheon Co., Missile Systems Div.
P.O. Box 740
Lowell, Massachusetts 01853

Kiremidjiam, Jaro
ESL, Inc.
495 Java Drive
Sunnyvale, California 94086

Kleiman, Herbert
MIT, Lincoln Laboratory
P.O. Box 73, 244 Wood Street
Lexington, Massachusetts 02173

Kliger, I. E.
Raytheon Co., Missile System Div.
Hartwell Road
Bedford, Massachusetts 01730

Klinglesmith, Dr. Daniel
NASA/GSFC, Code 685
Greenbelt, Maryland 20771

Knecht, John A., Code 3153
Naval Weapons Center
China Lake, California 93555

Kollodge, Jerome C.
Ball Aerospace Systems Div. of Ball Corp.
P.O. Box 1062, Boulder Industrial Park
Boulder, Colorado 80306

Kobler, Julian S.
U.S. Army Missile Command
ATTN: DRSMI-R
Redstone Arsenal, Alabama 35898

Konar, A. F.
Systems and Research Center
Honeywell, Inc.
2600 Ridgway Parkway, N.E.
Minneapolis, Minnesota 55413

Kopcsak, George C.
OSD/OUSDR&E (ET)
Room 3D1089, The Pentagon
Washington, D.C. 20301

Kopet, T. C.
Honeywell, Inc.
2600 Ridgway Parkway, N.E.
Minneapolis, Minnesota 55413

Koso, Alexander
Charles Stark Draper Laboratory
555 Technology Square
Cambridge, Massachusetts 02138

Kosovac, Stephen M.
General Research Corporation
307 Wynn Drive
Huntsville, Alabama 35805

Kovacev, Robert J.
Northrop Corporation
500 E. Orangethorpe
Anaheim, California 92801

Koval, K. A.
U.S. Air Force
AD/SD8E
Eglin AFB, Florida 32542

Kovel, Steven M.
Office of Missile Electronic Warfare
ATTN: DELEW-M-STG
White Sands Missile Range,
New Mexico 88002

Kovar, Joseph J., Code 3153
Naval Weapons Center
China Lake, California 93555

Krasnakevich, John R.
Kindler Division, Ketron, Inc.
One Broadway
Cambridge, Massachusetts 02142

Krich, Steven I
MIT, Lincoln Laboratory
P.O. Box 73
Lexington, Massachusetts 02173

Kruer, Mel, Code 5554
Naval Research Laboratories
Washington, D.C. 20375

Kuhn, Gerald D., D/832
Naval Avionics Center
6000 E. 21st Street
Indianapolis, Indiana 46218

Kulas, Chris
U.S. Army Missile Command
ATTN: DRDMI-TES
Redstone Arsenal, Alabama 35898

Kumar, Vas
Rockwell International
Satellite Systems Division
11214 Lakewood Blvd.
Downey, California 90241

Kwang-Shen Wu
4301 Connecticut Ave., N.W., Suite 132
Washington, D.C. 20008

Kylander, John C.
ITT Electro-Optical Products Div.
3700 East Pontiac Street
Ft. Wayne, Indiana 46807

Laats, Ain
Charles Stark Draper Laboratory
555 Technology Square
Cambridge, Massachusetts 02139

Laite, Peter J.
DES/British Embassy
3100 Massachusetts Ave., N.W.
Washington, D.C. 20008

Lander, William B.
Jaycor Company
205 South Whiting, Suite 500
Alexandria, Virginia 22304

Lawler, Bill
Boeing Aerospace Company
P.O. Box 1470
Huntsville, Alabama 35807

Layne, Charles R.
Martin-Marietta Aerospace
P.O. Box 5837, MP 304
Orlando, Florida 32805

Ledbetter, James D.
U.S. Air Force Academy
Frank J. Seiler Research Lab/NHL
U.S. Air Force Academy, Colorado 80840

Lee, Dr. Harry
Martin-Marietta Aerospace
P.O. Box 5837
Orlando, Florida 32805

Lemmer, John
Pattern Analysis & Recognition Corp.
220 Liberty Plaza
Rome, New York 13440

Lesnik, Abraham
Fairchild Republic Company
Conklin Street
Farmingdale, New York 11735

Levitt, Barry N.
Raytheon Co., Missile Systems Div.
Hartwell Road
Bedford, Massachusetts 01730

Light, David O. Jr.
U.S. Army Missile Command
ATTN: DRSMI-TEI(R&D), Bldg. 5400
Redstone Arsenal, Alabama 35898

Likeness, Barry K.
Sperry Microwave
13133 34th Street
Clearwater, Florida 33518

Lindberg, William J.
U.S. Army Missile Command
ATTN: DRSMI-RE
Redstone Arsenal, Alabama 35898

Lindquist, George H.
Nichols Research Corporation
2002 Hogback Road, Suite 9
Ann Arbor, Michigan 48104

Loefer, Gene R.
Georgia Institute of Technology/EES
EML, Baker Building
Atlanta, Georgia 30332

Loft, Ernest E.
Martin-Marietta Aerospace
P.O. Box 5837, MP 74
Orlando, Florida 32805

Mack, Harold
Ford Aerospace & Communications Corp.
P.O. Box A, Ford Road
Newport Beach, California 92663

MacKenzie, Gordon C.
Raytheon Co., Missile Systems Div.
Hartwell Road
Bedford, Massachusetts 01730

Malcolm, W. W.
U.S. Army Missile Command
Redstone Arsenal, Alabama 35898

Mall, Ben J.
CAI, Div. of Recon/Optical, Inc.
550 W. Northwest Highway
Barrington, Illinois 60010

Matuszewski, Henry
Bell Helicopter
P.O. Box 482
Fort Worth, Texas 76101

Maxwell, J. Robert
Environmental Research Institute of
Michigan, P. O. Box 8618
Ann Arbor, Michigan 48107

McCulloch, Terry M., MS 5/168
Hughes Aircraft Company
Centinela & Teale Streets
Culver City, California 90230

McDuffie, G. L.
Westinghouse Electric Corporation
80 Holiday Office Center
Huntsville, Alabama 35801

McIngvale, Pat H.
U.S. Army Missile Command
ATTN: DRSMI-TGC(R&D)
Redstone Arsenal, Alabama 35898

McIntosh, D.
Texas Instruments, Inc.
P.O. Box 222013, MS 3402
Dallas, Texas 75222

McShera, J. T. Jr.
Goodyear Aerospace Corporation
2112 Lewis Turner Blvd.
Ft Walton Beach, Florida 32548

McVey, Eugene S.
University of Virginia
Thornton Hall
Charlottesville, Virginia 22901

McWilliams, Joel K.
Texas Instruments, Inc.
P.O. Box 222013, MS 3402
Dallas, Texas 75222

Mead, Dr. Jaylee
NASA/GSFC, Code 680
Greenbelt, Maryland 20771

Mead, Wendell
Spectra Research Systems, Inc.
2212 Dupont Drive
Irvine, California 92715

Mekaru, Mark
Defense Mapping Agency
Advanced Technology Division
U.S. Naval Observatory, Bldg. 56
Washington, D.C. 20305

Mendelsohn, J.
Grumman Aerospace Corporation
Research Dept. A08/35
Bethpage, New York 11714

Merrill, Ronald B.
Nichols Research Corporation
7910 S. Memorial Parkway, Suite F
Huntsville, Alabama 35802

Metzer, 2Lt. Dennis H.
U.S. Air Force
AFEWC/ESAF
Kelly Air Force Base
San Antonio, Texas 78243

Mickelsen, Arlo D., Code 3912
Naval Weapons Center
China Lake, California 93555

Milgram, David L.
Lockheed Palo Alto Research Labs
Dept. 52-53, Bldg. 204
3251 Hanover Street
Palo Alto, California 94304

Miller, John M.
Boeing Aerospace Company
P.O. Box 1470
220 Wynn Drive
Huntsville, Alabama 35807

Milton, A. F.
Naval Research Laboratory
4555 Overlook Avenue, S. W.
Washington, D.C. 20375

Minor, Dr. Lewis G.
U.S. Army Missile Command
ATTN: DRSMI-REO
Redstone Arsenal, Alabama 35898

Minshew, Henry M.
General Research Corporation
307 Wynn Drive
Huntsville, Alabama 35805

Montgomery, Willard W.
Lockheed Missiles & Space Co., Inc.
P.O. Box 1103, West Station
Huntsville, Alabama 35807

Montori, John
Perkin-Elmer Corporation
1004 Wooster Heights Road, MS 803
Danberry, Connecticut 06810

Morgan, Hugh W.
Teledyne Brown Engineering
Cummings Research Park
Huntsville, Alabama 35807

Morris, Ed, MD 906
General Electric Company
901 Broad Street
Utica, New York 13502

Morris, Gerald L.
U.S. Army Missile Command
ATTN: DRSMI-KKL
Redstone Arsenal, Alabama 35898

Morris, John W. Jr.
Science Applications, Inc.
2109 W. Clinton Ave., Suite 800
Huntsville, Alabama 35805

Mostafavikashani, Hassan
Systems Control, Inc.
1801 Page Mill Road
Palo Alto, California 94304

Musa, Sam
Office of Under Secretary of Defense
Research & Advanced Technology
Room 3D1079, The Pentagon
Washington, D.C. 20301

Myers, Karen
MIT, Lincoln Laboratory
P.O. Box 73 (Room L-153)
Lexington, Massachusetts 02193

Nagy, T. A.
Systems & Applied Sciences Corp.
6811 Kenilworth Avenue
Riverdale, Maryland 20840

Narendra, P. M.
Honeywell, Inc.
2600 Ridgway Parkway, N.E.
Minneapolis, Minnesota 55413

Nelgner, Harry D.
Emerson Electric Company
8100 W. Florissant Avenue
St. Louis, Missouri 63136

Newman, Thomas G.
Texas Tech University
Department of Mathematics
Lubbock, Texas 79409

Noble, Stephen C.
Ampex Corporation
401 Broadway
Redwood City, California 94063

Noges, E.
Boeing Aerospace Company
P.O. Box 3999, MS 86-18
Seattle, Washington 98124

Noren, Keith E.
Teledyne Brown Engineering
Cummings Research Park
Huntsville, Alabama 35807

Ormsby, Dr. C. C.
Analytic Sciences Corporation
Reading, Massachusetts 01867

Ormsby, Ralph D.
Honeywell, Inc.
2600 Ridgway Parkway, N.E.
Minneapolis, Minnesota 55413

Orton, David A.
Rockwell International
3370 Miraloma Avenue, MS D/521 022-BB85
Anaheim, California 92803

Owens, W. R.
Analytics
2500 Maryland Road
Willow Grove, Pennsylvania 19090

Panda, Durga P.
Honeywell, Inc.
2600 Ridgway Parkway, N.E.
Minneapolis, Minnesota 55413

Panton, Dale J.
Control Data Corporation
1800 N. Beauregard Street
Alexandria, Virginia 22311

Parenti, Ronald R.
MIT, Lincoln Laboratory
P.O. Box 73, 244 Wood Street
Lexington, Massachusetts 02173

Pasik, David D.
Martin-Marietta Aerospace
P.O. Box 5837, MS 171
Orlando, Florida 32855

Peace, George M.
Environmental Research Inst. of Michigan
P.O. Box 8618
Ann Arbor, Michigan 48107

Peckins, Seymour
Singer-Kearfott
1150 McBride Avenue
Little Falls, New Jersey 07424

Penny, Morris M.
Lockheed Missiles & Space Co., Inc.
P.O. Box 1103, West Station
Huntsville, Alabama 35807

Pessoni, Douglas A.
Perkin-Elmer Corporation
Main Avenue, MS 200
Norwalk, Connecticut 06856

Peters, Edward G.
M & S Computing, Inc.
P.O. Box 5183
Huntsville, Alabama 35805

Petersen, George J.
GTE Sylvania, Inc.
Sylvania Systems Group
100 Ferguson Drive, P.O. Box 205
Mountain View, California 94042

Petty, Ernest L. Jr.
U.S. Army Missile Command
ATTN: DRSMI-DS(R&D)
Redstone Arsenal, Alabama 35898

Pfeiffer, H.
Goodyear Aerospace Corporation
1210 Massillon Road
Akron, Ohio 44315

Pinson, George
Boeing Aerospace Company
P.O. Box 1470
Huntsville, Alabama 35807

Pinson, L. J.
University of Tennessee
Space Institute
Tullahoma, Tennessee 37388

Pittenger, R. L.
Rockwell International
3370 Miraloma Avenue
Anaheim, California 92803

Pitruzello, Michael
U.S. Army Missile Command
ATTN: DRSMI-TGC(R&D)
Redstone Arsenal, Alabama 35898

Plunkett, Kenneth W.
U.S. Army Missile Command
ATTN: DRSMI-RG
Redstone Arsenal, Alabama 35898

Politopoulos, Anastasios, S.
Ford Aerospace & Communications Corp.
P.O. Box A, Ford Road
Newport Beach, California 92663

Poulsen, Peter D.
Science Applications, Inc.
2109 W. Clinton Avenue
Huntsville, Alabama 35805

Powell, Rex
U.S. Army Missile Command
ATTN: DRSMI-RDD
Redstone Arsenal, Alabama 35898

Prestwood, William T.
U.S. Army Missile Command
ATTN: DRSMI-NS, Bldg. 5681
Redstone Arsenal, Alabama 35898

Pridgen, Jack H.
Texas Instruments, Inc.
P.O. Box 222013
Dallas, Texas 75222

Prom, G. O.
Honeywell, Inc.
2600 Ridgway Parkway, N.E.
Minneapolis, Minnesota 55413

Pryor, P.
Systems Control, Inc.
1801 Page Mill Road
Palo Alto, California 94304

Purdy, D. R.
Honeywell, Inc.
2600 Ridgway Parkway, N.E.
Minneapolis, Minnesota 55413

Purkis, Ronald
Xerox Corporation
300 North Halstead Street, MS 1204
Pasadena, California 91107

Putriment, B. W.
U.S. Air Force
AD/SD8E
Eglin Air Force Base, Florida 32542

Radgowski, Charles
IIT Research Institute
10 West 35th Street
Chicago, Illinois 60616

Randall, N. C.
Analytics
2500 Maryland Road
Willow Grove, Pennsylvania 19090

Ranganath, H. S.
Auburn University
Electrical Engineering Department
Auburn, Alabama 36830

Rast, Gustaf J. Jr.
U.S. Army Missile Command
ATTN: DRSMI-REL, Bldg. 5400
Redstone Arsenal, Alabama 35898

Ratkovic, Joseph A.
The Rand Corporation
1700 Main Street
Santa Monica, California 90406

Rearick, Thomas C.
General Electric Company, AESD
901 Broad Street, MS 908
Utica, New York 13501

Reinhold, Ralph R.
Boeing Aerospace
P.O. Box 1470
Huntsville, Alabama 35807

Reischer, Benjamin
Night Vision & Electro-Optics Lab
Advanced Concepts Division
Ft. Belvoir, Virginia 22060

Reitan, R. C.
Honeywell, Inc.
2600 Ridgway Parkway, N.E.
Minneapolis, Minnesota 55413

Riedel, Frederick W.
Johns Hopkins University
Applied Physics Laboratory
Johns Hopkins Road
Laurel, Maryland 20810

Riedl, H. R.
Naval Surface Weapons Center
White Oak Laboratory
Silver Spring, Maryland 20910

Riley, Chalmers G.
U.S. Army Missile Command
ATTN: DRSMI-TG
Redstone Arsenal, Alabama 35898

Riley, Glenn E.
Georgia Institute of Technology/EES
EML, Baker Building
Atlanta, Georgia 30332

Rinehold, Ralph
Boeing Aerospace Corporation
P.O. Box 1470
Huntsville, Alabama 35807

Roberson, Donald L.
U.S. Army Missile Command
ATTN: DRSMI-ESP(R&D), Bldg. 5400
Redstone Arsenal, Alabama 35898

Rogers, R. W.
U.S. Air Force
AD/SD8E
Eglin Air Force Base, Florida 32542

Rolnick, B.
RCA, Advanced Technology Laboratories
Building 10-8
Camden, New Jersey 08102

Rosenfeld, Azriel
University of Maryland
College Park, Maryland 20742

Roth, Erving
Sperry Gyroscope Division
Great Neck, New York 11020

Rudnick, Jack J.
RCA, Advanced Technology Laboratories
Building 10-8
Camden, New Jersey 08102

Ruetmann, Mihkel
Aerospace Corporation
Building 115, MS 2246
2350 E. El Segundo Blvd.
El Segundo, California 90245

Russell, Robert F.
U.S. Army Missile Command
ATTN: DRSMI-TEG
Redstone Arsenal, Alabama 35898

Salter, W. E.
M & S Computing, Inc.
P.O. Box 5183
Huntsville, Alabama 35805

Sander, William A.
U.S. Army Research Office
P.O. Box 12211
Rip, North Carolina 27709

Sandhu, G.
General Research Corporation
307 Wynn Drive
Huntsville, Alabama 35805

Satterfield, Richard A.
Northrop Corporation
Electro-Mechanical Division
500 E. Orangethorpe Ave.
Anaheim, California 92801

Sauer, R. L.
McDonnell Douglas Astronautics
P.O. Box 516
St. Louis, Missouri 63166

Savol, A. M.
Boeing Aerospace Company
P.O. Box 3999, MS 86-18
Seattle, Washington 98124

Scales, Dr. John L.
HQ, ERADCOM
Harry Diamond Laboratories
2800 Powder Mill Road
Adelphi, Maryland 20783

Schaming, W. B.
RCA, Advanced Technology Laboratories
Building 10-8
Camden, New Jersey 08102

Schau, H. C.
Martin-Marietta Aerospace
P.O. Box 5837
Orlando, Florida 32855

Scheynayder, Cpt. Mike
U.S. Army Missile Command
ATTN: DRSMI-R
Redstone Arsenal, Alabama 35898

Schmerling, Erwin R.
NASA-Marshall Space Flight Center
Huntsville, Alabama 35812

Schmid, H. P.
General Dynamics/Pomona Division
P.O. Box 2507, MZ 44-44
Pomona, California 91766

Schmeider, David E.
Georgia Institute of Technology/EES
EML, Baker Building
Atlanta, Georgia 30332

Schneider, Hank
Sperry Microwave, MS 41
P.O. Box 4648
Clearwater, Florida 33518

Schnitzler, Alvin D.
Naval Research Laboratory
Washington, D.C. 20375

Schooler, Lanny D.
Martin-Marietta Aerospace
P.O. Box 5837, MP-51
Orlando, Florida 32855

Schrenk, William
U.S. Army Missile Command
ATTN: DRSMI-ROW
Redstone Arsenal, Alabama 35898

Schroeder, John W.
Aerodyne Research, Inc.
Crosby Drive
Bedford, Massachusetts 02170

Schroeder, M. O.
Honeywell, Inc.
2600 Ridgway Parkway, N.E.
Minneapolis, Minnesota 55413

Schultz, R. L.
Honeywell, Inc.
2600 Ridgway Parkway, N.E.
Minneapolis, Minnesota 55413

Selim, J. D.
U.S. Marine Corps
C3 Division, Development Center
MCDEC
Quantico, Virginia 22134

Sellers, G. W.
Rockwell International
Avionics & Missile Systems Group
Cedar Rapids, Iowa 52406

Serota, Ted R.
Southern Research Institute
2000 Ninth Avenue, South
Birmingham, Alabama 35205

Serreyn, David V.
Honeywell, Inc.
2600 Ridgway Parkway, N.E.
Minneapolis, Minnesota 55413

Seyrafi, Khalil
Electro-Optical Research Company
2029 Century Park East, Suite 422
Los Angeles, California 90067

Shapiro, Sol
Singer-Kearfott
150 Totowa Road
Wayne, New Jersey 07470

Sheats, L.
Raytheon Co., Missile Systems Div.
Hartwell Road
Bedford, Massachusetts 01730

Silver, Alan
Magnavox Government & Industrial
Electronics Company
46 Industrial Avenue
P.O. Box 615
Mahwah, New Jersey 07430

Sims, Richard
U.S. Army Missile Command
ATTN: DRSMI-REO
Redstone Arsenal, Alabama 35898

Sklansky, Jack
University of California, Irvine
Electrical Engineering Department
Irvine, California 92717

Smith, Charles P.
Raytheon Co., Missile Systems Div.
Hartwell Road
Bedford, Massachusetts 01730

Smith, Fred W.
Systems Control, Inc.
1801 Page Mill Road
Palo Alto, California 94304

Soland, D. E.
Honeywell, Inc.
2600 Ridgway Parkway, N.E.
Minneapolis, Minnesota 55413

Sondergard, Richard D.
Emerson Electric Company
8100 W. Florissant, MS 3842
St. Louis, Missouri 63136

Sparrow, B. D.
IBM Corporation
150 Sparkman Drive
Huntsville, Alabama 35805

Staggs, William D.
Northrop Corporation
500 E. Orangethorpe Ave.
Anaheim, California 92801

Stamboni, James J.
Raytheon Co., Missile Systems Div.
Hartwell Road
Bedford, Massachusetts 01730

Stanton, Mack J.
General Electric-TEMPO
816 State Street, PO Drawer QQ
Santa Barbara, California 93102

Statham, Richard B.
Teledyne Brown Engineering
Cummings Research Park
Huntsville, Alabama 35807

Statt, David J.
Eastman Kodak Company
121 Lincoln Avenue
Rochester, New York 14650

Steiner, J. M.
JMS Company

Stovall, Robert E.
Johns Hopkins University
Applied Physics Laboratory
Johns Hopkins Road
Laurel, Maryland 20904

Stutman, E. G.
Raytheon Co., Missile Systems Div.
Hartwell Road
Bedford, Massachusetts 01730

Svedlow, Dr. M.
Analytic Sciences Corporation
Reading, Massachusetts 01867

Takken, Edward H.
Naval Research Laboratory
4555 Overlook Avenue, S.W.
Washington, D.C. 20375

Tarbox, Tom C.
Bell Aerospace Company
Division of Textron Company
5719 Tannehill Circle
Huntsville, Alabama 35802

Taylor, Kenneth
Ford Aerospace & Communications Corp.
Aeronutronics Division
P.O. Box A, Ford Road
Newport Beach, California 92663

Taylor, R. L.
Northrop Corporation
500 E. Orangethorpe Avenue
Anaheim, California 92801

Tenbrink, Stephen C., Code 3924
Naval Weapons Center
China Lake, California 93555

Tennebaum, Dr. M.
SRI International
333 Ravenswood Avenue
Menlo Park, California 94025

Terry, Capt. Philip E.
U.S. Army Missile Command
ATTN: DRCPM-HFE
Redstone Arsenal, Alabama 35898

Thiede, E.
Honeywell, Inc.
2600 Ridgway Parkway, N.E.
Minneapolis, Minnesota 55413

Thomas, Albert L. Jr.
Southern Research Institute
2000 Ninth Avenue, South
Birmingham, Alabama 35205

Thorn, Capt. Charles T.
USAADS, Directorate of Combat Dev.
ATTN: ATSA-CD-SC-A
Ft. Bliss, Texas 79916

Tichenor, Vernon C.
Northrop Corporation
1800 Century Park East
Los Angeles, California 90067

Tierman, Kenneth E.
Ford Aerospace & Communications Corp.
P.O. Box A, Ford Road
Newport Beach, California 92663

Tisdale, Glenn E.
Westinghouse Defense Center
P.O. Box 746, MS 451
Baltimore, Maryland 21203

Tolitopoulos, Anastasios S.
Ford Aerospace & Communications Corp.
Aeronutronic Division
P.O. Box A, Ford Road
Newport Beach, California 92663

Trussell, Joel
Los Alamos Scientific Laboratory
Group M-8, MS 263
Los Alamos, New Mexico 87545

Tseng, David Y.
Hughes Research Laboratories
3011 Malibu Canyon Road
Malibu, California 90265

Tullia, Lt. T.
U.S. Army Missile Command
ATTN: DRSMI-TES
Redstone Arsenal, Alabama 35898

Tully, David B.
Teledyne Brown Engineering
Cummings Research Park
Huntsville, Alabama 35807

Utley, Kurt E.
RCA
Routes 52 and 3
Burlington, Massachusetts 01803

Varecka, Charles P.
Honeywell, Inc.
600 Second Street, North
Hopkins, Minnesota 55343

Ventre, Andrew J.
Computer Sciences Corporation
6022 Technology Drive
Huntsville, Alabama 35807

Viggh, Dr. M.
Analytic Sciences Corporation
Reading, Massachusetts 01867

Wagner, Robert G.
McDonnell Douglas Astronautics
Bldg. 106/2/E9, MS 34
P.O. Box 516
St. Louis, Missouri 63166

Waite, William F.
U.S. Army Missile Command
ATTN: DRSMI-TED(R&D), Bldg. 5400
Redstone Arsenal, Alabama 35898

Walker, Robert
Sanders Associates
Suite 1, Holiday Office Center
Huntsville, Alabama 35801

Walker, Billy J.
U.S. Army Missile Command
ATTN: DRSMI-TDK
Redstone Arsenal, Alabama 35898

Walter, John F. Jr.
Johns Hopkins University
Applied Physics Laboratory
Johns Hopkins Road
Laurel, Maryland 20810

Walter, Richard A.
Martin-Marietta Aerospace
P.O. Box 5837
Orlando, Florida 32855

Ware, C. L.
Teledyne Brown Engineering
Cummings Research Park
Huntsville, Alabama 35807

Warren, Frank
RCA, Advance Technology Laboratories
Cooper and Front Streets
Camden, New Jersey 08102

Watson, Harry J.
Teledyne Brown Engineering
Cummings Research Park
Huntsville, Alabama 35807

Weathers, Glen
M & S Computing, Inc.
P.O. Box 5183
Huntsville, Alabama 35805

Weiler, T.
Goodyear Aerospace
1210 Massillon Road
Akron, Ohio 44315

Welch, Albert B.
Vought Corporation
Advanced Technology Center
P.O. Box 226144
Dallas, Texas 75266

Werner, Richard E.
Martin-Marietta Aerospace
P.O. Box 5837, MP-74
Orlando, Florida 32855

Whitaker, John M.
U.S. Army Missile Command
ATTN: DRSMI-DS(R&D)
Redstone Arsenal, Alabama 35898

Whiteman, Peter
Harris Corporation, GISD
P.O. Box 37
Melbourne, Florida 32901

Whiting, Webster K.
Naval Air Systems Command
HQ, AIR54922B
Washington, D.C. 20361

Wiener, Commander Thomas F.
Defense Advanced Research Project Agency
1400 Wilson Blvd.
Arlington, Virginia 22209

Wilkins, Lowell H.
Naval Weapons Center
China Lake, California 93555

Williams, Bruce
Lockheed Missiles & Space Co., Inc.
P.O. Box 1103, West Station
Huntsville, Alabama 35807

Williams, D. F.
Computer Sciences Corporation
6022 Technology Drive
Huntsville, Alabama 35805

Williams, D. F.
U.S. Army Missile Command
ATTN: DRSMI-TG
Redstone Arsenal, Alabama 35898

Williams, George B.
Lockheed Missiles & Space Co., Inc.
P.O. Box 1103, West Station
Huntsville, Alabama 35807

Williams, Henry F.
Rockwell International
3370 Miraloma Ave., DF 39
Anaheim, California 92806

Williamson, Robert
Raytheon Co., Missile Systems Div.
Hartwell Road, MS M20-66
Bedford, Massachusetts 01730

Witsmeer, A. J.
Boeing Aerospace
P.O. Box 3999, MS 86-18
Seattle, Washington 98124

Wittmann, Horst R.
U.S. Army Research Office
P.O. Box 12211
Rip, North Carolina 27709

Wode, Melvin G.
Boeing Aerospace
P.O. Box 1470
Huntsville, Alabama 35807

Wohlers, M. R.
Grumman Aerospace Corporation
Research Dept., A08/35
Bethpage, New York 11714

Wright, Lloyd R.
System Planning Corporation
1500 Wilson Blvd.
Arlington, Virginia 22209

Yalkut, Carl R.
Office of Missile Electronic Warfare
ATTN: DELEW-M-STO
White Sands Missile Range, New Mexico 88002

Yoo, Jay K.
Mitre Corporation
1820 Dolly Madison Blvd.
McLean, Virginia 22101

Yutzi, Gerald N.
Rockwell International
3370 Miraloma Ave., MS D/521 022-BB85
Anaheim, California 92803

Zavodny, Ted
Technology Service Corporation
2811 Wilshire Blvd.
Santa Monica, California 90403

Zoracki, Dr. M. J.
Pattern Analysis & Recognition Corp.
228 Liberty Plaza
Rome, New York 13440

Zvolanek, Budimir
McDonnell Douglas Astronautics
Bldg. 106/2/E9, MS 34
P.O. Box 516
St. Louis, Missouri 63166

THE TACTICAL WEAPON GUIDANCE AND CONTROL INFORMATION ANALYSIS CENTER (GACIAC)

GACIAC is a DoD Information Analysis Center operated by IIT Research Institute under the sponsorship of the U.S. Army Missile Command. Its mission is to assist the tactical weapon guidance and control community by encouraging and facilitating the exchange and dissemination of technical data and information for the purpose of effecting coordination of research, exploratory development, and advanced technology demonstrations. To accomplish this, GACIAC's functions are to:

1. Develop a machine-readable bibliographic data base--currently containing over 27,000 entries;

2. Collect, review, and store pertinent documents in its field of interest--the library contains over 1500 reports in hard copy, and approximately one thousand more on microfiche;

3. Analyze, appraise and summarize information and data on selected subjects;

4. Disseminate information through the GACIAC Bulletin, bibliographies, state-of-the-art summaries, technology assessments, handbooks, special reports, and conferences;

5. Respond to technical inquiries related to tactical weapon guidance and control; and

6. Provide technical and administrative support to the Joint Service Guidance and Control Committee (JSGCC).

The products and services of GACIAC are available to qualified industrial users through a subscription plan or individual sales. Government personnel are eligible for products and services under block funding provided by the Army, Navy, and Air Force. A written request on government stationery is required to receive all the products as a government subscriber.

Further information regarding GACIAC services, products, participation plan, or additional copies of this Proceedings may be obtained by writing or calling: GACIAC, IIT Research Institute, 10 West 35th Street, Chicago, Illinois 60616, Area Code 312, 567-4519 or 567-4544.