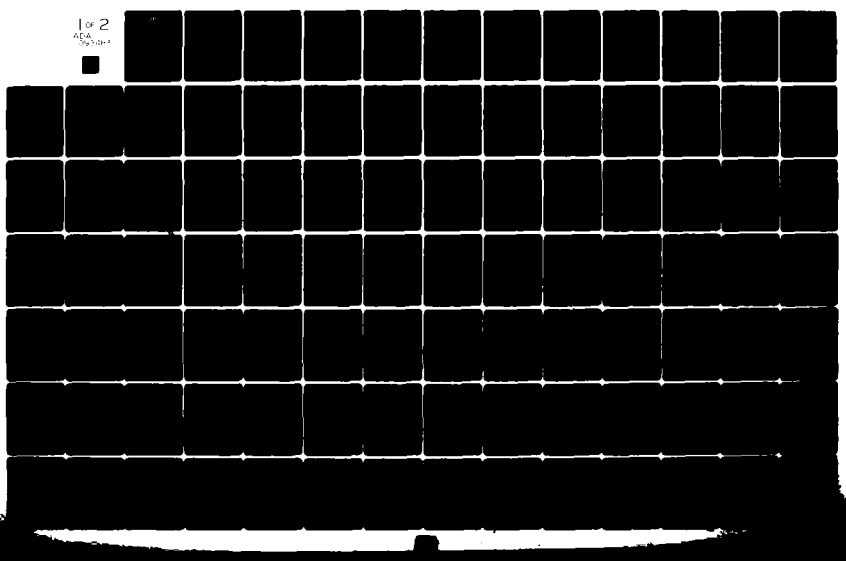
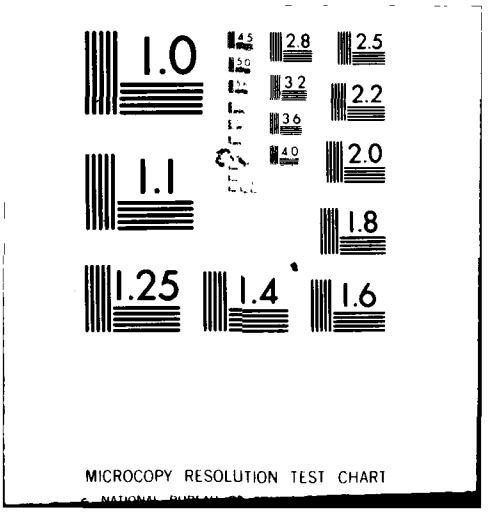
AD-A093 483 AUBURN UNIV ALA ENGINEERING EXPERIMENT STATION F/6 17/5
AUTOMATIC HANDOFF OF MULTIPLE TARGETS.(U)
SEP 80 J S BOLAND, H S RANGANATH DAAH01-80-C-0258
UNCLASSIFIED SBIE-AD-E950 071 NL




1.0
1.1
1.25
1.4
1.6
1.8
2.0
2.2
2.5
2.8
3.2
3.6
4.0
MICROCOPY RESOLUTION TEST CHART

LEVEL III

Automatic Handoff of Multiple Targets

by

J.S. Boland, III and H.S. Ranganath

Electrical Engineering Department
Auburn University
Auburn University, Alabama 36849

Final Technical Report - Phase I
For the Period 18 December 1979-30 September 1980

This research work was supported by
U.S. Army Missile Command
Redstone Arsenal, Alabama 35809
under Contract DAAH01-80-C-0258

Engineering Experiment Station
Auburn University
Auburn University, Alabama 36849

30 September 1980

Cleared for Public Release; Distribution Unlimited

AUTOMATIC HANDOFF OF MULTIPLE TARGETS

by

J. S. Boland, III and H. S. Ranganath

Electrical Engineering Department
Auburn University
Auburn University, Alabama 36849

Final Technical Report - Phase I
For the Period 18 December 1979-30 September 1980

This research work was supported by
U.S. Army Missile Command
Redstone Arsenal, Alabama 35809
under Contract DAAH01-80-C-0258

ENGINEERING EXPERIMENT STATION
Auburn University
Auburn University, Alabama 36849

30 September 1980

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER	2. GOVT ACCESSION NO. AD-A093 483	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) Automatic Handoff of Multiple Targets		5. TYPE OF REPORT & PERIOD COVERED Final Technical Report-Phase I 18 Dec 79-30 Sept 80
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) J. S. Boland, III and H. S. Ranganath		8. CONTRACT OR GRANT NUMBER(s) DAAH01-80-C-0258
9. PERFORMING ORGANIZATION NAME AND ADDRESS Engineering Experiment Station Auburn University Auburn University, AL 36849		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Headquarters, U.S. Army Missile Command ATTN: DRSMI-IYE/Oliver Redstone Arsenal, AL 35809		12. REPORT DATE 30 September 1980
		13. NUMBER OF PAGES 119
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) UNCLASSIFIED
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Cleared for Public Release; Distribution Unlimited

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

18. SUPPLEMENTARY NOTES

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

Correlation, Target Handoff, Image Correlation, Multiple Targets, Image Registration

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

In order to fully utilize the potential of the "fire and forget" class of helicopter-borne missiles, it is necessary to solve the technical problems associated with acquiring and handing off multiple targets from a precision pointing and tracking system (PTS) to several missile seekers simultaneously or

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

almost so in a short period of time. The multiple target problem is that of locating targets and missile seeker aim points within the PTS field of view, deciding which target is to be assigned to each missile, generating error signals to the torquers in order to slew the missile LOS such that its assigned target is in the center of its FOV, and initiating automatic seeker tracking.

The task of locating a given smaller image within a larger image is known as "image registration". A detailed comparison of the important multiple image registration methods based on the number of arithmetic operations for software implementation and the complexity of hardware for real time implementation is presented. New methods of accomplishing multiple image registration which are computationally more efficient than the most commonly used template matching techniques (correlation and sequential similarity detection algorithm) are described. Conclusions and recommendations are given.

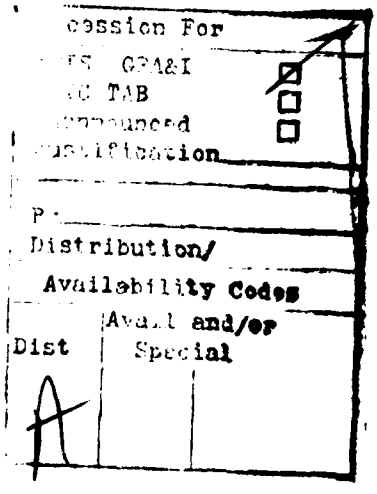


TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

I. INTRODUCTION

One concept which could potentially increase the firepower of the "fire-and-forget" class of helicopter borne missile systems would be to acquire and hand off multiple targets from a precision pointing and tracking system (PTS) to several missile seekers simultaneously, or almost so, in a short period of time. A typical over-all fire control configuration is shown in Figure 1-1. The pointing and tracking system typically consists of an optics train, line of sight (LOS) stabilization system, forward looking infra-red (FLIR) imaging system, manual and autotrack system, laser range finder and associated electronics. An imaging missile seeker could be an infrared type. It is assumed that during preflight checkout or during the actual flight, the lines of sight of all the missile seekers are aligned with the line of sight of the PTS. However, due to gyro drift, boresighting inaccuracies, vehicle vibration and flexure, etc., the seekers will not remain boresighted with the PTS. Since the PTS has a larger field of view (FOV) in both axes as compared to missile seekers, it is expected that the FOV of all the missile seekers will be located within the FOV of the PTS. The multiple target problem then becomes that of locating targets and missile seeker aim points within the PTS field of view, deciding which target is to be assigned to each missile, generating error signals to the torquers in order to slew the missile LOS such that its assigned target is in the center of its FOV, and initiating automatic seeker tracking.

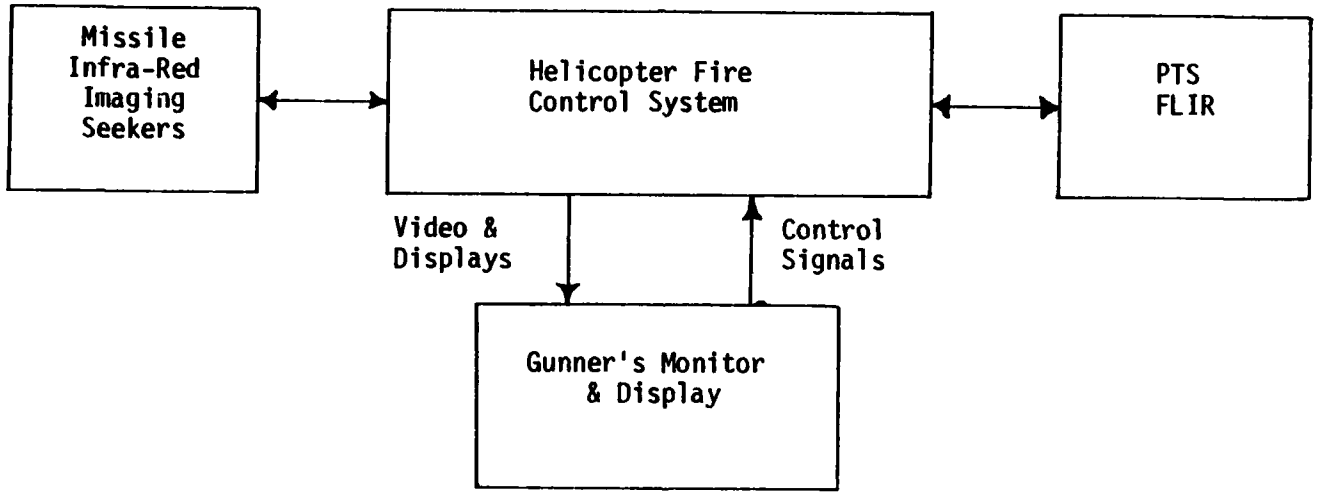


Figure 1-1. Helicopter missile fire control system.

In general, the task of locating a given smaller image within a larger image is known as "image registration". The smaller image is referred to as the window or the reference and the larger image is called the search area. With the above notation, in the multiple target problem, the image obtained from the PTS sensor is the search area and the image obtained from each missile seeker is a window. Therefore, there is one search area and more than one, say n, windows. It is assumed that all the n windows are completely located within the search area. Now the problem of multiple image registration can be defined as that of finding n subimages of the search area which best match the n windows. Even though very little attention has been given to date to the problem of multiple image registration, a considerable amount of work has been done in the area of single image registration. Several interesting problems such as map matching, cloud motion tracking, ship and aircraft identification are solved through digital image registration.

In Chapter II, the problem of single image registration is precisely defined and varuous existing methods of accomplishing digital image registration are described. The inherent problem associated with registration algorithms is their high computational cost. An algorithm which is computationally efficient for single image registration may not be efficient for multiple image registration. A detailed comparison of the important multiple image registration methods based on the number of arithmetic operations for software implementation and the complexity of hardware for real-time implementation is presented in Chapter III. New methods of accomplishing multiple image registration which are computationally more efficient than the most commonly used template matching

techniques (correlation and sequential similarity detection algorithm) are described in Chapter IV. Conclusions and recommendations are given in Chapter V.

This page intentionally left blank.

II. EXISTING METHODS FOR DIGITAL IMAGE REGISTRATION

The problem of locating a given image within a larger image uses techniques and algorithms fundamental to the disciplines of image processing and pattern recognition. Template matching methods such as "correlation" and "sequential similarity detection algorithms" are widely used for the determination of local similarity between two images [1] - [7]. The inherent problem associated with the above two methods or any image registration method is high computational time. Several schemes such as "two-stage template matching" and "course-fine template matching" have been proposed to speed-up template matching methods [8], [9]. The "method of invariant moments" which is widely used in classifying an unknown pattern as one of several known patterns can also be used to accomplish digital image registration [10] - [14]. Various methods of accomplishing single digital image registration are described in this chapter.

Problem of Digital Image Registration

Let two images, S the search area and W the window, be defined as shown in Figure 2-1. S is a MxN array of digital picture elements (pixels) which may assume one of G possible levels on the gray scale, i.e.,

$$0 \leq S(i,j) \leq G-1 \qquad (2\text{-}1)$$

for $1 \leq i \leq M$ and $1 \leq j \leq N$

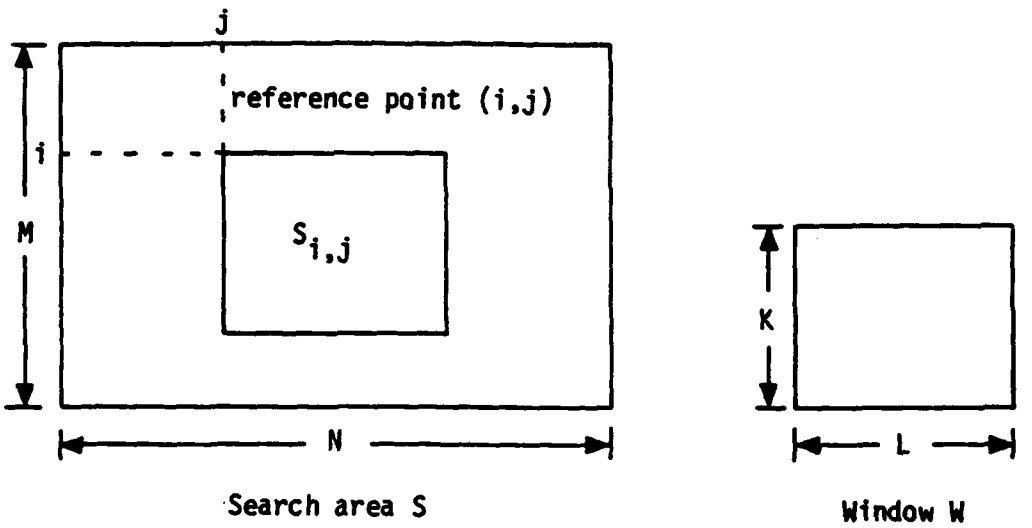


Figure 2-1. Search area and window.

W is a KxL (K < M and L < N) array of pixels having the same gray scale range.

$$0 \leq W(\ell,m) \leq G-1 \qquad (2\text{-}2)$$

for $1 \leq \ell \leq K$ and $1 \leq m \leq L$

Let $S_{i,j}$ denote each unique KxL subimage of S whose upper left corner coordinates are (i,j). Then (i,j) is also called the reference point of subimage $S_{i,j}$ and the (M-K+1)(N-L+1) reference points corresponding to the (M-K+1)(N-L+1) possible subimages of S are called allowable reference points.

$$S_{i,j}(\ell,m) = S(i+\ell-1\ ,\ j+m-1) \qquad (2\text{-}3)$$

for $1 \leq \ell \leq K$ and $1 \leq m \leq L$

$1 \leq i \leq M-K+1$ and $1 \leq j \leq N-L+1$

When S and W do not differ in pixel resolution and rotation (or have been preprocessed to equalize the pixel spatial resolution), digital image registration is a search over the allowed range of reference points to find the subimage $S_{i*,j*}$ which best matches the window W. Existing techniques for registering an image within a larger image and schemes to speed-up these techniques are presented in this chapter.

Correlation

When two images do not differ in pixel resolution and rotation, the method most widely used for image registration is cross-correlation [1] - [6]. The elements of the unnormalized cross-correlation surface, R(i,j), are defined to be

$$R(i,j) = \sum_{\ell=1}^{K} \sum_{m=1}^{L} W(\ell,m)\, S_{i,j}(\ell,m) \qquad (2\text{-}4)$$

for $1 \leq i \leq M-K+1$, $1 \leq j \leq N-L+1$

In most cases, coordinates of the maximum value of correlation surface indicate image registration. However, since R(i,j) is a cross correlation, it is possible that the maximum value of correlation surface does not indicate true image registration. This is illustrated below. Consider an ideal case where W exactly matches subimage S_{i^*,j^*}. Then

$$R(i^*,j^*) = \sum_{\ell=1}^{K} \sum_{m=1}^{L} W^2(\ell,m) \qquad (2\text{-}5)$$

Consider a nonmatching reference point $(\hat{i},\hat{j})$ where

$$S_{\hat{i},\hat{j}}(\ell,m) = \max_{(\ell,m)} W(\ell,m) = W_{max} \qquad (2\text{-}6)$$

for $1 \leq \ell \leq K$, $1 \leq m \leq L$.

Correlation value, $R(\hat{i},\hat{j})$, is given by

$$R(\hat{i},\hat{j}) = W_{max} \sum_{\ell=1}^{K} \sum_{m=1}^{L} W(\ell,m) \qquad (2\text{-}7)$$

It is easy to show that

$$R(\hat{i},\hat{j}) > R(i^*,j^*) \qquad (2\text{-}8)$$

Therefore, even in the ideal case a search for a maximum over the correlation surface does not necessarily yield true registration. In order that the maximum value of the correlation function indicate true image registration W and $S_{i,j}$ must be normalized. Elements of the normalized cross-correlation surface are defined to be

$$R(i,j) = \frac{\sum_{\ell=1}^{K}\sum_{m=1}^{L} W(\ell,m)\, S_{i,j}(\ell,m)}{\left[\sum_{\ell=1}^{K}\sum_{m=1}^{L} W^2(\ell,m)\right]^{\frac{1}{2}} \left[\sum_{\ell=1}^{K}\sum_{m=1}^{L} S^2_{i,j}(\ell,m)\right]^{\frac{1}{2}}} \qquad (2\text{-}9)$$

for $\quad 1 \le i \le M-K+1 \;,\; 1 \le j \le N-L+1$

This obviously involves more computation than the unnormalized method given by (2-4). In spite of its high computational cost, correlation is widely used in image registration for the following reasons:

1. Correlation appears to be a natural solution for the mean-square-error criterion.
2. Digital hardware and analog optical devices implement correlation easily.

In general, the amount of computation associated with any similarity detection method is proportional to the number of pixels in the window and the number of pixels in the allowable search area. In the correlation method each of the KL pixels in W is compared with the corresponding pixel in $S_{i,j}$ to compute R(i,j). Since the correlation function has (M-K+1)(N-L+1) elements a total of KL(M-K+1)(N-L+1) pairs of pixels are compared. Thus the total computation time is roughly proportional to KL(M-K+1)(N-L+1). The approximate number of arithmetic operations required to compute the normalized correlation surface is derived in Chapter III. Several modified versions of the standard correlation algorithm exist and each version has its own advantages and disadvantages. Vector correlation, feature matching correlation and hybrid correlation algorithms are briefly described on the next pages

Vector Correlation Algorithm

The standard correlation algorithm, as a measure of similarity, computes the cross correlation surface between the window W and the search area S by using only intensity levels for pixels of W and S. The vector correlation method computes the cross correlation surface between W and S based on gradient as well as gray scale values of pixels [15]. Let S(i,j) and GS(i,j) be the pixel and gradient values of the (i,j)th pixel of the search area. Similarly, W(ℓ,m) and GW(ℓ,m) are the pixel and gradient values of the (ℓ,m)th pixel of the window. Now, a two-dimensional vector consisting of intensity and gradient values can be associated with each pixel of S and W. Let $V^S(i,j)$ be the vector associated with the (i,j)th pixel of S, i.e.,

$$V^S(i,j) = \begin{bmatrix} S(i,j) \\ GS(i,j) \end{bmatrix} \tag{2-10}$$

for $1 \leq i \leq M$ and $1 \leq j \leq N$

Similarly, $V^W(\ell,m)$ denotes the vector associated with the (ℓ,m)th pixel of W.

$$V^W(\ell,m) = \begin{bmatrix} W(\ell,m) \\ GW(\ell,m) \end{bmatrix} \tag{2-11}$$

for $1 \leq \ell \leq K$ and $1 \leq m \leq L$

Let

$$V^S_{i,j}(\ell,m) = V^S(i+\ell-1,\ j+m-1) \tag{2-12}$$

for $1 \leq \ell \leq K$, $1 \leq m \leq L$

and $1 \leq i \leq M-K+1$, $1 \leq j \leq N-L+1$

Elements of the unnormalized vector correlation surface of S and W are defined to be

$$R(i,j) = \sum_{\ell=1}^{K} \sum_{m=1}^{L} (V^W(\ell,m))^T \, (V^S_{i,j}(\ell,m)) \tag{2-13}$$

for $1 \leq i \leq M-K+1$, $1 \leq j \leq N-L+1$

Elements of the normalized vector correlation surface are defined to be

$$R(i,j) = \frac{\sum_{\ell=1}^{K} \sum_{m=1}^{L} (V^W(\ell,m))^T \, (V^S_{i,j}(\ell,m))}{[\sum_{\ell=1}^{K} \sum_{m=1}^{L} (V^W(\ell,m))^T \, (V^W(\ell,m))]^{\frac{1}{2}}} \cdot \tag{2-14}$$

$$\frac{1}{[\sum_{\ell=1}^{K} \sum_{m=1}^{L} (V^S_{i,j}(\ell,m))^T \, (V^S_{i,j}(\ell,m))]^{\frac{1}{2}}}$$

for $1 \leq i \leq M-K+1$, $1 \leq j \leq N-L+1$

It can be easily shown that

$$R(i,j) = \frac{\sum_{\ell=1}^{K} \sum_{m=1}^{L} [W(\ell,m) S_{i,j}(\ell,m) + GW(\ell,m) \, GS_{i,j}(\ell,m)}{[\sum_{\ell=1}^{K} \sum_{m=1}^{L} [W^2(\ell,m) + GW^2(\ell,m)]]^{\frac{1}{2}}} \cdot \tag{2-15}$$

$$\frac{1}{[\sum_{\ell=1}^{K} \sum_{m=1}^{L} [S^2_{i,j}(\ell,m) + GS^2_{i,j}(\ell,m)]]^{\frac{1}{2}}}$$

for $1 \leq i \leq M-K+1$, $1 \leq j \leq N-L+1$

Since the vector correlation method is based on more independent information than the standard correlation method, it is expected to have a better

performance. However, the vector correlation method involves considerably more computation than the standard correlation method.

Feature Matching and Hybrid Correlation Algorithms

Since images W and S are obtained from two different sensors, a certain amount of preprocessing is necessary before computation of the correlation surface. In addition to the pixel spatial resolution equalization preprocessing, if the two sensors differ in dc gain and bias, each image can be preprocessed such that its mean pixel value is zero and standard deviation of pixel values is unity. This process is called intensity level normalization. Normalized images can then be correlated. There are two variations of the standard correlation algorithm suggested by the Rand Corporation [16]. Based on the preprocessing technique used, the algorithm is called either the feature matching correlation algorithm or the hybrid correlation algorithm.

Feature Matching Correlation Algorithm. In this method, the window (reference) is segmented into homogeneous regions. A homogeneous region is defined as a set of spatially connected pixels whose pixel values remain almost constant over the region. Each of the homogeneous regions is then preprocessed separately based on its characteristic. For example, intensity level normalization of a homogeneous region is accomplished by subtracting the mean pixel value of the region from each pixel and by normalizing with respect to the variance of pixel values. Similarly, the search area is also segmented into homogeneous regions and each homogeneous region is preprocessed separately. The preprocessed window and search areas are then correlated using a standard correlation

algorithm. Simulation results reported in Reference [16] indicate that this method yields a sharper correlation peak as compared to the standard correlation method. This may be due to the enhancement of high frequency content of each homogeneous region. The feature matching correlation also compensates for contrast reversals between the corresponding homogeneous regions of the window and the search area. However, depending on the scene and sensor resolution it may not be possible to decompose all scenes into homogeneous regions. Such scenes are called non-homogeneous scenes. Even if a scene is composed of homogeneous regions, it is extremely difficult to accomplish segmentation in real time.

Hybrid Correlation Algorithm. In hybrid correlation only the window is segmented into homogeneous regions. Each subimage of the search area is assumed to be the matching subimage and is segmented identically as the window. The correlation between the window and the subimage is computed by matching each homogeneous region of the window with its corresponding region in the subimage and by combining the partial results additively. Simulation results in Reference [16] show that this method is better than the standard correlation algorithm, but not as good as the feature matching correlation. However, the hybrid correlation algorithm has the advantage of not segmenting the search area and thus requires less computation as compared to the feature matching correlation.

The Fast Fourier Transform Method of Computing Correlation Function

The convolution theorem of fourier analysis states that convolution in the time or space domain is equivalent to multiplication in the temporal or spatial frequency domain. Since correlation is a form of convolution, an alternate method of computing the correlation function thus exists [17], [18]. Let X and Y be two images of the same size. Then the cross correlation between X and Y is given by

$$R(i,j) = \text{IFFT}\{\underline{X}(U,V)\ \underline{Y}^*(U,V)\} \tag{2-16}$$

where,

$\underline{X}(U,V)$ is the discrete fourier transform of X

$\underline{Y}^*(U,V)$ is the complex conjugate of the discrete fourier transform of Y

IFFT signifies the inverse fourier transform operation

The size of the correlation surface is the same as that of X or Y. However, R(0,0) is the only valid element and other elements are ignored.

Since W is a KxL array and S is a MxN array, Equation (2-16) cannot be directly used for the computation of the correlation function. This problem can be solved by padding W with zeros as described below.

$$0 \le S(i,j) \le G-1 \text{ , for } 0 \le i \le M-1 \text{ , } 0 \le j \le N-1 \tag{2-17}$$

and

$$0 \le W(\ell,m) \le G-1 \text{ , for } 0 \le \ell \le K-1 \text{ , } 0 \le m \le L-1 \tag{2-18}$$

Construct a new image W_1 of size MxN by padding W with zeros.

$$W_1(\ell,m) = \begin{cases} W(\ell,m) \text{ , for } 0 \le \ell \le K-1 \text{ , } 0 \le m \le L-1 \\ 0 \text{ , for } M > \ell \ge K \text{ or } N > m \ge L \text{ .} \end{cases} \tag{2-19}$$

Let $\underline{S}(U,V)$ and $\underline{W}_1(U,V)$ be the two dimensional DFT's of S and W. Now,

$$R(i,j) = IFFT[\underline{S}(U,V) \cdot \underline{W}_1^*(U,V)] \qquad (2\text{-}20)$$

for $0 \leq i \leq M-1$, $0 \leq j \leq N-1$

R(i,j) is valid for i = 0,1,2,...,M-K and j = 0,1,2,...,N-L and other values are ignored.

To compute the DFT or IDFT of an array of size MxN, MN $\log_2$MN complex multiplications and complex additions must be performed. MN complex multiplications are needed to multiply the DFT's of S and W_1. Since the FFT method yields the unnormalized correlation surface, it must be normalized. For large M and N, the FFT method of computing the correlation surface requires fewer calculations as compared to the direct approach. However, this method requires an additional memory of 4 MN real words which may not be feasible for large values of M and N. It is difficult to implement this method in real time due to hardware limitations.

Sequential Similarity Detection Algorithm

In this algorithm, a search over each of the (M-K+1)(N-L+1) reference points is performed as in correlation. However, the criterion for similarity at reference points is significantly different from that of the correlation method. The unnormalized error e'(i,j,ℓ,m) and the normalized error e(i,j,ℓ,m) between the pixel W(ℓ,m) and its corresponding pixel in $S_{i,j}$ are defined as

$$e'(i,j,\ell,m) \triangleq |S_{i,j}(\ell,m) - W(\ell,m)| \qquad (2\text{-}21)$$

$$e(i,j,\ell,m) \triangleq |S_{i,j}(\ell,m) - \hat{S}_{i,j} - W(\ell,m) + \hat{W}| \qquad (2\text{-}22)$$

where $$\hat{W} = \frac{1}{KL} \sum_{\ell=1}^{K} \sum_{m=1}^{L} W(\ell,m) \tag{2-23}$$

and

$$\hat{S}_{i,j} = \frac{1}{KL} \sum_{\ell=1}^{K} \sum_{m=1}^{L} S_{i,j}(\ell,m) \tag{2-24}$$

The correlation method yields the correlation surface as a measure of similarity, while the sequential similarity method computes the error surface as a measure of dissimilarity. The normalized error, E(i,j), associated with reference point (i,j) is defined as

$$E(i,j) = \sum_{\ell=1}^{K} \sum_{m=1}^{L} e(i,j,\ell,m) \tag{2-25}$$

for $1 \leq i \leq M-K+1$, $1 \leq j \leq N-L+1$

Now the problem of digital image registration reduces to the problem of finding S_{i^*,j^*} such that

$$E(i^*,j^*) < E(i,j) \text{ for } 1 \leq i \leq M-K+1 \text{ and } i \neq i^* \quad 1 \leq j \leq N-L+1 \text{ and } j \neq j^* \tag{2-26}$$

In general, computation of error is simpler than computation of correlation since addition takes less time than multiplication and is easier to implement. There are methods such as Constant Threshold Sequential Similarity Detection Algorithm" and "Monotonic Increasing Threshold Sequence Algorithms" suggested by Barnea and Silverman which further reduce the number of additions [7]. These methods are based on some kind of guess work or statistical assumptions and cannot be generalized. The number of arithmetic operations required to implement the SSDA method in its entirety is computed in Chapter III.

Moments Method

Given a two dimensional continuous function f(x,y), the moment m_{pq} of order (p+q) is defined by the relationship

$$m_{pq} = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} x^p y^q f(x,y)\, dx\, dy \qquad (2\text{-}27)$$

for $p,q = 0,1,2,3,\ldots$

A uniqueness theorem states that if f(x,y) is piecewise continuous and has non-zero values only in a finite region of the x-y plane, then the moments of all order exist and the moment sequence, $\{m_{pq}\}$, is uniquely determined by f(x,y) and conversely, $\{m_{pq}\}$ uniquely determines f(x,y) [19].

The central moments can be expressed as

$$\mu_{pq} = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} (x-\bar{x})^p\, (y-\bar{y})^q\, f(x,y)\, dx\, dy \qquad (2\text{-}28)$$

for $p,q = 0,1,2,\ldots$

where $\bar{x} = \dfrac{m_{10}}{m_{00}}$, $\bar{y} = \dfrac{m_{01}}{m_{00}}$

For a digital image these moments are given by

$$\mu_{pq} = \sum_x \sum_y (x-\bar{x})^p\, (y-\bar{y})^q\, f(x,y) \qquad (2\text{-}29)$$

The normalized central moments are defined as

$$\eta_{pq} = \frac{\mu_{pq}}{\mu_{00}^{\frac{p+q}{2}+1}} \text{ , for } p,q = 2,3,4,\ldots \qquad (2\text{-}30)$$

Hu has derived a set of seven invariant moments from second and third order central moments [10]. They are given by

$$\Phi_1 = \eta_{20} + \eta_{02} \tag{2-31}$$

$$\Phi_2 = (\eta_{20} - \eta_{02})^2 + 4\,\eta_{11}^2 \tag{2-32}$$

$$\Phi_3 = (\eta_{30} - 3\eta_{12})^2 + (3\eta_{21} - \eta_{03})^2 \tag{2-33}$$

$$\Phi_4 = (\eta_{30} + \eta_{12})^2 + (\eta_{21} + \eta_{03})^2 \tag{2-34}$$

$$\Phi_5 = (\eta_{30} - 3\eta_{12})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] + (3\eta_{21} - \eta_{03})(\eta_{21} + \eta_{03})[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] \tag{2-35}$$

$$\Phi_6 = (\eta_{20} - \eta_{02})[(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] + 4\eta_{11}(\eta_{30} + \eta_{12})(\eta_{21} + \eta_{03}) \tag{2-36}$$

$$\Phi_7 = (3\eta_{12} - \eta_{30})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] + (3\eta_{12} - \eta_{03})(\eta_{21} + \eta_{03})[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] \tag{2-37}$$

This set of moments has been shown invariant to translation, rotation, reflection and scale change [10].

The method of moments is used for automatic classification of an unknown pattern as one of several known patterns. Let $a_1, a_2, \ldots, a_k$ be k known patterns whose invariant moments are also known. Let x be a given unknown pattern to be classified as one of the k known patterns. Now, the moments method consists of computing the invariant moments for

x and comparing them with the invariant moments of a_1, a_2, ..., a_k. x is classified as a_i if the invariant moments of x best matches, according to some prespecified rule, the invariant moments of a_i. The moments method is widely used in applications such as visual and digital pattern recognition, recognition of two dimensional patterns with linear distortion, aircraft and ship identification and character recognition [10] - [14]. It is not widely used for image registration because of the enormous computation involved. However, this method looks promising for the multiple image registration problem for the following reasons.

1. If the images do not differ in rotation and resolution, it is not necessary to compute the normalized central moments or the invariant moments. The moment sequence $\{m_{pq}\}$ determines $f(x,y)$ uniquely and can be used to characterize each subimage of S.
2. It may be possible to obtain good registration results using very few moments. This trade-off should be investigated in a follow-on program using simulation of real digitized scenes.
3. If the correlation or SSDA method is used for multiple image registration, the correlation surface or error surface has to be determined for each of the windows separately. In other words, computation is directly proportional to the number of windows. For the moments

method, however, once the moments are computed for all subimages and windows, matching is accomplished with negligible computation.

As a result, the moments method may prove economical for the multiple image registration problem.

Hough Transformation for Digital Image Registration

One possible way of accomplishing digital image registration is by detecting the predominant lines in the window and the search areas and using that information to find the subimage of S which best matches the window. Consider an image consisting of a number of discrete white points (edge points) on a black background. The problem is to detect the groups of colinear or almost colinear edge points (white points). Of course, the problem can be solved by testing the lines formed by all pairs of edge points. However, computation required for n points is approximately proportional to n^2 and may be prohibitive for large n.

In 1962, Hough proposed an ingenious method of detecting lines in binary images [20], [21]. He replaced the original problem of finding colinear points by a mathematically equivalent problem of finding concurrent lines. A straight line is given by the equation

$$y = mx + c \tag{2-38}$$

where m is the slope and c is the intercept. Any line can be uniquely identified by its slope and intercept. Therefore, a line in the x-y plane maps into a point in the m-c parameter plane and vice versa. Similarly, a point in the x-y plane maps into a line in the parameter plane.

This can be seen by letting x and y be constants and solving for m as a function of c in Equation (2-38). The result is

$$m = \frac{1}{x}c + y/x = ac + b \tag{2-39}$$

where a and b are constants. In the m vs. c plane, (2-39) is the equation of a straight line. Thus, n points on a straight line in x-y plane are transformed to n lines which intersect at a common point in the m-c plane. Therefore, the problem of finding colinear points in the x-y plane is equivalent to that of finding concurrent lines in m-c plane.

Slope and intercept both being unbounded complicate the application of the Hough transformation for line detection. In order to overcome this problem, Duda and Hart suggested the use of an angle-radius rather than slope-intercept parameter plane [22]. A straight line can be uniquely specified by the angle θ of its normal and its algebraic distance ρ from the origin as shown in Figure 2-2. The line can now be represented as

$$x \cos \theta + y \sin \theta = \rho \tag{2-40}$$

where θ is bounded and takes on values between 0 and 2π and ρ is less than or equal to R, where R depends on the size of the image. From Equation (2-40) the following properties of the Hough transformation can be easily verified.

1. A point in the image plane (x-y plane) corresponds to a sinusoidal curve in the parameter plane. This can be seen by letting x and y be constants in Equation

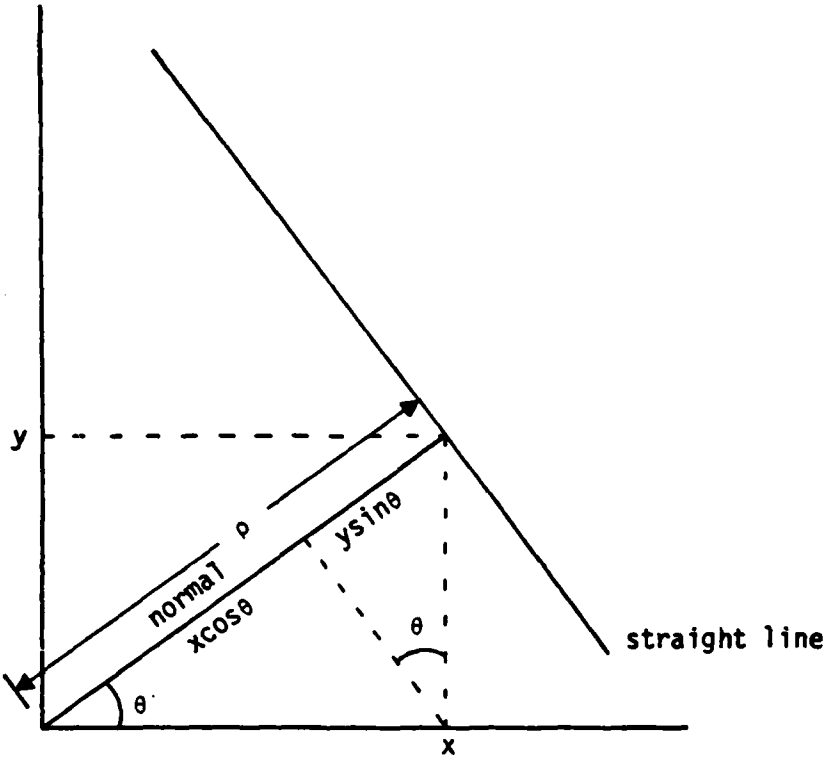


Figure 2-2. The normal co-ordinates of a straight line.

(2-40) and solving for ρ in terms of θ. The result is $\rho = K \sin(\theta + \emptyset)$ where $K = \sqrt{x^2 + y^2}$ and $\emptyset = \arctan x/y$.

2. A point in the parameter plane corresponds to a line in the image plane. This can be seen by letting ρ and θ be constants in Equation (2-40). The result is $y = ax + b$ where $b = \rho/\sin\theta$ and $a = -\cos\theta/\sin\theta$.
3. Points lying on a straight line in the image plane map into sinusoidal curves in the parameter plane each passing through a common point.
4. Points lying on the same sinusoidal curve in the parameter plane correspond to a family of lines through one point in the image plane.

Therefore, if all the edge points are mapped to the parameter plane, the problem of finding colinear edge points in the x-y image plane becomes that of finding concurrent sinusoidal curves in the parameter plane. The point of intersection of these sinusoidal curves uniquely identifies the straight line edge in the image plane.

Detection of Lines in Digital Images

Let F be a digital image of size KxL whose pixels can assume one of G possible levels on the gray scale. Let GF be the gradient image of F which can be computed using any of the known edge detection algorithms (e.g., Sobel edge detector or Roberts cross operator). GF is then transformed into a binary image by setting all pixels with gradient values greater than a predetermined threshold to one (edge points) and all remaining pixels to zero (non-edge points). Let n be the number of edge

points in the binary image. Suppose all edge points are mapped into their corresponding sinusoidal curves in the parameter plane. In general, these n curves will intersect at $\frac{n(n-1)}{2}$ points corresponding to $\frac{n(n-1)}{2}$ possible lines. Exactly colinear subsets of edge points can be found, in principle, by finding coincident points in the parameter plane. Unfortunately, this method is exhaustive and computation grows quadratically with the number of edge points.

When it is not necessary to determine the lines exactly, following Hough's basic proposal, the ρ-θ plane can be quantized into a quadruled grid on the basis of an acceptable error in ρ and θ. The quantization is confined to the region $0 \leq \theta \leq 2\pi$ and $0 \leq \rho \leq R$, where R depends on the size of the image. Assume that an accumulator is placed in each cell of the grid. For each edge point (x_i, y_i), the sinusoidal curve given by Equation (2-40) is entered in the grid by incrementing the count in each accumulator along the curve. When all edge points are mapped, each accumulator contains the number of curves through it. A count of k in accumulator cell (θ_i,ρ_j) means that precisely k edge points lie (to within the grid quantization error) along the line whose normal co-ordinates are θ_i and ρ_j. However, the exact location of these k edge points on this line in the x-y plane is not known (i.e., it is not known if the edge points are adjacent or widely separated). To determine this, some sort of connectivity test must be used.

Digital Image Registration

Step 1: Transform the window and the search area to binary images as described previously.

Step 2: Divide the θ-ρ parameter plane into a quadruled grid on the basis of acceptable error in θ and ρ.

Step 3: Map each edge point in the reference into its sinusoidal curve in θ-ρ plane. Even if the acceptable error in θ and ρ is moderate (5 degrees and 3 pixels), the parameter plane will have more than KL cells. As a result any attempt to use all the information in the θ-ρ plane will increase computation. In order to accomplish data reduction only the predominant lines (cells with relatively high count) are retained as features of the window and the remaining information is ignored.

Step 4: Determine the predominant lines present in each subimage of the search area by repeating the procedure outlined in Step 3.

Step 5: Match the line features of the window with those of each subimage according to some predetermined criterion. One way of doing this in the θ-ρ plane is given in Reference [23].

It is felt that Hough's method of image matching is not very sensitive to slight geometric distortion and rotation and is also insensitive to small differences in pixel resolution of the window and the search area. However, this method requires a highly reliable algorithm to quantize gradient images to two levels. In the parameter plane, cell (θ_i,ρ_j) represents the line in the picture plane whose normal co-ordinates are θ_i and ρ_j and the corresponding count gives the total number of edge points on that line. The exact location of edge points on the line is not known. Therefore, in order to determine a true line (i.e., points

are adjacent on a line) and prevent isolated edge points from affecting the result, Hough's method may have to be coupled with some kind of connectivity test.

Schemes to Speedup Template Matching

"Template matching" is the common terminology used for correlation and sequential similarity detection algorithms. In template matching, each of the KL pixels in W is compared with its corresponding pixel in $S_{i,j}$ to compute the measure of similarity (correlation) or dissimilarity (sequential similarity detection algorithm). Therefore, the total amount of computation is roughly proportional to the product of the number of pixels in the reference and the number of allowable reference points in the search area. Since there are KL pixels in W and (M-K+1)(N-L+1) allowable reference points in S, computation is proportional to KL(M-K+1)(N-L+1). Three popular schemes of accomplishing savings in computation and speeding up template matching are presented in this section. All methods accomplish savings in computation by reducing the total number of pixel pair comparisons. These methods are computationally more efficient in terms of the number of arithmetic operations required (software implementation) but may not enjoy any advantage in real time implementation using special purpose hardware.

Two-Stage Template Matching

The two-stage template matching technique, suggested by Rosenfeld and Vanderbrug, searches for $S_{i*,j*}$ which best matches W in two stages [8].

Stage 1: In the first stage, some subimage W' of size pxq from W and its corresponding subimage $S'_{i,j}$ of the same size in $S_{i,j}$ are matched to compute a measure of similarity (correlation) or dissimilarity (SSDA) between W and $S_{i,j}$ for i = 1,2,...,M-K+1 and j = 1,2,... N-L+1. W' and $S'_{i,j}$ are shown in Figure 2-3. The net effect of step one is to find the (M-K+1) by (N-L+1) correlation surface with a reduced reference array size.

Stage 2: In the second stage, all reference points, (i,j) for which the measure of similarity is less than (correlation) or the measure of dissimilarity is greater than (SSDA) a predetermined threshold T are discarded as non-match points. At the remaining reference points, the window W is matched with $S_{i,j}$ in its entirety. The method of finding $S_{i*,j*}$ is the same as before. The computational savings in this two-stage recognition procedure results from not having to match the entire template at each reference point. Savings in computation depend on the size of W' and threshold T.

Rosenfeld has suggested a method to determine optimal values for the size of W' and T for a given W and S. His analytical model is based on many simplifying assumptions which are rarely true for typical military type images. Improper selection of W' and T can lead to the possibility of discarding the true registration point in the first stage itself.

Course-fine Template Matching

Course-fine template matching is also a two-stage matching algorithm [9]. For the first stage, the spatial resolution of both image

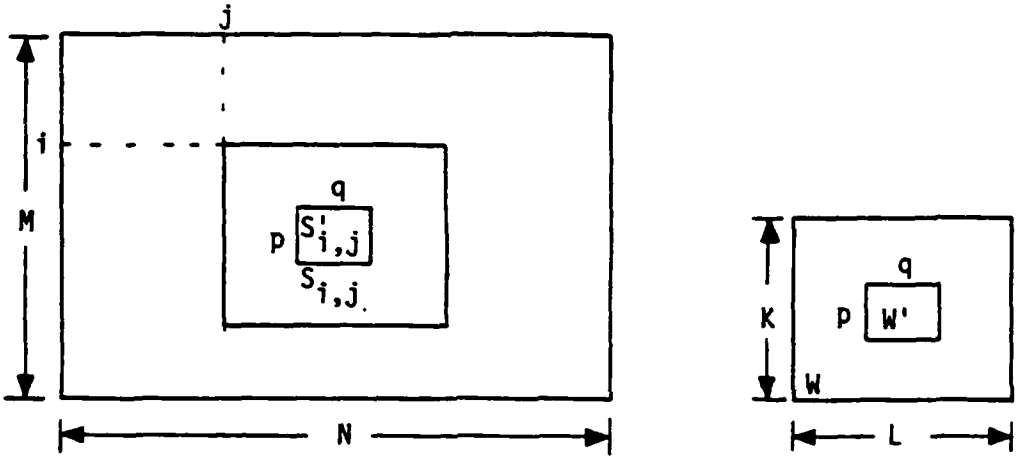


Figure 2-3. Layout for two stage template matching.

arrays is reduced by replacing each pxq subarray by its average. This yields a reference array, W', of size K/p x L/q and a search area, S', of size M/p x N/q. The amount of computation required to find the course correlation surface for W' and S' is proportional to $\frac{KL}{pq}\left(\frac{M}{p} - \frac{K}{p} + 1\right)\left(\frac{N}{q} - \frac{L}{q} + 1\right)$. At each of the correlation surface peaks for which the correlation value is greater than T, the original reference W is correlated with the original search array. The largest value of this correlation computation is treated as the registration point. Savings in computation depend on p, q and T. This technique can also be used with the SSDA method.

Hierarchical Search Method

This technique is a generalization of the "course-fine template matching" scheme. In this method the search for $S_{i*,j*}$ is done in n-resolution levels [24], [25]. From a given window W and the search area S, a set of windows $\{W^1, W^2, \ldots W^n\}$ and a set of search areas $\{S^1, S^2, \ldots, S^n\}$ are created as shown in Figure 2-4. Resolution of the window and search area in the X or Y direction at any level is twice the resolution of the window and search area for the next level, respectively. W^i can be created from W^{i-1} by dividing W^{i-1} into blocks of size 2x2 and treating each block as a pixel with value equal to the average of its four pixels. Similarly, S^i can be created from S^{i-1}. Matching starts at the lowest resolution level (level-n) where W^n of size $\frac{K}{2^n} \times \frac{L}{2^n}$ is matched with S^n of size $\frac{M}{2^n} \times \frac{N}{2^n}$. The amount of computation at this level, therefore, is proportional to $\frac{K}{2^n}\frac{L}{2^n}\left(\frac{M}{2^n} - \frac{K}{2^n} + 1\right)\left(\frac{N}{2^n} - \frac{L}{2^n} + 1\right)$. Based on some predetermined criterion, only the most promising test locations are selected for testing in $(n-1)^{th}$ level.

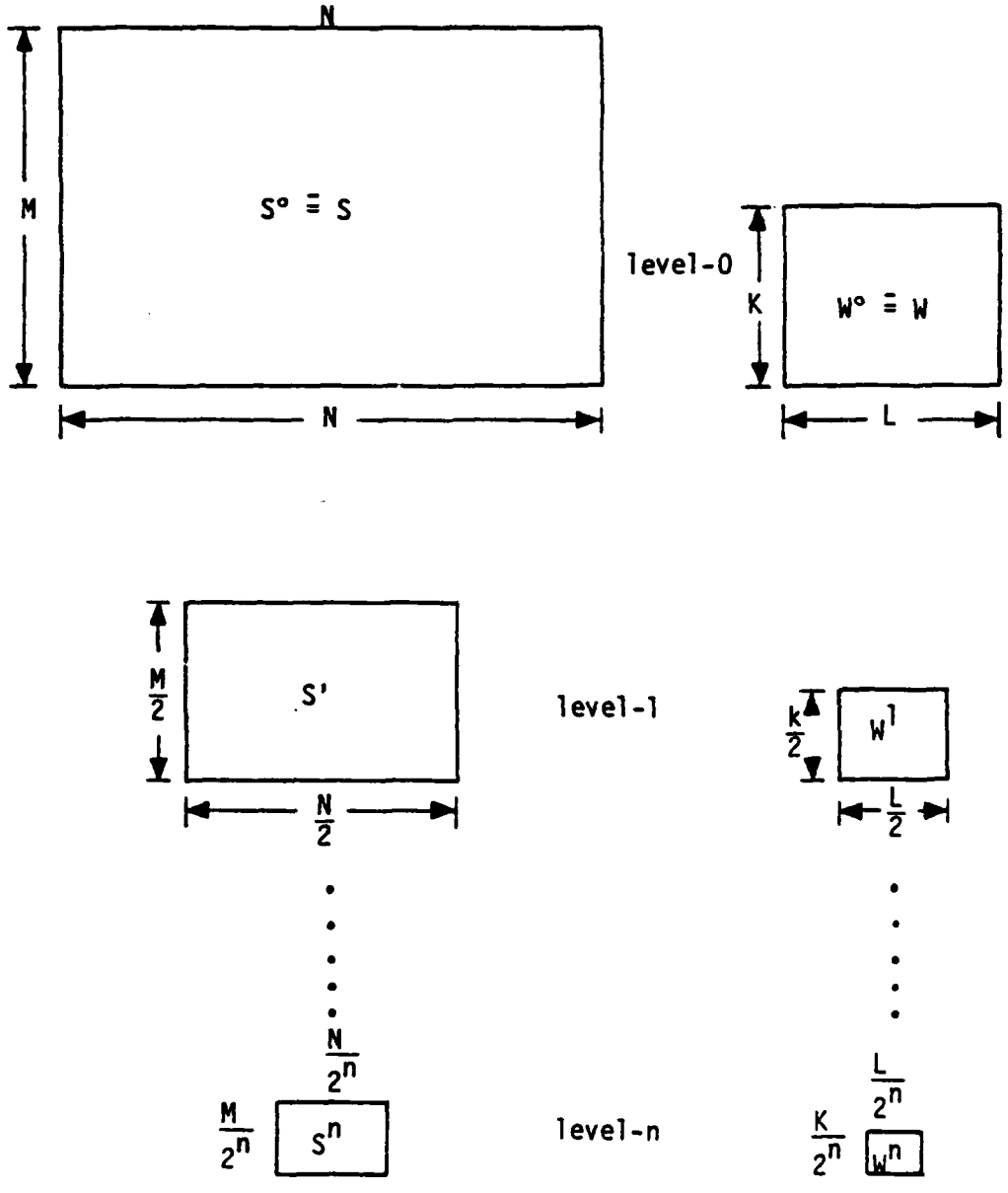


Figure 2-4. Search area and window for various levels.

In the $(n-1)^{th}$ level, W^{n-1} is matched with S^{n-1} only at locations selected in n^{th} level. This procedure continues from the $(n-1)^{th}$ level to the $(n-2)^{th}$ level and so on. Finally at the 0^{th} level, the registration point is identified. Savings in computation depends on the number of levels used and the criteria used to select promising test locations at each level. It should be pointed out, however, that the more points eliminated at each level, the greater is the possibility of obtaining a false match.

In this chapter various existing methods of accomplishing digital image registration are presented. The most commonly used method is cross-correlation. There exist two independent ways of computing the correlation surface (Direct method and Fast fourier transform method). The FFT method requires a large amount of memory for software implementation and is very complex for real time hardware implementation. The direct method requires less memory as compared to the FFT method, involves no complex multiplications or complex additions, and can be easily implemented in real time using digital hardware. Even though the vector correlation algorithm is expected to yield better performance than the standard correlation algorithm, its use is limited by the large amount of computation required to implement the method (more than twice the computation needed by the standard correlation algorithm). Due to the difficulty encountered in the automatic segmentation of digital images into homogeneous regions, the two variations of the standard correlation algorithm, namely, feature matching correlation and hybrid correlation algorithms may not be of any significant use. The other template matching method, sequential similarity detection algorithm, computes an error surface as

a measure of dissimilarity between the window and the search area. Since this method requires addition and a few division operations, it can be easily implemented (addition and subtraction operations are simpler than multiplication and division operations).

Algorithms which accomplish image registration by matching moments (moments method) or straight line edge content of the window and the search area are called feature matching algorithms. The moments method, which is computationally inefficient for single image registration, looks promising for multiple image registration. The necessity of a highly efficient algorithm to transform a digital image to a binary image (edge and non-edge pixels) and a connectivity test to identify true straight line edges (i.e., composed of adjacent edge pixels) makes the use of Hough's transformation for digital image registration less attractive. Therefore, it is concluded that the standard correlation algorithm, the SSDA and the moments method are more promising using present state-of-the-art hardware. A detailed comparison of the above three methods is presented in the next chapter. It should be pointed out, however, that very large scale integrated circuits developed to perform a specialized task may at some future date make any of the algorithms discussed above as being computationally inefficient feasible. Because of this possibility, a follow-on program should investigate the computational accuracy of some of these methods.

III. COMPARISON OF METHODS FOR MULTIPLE IMAGE REGISTRATION

Problem of Multiple Image Registration

Let the search area S and n windows W_1, W_2, ..., W_n be defined as shown in Figure 3-1. S is a MxN array of digital picture elements which may assume one of G possible levels on the gray scale.

$$0 \leq S(i,j) \leq G-1 \tag{3-1}$$

for $1 \leq i \leq M,\ 1 \leq j \leq N$

W_k is a KxL array of pixels having the same gray scale range.

$$0 \leq W_k(\ell,m) \leq G-1 \tag{3-2}$$

for $1 \leq \ell \leq K,\ 1 \leq m \leq L$ and $k = 1, 2, \ldots, n$

Each KxL subimage of S can be uniquely identified by its upper left corner's coordinates. Let $S_{i,j}$ denote the KxL subimage of S whose upper left corner is (i,j).

$$S_{i,j}(\ell,m) = S(i+\ell-1,\ j+m-1) \tag{3-3}$$

for $1 \leq \ell \leq K,\ 1 \leq m \leq L$

and $1 \leq i \leq M-K+1,\ 1 \leq j \leq N-L+1$

If S and W_k do not differ in pixel resolution and rotation, the multiple image registration problem reduces to that of finding (i_k^*, j_k^*) such that $S_{i_k^*,j_k^*}$ best matches W_k, for $k = 1, 2, \ldots, n$.

Correlation and sequential similarity detection algorithms are commonly used for the registration of a smaller image within a larger

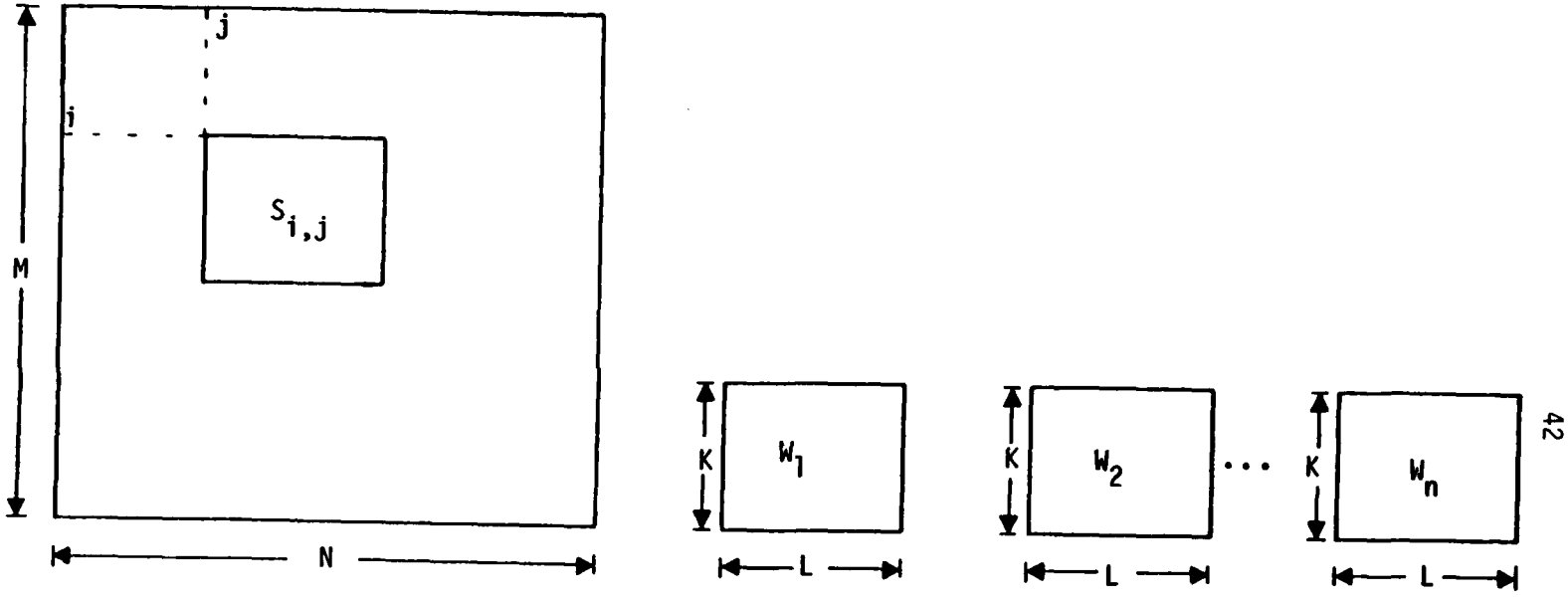


Figure 3-1. Search area and windows.

image. These methods have been proven to be reliable and computationally efficient for the problem of single image registration. However, a method which is efficient for single image registration may not be efficient for multiple image registration. In general, in order to compare the computational efficiency of different algorithms, the number of arithmetic operations, memory requirement, computational speed and complexity of implementation must be considered. The method which requires fewer arithmetic operations for software implementation may be complex or even infeasible for hardware implementation. It is extremely difficult to arrive at a valid means of comparison without knowing, exactly, how the methods are implemented. Therefore, the problem of multiple image registration is studied from both software as well as hardware points of view, independently.

Comparison of Software Implementations

If the algorithms are implemented entirely using software, the amount of core memory required and the number of arithmetic operations to be performed can be used as a means of comparison. It is assumed that:

a. Multiplications, divisions and squaring operations are equivalent (i.e., 1 multiplication = 1 division = 1 squaring).

b. Enough memory is available to store the intermediate results for future use.

c. Computation time is directly proportional to the number of arithmetic operations performed to implement the method.

Multiplication and addition requirements for the correlation, the SSDA and the moments methods are derived in the following sections.

Correlation Algorithm

Elements of normalized correlation function of search area S and window W_k are defined to be

$$R_k^2(i,j) = \frac{[\sum_{\ell=1}^{K}\sum_{m=1}^{L} W_k(\ell,m)\, S_{i,j}(\ell,m)]^2}{[\sum_{\ell=1}^{K}\sum_{m=1}^{L} W_k^2(\ell,m)][\sum_{\ell=1}^{K}\sum_{m=1}^{L} S_{i,j}^2(\ell,m)]} \tag{3-4}$$

for $1 \le i \le M-K+1$, $1 \le j \le N-L+1$

and $k = 1, 2, \ldots, n$

Let

$$A_{i,j}^k = [\sum_{\ell=1}^{K}\sum_{m=1}^{L} W_k(\ell,m)\, S_{i,j}(\ell,m)]^2 \tag{3-5}$$

$$B_k = \sum_{\ell=1}^{K}\sum_{m=1}^{L} W_k^2(\ell,m) \tag{3-6}$$

$$C_{i,j} = \sum_{\ell=1}^{K}\sum_{m=1}^{L} S_{i,j}^2(\ell,m) \tag{3-7}$$

Therefore,

$$R_k^2(i,j) = \frac{A_{i,j}^k}{B_k\, C_{i,j}} \tag{3-8}$$

for $1 \le i \le M-K+1$, $1 \le j \le N-L+1$

and $k = 1, 2, \ldots, n$

Equations for the number of multiplications and additions in terms of M, N, K, L and n are derived in the four steps below:

1. Computation of $R_k^2(i,j)$ from $A_{i,j}^k$, B_k and $C_{i,j}$ requires one multiplication and one division or two equivalent multiplications. Since there are (M-K+1)(N-L+1) elements in each of the n correlation surfaces, a total of 2n(M-K+1)(N-L+1) equivalent multiplications are required for this stage.
2. To compute $A_{i,j}^k$, (KL+1) multiplications and (KL-1) additions are performed. Since, for each of the n windows, $A_{i,j}^k$ must be computed at (M-K+1)(N-L+1) reference points, this task requires a total of (M-K+1)(N-L+1)(KL+1)n multiplications and (M-K+1)(N-L+1)(KL-1)n additions.
3. B_k is computed for each of the n windows and thus requires KLn multiplications and (KL-1)n additions.
4. $C_{i,j}$ for i = 1, 2, ..., M-K+1 and j = 1, 2, ..., N-L+1 can be computed in many ways. In this report, it is assumed that the $S_{i,j}^2(\ell,m)$ values, for $1 \leq \ell \leq K$ and $1 \leq m \leq L$, are first computed and stored.

Then $C_{i,j}$ is computed as

$$C_{i,j} = \sum_{\ell=1}^{K} \sum_{m=1}^{L} S_{i,j}^2(\ell,m) \qquad (3\text{-}9)$$

for $1 \leq i \leq M-K+1$ and $1 \leq j \leq N-L+1$

Computation of $C_{i,j}$'s is done only once and the values are stored in memory for later use. The above tasks require MN multiplications and

(M-K+1)(N-L+1)(KL-1) additions. This technique requires an additional MN+(M-K+1)(N-L+1) memory locations.

By adding the partial results of the above four steps,

$$\text{Total number of additions} = [(M-K+1)(N-L+1) + 1][(KL-1)n] + (M-K+1)(N-L+1)(KL-1) \quad (3\text{-}10)$$

$$\text{Total number of multiplications} = [(M-K+1)(N-L+1)(KL+3)+KL]n + MN \quad (3\text{-}11)$$

For $M = 240$, $N = 256$, $K = L = 32$,

$$\text{Total number of additions} = 48107598n + 48106575 \quad (3\text{-}12)$$

$$\text{Total number of multiplications} = 48295699n + 61440 \quad (3\text{-}13)$$

Sequential Similarity Detection Algorithm

The correlation method yields the correlation surface as a measure of similarity, while the sequential similarity method computes the error surface as a measure of dissimilarity. The normalized error, $e_k(i,j,\ell,m)$, between $W_k(\ell,m)$ and $S_{i,j}(\ell,m)$ is defined as

$$e_k(i,j,\ell,m) \triangleq |S_{i,j}(\ell,m) - \hat{S}_{i,j} - W_k(\ell,m) + \hat{W}_k| \quad (3\text{-}14)$$

where

$$\hat{W}_k = \frac{1}{KL} \sum_{\ell=1}^{K} \sum_{m=1}^{L} W_k(\ell,m) \quad (3\text{-}15)$$

and

$$\hat{S}_{i,j} = \frac{1}{KL} \sum_{\ell=1}^{K} \sum_{m=1}^{L} S_{i,j}(\ell,m) \tag{3-16}$$

The normalized error, $E_k(i,j)$, associated with the reference point (i,j) is defined as

$$E_k(i,j) = \sum_{\ell=1}^{K} \sum_{m=1}^{L} e_k(i,j,\ell,m) \tag{3-17}$$

for $\quad 1 \leq i \leq M-K+1 \ , \ 1 \leq j \leq N-L+1$

To register W_k within S, (i_k^*,j_k^*) must be found such that

$$E_k(i_k^*,j_k^*) < E_k(i,j) \text{ for, } 1 \leq i \leq M-K+1,\ i \neq i_k^* \text{ and } 1 \leq j \leq N-L+1,\ j \neq j_k^* \tag{3-18}$$

Equations for the number of multiplications and additions required to register n windows using the SSDA method are derived in the following steps:

1. To compute each $\hat{W}_k$, (KL-1) additions and one multiplication (division) are required. Since there are n such windows, a total of (KL-1)n additions and n multiplications are performed to compute $\hat{W}_k$ for k = 1, 2, 3, ..., n.
2. It is assumed that $\hat{S}_{i,j}$ is computed independently at each reference point and stored in memory for later use. This task needs (M-K+1)(N-L+1)(KL-1) additions and (M-K+1)(N-L+1) multiplications and requires (M-K+1)(N-L+1) memory locations.
3. Computation of $E_k(i,j)$ requires (3KL)+(KL-1) = (4KL-1) additions. Since there are (M-K+1)(N-L+1) elements

in each of the n error surfaces, a total of (M-K+1)(N-L+1)(4KL-1)n additions are performed to compute n error surfaces.

Therefore,

$$\text{Total number of additions} = [(M-K+1)(N-L+1)(4KL-1)+(KL-1)]n + (M-K+1)(N-L+1)(KL-1) \quad (3\text{-}19)$$

$$\text{Total number of multiplications} = (M-K+1)(N-L+1) + n \quad (3\text{-}20)$$

For $M = 240$, $N = 256$, $K = L = 32$

$$\text{Total number of additions} = 192568393n + 48106575 \quad (3\text{-}21)$$

$$\text{Total number of multiplications} = 47025 + n \quad (3\text{-}22)$$

The number of multiplications required is negligible when compared to the number of additions.

Moments Method

Given a two dimensional function f(x,y), the moment m_{pq} of order (p+q) is defined by the relation,

$$m_{pq} = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} x^p y^p f(x,y)dx\, dy \quad (3\text{-}32)$$

for $p,q = 0, 1, 2, 3, \ldots$

A uniqueness theorem states that if f(x,y) is piecewise continuous and has non-zero value only in a finite region of X-Y plane, then the moments of

all order exist and the moment sequence, $\{m_{pq}\}$, is uniquely determined by f(x,y) and conversely, $\{m_{pq}\}$ uniquely determines f(x,y) [10]. For a digital image these moments are given by

$$m_{pq} = \sum_{x} \sum_{y} x^p y^q f(x,y) \tag{3-24}$$

for $p,q = 0, 1, 2, 3, \ldots$

Since the digital image satisfies all the conditions required by the uniqueness theorem as stated above, moments of all order exist and the moment sequence $\{m_{pq}\}$ can be used to accomplish digital image registration as described in the three steps below.

1. Let $\{m_{pq}^k\}$ denote the moment sequence for the window W_k. Compute all moments of order less than or equal to r where r is a predetermined number, for each of the n windows.

$$m_{pq}^k = \sum_{\ell=1}^{K} \sum_{m=1}^{L} \ell^p m^q W_k(\ell,m) \tag{3-25}$$

for $p,q = 0, 1, 2, 3, \ldots, r$, such that $p+q \leq r$

and $k = 1, 2, \ldots, n$.

Although the reliability of this method increases with increasing r, the amount of computation increases with r and therefore a trade-off exists. For some applications, the moments method has been found to be successful for r equal two or three [10] - [14].

2. Compute all moments of order r and less for each of the (M-K+1)(N-L+1) subimages of S.

$$m_{pq}^{ij} = \sum_{\ell=1}^{K} \sum_{m=1}^{L} \ell^p m^q S_{i,j}(\ell,m) \qquad (3\text{-}26)$$

for $p,q = 0, 1, 2, \ldots, r$, such that $p+q \leq r$

and $1 \leq i \leq M-K+1$, $1 \leq j \leq N-L+1$.

3. To register W_k within S, find (i_k^*, j_k^*) such that

$$\sum_{pq} (m_{pq}^k - m_{pq}^{i_k^*, j_k^*})^2 < \sum_{pq} (m_{pq}^k - m_{pq}^{ij})^2 \qquad (3\text{-}27)$$

for $1 \leq i \leq M-K+1$, $i \neq i_k^*$

and $1 \leq j \leq N-L+1$, $j \neq j_k^*$.

To register n windows W_1, W_2, ..., W_n within S, moments for the n windows and (M-K+1)(N-L+1) subimages are computed only once (step 1 and 2) and step 3 is repeated n times (once for each window). The amount of computation associated with step 3 is negligible when compared to the amount of computation associated with steps 1 and 2. Therefore, although this method is computationally inefficient for n=1, its efficiency with respect to other methods increases for large n.

The total number of multiplications and additions required to implement this method depends on the number of moments used to characterize each KxL subimage. Equations for the number of additions and multiplications required to register n windows are derived for the following two cases:

a. Case 1: All moments of order two and less are used to accomplish image matching. To compute these six moments for each KxL subimage, 6(KL-1) additions and 8KL multiplications must be performed (see Table 3-1). There are n windows and (M-K+1)(N-L+1) subimages of size KxL.

Table 3-1

Computation of moments of order three and less.

Moment	Additions	Multiplications
m_{00}	(KL-1)(M-K+1)(N-L+1)	0
m_{01}	(KL-1)(M-K+1)(N-L+1)	KL(M-K+1)(N-L+1)
m_{10}	(KL-1)(M-K+1)(N-L+1)	KL(M-K+1)(N-L+1)
m_{11}	(KL-1)(M-K+1)(N-L+1)	2KL(M-K+1)(N-L+1)
m_{02}	(KL-1)(M-K+1)(N-L+1)	2KL(M-K+1)(N-L+1)
m_{20}	(KL-1)(M-K+1)(N-L+1)	2KL(M-K+1)(N-L+1)
m_{12}	(KL-1)(M-K+1)(N-L+1)	3KL(M-K+1)(N-L+1)
m_{21}	(KL-1)(M-K+1)(N-L+1)	3KL(M-K+1)(N-L+1)
m_{30}	(KL-1)(M-K+1)(N-L+1)	3KL(M-K+1)(N-L+1)
m_{03}	(KL-1)(M-K+1)(N-L+1)	3KL(M-K+1)(N-L+1)

To register each of n windows within S, 11(M-K+1)(N-L+1) additions and 6(M-K+1)(N-L+1) multiplications must be performed (step 3). Therefore,

$$\text{Total number of additions} = 6(KL-1)(M-K+1)(N-L+1) + [11(M-K+1)(N-L+1) + 6(KL-1)]n \quad (3\text{-}28)$$

$$\text{Total number of multiplications} = 8KL(M-K+1)(N-L+1) + [8KL + 6(M-K+1)(N-L+1)]n \quad (3\text{-}29)$$

For $M = 240$, $N = 256$, $K = 32$ and $L = 32$

$$\text{Total number of additions} = 288639450 + 523413n \quad (3\text{-}30)$$

$$\text{Total number of multiplications} = 385228800 + 290342n \quad (3\text{-}31)$$

b. Case 2: All moments of order three and less are used to accomplish image matching. Following the steps of case 1 it can be shown that

$$\text{Total number of additions} = 481065750 + 903705n \quad (3\text{-}32)$$

$$\text{Total number of multiplications} = 963072000 + 490730n \quad (3\text{-}33)$$

Comparison of Multiplication and Addition Requirements

The number of additions and multiplications which are required to implement correlation, SSDA and moments method for various values of n, are shown in Figures 3-2 and 3-3, respectively. From Figures 3-2 and 3-3 it is clear that if n is greater than 7, moments method with r equal two requires less computation when compared to the correlation method.

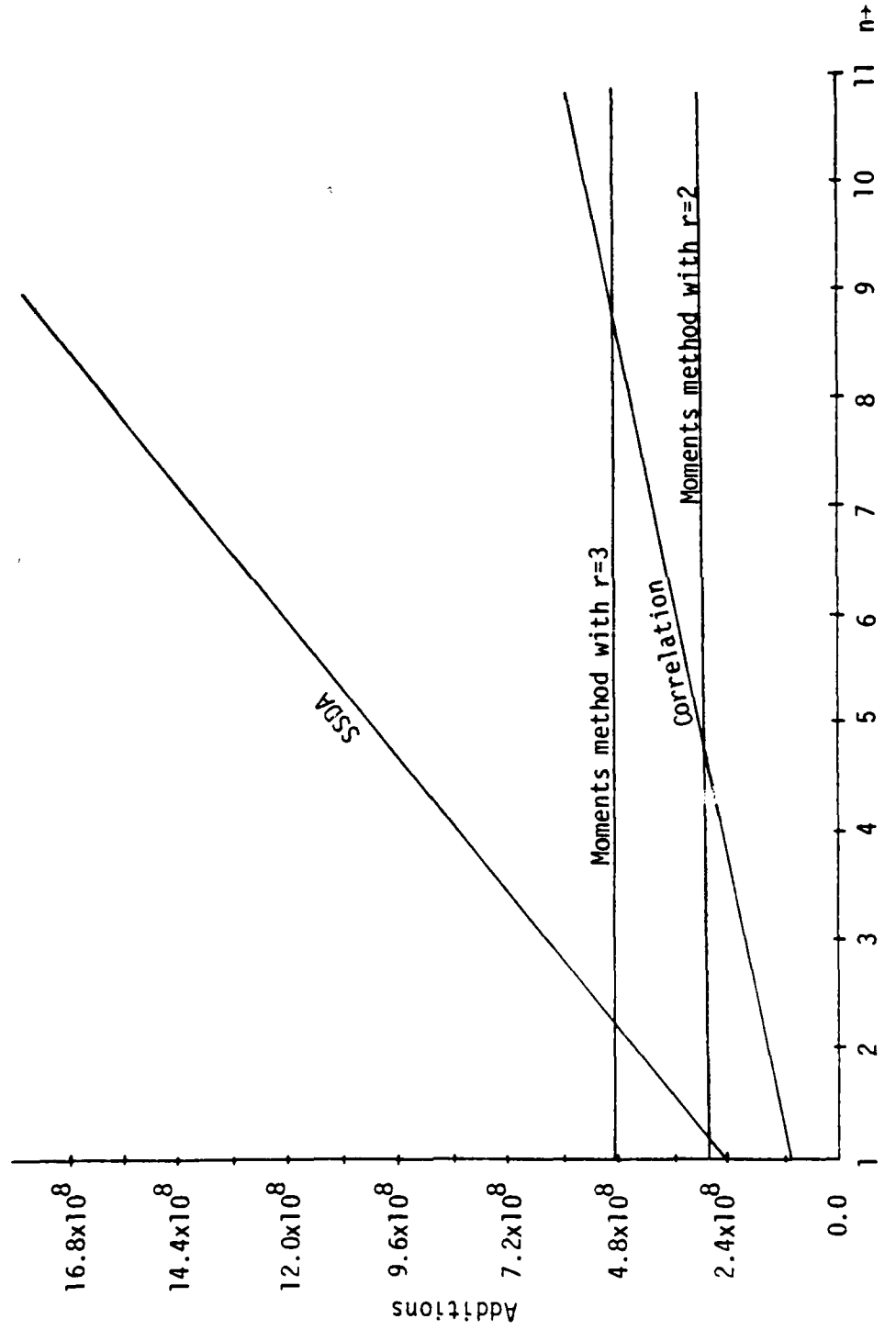


Figure 3-2. Number of additions to register n windows.

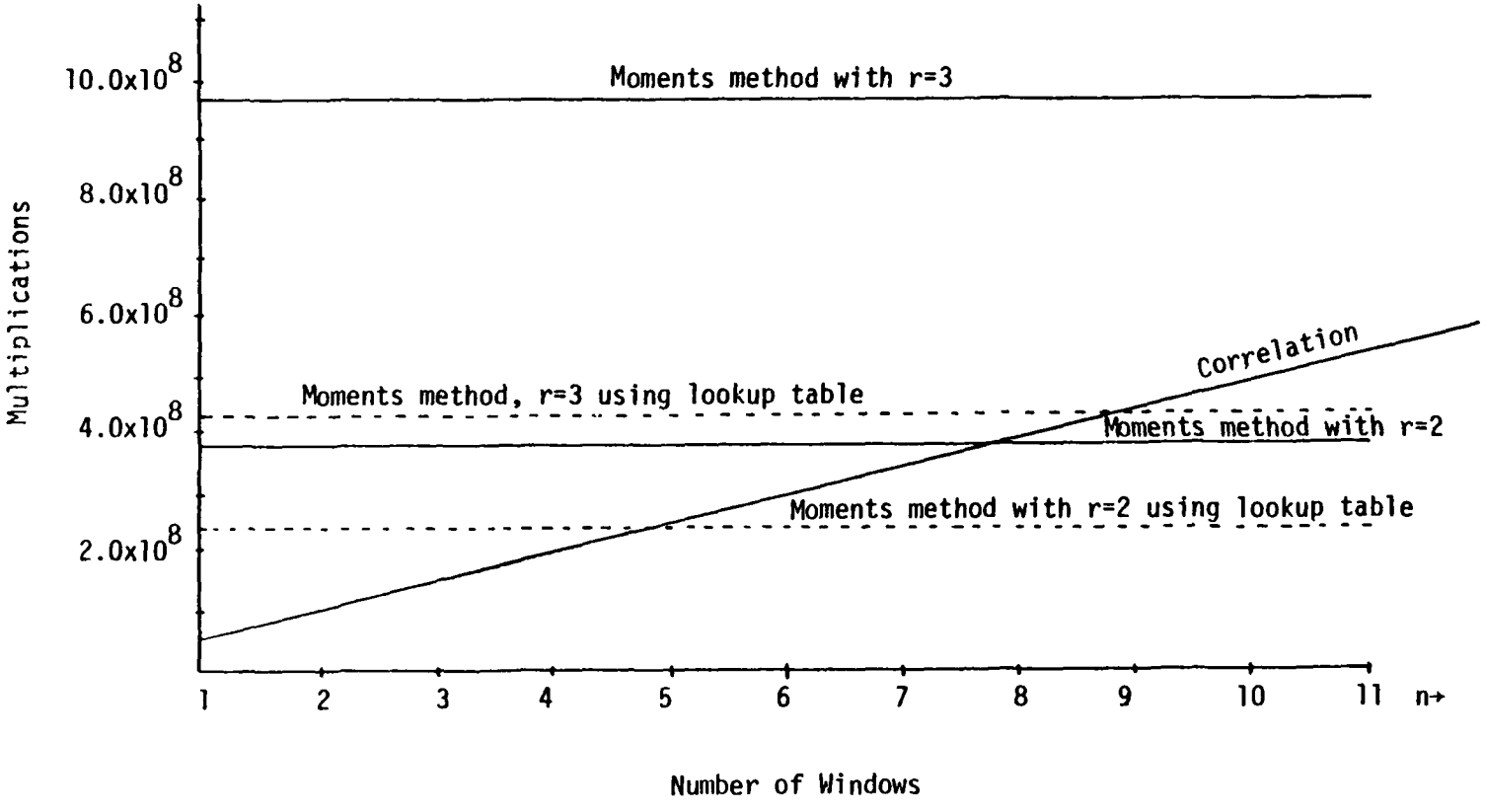


Figure 3-3. Number of multiplications to register n windows.

However, if r equals 3, n must be greater than 19 for the moments method to have a computational advantage over the correlation method. Substantial savings in computation can be accomplished for the moments method by having lookup tables for each of the moments as described below. The moment m_{pq} of a two-dimensional discrete function f(x,y) of size KxL is given by

$$m_{pq} = \sum_{\ell=1}^{K} \sum_{m=1}^{L} \ell^p m^q f(\ell,m) \tag{3-34}$$

$\ell^p m^q$ is a constant for given values of ℓ, m, p and q.

Let, $$K_{\ell m}^{pq} = \ell^p m^q \tag{3-35}$$

Now, if $K_{\ell m}^{pq}$ for ℓ = 1, 2, 3, ..., K and m = 1, 2, ..., L are precomputed and stored in memory, m_{pq} can be computed by performing, only, KL multiplications and (KL-1) additions no matter what the values of p and q are. Assuming that such look-up tables are available for all moments of order 3 and less, equations for the number of additions and multiplications are recomputed.

For r equal two;

Number of additions = (3-36)

$$6(KL-1)(M-K+1)(N-L+1)+[11(M-K+1)(N-L+1) + 6(KL-1)]n$$

Number of multiplications =

$$5KL(M-K+1)(N-L+1)+[5KL+6(M-K+1)(N-L+1)]n \tag{3-37}$$

By substituting values for M, N, K and L,

Number of additions = (3-38)

$$288639450 + 523413n$$

Number of multiplications = (3-39)

240768000 + 287270n

Similarly, if r equal three,

Number of additions = (3-40)

481065750 + 903705n

Number of multiplications = (3-41)

433382400 + 479466n

The number of additions remained the same, but the number of multiplications is reduced substantially. This fact is graphically shown, in dotted lines, in Figure 3-3. In general, an accurate comparison of computation time required by different methods is not possible. This is because the ratio of multiplication time to addition time depends on a number of factors such as machine used, bit length and algorithm used to accomplish multiplication of two numbers. However, the previous analysis shows that the moments method is computationally feasible and takes less number of arithmetic operations than the correlation method if the number of windows to be registered is sufficiently large. A direct comparison of the SSDA and moments methods is difficult because the SSDA method requires more additions while the moments method requires more multiplications. Since the ratio of the time required to perform one real multiplication to the time required to perform one real addition is not known, one single measure of comparison cannot be determined. In general,

real multiplication time = a x real addition time,

where the value of "a" depends on the machine, its bit length and algorithm used to implement multiplication. It is bounded by the bit length,

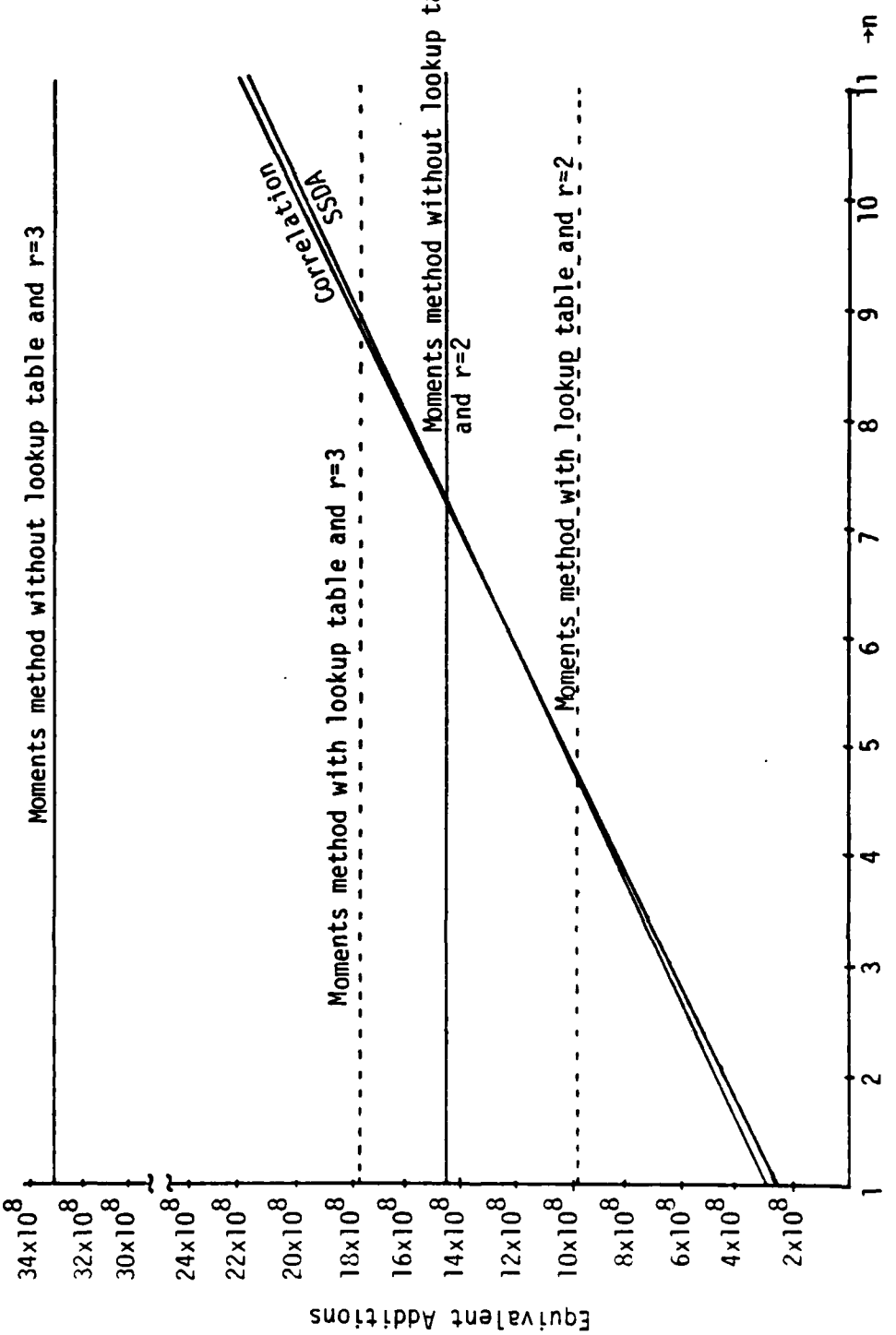


Figure 3-4. Number of equivalent additions to register n windows.

however (i.e., a $\leq$ bit length). Assuming that 'a' equals three, for a machine such as the IBM 370 the total number of equivalent real additions required to implement correlation, SSDA and moments methods (for r equal 2 and 3) for different values of n are plotted in Figure 3-4. If all moments of order two or less are used to characterize each KxL subarray, the moments method using lookup tables is computationally more efficient than the other methods, for n greater than or equal to five.

Comparison for Hardware Implementation

To accomplish multiple image registration in real time (or almost so), image matching algorithms must be implemented using fast hardware. A given algorithm can be implemented in many ways. Three simple schematics shown in Figure 3-5, 3-6, 3-7, one for each method, are used for comparison of the complexity of implementation.

Correlation Algorithm

Correlation between the window W_k and the subimage $S_{i,j}$ is computed in four stages as described below.

Stage 1: In the first stage, $S^2_{i,j}(\ell,m)$, $W^2_k(\ell,m)$ and the product $S_{i,j}(\ell,m)W_k(\ell,m)$ are computed for $\ell = 1, 2, 3, \ldots, K$ and $m = 1, 2, \ldots, L$. This requires a total of 3KL two-input multipliers (three multipliers for each pixel pair).

Stage 2: In the second stage, $\sqrt{A^k_{i,j}} = \sum_{\ell=1}^{K}\sum_{m=1}^{L} S_{i,j}(\ell,m)W_k(\ell,m)$, $B_k = \sum_{\ell=1}^{K}\sum_{m=1}^{L} W^2_k(\ell,m)$ and $C_{i,j} = \sum_{\ell=1}^{K}\sum_{m=1}^{L} S^2_{i,j}(\ell,m)$ are computed using three KL-input adders.

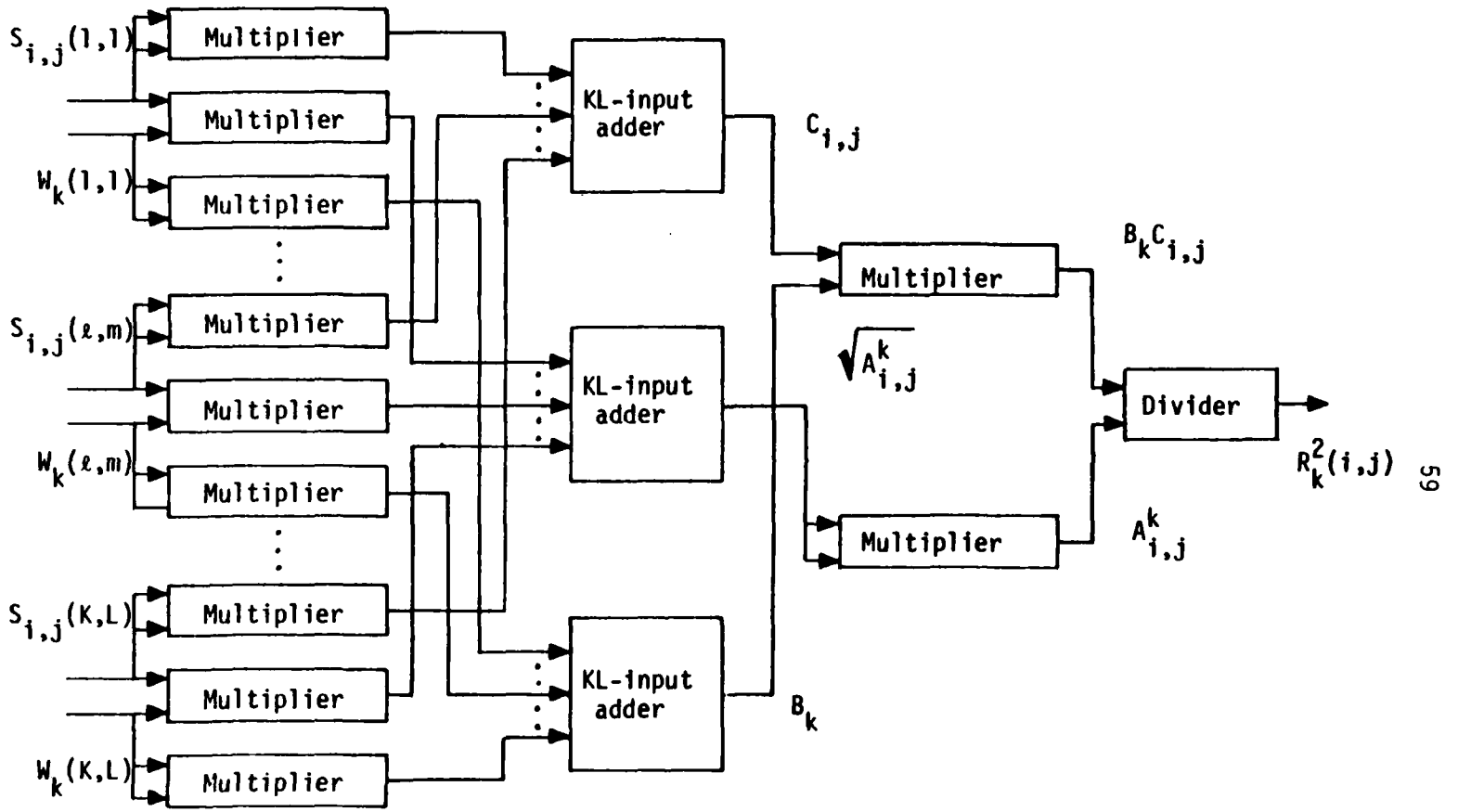


Figure 3-5. Schematic for correlation method.

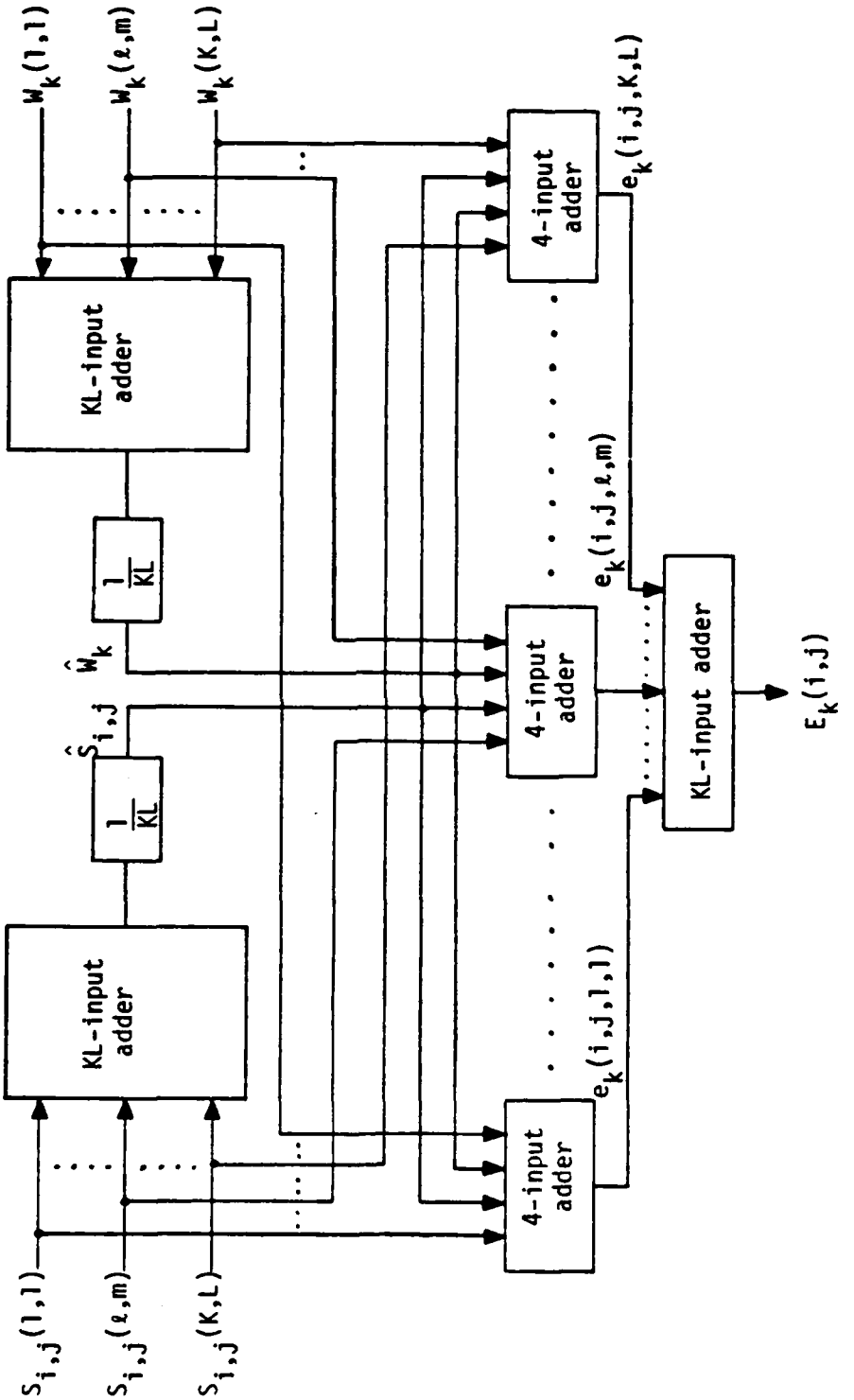


Figure 3-6. Schematic for sequential similarity detection algorithm.

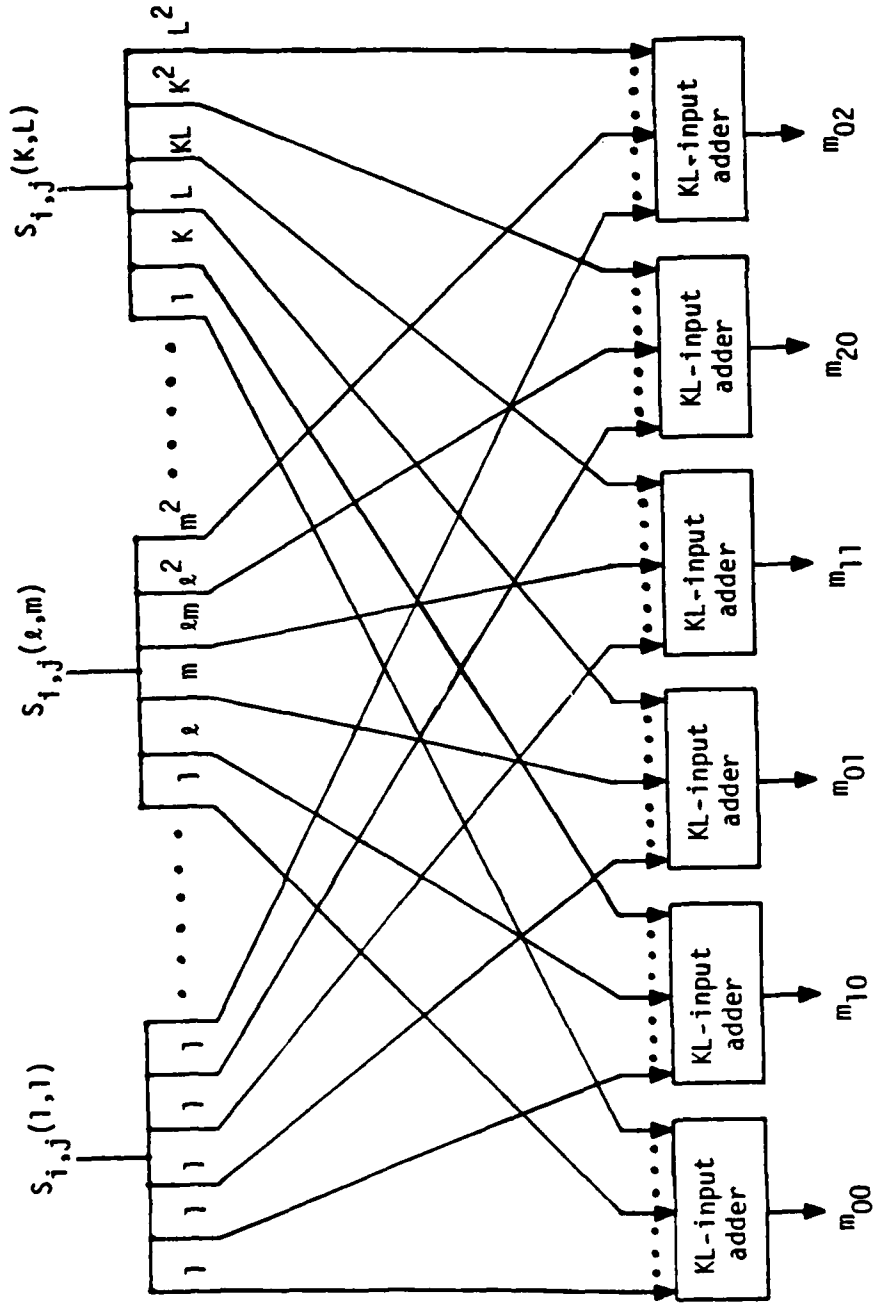


Figure 3-7. Schematic for the moments method.

Stage 3: In the third stage, two two-input multipliers compute $A^k_{i,j}$ and the product $B_k C_{i,j}$.

Stage 4: Finally, a divider computes $R^2_k(i,j)$ using the results obtained in Stage 3.

Therefore, 3KL+3 two-input multipliers and three KL-input adders are required to compute $R^2_k(i,j)$ from $S_{i,j}$ and W_k.

Sequential Similarity Detection Algorithms

The error between the window W_k and subimage $S_{i,j}$ is computed in three stages as described below.

Stage 1: In the first stage, $\hat{S}_{i,j}$ and $\hat{W}_k$ are computed using two KL-input adders and two dividers.

Stage 2: In the second stage, $e_k(i,j,\ell,m)$ for $\ell = 1, 2, \ldots, K$ and $m = 1, 2, \ldots, L$ are computed using KL four-input adders.

Stage 3: Finally, one KL-input adder computes $E_k(i,j)$ as shown in Figure 3-6.

Therefore, SSDA implementation requires three KL-input adders, KL four-input adders and two multipliers.

Moments Method

A schematic for the computation of all moments of order two and less is shown in Figure 3-7. Moments of $S_{i,j}$ can be computed in two stages.

Stage 1: In the first stage, the products $\ell S_{i,j}(\ell,m)$, $mS_{i,j}(\ell,m)$, $\ell m S_{i,j}(\ell,m)$, $\ell^2 S_{i,j}(\ell,m)$ and $m^2 S_{i,j}(\ell,m)$ are computed for $\ell = 1, 2, \ldots, K$ and $m = 1, \ldots, L$. 5KL two-input multipliers are required for the above purpose.

Stage 2: In the second stage, six KL-input adders compute m_{00}, m_{10}, m_{01}, m_{11}, m_{20}, and m_{02}.

Therefore, the moments method requires a total of 4KL-5(K+6)+6 two-input multipliers and six KL-input adders to compute all moments of order two and less.

From Figures 3-5, 3-6, and 3-7, it can be seen that all three methods can be easily implemented using multipliers and adders (i.e., none of the methods involve function evaluation). The moments method looks simpler than the other two methods with just one level of multiplications and one level of additions. In general, no method has any significant advantage over others as far as the complexity of implementation is concerned.

If the video images are sampled at 5 MHz and if it is required to register all n windows simultaneously (or almost so) between sampling, more than one correlator (probably n correlators) may have to be used in parallel when implementing the correlation or SSDA method. Therefore, if n is large, these two methods may prove uneconomical. The maximum number of parallel units that can be used is also restricted by cost and the volume of space and power available. For the moments method, however, if there is one unit of hardware to compute all the required moments, the degree of mismatch between $S_{i,j}$ and each of the n windows can be computed almost simultaneously as shown in Figure 3-8. The hardware required to compute $\sum_{pq} (m_{pq}^{k} - m_{pq}^{ij})^2$ for k = 1, 2, ..., n is negligible and therefore moment's method accomplishes multiple image registration using relatively less hardware as compared to the correlation and SSDA methods when n is large.

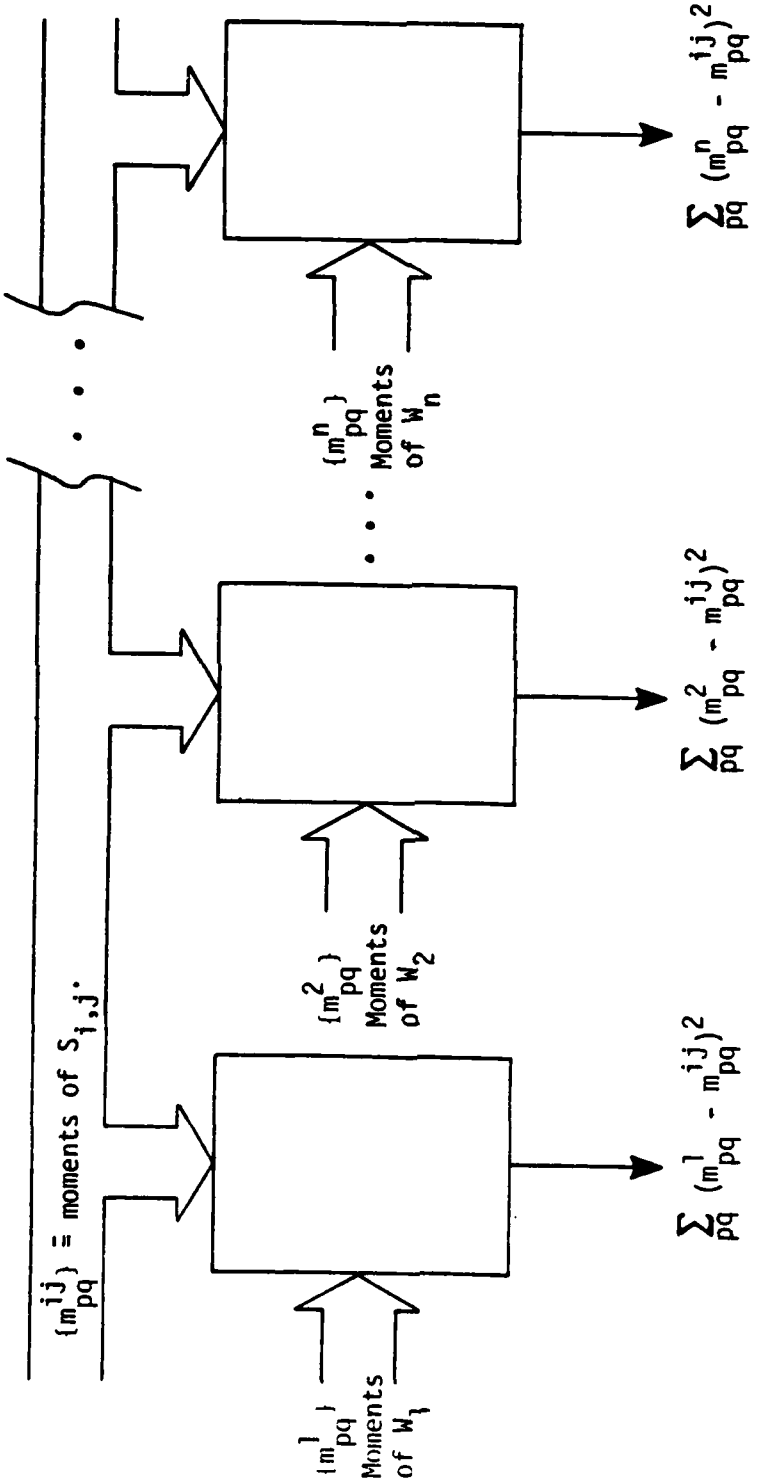


Figure 3-8. Parallel computation of dissimilarity between $S_{i,j}$ and each of the n windows.

In this chapter, computational efficiency of the correlation, the SSDA and the moments method is compared from software as well as hardware points of view, independently. It is found that the moments method becomes more efficient as the number of windows increases. It takes less computation time if implemented using software and less hardware for real time implementation if the number of windows is sufficiently large. In general, feature matching algorithms are expected to be more efficient than template matching algorithms when the number of windows is large. Modifications of the moments method and two new feature matching algorithms are presented in the next chapter.

IV. NEW METHODS FOR DIGITAL IMAGE REGISTRATION

For multiple image registration, it has been shown that the moments method is computationally more efficient than template matching algorithms if the number of windows is sufficiently large. This is because computation for template matching algorithms is directly proportional to the number of windows whereas in feature matching algorithms features are extracted for all subimages of the search area and windows only once and the matching procedure is repeated once for each window. Computation required to match the features is negligible compared to that required to compute the features.

In general, feature matching algorithms for multiple image registration are expected to be more efficient than template matching algorithms. For this reason, an effort was made to improve the moments method and to develop new feature matching algorithms. Two new feature matching algorithms, one based on intraset and interset distances [26], and the other based on correlation between adjacent pixels [27], are presented in this chapter. The computational efficiency of each of the new methods is compared with that of existing methods. A technique of making the standard correlation algorithm more suitable for multiple registration is also described.

Moments Method

In order to obtain meaningful results from any of the image registration algorithms, it is necessary to preprocess windows and subimages

of the search area such that the mean and the standard deviation of their pixel values are equal. This is called intensity level normalization and is required when the window and the search area are obtained from sensors with different d.c. gain and bias. Ideally, each KxL subarray of the search image should be preprocessed to have zero mean and unity standard deviation. This, however, would require too much additional computation. Fortunately, normalization can be incorporated into the moments method with almost no additional computation. Two cases where it is not possible to use the moment sequence $\{m_{pq}\}$ directly to accomplish image registration without intensity level normalization are given below. Means of incorporating normalization within the algorithm without actually changing the pixel values are also described.

Case 1: Let W and $S_{i*,j*}$ be the window and its matching subimage of the search area, respectively. Since the two images are obtained from different sensors, the corresponding pixels may not have the same pixel value. Assume that W and $S_{i*,j*}$ are related by the following equation.

$$W(\ell,m) = c\, S_{i*,j*}(\ell,m) \;,\; 1 \leq \ell \leq K,\; 1 \leq m \leq L \qquad (4\text{-}1)$$

where c is a constant.

The moment, m^{W}_{pq}, of the window is given by

$$m^{W}_{pq} = \sum_{\ell=1}^{K} \sum_{m=1}^{L} \ell^{p} m^{q}\, W(\ell,m) \qquad (4\text{-}2)$$

for $p,q = 0, 1, 2, \ldots$

The moment, $m^{S_{i*,j*}}_{pq}$, of the subimage $S_{i*,j*}$ is given by

$$m_{pq}^{S_{i*,j*}} = \sum_{\ell=1}^{K} \sum_{m=1}^{L} \ell^p m^q \, S_{i*,j*}(\ell,m) \qquad (4\text{-}3)$$

From Equations (4-1), (4-2) and (4-3), it is clear that

$$m_{pq}^{W} = c \, m_{pq}^{S_{i*,j*}} \quad \text{for} \quad p,q = 0, 1, 2, \ldots$$

Any attempt to match the moment sequences $\{m_{pq}^{W}\}$ and $\{m_{pq}^{S_{i*,j*}}\}$ directly leads to false registration. However,

$$\frac{m_{pq}^{W}}{m_{00}^{W}} = \frac{m_{pq}^{S_{i*,j*}}}{m_{00}^{S_{i*,j*}}} \quad \text{for} \quad p,q = 0, 1, 2, \ldots \qquad (4\text{-}4)$$

Therefore, the sequences $\{\frac{m_{pq}^{W}}{m_{00}^{W}}\}$ and $\{\frac{m_{pq}^{S_{i,j}}}{m_{00}^{S_{i,j}}}\}$ must be matched to accomplish registration.

Case 2: Assume that $S_{i*,j*}$ and W are related as described by Equation (4-5).

$$W(\ell,m) = c \, S_{i*,j*}(\ell,m) + d \;, \; 1 \leq \ell \leq K \text{ and } 1 \leq m \leq L \qquad (4\text{-}5)$$

where c and d are constants.

For this case,

$$m_{pq}^{W} \neq m_{pq}^{S_{i*,j*}} \quad \text{for,} \; p,q = 0, 1, 2, \ldots \qquad (4\text{-}6)$$

Let $\hat{W}$ and $\hat{S}_{i*,j*}$ be the mean pixel values of W and $S_{i*,j*}$, respectively. Suppose, W' is obtained from W by subtracting its mean pixel value $\hat{W}$ from each pixel. $S'_{i*,j*}$ is obtained from $S_{i*,j*}$, similarly. Now, W' and $S'_{i*,j*}$ are related by the following equation.

$$W'(\ell,m) = c \, S'_{i*,j*}(\ell,m) \qquad (4\text{-}7)$$

for $\quad 1 \leq \ell \leq K$ and $1 \leq m \leq L$

The following relations between the moments of W' and S'_{i^*,j^*} can be easily derived

$$m^{W'}_{00} = m^{S'_{i^*,j^*}}_{00} = 0 \tag{4-8}$$

and $$m^{W'}_{pq} = c\, m^{S'_{i^*,j^*}}_{pq} \quad , \text{ for } p,q = 0, 1, 2, \ldots \tag{4-9}$$

Therefore,

$$\frac{m^{W'}_{pq}}{m^{W'}_{rs}} = \frac{m^{S'_{i^*,j^*}}_{pq}}{m^{S'_{i^*,j^*}}_{rs}} \tag{4-10}$$

for $p,q = 0, 1, 2, \ldots$

provided that r and s are not simultaneously equal to zero. Therefore, for this case the sequences $\{\frac{m^{W'}_{pq}}{m^{W'}_{rs}}\}$ and $\{\frac{m^{S'_{i,j}}_{pq}}{m^{S'_{i,j}}_{rs}}\}$ can be matched to accomplish image registration. However,

$$\frac{m^{W'}_{pq}}{m^{W'}_{rs}} = \frac{\sum_{\ell=1}^{K}\sum_{m=1}^{L} \ell^p m^q \,(W(\ell,m) - \hat{W})}{\sum_{\ell=1}^{K}\sum_{m=1}^{L} \ell^r m^s \,(W(\ell,m) - \hat{W})} \tag{4-11}$$

$$= \frac{\sum_{\ell=1}^{K}\sum_{m=1}^{L} \ell^p m^q\, W(\ell,m) - \hat{W} \sum_{\ell=1}^{K}\sum_{m=1}^{L} \ell^p m^q}{\sum_{\ell=1}^{K}\sum_{m=1}^{L} \ell^r m^s\, W(\ell,m) - \hat{W} \sum_{\ell=1}^{K}\sum_{m=1}^{L} \ell^r m^s}$$

$$= \frac{m^W_{pq} - \hat{W} \sum_{\ell=1}^{K} \sum_{m=1}^{L} \ell^p m^q}{m^W_{rs} - \hat{W} \sum_{\ell=1}^{K} \sum_{m=1}^{L} \ell^r m^s}$$

Now,

$$\hat{W} = \frac{1}{KL} \sum_{\ell=1}^{K} \sum_{m=1}^{L} W(\ell,m) = \frac{m^W_{00}}{KL} \tag{4-12}$$

Let K_{pq} and K_{rs} be constants defined as

$$K_{pq} = \frac{1}{KL} \sum_{\ell=1}^{K} \sum_{m=1}^{L} \ell^p m^q \tag{4-13}$$

and $$K_{rs} = \frac{1}{KL} \sum_{\ell=1}^{K} \sum_{m=1}^{L} \ell^r m^s \tag{4-14}$$

From Equations (4-11) through (4-14)

$$\frac{m^{W'}_{pq}}{m^{W'}_{rs}} = \frac{m^W_{pq} - K_{pq} m^W_{00}}{m^W_{rs} - K_{rs} m^W_{00}} \tag{4-15}$$

K_{pq} and K_{rs} are constants which can be precomputed and stored. Therefore, it is clear from Equation (4-15) that if sequences $\{\frac{m^W_{pq} - K_{pq} m^W_{00}}{m^W_{rs} - K_{rs} m^W_{00}}\}$ and $\{\frac{m^{S_{i,j}}_{pq} - K_{pq} m^{S_{i,j}}_{00}}{m^{S_{i,j}}_{rs} - K_{rs} m^{S_{i,j}}_{00}}\}$ are used for image registration, intensity level normalization is accomplished with almost no additional computation. This makes the moments method more reliable without excessive additional computation.

Distance Measures for Digital Image Registration

Pattern classification by distance measures is one of the earliest concepts in automatic pattern recognition. The motivation for using distance functions as a classification tool follows from the fact that the most obvious way of establishing a measure of similarity between pattern vectors, which can also be considered as points in Euclidean space, is by determining their proximity. Common distance measures used for preprocessing and feature extraction, a new registration method based on intraset and interset distances, the number of arithmetic operations required to implement this method and the relation between distance measures and moments of a digital image are described in this section.

Distance Measures

Point-to-Point Distance. Let P_i and P_j be any two points in the two-dimensional x-y plane. Each point can be uniquely identified by its x and y co-ordinates, i.e.,

$$P_i = \begin{bmatrix} X_i \\ Y_i \end{bmatrix}$$

$$P_j = \begin{bmatrix} X_j \\ Y_j \end{bmatrix}$$

The square of the distance between points P_i and P_j is given by

$$D^2(P_i,P_j) = (X_i-X_j)^2 + (Y_i-Y_j)^2 \qquad (4\text{-}16)$$

Point-to-set-distance. Let P_o be a point and S_a be a set of 'a' points in the two-dimensional plane, i.e.,

$$P_o = \begin{bmatrix} X_o \\ Y_o \end{bmatrix}$$

$$S_a = \{P_1, P_2, \ldots, P_{a-1}, P_a\}$$

The distance between the point P_o and set S_a, $D^2(P_o,S_a)$, is defined as the mean square distance between the point P_o and the 'a' members of the set S_a. Therefore,

$$D^2(P_o,S_a) = \frac{1}{a} \sum_{i=1}^{a} [(X_o-X_i)^2 + (Y_o-Y_i)^2] \tag{4-17}$$

Intraset distance. The intraset distance of a set is defined as the mean square distance between the points of that set [26]. The mean square distance, $D^2(P_j,S_a)$, between a fixed point P_j and all other a-1 points of the set S_a is given by

$$D^2(P_j,S_a) = \frac{1}{a-1} \sum_{\substack{i=1 \\ i\neq j}}^{a} [(X_j-X_i)^2 + (Y_j-Y_i)^2] \tag{4-18}$$

Since $D^2(P_j,P_j)$ is zero, $D^2(P_j,S_a)$ can be written as

$$D^2(P_j,S_a) = \frac{1}{a-1} \sum_{i=1}^{a} [(X_j-X_i)^2 + (Y_j-Y_i)^2] \tag{4-19}$$

Therefore, the intraset distance of S_a, $D^2(S_a)$, is given by

$$D^2(S_a) = \frac{1}{a(a-1)} \sum_{j=1}^{a} \sum_{i=1}^{a} [(X_j-X_i)^2 + (Y_j-Y_i)^2] \tag{4-20}$$

Equation (4-20) can be easily reduced to a simple closed form in terms of statistical properties as described below:

$$D^2(S_a) = \frac{1}{a(a-1)} \sum_{j=1}^{a} \sum_{i=1}^{a} (X_j - X_i)^2 + \frac{1}{a(a-1)} \sum_{j=1}^{a} \sum_{i=1}^{a} (Y_j - Y_i)^2 \tag{4-21}$$

In Equation (4-21) $\frac{1}{a(a-1)} \sum_{j=1}^{a} \sum_{i=1}^{a} (X_j - X_i)^2$ and $\frac{1}{a(a-1)} \sum_{j=1}^{a} \sum_{i=1}^{a} (Y_j - Y_i)^2$ will be referred to as the X-component and Y-component of intraset distances, respectively.

Let

$$\bar{X} = \frac{1}{a} \sum_{i=1}^{a} X_i \text{ and } \bar{Y} = \frac{1}{a} \sum_{i=1}^{a} Y_i \tag{4-22}$$

Then

$$\sum_{j=1}^{a} \sum_{i=1}^{a} (X_j - X_i)^2 = \sum_{j=1}^{a} \sum_{i=1}^{a} (X_j - \bar{X} - X_i + \bar{X})^2$$

$$= \sum_{j=1}^{a} \sum_{i=1}^{a} [(X_j - \bar{X})^2 + (X_i - \bar{X})^2 - 2(X_j - \bar{X})(\bar{X}_i - X) \tag{4-23}$$

It can be shown that

$$\sum_{j=1}^{a} \sum_{i=1}^{a} (X_j - \bar{X})(X_i - \bar{X}) = 0 \tag{4-24}$$

Therefore,

$$\sum_{j=1}^{a} \sum_{i=1}^{a} (X_j - X_i)^2 = \sum_{j=1}^{a} \sum_{i=1}^{a} (X_j - \bar{X})^2 + \sum_{j=1}^{a} \sum_{i=1}^{a} (X_i - \bar{X})^2 \tag{4-25}$$

$$= a \sum_{j=1}^{a} (X_j - \bar{X})^2 + a \sum_{i=1}^{a} (X_i - \bar{X})^2$$

$$= a^2 \sigma_x^2 + a^2 \sigma_x^2$$

$$= 2a^2 \sigma_x^2$$

where

$$\sigma_x^2 = \frac{1}{a} \sum_{i=1}^{a} (X_i - \bar{X})^2 = \frac{1}{a} \sum_{j=1}^{a} (X_j - \bar{X})^2 \qquad (4\text{-}26)$$

Therefore, the X-component of intraset distance, $D_x^2(S_a)$, is given by

$$D_x^2(S_a) = \frac{2a}{a-1} \sigma_x^2 \qquad (4\text{-}27)$$

Similarly, the Y-component of intraset distance is given by

$$D_y^2(S_a) = \frac{2a}{a-1} \sigma_y^2 \qquad (4\text{-}28)$$

where

$$\sigma_y^2 = \frac{1}{a} \sum_{i=1}^{a} (Y_i - \bar{Y})^2 \qquad (4\text{-}29)$$

From equations (4-20), (4-27) and (4-28),

$$D^2(S_a) = \frac{2a}{a-1} (\sigma_x^2 + \sigma_y^2) \qquad (4\text{-}30)$$

Equation (4-30) expresses intraset distance in terms of variances associated with the X and Y coordinates of the set points.

Interset distance. Let S_a be a set of 'a' points and S_b be a set of 'b' points in the two-dimensional plane. Then, the interset distance between sets S_a and S_b is defined as the distance between their centroids [26]. If $(\overline{X}_a, \overline{Y}_a)$ and $(\overline{X}_b, \overline{Y}_b)$ are the centroids of set S_a and S_b, respectively, the interset distance squared between S_a and S_b is given by

$$D^2(S_a,S_b) = (\overline{X}_a-\overline{X}_b)^2 + (\overline{Y}_a-\overline{Y}_b)^2 \tag{4-31}$$

Feature Vector for Digital Image

Let F be a KxL array of digital pixels which may assume one of G gray levels.

$$1 \leq F(X,Y) \leq G \ , \ 1 \leq X \leq K, \ 1 \leq Y \leq L \tag{4-32}$$

Let S^g denote the set of pixels with pixel value equal to g and n_g be the number of elements in S^g. Now, digital image F can be considered as the union of sets $S^1, S^2, \ldots, S^G$.

$$F = S^1 \cup S^2 \cup \ldots \cup S^G \tag{4-33}$$

Each element of the set S^g, which is just a point in the two-dimensional X-Y plane, is uniquely specified by its coordinates.

$$S^g = \{p_1^g, p_2^g, \ldots, p_{n_g}^g\} \tag{4-34}$$

where

$$p_i^g = \begin{bmatrix} x_i^g \\ y_i^g \end{bmatrix}, \text{ for } i = 1, 2, \ldots, n_g \tag{4-35}$$

and $g = 1, 2, \ldots, G$

Each of the G sets can be characterized by its intraset distance which is defined as the mean square distance between its elements. From Equation 4-3 the intraset distance of the set S^g is given by

$$D^2(S^g) = D_x^2(S^g) + D_y^2(S^g) \tag{4-36}$$

$$= \frac{2n_g}{n_g-1} (\sigma_{x^g}^2 + \sigma_{y^g}^2)$$

where

$$\sigma_{x^g}^2 = \frac{1}{n_g} \sum_{i=1}^{n_g} (X_i^g - \bar{X}^g)^2 \tag{4-37}$$

and

$$\sigma_{y^g}^2 = \frac{1}{n_g} \sum_{i=1}^{n_g} (Y_i^g - \bar{Y}^g)^2 \tag{4-38}$$

The location of one set with respect to the other can be characterized by the interset distance between them which is defined as the distance between their centroids. Since there are G sets in F, there exist $\frac{G(G-1)}{2}$ distinct interset and G intraset distances. A $\frac{G(G+1)}{2}$ -dimensional feature vector consisting of G intraset and $\frac{G(G-1)}{2}$ interset distances maps the digital image F to a point in the $\frac{G(G+1)}{2}$ -dimensional Euclidean space.

Derivation of Multiplication and Addition Requirements

Consider the digital image F of size KxL described by Equations (4-32) and (4-33). Intraset distance of the set S^g is given by Equation (4-36). The X-component of the intraset distance, $D_x^2(S^g)$, can be written as

$$D_x^2(S^g) = \frac{2\,n_g}{n_g-1}\,\sigma_x{}^{g\,2} \tag{4-39}$$

$$= \frac{2\,n_g}{n_g-1}\,\frac{\sum_{i=1}^{n_g}(x_i^g - \bar{x}^g)^2}{n_g}$$

$$= \frac{2}{n_g-1}\sum_{i=1}^{n_g}(x_i^g - \bar{x}^g)^2$$

$$= \frac{2}{n_g-1}\left[\sum_{i=1}^{n_g}(x_i^g)^2 - n_g(\bar{x}^g)^2\right]$$

where

$$\bar{x}^g = \frac{1}{n_g}\sum_{i=1}^{n_g} x_i^g \tag{4-40}$$

The multiplication and addition requirements to compute the feature vector are derived in the steps below. It is assumed that multiplications, divisions and squaring operations are equivalent.

1. From Equation (4-40), it is clear that n_g-1 additions and one multiplication (division) are required to compute $\bar{x}^g$.
2. Computation of $D_x^2(S^g)$ as described by Equation (4-39) requires n_g+4 multiplications and n_g+1 additions. Therefore, a total of $2n_g$ additions and (n_g+5) multiplications are needed for the computation of $D_x^2(S^g)$.
3. Similarly, $2n_g$ additions and (n_g+5) multiplications are required for the computation of $D_y^2(S^g)$.
4. Adding results of steps 2 and 3, $4n_g$ additions and $2n_g$ + 10) multiplications must be performed to compute $D^2(S^g)$.

5. Since there are G sets in the image F,

$$\text{Total number of additions} = \sum_{g=1}^{G} (4n_g) = 4KL \qquad (4\text{-}41)$$

$$\text{Total number of multiplications} = \sum_{g=1}^{G} (2n_g+10) = 2KL+10G \qquad (4\text{-}42)$$

6. The interset distance between two sets is the distance between their centroids (See equation (4-31)). Two multiplications and three additions are required to compute each of the $\frac{G(G-1)}{2}$ interset distances.

$$\text{Total number of additions} = \frac{3G(G-1)}{2} \qquad (4\text{-}43)$$

$$\text{Total number of multiplications} = G(G-1) \qquad (4\text{-}44)$$

Therefore, in order to compute the feature vector for a KxL array of pixels which can assume one of G gray levels, a total of $4KL+\frac{3G(G-1)}{2}$ additions and $2KL+G(G+9)$ multiplications must be performed.

Multiple Image Registration

A method of accomplishing multiple image registration using feature vectors of windows and subimages of S is described below.

Step 1: Compute the feature vector for each of the n windows and each of $(M-K+1)(N-L+1)$ subimages of S. Let, V^k and $V^{i,j}$ denote feature vectors of the window W_k and subimage $S_{i,j}$, respectively.

$$V^k = \begin{bmatrix} V_1^k \\ V_2^k \\ . \\ . \\ . \\ V^k_{\frac{G(G+1)}{2}} \end{bmatrix} \quad \text{for } k = 1, 2, \ldots, n \tag{4-45}$$

$$V^{i,j} = \begin{bmatrix} V_1^{i,j} \\ V_2^{i,j} \\ . \\ . \\ . \\ V^{i,j}_{\frac{G(G+1)}{2}} \end{bmatrix} \quad \begin{array}{l} \text{for, } 1 \leq i \leq M-K+1 \\ \text{and, } 1 \leq j \leq N-L+1 \end{array} \tag{4-46}$$

The above vectors can be considered as points in $\frac{G(G+1)}{2}$ -dimensional Euclidean space.

Step 2: To register W_k within S, find the subimage $S_{i_k^*,j_k^*}$ whose feature vector best matches that of W_k. In other words, find $S_{i_k^*,j_k^*}$ such that

$$||V^k - V^{i_k^*,j_k^*}||^2 < ||V^k - V^{i,j}||^2 \tag{4-47}$$

for $\quad 1 \leq i \leq M-K+1, \; i \neq i_k^*$

and $\quad 1 \leq j \leq N-L+1, \; j \neq j_k^*$

where, $||V_k - V^{i,j}||^2$ is the square of the distance between vectors V_k and $V^{i,j}$.

To register n windows W_1, W_2, ..., W_n within S, the feature vectors for n windows and (M-K+1)(N-L+1) subimages are computed only once (step 1) and step 2 is repeated n times, once for each window. The computation associated with step 2 is small when compared to the computation associated with step 1. Therefore, although this method is computationally inefficient for n=1, its efficiency with respect to the correlation method increases for large n. Equations for the number of additions and multiplications required to register n windows are derived below.

In step 1 (M-K+1)(N-L+1)+n feature vectors are computed. $4KL+\frac{3G(G-1)}{2}$ additions and 2KL+G(G+9) multiplications are required to compute one feature vector. Therefore,

$$\left.\begin{array}{l}\text{Number of additions}\\ \text{in step 1}\end{array}\right\} = [4KL + \frac{3G(G-1)}{2}][(M-K+1)(N-L+1)+n] \quad (4\text{-}48)$$

$$\left.\begin{array}{l}\text{Number of multipli-}\\ \text{cations in step 1}\end{array}\right\} = [2KL + G(G+9)][(M-K+1)(N-L+1)+n] \quad (4\text{-}49)$$

In step 2, G^2+G-1 additions and $\frac{G(G+1)}{2}$ multiplications are performed to compute $||V^k - V^{i,j}||^2$. There are (M-K+1)(N-L+1) reference points and n windows. Therefore,

$$\left.\begin{array}{l}\text{Number of additions}\\ \text{in step 2}\end{array}\right\} = (G^2+G-1)(M-K+1)(N-L+1)n \quad (4\text{-}50)$$

$$\left.\begin{array}{l}\text{Number of multipli-}\\ \text{cations in step 2}\end{array}\right\} = \frac{G(G+1)}{2}(M-K+1)(N-L+1)n \quad (4\text{-}51)$$

From Equations (4-48) through (4-51),

$$\text{Total number of additions} = \quad (4\text{-}52)$$

$$[4KL+\frac{3G(G-1)}{2}][(M-K+1)(N-L+1)+n] + (G^2+G-1)(M-K+1)(N-L+1)n$$

Total number of multiplications = (4-53)

$$[2KL+G(G+9)][(M-K+1)(N-L+1)+n] + \frac{G(G+1)}{2}(M-K+1)(N-L+1)n$$

For M = 240, N = 256 and K = L = 32,

Total number of additions = (4-54)

$$[47025\ (G^2+G-1) + 1.5\ G(G-1) + 4096]n + 70537.5\ G(G-1) + 192614400$$

Total number of multiplications = (4-55)

$$[2048 + G(G+9) + 23512.5(G+1)G]n + 47025\ G(G+9) + 96307200$$

The number of additions and multiplications which are required to implement distance and other methods for various values of n and G, are shown in Figures 4-1 and 4-2, respectively. From Figures 4-1 and 4-2 it is clear that if n is greater than 4, the distance method requires less computation when compared to the correlation method. In order to have one single measure of comparison, the time ratio relating multiplication and addition operations can be used. For machines like the IBM 370, multiplication time is three times the addition time. With the above assumption, the number of equivalent additions required to implement correlation and distance methods for various values of n and G is shown in Figure 4-3. Figures 4-1, 4-2, and 4-3 prove that the distance method is computationally more efficient than the correlation method if the number of windows is large.

Relation between Distance Measures and Moments

Let F be a KxL array of digital pixels which may assume one of G possible levels on the gray scale.

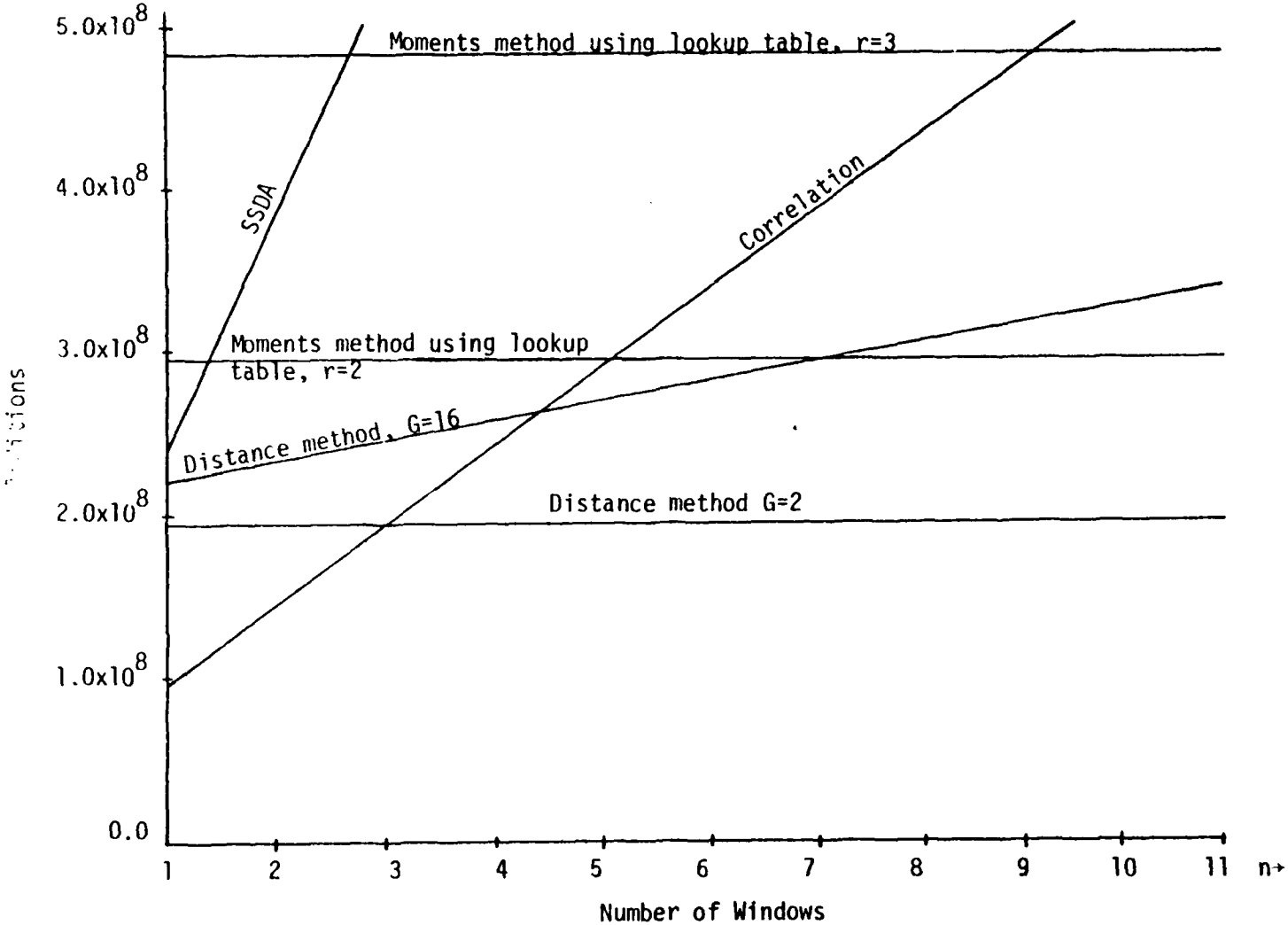


Figure 4-1. Number of additions to register n windows.

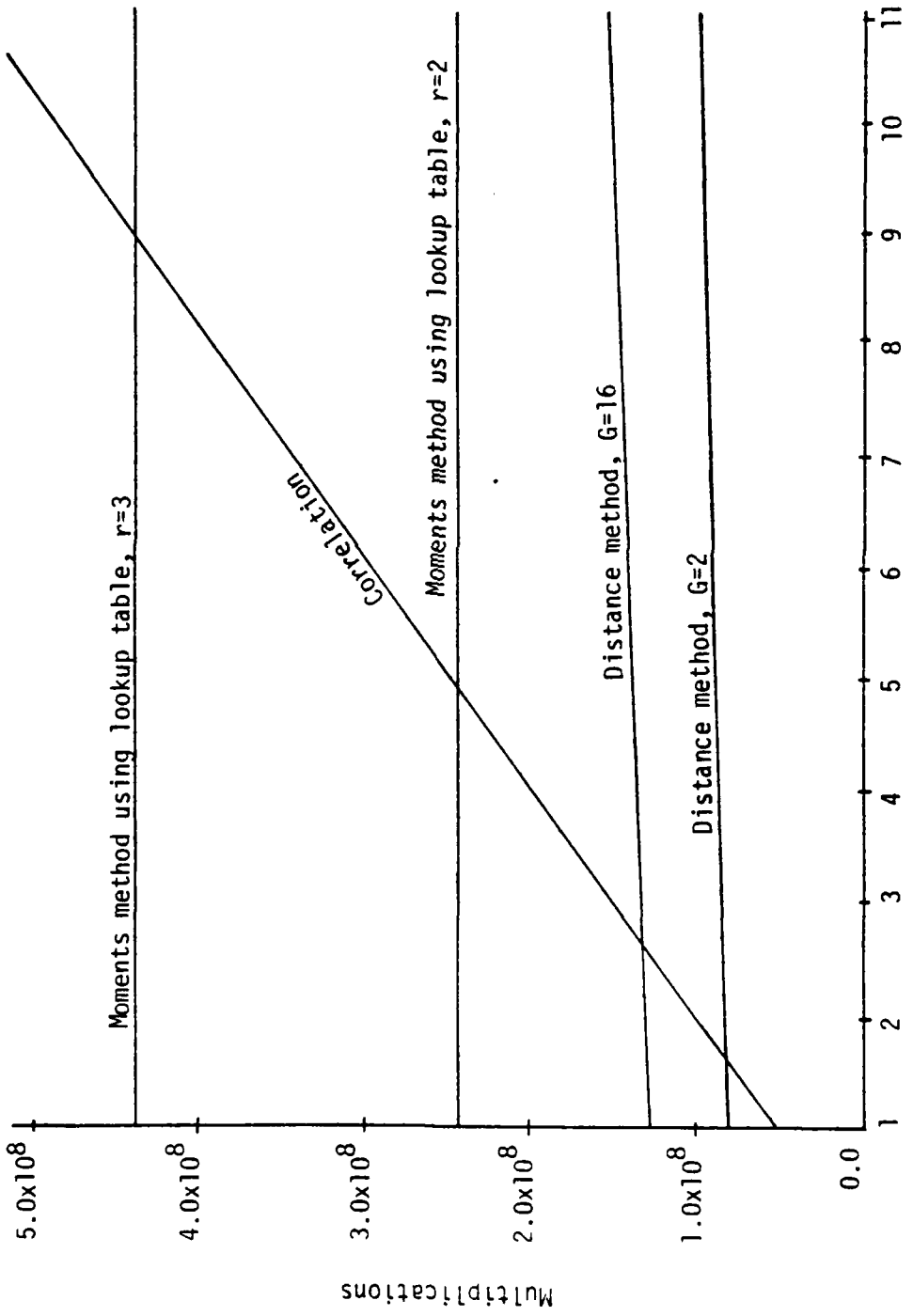


Figure 4-2. Number of multiplications to register n windows.

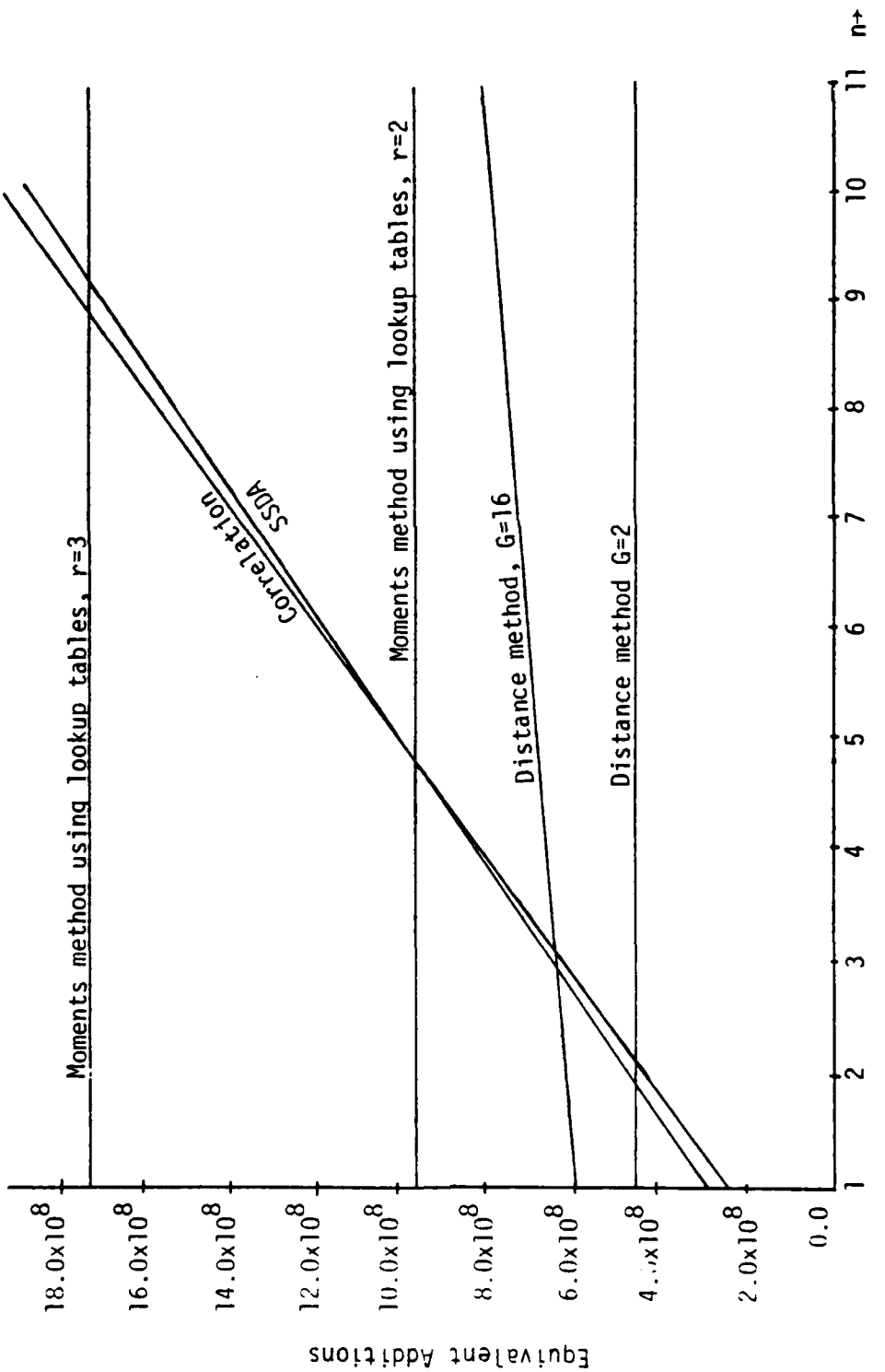


Figure 4-3. Number of equivalent additions to register n windows.

$$1 \leq F(X,Y) \leq G \tag{4-56}$$

for $1 \leq X \leq K$, $1 \leq Y \leq L$

Let S^g denote the set of pixels with pixel value equal to g and n_g be the number of elements in S^g. Now, digital image F can be considered as the union of sets S^1, S^2, ..., S^G.

$$F = S^1 \cup S^2 \cup \ldots \cup S^G \tag{4-57}$$

Each element, which is just a point in the two dimensional x-y plane of the set S^g, is uniquely specified by its coordinates.

$$S^g = \{P_1^g, P_2^g, P_3^g, \ldots P_{n_g}^g\}$$

where

$$P_i^g = \begin{bmatrix} X_i^g \\ Y_i^g \end{bmatrix} \text{ , for } i = 1, 2, \ldots, n_g \tag{4-58}$$

and $g = 1, 2, \ldots, G$

The set S^g can be considered as a two dimensional discrete function whose value is equal to g at P_1^g, P_2^g, ..., $P_{n_q}^g$ and zero elsewhere in the x-y plane. Therefore, the central moments of S^g are given by

$$\mu_{20}^g = \sum_{i=1}^{n_g} g(X_i^g - \overline{X}^g)^2 \tag{4-59}$$

$$\mu_{02}^g = \sum_{i=1}^{n_g} g(Y_i^g - \overline{Y}^g)^2 \tag{4-60}$$

$$\mu_{00}^{g} = \sum_{i=1}^{n_g} g \quad = n_g \cdot g \tag{4-61}$$

From Equations (4-59), (4-60) and (4-61),

$$\frac{\mu_{20}^{g}+\mu_{02}^{g}}{\mu_{00}^{g}} = \frac{1}{n_g} \sum_{i=1}^{n_g} [(x_i^g-\bar{x}^g)^2 + (y_i^g-\bar{y}^g)^2] \tag{4-62}$$

The intraset distance of the set S^g is given by Equations (4-36) through (4-38).

$$D^2(S^g) = \frac{2}{n_g-1} \left(\sum_{i=1}^{n_g} [(x_i^g-\bar{x}^g)^2 + (y_i^g-\bar{y}^g)^2]\right) \tag{4-63}$$

Therefore,

$$D^2(S^g) = \frac{2n_g}{n_g-1} \left(\frac{\mu_{20}^{g}+\mu_{02}^{g}}{\mu_{00}^{g}}\right) \tag{4-64}$$

Equation (4-64) shows the relation between the intraset distance and moments of the set S^g. Since $(\mu_{20}^{g}+\mu_{02}^{g})$ is invariant under translation and rotation [10], the intraset distance is also invariant under translation and rotation. If each point P_i^g is considered as a point mass of value g, $\frac{\mu_{20}^{g}+\mu_{02}^{g}}{\mu_{00}^{g}}$ is the square of the radius of gyration of the mass distribution about its centroid. Therefore,

$$D^2(S^g) = \left(\frac{2n_g}{n_g-1}\right) * r^2 \tag{4-65}$$

where, r is the radius of gyration of S^g about its centroid.

The interset distance between the sets S^g and S^h is defined as the distance between their centroids.

$$D^2(S^g,S^h) = (\bar{X}^g-\bar{X}^h)^2 + (\bar{Y}^g-\bar{Y}^h)^2 \tag{4-66}$$

$$= \left(\frac{m^g_{10}}{m^g_{00}} - \frac{m^h_{10}}{m^h_{00}}\right)^2 + \left(\frac{m^g_{01}}{m^g_{00}} - \frac{m^h_{01}}{m^h_{00}}\right)^2$$

Equation (4-66) shows the relation between the interset distance between the sets S^g and S^h, and their moments. Simulation results given in Reference [16] show that matching homogeneous regions separately and combining the partial results additively yields sharper correlation peaks. If the image is composed of homogeneous regions as described in Reference [16], all pixels of the homogeneous region normally fall into the same set when the image is segmented based on pixel values. Computing intraset distance, in some sense, is the same as processing each homogeneous region separately. The relative location of homogeneous regions with respect to each other is determined by interset distances. Therefore, the distance method is expected to perform better than the moments method if the scene is composed of homogeneous regions.

Correlation of Adjacent Pixels for Image Registration

Let W be a digital image of size KxL whose pixels can assume one of G possible levels on the gray scale.

$$0 \leq W(\ell,m) \leq G-1 \tag{4-67}$$

for $\quad 1 \leq \ell \leq K$ and $1 \leq m \leq L$

The normalized correlation between adjacent pixels of the ℓ^{th} row of the digital image W is given by the ensemble average

$$\rho_\ell = \frac{E\{W(\ell,m)W(\ell,m+1)\}}{E\{W^2(\ell,m)\}} \tag{4-68}$$

Since there are L elements in any row of W, ρ_ℓ can be approximated by the spatial average

$$\rho_\ell = \frac{\frac{1}{L-1}\sum_{m=1}^{L-1} W(\ell,m)W(\ell,m+1)}{\frac{1}{L}\sum_{m=1}^{L} W^2(\ell,m)} \tag{4-69}$$

where ρ_ℓ always lies between zero and one. When all pixels in the ℓ^{th} row have the same value, ρ_ℓ is one. The value of ρ_ℓ, in some sense, is related to the difference in pixel values of the adjacent pixels. Values of adjacent pixels are highly correlated for most images except at edge pixels. If pixel $W(\ell,m)$ is of a certain gray level, then the adjacent pixel $W(\ell,m+1)$ along the scan line ℓ, is likely to have a similar value. This property of ρ_ℓ has been used in image coding and transmission in the past [27, pp. 278-281]. A new method of accomplishing image registration using normalized correlation of adjacent pixels of rows and columns of digital images is presented next.

Feature Vector for a Digital Image

Since in Equation (4-69) L/(L-1) is a constant and can be dropped without losing any information, the normalized correlation between adjacent pixels of the ℓ^{th} row is given by

$$\rho_{\ell} = \frac{\sum_{m=1}^{L-1} W(\ell,m)W(\ell,m+1)}{\sum_{m=1}^{L} W^2(\ell,m)} \qquad (4\text{-}70)$$

for $\ell = 1, 2, \ldots, K$

Let σ_m be the normalized correlation between the adjacent pixels of the m^{th} column.

$$\sigma_m = \frac{\sum_{\ell=1}^{K-1} W(\ell,m)W(\ell+1,m)}{\sum_{\ell=1}^{K} W^2(\ell,m)} \qquad (4\text{-}71)$$

for $m = 1, 2, \ldots, L$

The (K+L)-dimensional feature vector, V_w, of the digital image W is given by

$$V_w^T = [\rho_1 \ \rho_2 \cdots \rho_K \ \sigma_1 \ \sigma_2 \cdots \sigma_L] \qquad (4\text{-}72)$$

Multiple Image Registration

A method of accomplishing multiple image registration using feature vectors of windows and subimages of the search area S is described below.

Step 1: Compute the feature vector for each of the n windows and each of (M-K+1)(N-L+1) subimages of S. Let V^k and $V^{i,j}$ denote feature vectors of the window W_k and subimage $S_{i,j}$, respectively.

Step 2: To register W_k within S, find the subimage $S_{i_k^*,j_k^*}$ whose feature vector best matches that of W_k. In other words, find $S_{i_k^*,j_k^*}$ such that

$$||V^k - V^{i_k^*, j_k^*}||^2 < ||V^k - V^{i,j}||^2 \qquad (4\text{-}73)$$

for $1 \leq i \leq M-K+1$, $i \neq i_k^*$, and $1 \leq j \leq N-L+1$, $j \neq j_k^*$

where $||V^k - V^{i,j}||^2$ is the square of the distance between the vectors V^k and $V^{i,j}$.

To register windows W_1, W_2, ..., W_n within S, the feature vector for n windows and (M-K+1)(N-L+1) subimages are computed only once (Step 1) and Step 2 is repeated n times, once for each window. Since the computation associated with Step 2 is small compared to the computation associated with Step 1, this method is more efficient for multiple image registration than the correlation and SSDA methods.

Derivation of Multiplication and Addition Requirements

Equations for the number of multiplications and additions in terms of M, N, K, L and n are derived in the seven steps below.

1. From Equation (4-70), it is clear that 2L multiplications and 2L-3 additions must be performed to compute ρ_ℓ. Similarly, 2K multiplications and 2K-3 additions are needed to compute σ_m. Since there are K rows and L columns, a total of 4KL multiplications and (4KL-3K-3L) additions are required to compute the feature vector for a KxL array.
2. Let $\rho_\ell(i,j)$ and $\sigma_m(i,j)$ denote the normalized correlation between adjacent pixels of the ℓ^{th} row and the m^{th} column of the subimage $S_{i,j}$, respectively. In

order to compute the feature vector for the subimage $S_{1,1}$, 4KL multiplications and 4KL-3K-3L additions are required as outlined in Step 1.

3. Once $V^{1,1}$ is computed, $V^{1,2}$ can be computed with very few arithmetic operations as shown below.

$$\rho_\ell(1,1) = \frac{\sum_{m=1}^{L-1} S_{1,1}(\ell,m)S_{1,1}(\ell,m+1)}{\sum_{m=1}^{L} S_{1,1}^2(\ell,m)} \tag{4-74}$$

$$= \frac{A_\ell(1,1)}{B_\ell(1,1)}$$

where

$$A_\ell(1,1) = \sum_{m=1}^{L-1} S_{1,1}(\ell,m)S_{1,1}(\ell,m+1) \tag{4-75}$$

$$B_\ell(1,1) = \sum_{m=1}^{L} S_{1,1}^2(\ell,m) \tag{4-76}$$

Now,

$$\rho_\ell(1,2) = \frac{\sum_{m=1}^{L-1} S_{1,2}(\ell,m)S_{1,2}(\ell,m+1)}{\sum_{m=1}^{L} S_{1,2}^2(\ell,m)} \tag{4-77}$$

$$= \frac{A_\ell(1,1)-S_{1,1}(\ell,1)S_{1,1}(\ell,2) + S_{1,2}(\ell,L-1)S_{1,2}(\ell,L)}{B_\ell(1,1)-S_{1,1}^2(\ell,1) + S_{1,2}^2(\ell,L)}$$

From Equation (4-77), $\rho_\ell(1,2)$ can be computed from $\rho_\ell(1,1)$ by performing only 5 multiplications and 4 additions. In general, $\rho_\ell(i,j+1)$ can be computed from $\rho_\ell(i,j)$ by performing 5 multiplications and 4 additions.

Since there are K rows, computation of $\rho_{\ell}(i,j+1)$ from $\rho_{\ell}(i,j)$ for $\ell = 1, 2, \ldots, K$ requires a total of 5K multiplications and 4K additions. Also,

$$\sigma_m(i,j+1) = \sigma_{m+1}(i,j) \text{ for } m = 1, 2, \ldots, L-1 \qquad (4\text{-}78)$$

Therefore, it is necessary to compute $\sigma_L(i,j+1)$ which needs 2K multiplications and 2K-3 additions. In other words, computation of the feature vector $V^{i,j+1}$ from $V^{i,j}$ requires 7K multiplications and 6K-3 additions. However, in order to compute $V^{i,j+1}$ from $V^{i,j}$, $V^{i,j}$ and the first column of $S_{i,j}$ must be stored. Since $V^{i,j}$ is a (K+L)-dimensional vector and $S_{i,j}$ is a KxL array, 2K+L additional memory locations are required.

4. Similarly, computation of $V^{i+1,j}$ from $V^{i,j}$ requires 7L multiplications and 6L-3 additions and K+2L additional memory locations.

5. There are (M-K+1)(N-L+1) allowable reference points in the search area. The feature vector associated with the first reference point (1,1) is computed as described in Step 2. This requires 4KL multiplications and 4KL-3K-3L additions. $V^{2,1}, \ldots, V^{M-K+1,1}$ are computed using the procedure similar to the one in Step 3. This requires (M-K)(6L-3) additions and 7L(M-K) multiplications. For line i, the feature vectors $V^{i,2}, V^{i,3}, \ldots, V^{i,N-L+1}$ can be computed using the procedure outlined in Step 3 with 7K(N-L)

multiplications and (6K-3)(N-L) additions, and there are (M-K+1) such lines. Therefore, to compute (M-K+1)(N-L+1) feature vectors:

$$\text{Number of multiplications} = 4KL + 7(M-K)[K(N-L)+L] \quad (4\text{-}79)$$

$$\text{Number of additions} = 4KL - 3K - 3L + (M-K)(N-L)(6K-3) + (M-K)(6L-3) \quad (4\text{-}80)$$

6. The computation of n feature vectors for n windows needs 4KLn multiplications and (4KL-3K-3L)n additions.
7. (2K+2L-1) additions and (K+L) multiplications must be performed to compute $||V^k - V^{i,j}||^2$. Since there are (M-K+1)(N-L+1) reference points and n windows,

$$\text{Number of multiplications} = (K+L)(M-K+1)(N-L+1)n \quad (4\text{-}81)$$

$$\text{Number of additions} = (2K+2L-1)(M-K+1)(N-L+1)n \quad (4\text{-}82)$$

From Step 5 through Step 7,

$$\text{Total number of multiplications} = \{4KL+(M-K+1)(N-L+1)(K+L)\}n + 4KL+7(M-K)[K(N-L)+L] \quad (4\text{-}83)$$

$$\text{Total number of additions} = \{(2K+2L-1)(M-K+1)(N-L+1)+(4KL-3K-3L)\}n + (M-K)(N-L)(6K-3)+(M-K)(6L-3)+4KL-3K-3L \quad (4\text{-}84)$$

For M = 240, N = 256, K = 32, and L = 32,

$$\text{Total number of multiplications} = 3013696n + 10487296 \quad (4\text{-}85)$$

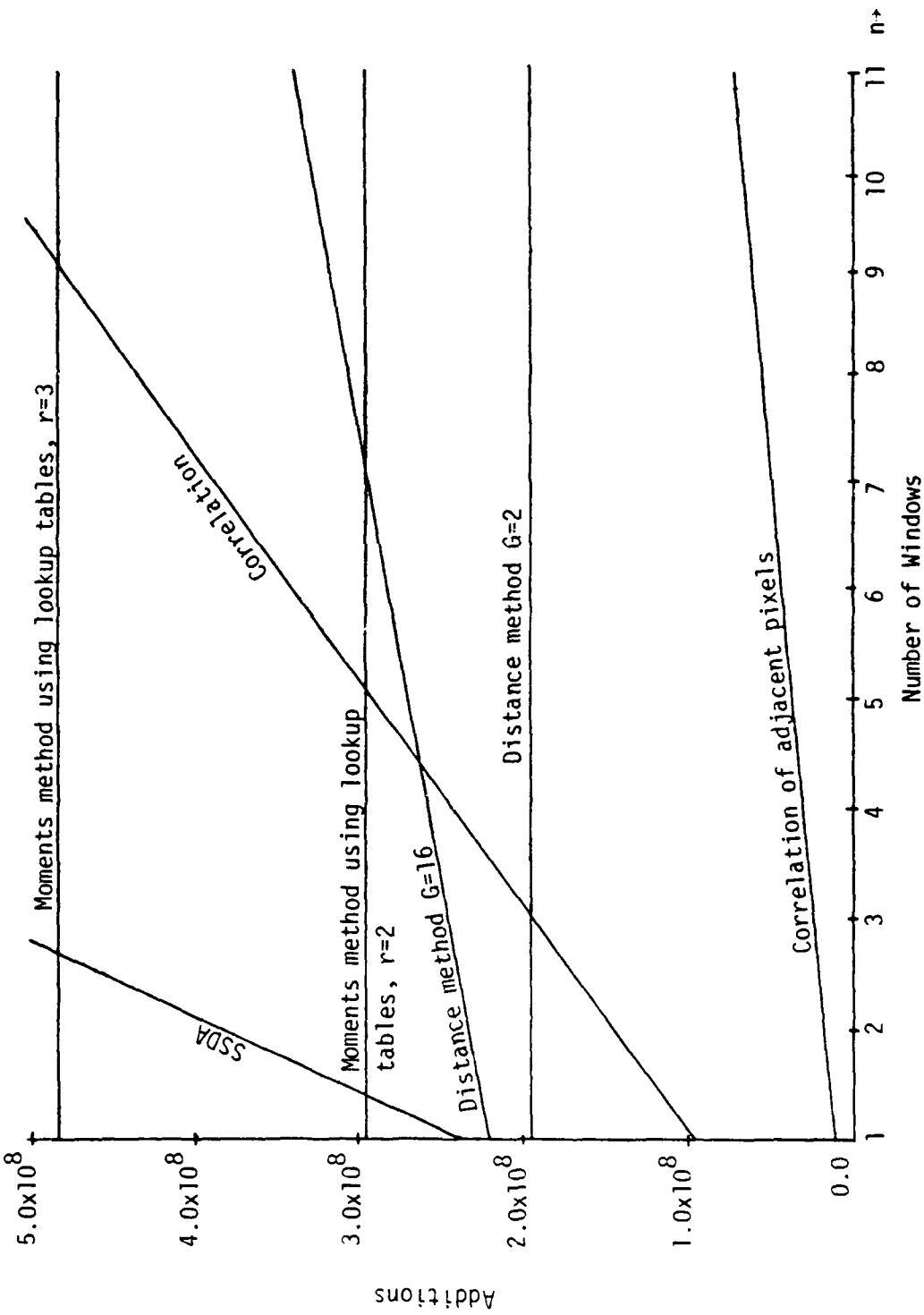


Figure 4-4. Number of additions to register n windows.

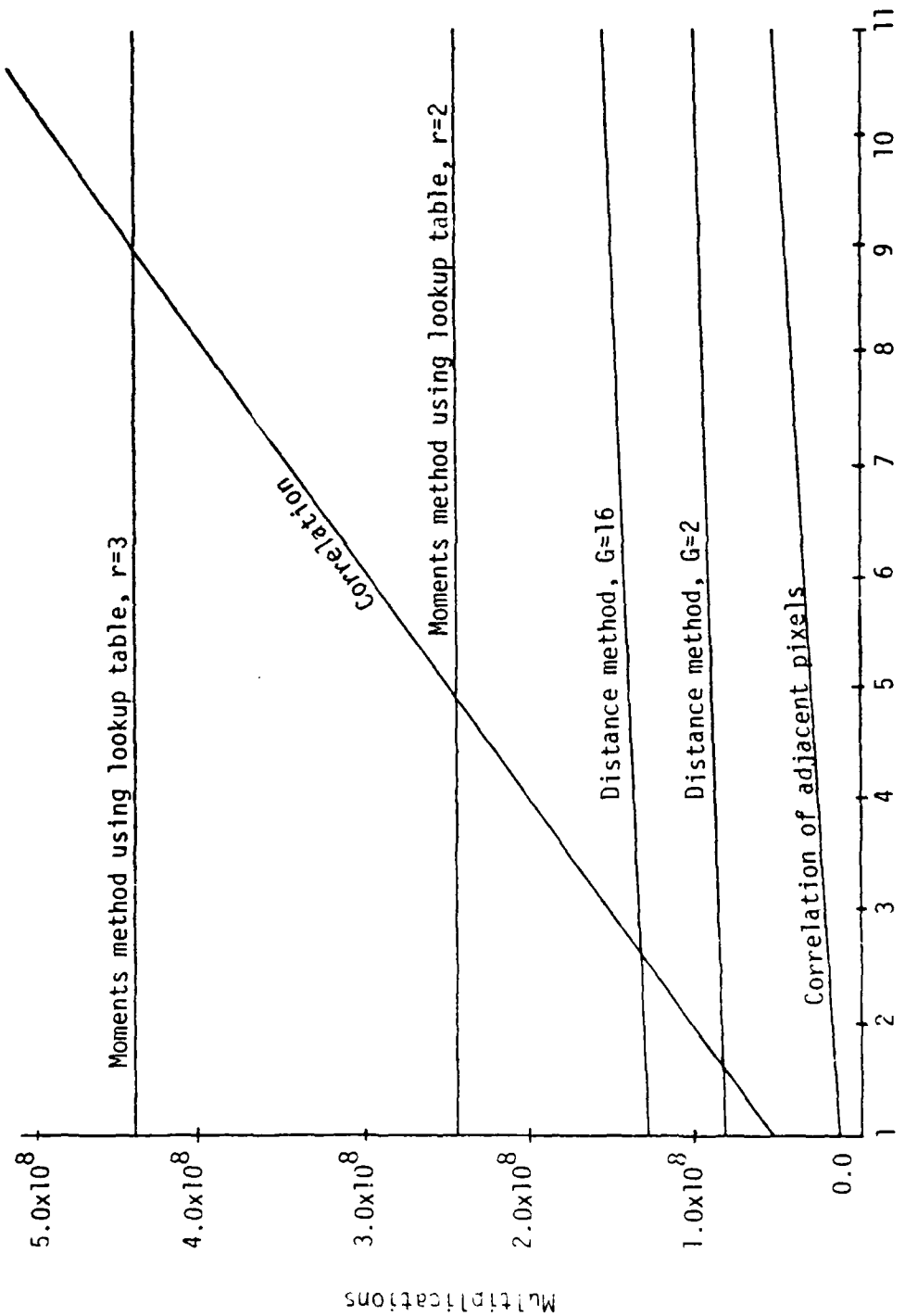
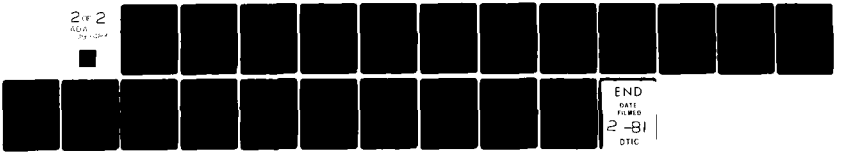
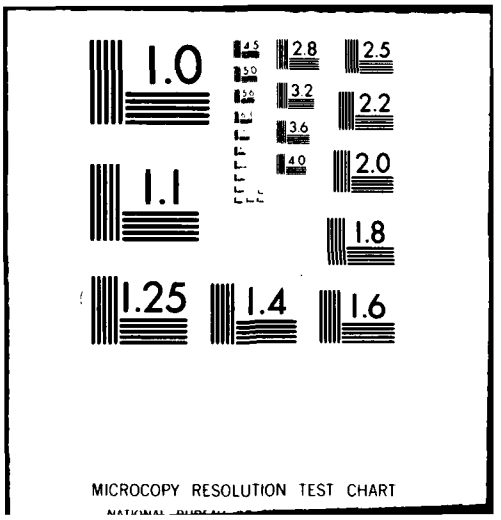


Figure 4-5. Number of multiplications to register n windows.

AD-A093 483 AUBURN UNIV ALA ENGINEERING EXPERIMENT STATION F/G 17/5
AUTOMATIC HANDOFF OF MULTIPLE TARGETS.(U)
SEP 80 J S BOLAND, H S RANGANATH DAAH01-80-C-0258
UNCLASSIFIED SBIE-AD-E950 071 NL



END
DATE FILMED
2-81
DTIC


1.0
1.1
1.25
1.4
1.6
1.8
2.0
2.2
2.5
2.8
3.2
3.6
4.0
MICROCOPY RESOLUTION TEST CHART

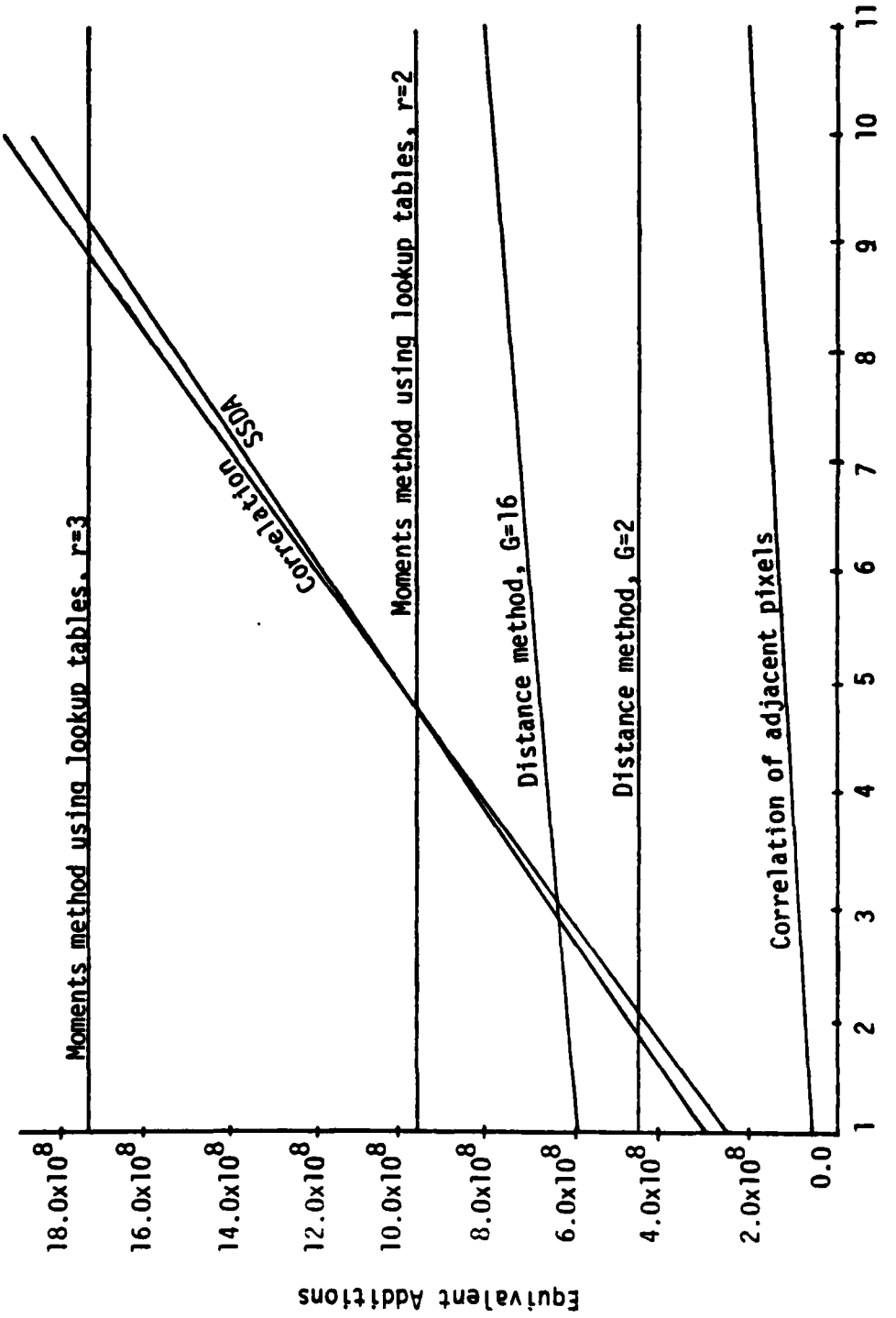


Figure 4-6. Number of equivalent additions to register n windows.

Total number of additions = (4-86)

5976079n + 8849104

The number of additions and multiplications which are required to implement several registration methods for various values of n are shown in Figures (4-4) and (4-5), it is obvious that the new method based on correlation between adjacent pixels is computationally more efficient than any of the methods previously considered. Since $V^{i,j+1}$ or $V^{i+1,j}$ can be computed from $V^{i,j}$ with few arithmetic operations, this method is promising for real time implementation. A few simulations were run using images from similar sensors and the above method was successful.

Feature Extraction Technique for Fast Image Registration

When two images do not differ in pixel resolution and rotation, the method most widely used for image registration is cross-correlation. The elements of the normalized cross-correlation surface are defined to be

$$R(i,j) = \frac{\sum_{\ell=1}^{K}\sum_{m=1}^{L} W(\ell,m)\, S_{i,j}(\ell,m)}{\left[\sum_{\ell=1}^{K}\sum_{m=1}^{L} W^2(\ell,m)\right]^{\frac{1}{2}}\left[\sum_{\ell=1}^{K}\sum_{m=1}^{L} S_{i,j}^2(\ell,m)\right]^{\frac{1}{2}}} \qquad (4\text{-}87)$$

for $1 \le i \le M-K+1$, $1 \le j \le N-L+1$

In general, the amount of computation associated with any similarity detection method is proportional to the number of pixels in the window and the number of pixels in the allowable search area. For the cross-correlation method, each of the KL pixels in W is compared with the corresponding KL pixels in $S_{i,j}$ to compute R(i,j). Since the

correlation function has (M-K+1)(N-L+1) elements, a total of KL(M-K+1)(N-L+1) pairs of pixels are compared. Thus the computation time is roughly proportional to KL(M-K+1)(N-L+1). Schemes such as "two-stage template matching" and "course-fine template matching" have been proposed to speedup correlation method and are described in Chapter I. A feature extraction technique presented in this Section selects a set of pixels, W', from the window W to be used in correlation and thus achieves significant savings in computation with little effect on the correlator accuracy and reliability.

All methods which speedup correlation accomplish the task by somehow reducing the total number of pixel pair comparisons during computation of the correlation surface. Feature extraction, one of the fundamental methods for data compression in the field of pattern recognition can be used to accomplish the same. An ideal feature is required to have the following properties:

1. The feature should retain, from the original pattern, as much information as possible.
2. The feature should accomplish as much data reduction as possible.
3. The feature should be invariant or depend on some invariant properties of the original pattern in a known way.

In practice the first two above are conflicting properties. However, striking a balance with consistent and acceptable accuracy for recognition of the original pattern can be accomplished.

Consider the subset of pixels, W', from W as a feature based on the mean and standard deviation of pixel values of W as shown in Figure 4-7. W' is a set of all pixels in W such that W(ℓ,m) is either greater than the mean plus a constant ρ times the standard deviation ($\mu+\rho\sigma$) or less than ($\mu-\rho\sigma$), where ρ is the scale factor greater than zero. This feature retains pixels from W whose pixel values are relatively low or high as compared to the mean. Data reduction is accomplished by deleteing all pixels in W whose pixel values range from ($\mu-\rho\sigma$) to ($\mu+\rho\sigma$). The amount of information retained and the number of pixels n in W', depends on the scale factor ρ and therefore is controllable. Finally, the feature depends on statistics of the window in a known way.

Elements of the correlation surface are computed by comparing only the n pixels from W which belong to feature set W' with their corresponding pixels in the subimages of S.

$$R(i,j) = \frac{\sum_{(\ell,m)} W(\ell,m)\, S_{i,j}(\ell,m)}{\left[\sum_{(\ell,m)} W^2(\ell,m)\right]^{\frac{1}{2}}\left[\sum_{(\ell,m)} S^2_{i,j}(\ell,m)\right]^{\frac{1}{2}}} \qquad (4\text{-}88)$$

for all (ℓ,m) such that W(ℓ,m) belongs to W' and for

$$1 \leq i \leq M-K+1,\ 1 \leq j \leq N-L+1$$

Now the computation time is roughly proportional to n(M-K+1)(N-L+1). The percent savings in computation time depends on the histogram of W and the scale factor ρ and therefore is scene dependent. However, simulation results presented in the next section indicate a savings of 50 to 75 percent if ρ is set to one. The percent savings in computation time, T, can be computed using the relation

$$T = \left(\frac{KL-n}{KL}\right)*100\% \tag{4-89}$$

In real world problems, n depends on the scene and nothing much can be said about it without knowledge of the histogram of W. However, in order to gain some insight into this method, consider the following two examples.

Example 1: Let W be an image whose pixel values assume a unimodal gaussian distribution with mean μ and standard deviation σ; i.e.,

$$p(x) = \frac{1}{\sqrt{2\pi}\,\sigma} \exp\left[\frac{(x-\mu)^2}{2\sigma^2}\right] \tag{4-90}$$

Then,

$$\frac{\text{number of pixels in W'}}{\text{number of pixels in W}} = 1 - \int_{x=\mu-\rho\sigma}^{\mu+\rho\sigma} p(x)dx \tag{4-91}$$

$$= 1 - \left[\text{erf}\left(\frac{\mu+\rho\sigma-\mu}{\sigma}\right) - \text{erf}\left(\frac{\mu-\rho\sigma-\mu}{\sigma}\right)\right]$$

$$= 1 - 2\,\text{erf}(\rho)$$

where

$$\text{erf}(\rho) = \frac{1}{\sqrt{2\pi}} \int_0^{\rho} e^{-x^2/2} dx$$

Therefore, $T = 200\,\text{erf}(\rho)\%$

A plot of T versus ρ is shown in Figure 4-8. For small values of ρ, the curve is almost linear indicating rapid reduction in computation time as ρ is increased from zero. For large values of ρ the curve becomes flat yielding small savings in computation for corresponding increases in ρ.

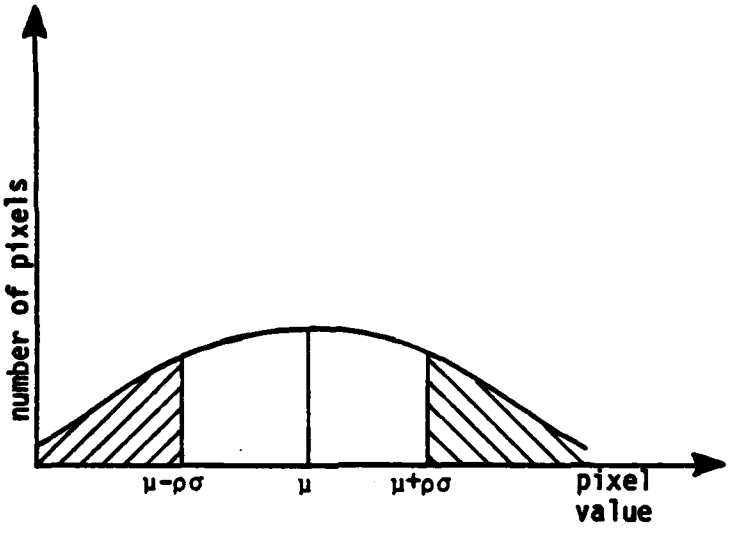


Figure 4-7. Histogram of the window.

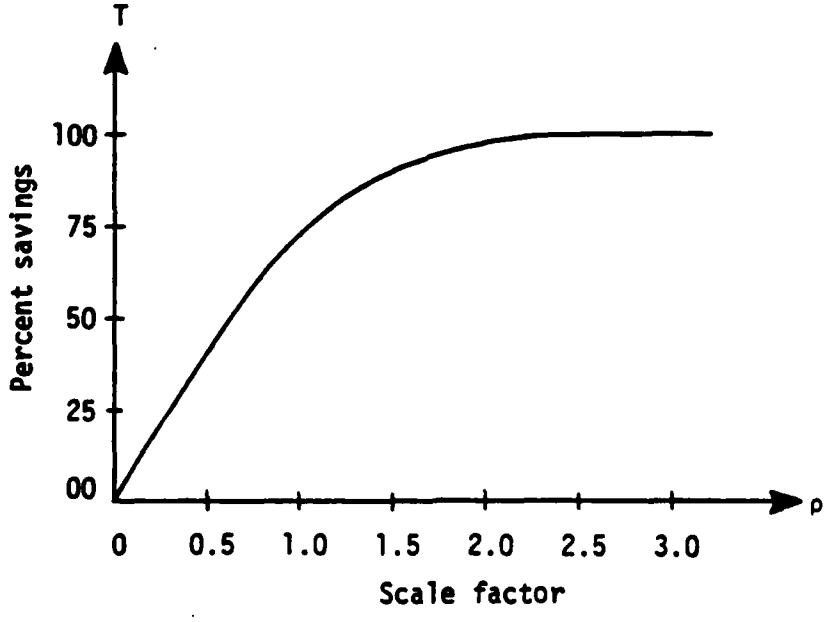


Figure 4-8. Plot of percent savings in computation T versus scale factor ρ.

Example 2: Let W be a bimodal image which means that the histogram of W is a mixture of two unimodal histograms $p_1(x)$ and $p_2(x)$. μ_1 is the mean of $p_1(x)$ and σ_1 is the standard deviation about μ_1. Similarly μ_2 is the mean of $p_2(x)$ and σ_2 is the standard deviation about μ_2. If P_1 and P_2 are apriori probabilities of two principal brightness levels, then the histogram of W, $p(x)$, can be expressed as [26, pp. 326-328]

$$p(x) = P_1\, p_1(x) + P_2\, p_2(x) \tag{4-92}$$

where $P_1 + P_2 = 1$

For the gaussian case

$$p_1(x) = \frac{1}{\sqrt{2\pi}\,\sigma_1} \exp\left[\frac{-(x-\mu_1)^2}{2\sigma_1^2}\right] \tag{4-93}$$

$$p_2(x) = \frac{1}{\sqrt{2\pi}\,\sigma_2} \exp\left[\frac{-(x-\mu_2)^2}{2\sigma_2^2}\right] \tag{4-94}$$

Let μ and σ^2 be the mean and variance of pixel values of W.

$$\mu = E[x] = P_1\mu_1 + P_2\mu_2 \tag{4-95}$$

$$E[x^2] = \int_{-\infty}^{\infty} [P_1 p_1(x) + P_2 p_2(x)]x^2\, dx \tag{4-96}$$

$$= P_1 \int_{-\infty}^{\infty} x^2 p_1(x)\, dx + P_2 \int_{-\infty}^{\infty} x^2 p_2(x)\, dx$$

$$= P_1\,(\mu_1^2 + \sigma_1^2) + P_2(\mu_2^2 + \sigma_2^2)$$

The variance σ^2 is given by:

$$\sigma^2 = E[x^2] - \mu^2 \tag{4-97}$$

$$= P_1(\mu_1^2 + \sigma_1^2) + P_2(\mu_2^2 + \sigma_2^2) - (P_1\mu_1 + P_2\mu_2)^2$$

$$= P_1\mu_1^2 - P_1^2\mu_1^2 + P_2\mu_2^2 - P_2^2\mu_2^2 - 2P_1P_2\mu_1\mu_2 + P_1\sigma_1^2 + P_2\sigma_2^2$$

$$= P_1\mu_1^2(1 - P_1) + P_2\mu_2^2(1 - P_2) - 2P_1P_2\mu_1\mu_2 + P_1\sigma_1^2 + P_2\sigma_2^2$$

$$= P_1P_2(\mu_1 - \mu_2)^2 + P_1\sigma_1^2 + P_2\sigma_2^2$$

The percent savings in computation T is given by:

$$T = \int_{\mu-\rho\sigma}^{\mu+\rho\sigma} p(x)\,dx \tag{4-98}$$

$$= \int_{\mu-\rho\sigma}^{\mu+\rho\sigma} [P_1p_1(x) + P_2p_2(x)]\,dx$$

$$= P_1 \int_{\mu-\rho\sigma}^{\mu+\rho\sigma} p_1(x)\,dx + P_2 \int_{\mu-\sigma\rho}^{\mu+\rho\sigma} p_2(x)\,dx$$

$$= P_1\left\{\operatorname{erf}\left[\frac{\mu+\rho\sigma-\mu_1}{\sigma_1}\right] - \operatorname{erf}\left[\frac{\mu-\rho\sigma-\mu_1}{\sigma_1}\right]\right\}$$

$$+ P_2\left\{\operatorname{erf}\left[\frac{\mu+\rho\sigma-\mu_2}{\sigma_2}\right] - \operatorname{erf}\left[\frac{\mu-\rho\sigma-\mu_2}{\sigma_2}\right]\right\}$$

Substituting Equation (4-95) into the above equation yields

$$T = P_1\left\{\operatorname{erf}\left[\frac{P_2(\mu_2-\mu_1)+\rho\sigma}{\sigma_1}\right] - \operatorname{erf}\left[\frac{P_2(\mu_2-\mu_1)-\rho\sigma}{\sigma_1}\right]\right\}$$

$$+ P_2\left\{\operatorname{erf}\left[\frac{P_1(\mu_1-\mu_2)+\rho\sigma}{\sigma_2}\right] - \operatorname{erf}\left[\frac{P_1(\mu_1-\mu_2)-\rho\sigma}{\sigma_2}\right]\right\}$$

where $\sigma^2 = P_1P_2(\mu_1-\mu_2)^2 + P_1\sigma_1^2 + P_2\sigma_2^2$

Notice T depends on the parameters μ_1, μ_2, σ_1 and σ_2 of the image and of course on the scale factor ρ. T is scene dependent and general conclusions cannot be drawn without knowledge of the histogram of W.

Simulation Results

Scenes used for simulation were obtained from sensors sensitive in the visual spectrum (day TV sensors). Because of the difference in the sensors, the two images were preprocessed such that they have the same spatial resolution. An algorithm to accomplish this is given in Reference [4]. W is a 32x32 array of pixels and S is a 120x120 array of pixels extracted from preprocessed high and low resolution images, respectively. Since a one-bit or two-level correlator is used in this work, W' and S must be quantized to two levels. The mean, μ, and standard deviation, σ, of W are first computed. Pixels are quantized to 1 or 0 if they are above $\mu+\rho\sigma$ or below $\mu-\rho\sigma$, respectively. The locations of pixels within W with values between $\mu-\rho\sigma$ and $\mu+\rho\sigma$ are stored and masked out of the correlation process. Also, each subimage, $S_{i,j}$ of S was quanitzed to ones or zeros depending on whether the pixel value was above or below the average value of the subimage, $\mu_{i,j}$, as given below

$$S_{i,j}(\ell,m) = \begin{cases} 1 & \text{, if } S_{i,j}(\ell,m) \geq \mu_{i,j} \\ 0 & \text{, otherwise} \end{cases} \qquad (4\text{-}99)$$

A number of simulations were run using typical scenes for different values of ρ. As a figure of merit, the correlation surface signal-to-noise ratio was computed as

$$SNR = \frac{R_{max} - R_{avg}}{R_{sigma}} , \qquad (4\text{-}100)$$

where R_{max} is the maximum value of correlation surface,

R_{avg} is the average value of correlation surface,

R_{sigma} is the standard deviation of correlation surface.

Correlation was found successful with the highest value in the correlation surface indicating the true registration for values of ρ ranging from 0 to 1.4. When ρ was increased beyond 1.4, correlation was unsuccessful. Percent savings in computation time and signal-to-noise ratio of the correlation surface for various values of ρ ranging from zero to one are tabulated in Table 4-1 and Table 4-2, respectively. From the above simulation results, it is concluded that substantial savings in computation (50 to 75 percent) is accomplished with little effect on correlation accuracy and reliability.

Generalization

Since the new feature extraction technique reduces the amount of data to be processed without altering the structure, it is applicable not only to correlation but to many other similarity detection methods. The following generalizations can be made:

1. By using n elements of W' as test points in sequential similarity detection algorithm suggested by Barnea and Silverman, the problem of digital image registration reduces to the problem of finding $S_{i*,j*}$ such that

Table 4-1

Percent savings in computation.

Scene	ρ=0.00	ρ=0.25	ρ=0.50	ρ=0.75	ρ=1.00
1	00.00	16.89	33.30	57.92	75.78
2	00.00	13.21	28.90	48.82	71.38
3	00.00	3.61	12.03	27.63	51.30
4	00.00	19.02	27.52	53.21	70.50
5	00.00	20.00	37.42	55.34	73.24

Table 4-2

SNR of correlation surface.

Scene	ρ=0.00	ρ=0.25	ρ=0.50	ρ=0.75	ρ=1.00
1	5.885	5.609	4.907	4.669	3.934
2	7.175	6.101	5.045	4.356	4.321
3	6.775	6.769	6.650	6.347	5.236
4	7.619	7.20	6.64	6.13	5.31
5	8.197	8.14	7.91	7.51	6.85

$$e_{i^*,j^*} < e_{i,j} \text{ for } 1 \le i \le M-K+1 \text{ and } i \ne i^* \quad (4\text{-}101)$$
$$1 \le j \le N-L+1 \text{ and } j \ne j^*$$

In (4-101) $e_{i,j}$ can be computed using Equation (4-102)

$$e_{i,j} = \sum_{(\ell,m)} |S_{i,j}(\ell,m) - \hat{S}_{i,j} - W(\ell,m) + \hat{W}| \quad (4\text{-}102)$$

where

$$\hat{S}_{i,j} = \frac{1}{n} \sum_{(\ell,m)} S_{i,j}(\ell,m) \quad (4\text{-}103)$$

and

$$\hat{W} = \frac{1}{n} \sum_{(\ell,m)} W(\ell,m) \quad (4\text{-}104)$$

for all (ℓ,m) such that $W(\ell,m)$ belongs to W' and for $1 \le i \le M-K+1$, $1 \le j \le N-L+1$

A few simulations were run and the above method was found successful.

2. For the moments method, moments can be computed based on n elements of W' rather than all the KL elements of W.

3. This method can be combined with the improved method for correlating similar sensor images, suggested in Reference [30] to improve the probability of finding the true peak.

   Step 1: Compute the Cross-Correlation surface by comparing only the n pixels from W which belong to the feature set W' with their corresponding pixels in the subimages of S (using Equation (4-88)).

Step 2: Identify a predetermined number of highest peaks and coordinates of their occurence from the correlation surface. Let (I_1,J_1), (I_2,J_2) ... , (I_{k-1},J_{k-1}) and (I_k,J_k) be the coordinates of the first k peaks. It is assumed that the true registration point is one among them. Recompute $R(I_1,J_1)$, $R(I_2,J_2)$, ... , $R(I_{k-1},J_{k-1})$ and $R(I_k,J_k)$ by comparing all the KL pixels of W with the corresponding pixels in the subimages of S beginning at (I_1,J_1), (I_2,J_2), ... , (I_{k-1},J_{k-1}) and (I_k,J_k), respectively (using Equation (4-87)). Savings in computation is accomplished in Step 1. Step 2 increases the probability of finding the true peak and reduces the possibility of false registration.

V. CONCLUSIONS AND RECOMMENDATIONS

Most of the conclusions and recommendations presented in this chapter have been given and justified within the first five chapters of this dissertation.

Conclusions

1. Various methods for accomplishing digital image registration are presented in Chapter II. The most commonly used method is cross-correlation. There exist two independent ways of computing the correlation surface (Direct method and the FFT method). Even though the FFT method requires less computation compared to the direct method, it requires a large amount of memory for software implementation and is very complex for real time hardware implementation. The direct method requires less memory compared to the FFT method, involves no complex multiplications or complex additions and can be easily implemented using digital hardware.
2. Even though the vector correlation is expected to yield better performance than the standard correlation algorithm, its use is limited by the large amount of computation required to implement the method (more than twice the computation required by the standard correlation algorithm).
3. Due to the difficulty encountered in the automatic segmentation of digital images into homogeneous regions, feature matching

correlation and hybrid correlation algorithms may not be of any significant use (especially for hardware implementation).

4. Sequential similarity detection algorithm computes error surface as a measure of dissimilarity between the window and subimages of the search area. Since this method requires only addition and a few division operations, implementation is simpler than that of the correlation method (addition or subtraction is simpler than multiplication or division).
5. All the three popular techniques which speedup template matching (two-stage template matching, course-fine template matching and hierarchical search method), accomplish savings in computation by reducing the total number of pixel pair comparisons. The methods are computationally more efficient in terms of the number of arithmetic operations required (software implementation) but may not enjoy any advantage in real-time implementation using special purpose hardware.
6. The necessity of a highly efficient algorithm to transform digital images to binary images (edge and non-edge pixels) and a connectivity test to identify true straight line edges (composed of adjacent pixels) makes the use of the Hough transformation for digital image registration less attractive.
7. The moments of an image or subimage can be easily computed by performing multiplications and additions only. When the window and the search area do not differ in rotation or pixel spatial resolution, it is not necessary to compute Hu's invariant moments and the ordinary moment sequence can be directly used for scene matching.

8. For the reasons given in 1 through 7, it is concluded that the standard correlation algorithm, the SSDA and the moments method are more promising using present state-of-the-art hardware.
9. In Chapter III, the computational efficiency of the correlation, the SSDA and the moments methods is compared from software as well as hardware points of view, independently, for the multiple image registration problem. It is found that the moments method takes less computation time if implemented in software and less hardware for real-time implementation if the number of windows is sufficiently large. Moments of any order can be computed with one level of multiplications and one level of additions. Therefore, the moments method is promising for real-time implementation.
10. In order to obtain better results from any of the image registration algorithms, the window and each subimage of the search area should be preprocessed to have zero mean and unity standard deviation. This is called intensity level normalization and would require too much additional computation. In Chapter IV, it is shown that intensity level normalization can be embedded within the moments method with almost no additional computation. This makes the moments method more accurate without excessive additional computation.
11. Two new feature matching algorithms, one based on interset and intraset distances, and the other based on correlation of adjacent pixels are developed in Chapter IV. The distance method is computationally more efficient (in terms of the number of arithmetic operations) than the correlation or the SSDA method if the number of windows is greater than three.

12. It has been shown that the intraset distance of a set is a function of zero and second order central moments of the set. The interset distance between two sets is a function of zero and first order moments of both the sets. When an image is composed of homogeneous regions, all pixels of a homogeneous region normally fall into the same set when the image is segmented based on pixel values. Computing intraset and interset distances, in some sense, is the same as computing moments of each homogeneous region, separately. In general, processing each homogeneous region separately and combining the partial results additively yields better performance. Therefore, the distance method is expected to perform better than the moments method if the scene is composed of homogeneous regions.
13. For multiple image registration, the algorithm based on correlation of adjacent pixels is computationally more efficient than any of the algorithms considered. Even for single image registration, this method requires less number of arithmetic operations than the standard correlation algorithm. Since the feature vector associated with reference point (i,j+1) or (i+1,j) can be computed from the feature vector associated with the reference point (i,j) with very few arithmetic operations, this method is very promising for real-time implementation.
14. From 9, 11 and 13, it is concluded that an algorithm which is computationally efficient for single image registration may not be efficient for multiple image registration. It is also concluded that the computation for template matching algorithms is directly proportional to the number of windows whereas in feature matching

algorithms features are extracted for all subimages and windows only once, and the matching procedure is repeated once for each window. Since the computation required to match the features is negligible compared to that required to compute features, in general, feature matching algorithms are expected to be more efficient than template matching algorithms if the number of windows is sufficiently large.

15. There are many other feature matching image registration algorithms reported in the literature that have not been mentioned in this report. The reason for eliminating these algorithms was either they were too computationally complex to be implemented in real-time hardware in the near future or they were not as computationally accurate as the methods presented. Special purpose VLSIC developed at some future date may make some of these algorithms feasible.
16. A feature extraction technique based on the mean and standard deviation of pixel values of the window accomplishes 50 to 75 percent savings in computation with very little effect on registration accuracy. Since the feature extraction technique reduces the amount of data to be processed without altering the structure, it is applicable not only to correlation but to many other image registration algorithms.

Recommendations for Future Work

1. Even though the correlation, the SSDA and the moments methods are more promising using present state-of-the-art hardware, very large scale integrated circuits developed to perform a specialized task may at some future date make any of the methods discussed in Chapter

II feasible. Because of this possibility, a follow-on program should investigate the computational accuracy of these methods.

2. It may be possible to obtain good registration results using very few moments. This trade-off should be investigated through simulation of typical military-type digitized scenes.
3. If the images are quantized to two levels (0 and 1) or three levels (-1, 0 and +1), the computation of moments involves no multiplications. Therefore, performance of the moments method with two and three level images as inputs should be studied.
4. Accuracy, reliability and sensitivity to noise of all three feature matching methods in Chapter IV should be determined through simulation.
5. The effect of quantization of images to two or three levels on the performance of feature matching algorithms should be investigated.
6. The potential of the method based on correlation of adjacent pixels for real-time hardware implementation should be studied.

REFERENCES

[1] J. A. Leese, C. S. Novak and B. B. Clark, "An automated technique for obtaining cloud motion from geosynchronous satellite data using cross-correlation," J. Appl. Meteorol. Vol. 10, pp. 110-132, February 1971.

[2] L. Novak, "Correlation algorithm for radar map matching," Proc. IEEE Conference on Decision and Control, Clearwater, FL., pp. 780-790, December 1-3, 1976.

[3] R. Y. Wong, "Scene matching tecnhiques," Proc. IEEE Southeastcon, Atlanta, GA, pp. 238-241, April 10-12, 1978.

[4] J. S. Boland, III, L. J. Pinson, G. R. Kane, M. A. Honnell and E. G. Peters, "Automatic target hand-off using correlation techniques," Final Technical Report Contract DAAH-1-76-C-0369, U. S. Army Missile Research and Development Command, Redstone Arsenal, AL, January 31, 1977.

[5] S. L. Tanimoto, "A comparison of some image searching methods," Proc. IEEE Conference on Pattern Recognition and Image Processing, Chicago, Illinois, pp. 280-286, May 31-June 2, 1978.

[6] S. A. Dudani, J. A. Jenney and B. L. Bullock, "Correlation and alternatives for scene matching," Proc. IEEE Conference on Decision and Control, Clearwater, FL, pp. 774-779, December 1-3, 1976.

[7] D. I. Barnea and H. F. Silverman, "A class of algorithms for fast digital image registration," IEEE Trans. Comput., Vol. C-21, pp. 179-186, February 1972.

[8] G. J. Vanderbrug and A. Rosenfeld, "Two-stage template matching," IEEE Trans. Comput., Vol. C-26, pp. 384-393, April 1977.

[9] A. Rosenfeld and G. J. Vanderbrug, "Course-fine template matching," IEEE Trans. Systems, Man, and Cybernetics, SMC-7, No. 2, pp. 104-107, February 1977.

[10] M. K. Hu, "Visual pattern recognition by moment invariants," IRE Trans., IT-8, pp. 179-187, February 1962.

[11] F. L. Alt, "Digital pattern recognition by moments," J. of the Assoc. for Comp. Machinery, pp. 240-246, April 1962.

[12] K. Udagawa, J. Toriwaki and K. Sagino, "Normalization and Recognition of two dimensional patterns with linear distortions with moments," Electron. Commun. Jap., Vol. 47, pp. 34-46, June 1964.

[13] S. A. Dudani, K. J. Breeding and R. B. McGhee, "Aircraft identification by moment invariants," IEEE Trans. on Comput., pp. 39-46, January 1977.

[14] F. W. Smith and M. H. Wright, "Automatic ship photo interpretation by the method of moments," IEEE Trans. on Comput., Vol. C-20, pp. 1089-1094, September 1971.

[15] D. P. Panda and A. Rosenfeld, "Image segmentation by pixel classification in gray level/edge value space," IEEE Trans. Comput., Vol. C-27, pp. 875-879, September 1978.

[16] J. A. Ratkovic, "Hybrid correlation algorithms - a bridge between feature matching and image correlation," The Rand Paper Series, November 1979.

[17] J. W. Cooley, A. W. Lewis and P. D. Welch, "Application of the fast fourier transform to computation of fourier integrals, fourier series, and convolution integrals," IEEE Trans. of Audio and Electroacoustics, Vol. AU-15, No. 2, pp. 79-84, June 1967.

[18] P. E. Anuta, "Spatial registration of multispectral and multitemporal digital imagery using FFT technique," IEEE Trans. on Geoscience Electronics, Vol. GE-8, pp. 353-368, October 1970.

[19] A. Papoulis, "Probability, Random Variables and Stochastic Processes," McGraw-Hill Book Company, N. Y., 1965.

[20] P. V. C. Hough, "Method and means of recognizing complex patterns," U. S. Patent 3,069,654, December 1962.

[21] A. Rosenfeld, "Picture Processing by Computer," Academic Press, New York, 1969.

[22] R. O. Duda and P. E. Hart, "Use of the Hough transformation to detect lines and curves in pictures," Computer Graphics and Image Processing, Vol. 15, pp. 11-15, January 1972.

[23] C. C. Ormsby, "Advanced scene matching techniques," Proc. IEEE National Aerospace and Electronics Conference, Dayton, Ohio, pp. 68-76, May 15-17, 1979.

[24] R. Y. Wong and E. L. Hall, "Sequential hierarchical scene matching," IEEE Trans. on Comput., Vol. C-27, pp. 359-366, April 1978.

[25] E. L. Hall and R. Y. Wong, "Hierarchical search for image matching," Proc. IEEE Conference on Decision and Control, Clearwater, FL, pp. 791-796, December 1976.

[26] J. T. Tou and R. C. Gonzalez, "Pattern Recognition Principles," Addison-Wesley Publishing Company, Inc., 1974.

[27] R. C. Gonzalez and P. Wintz, "Digital Image Processing," Addison-Wesley Publishing Company, Inc., 1977.

[28] J. S. Boland, III, L. J. Pinson, and E. G. Peters, "Automatic target hand-off for non-compatible image systems," Final Technical Report under contract DAAK40-77-G-0156, U. S. Army Missile Research and Development Command, Redstone Arsenal, AL, September 1976.

[29] J. S. Boland, III, and H. S. Ranganath, "Correlation algorithm development," Final Technical Report under Contract DAAK40-79-M-0016, U. S. Army Missile Research and Development Command, Redstone Arsenal, AL, April 1979.

[30] J. S. Boland, III, and H. S. Ranganath, "Quantization effects on target hand-off from TV-to-IR digitized scenes," Final Technical Report under Contract DAAK40-79-M-0104, U. S. Army Missile Command, Redstone Arsenal, AL, September 1979.


DATE:
ILME