CRINC REMOTE SENSING LABORATORY

AD A053240

AD No.
DDC FILE COPY

ETL-0118
RADAR IMAGE SIMULATION: VALIDATION OF THE POINT SCATTERING METHOD

Volume Two

J. C. Holtzman
V. H. Kaupp
J. L. Abbott
V. S. Frost
E. E. Komp
E. C. Davison

DDC
APR 27 1978

September, 1977

Approved for public release; distribution unlimited

Prepared for:

U. S. ARMY ENGINEER TOPOGRAPHIC LABORATORIES
Fort Belvoir, Virginia 22060
CONTRACT DAAG53-76-C-0154

THE UNIVERSITY OF KANSAS CENTER FOR RESEARCH, INC.
2291 Irving Hill Drive—Campus West Lawrence, Kansas 66045

ETL - 0118

RADAR IMAGE SIMULATION: VALIDATION OF THE POINT SCATTERING METHOD VOLUME TWO

J. C. HOLTZMAN

V. H. KAUPP

J. L. ABBOTT

V. S. FROST

E. E. KOMP

E. C. DAVISON

DDC
APR 27 1978
F

September, 1977

Approved for public release; distribution unlimited

Prepared for:

U. S. Army Engineer Topographic Laboratories
Fort Belvoir, Virginia 22060

CONTRACT DAAG53-76-C-0154

DESTROY THIS REPORT WHEN NO LONGER NEEDED.
DO NOT RETURN IT TO THE ORIGINATOR.

THE FINDINGS IN THIS REPORT ARE NOT TO BE CONSTRUED AS AN OFFICIAL DEPARTMENT OF THE ARMY POSITION UNLESS SO DESIGNATED BY OTHER AUTHORIZED DOCUMENTS.

THE CITATION IN THIS REPORT OF TRADE NAMES OF COMMERCIALLY AVAILABLE PRODUCTS DOES NOT CONSTITUTE OFFICIAL ENDORSEMENT OR APPROVAL OF THE USE OF SUCH PRODUCTS.

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER: ETL-0118
2. GOVT ACCESSION NO.
3. RECIPIENT'S CATALOG NUMBER

4. TITLE (and Subtitle): RADAR IMAGE SIMULATION: VALIDATION OF THE POINT SCATTERING MODEL. VOLUME II.

5. TYPE OF REPORT & PERIOD COVERED: Contract Report

6. PERFORMING ORG. REPORT NUMBER: Remote Sensing Lab 319-28

7. AUTHOR(s): J. C. Holtzman, V. S. Frost, V. H. Kaupp, E. E. Komp, J. L. Abbott, E. C. Davison

8. CONTRACT OR GRANT NUMBER(s): DAAG53-76-C-0154

9. PERFORMING ORGANIZATION NAME AND ADDRESS:
University of Kansas Center for Research, Inc.
2291 Irving Hill Drive
Lawrence, Kansas 66045

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

11. CONTROLLING OFFICE NAME AND ADDRESS:
U.S. Army Engineer Topographic Laboratories
Fort Belvoir, Virginia 22060

12. REPORT DATE: Sep[redacted] 1977

13. NUMBER OF PAGES

14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office): RSL-TR-319-28

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

15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved 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)

Radar, Image, Simulation, Digital, Side-Looking Airborne, Plan Position Indicator, SLAR, PPI, Reflectivity, Backscatter, Data Base

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

The technical details of all aspects of the radar image simulation are reported. In particular, the activities associated with the Point Scattering Method are discussed. They include: (1) construction of a ground truth data base, i.e., the terrain model which incorporates elevation and dielectric behavior; (2) digitization of the terrain information to build a digital matrix; (3) formation of a backscatter data catalogue; (4) radar device modeling; and (5) problems and solutions inherent in image handling and analysis.

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

Unclassified
SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

406 688

THE UNIVERSITY OF KANSAS CENTER FOR RESEARCH, INC.

2291 Irving Hill Drive - Campus West
Lawrence, Kansas 66045

Telephone:

ETL - 0118

RADAR IMAGE SIMULATION: VALIDATION OF THE POINT SCATTERING METHOD VOLUME TWO

J. C. HOLTZMAN
V. H. KAUPP
J. L. ABBOTT
V. S. FROST
E. E. KOMP
E. C. DAVISON

RSL Technical Report 319-28

September, 1977

Approved for public release; distribution unlimited

Prepared for:

U. S. Army Engineer Topographic Laboratories
Fort Belvoir, Virginia, 22060

CONTRACT DAAG53-76-C-0154

ABSTRACT

The technical details of all aspects of the radar image simulation are reported. In particular, the activities associated with the Point Scattering Method are discussed. They include: (1) construction of a ground truth data base, i.e., the terrain model which incorporates elevation and dielectric behavior; (2) digitization of the terrain information to build a digital matrix; (3) formation of a backscatter data catalogue; (4) radar device modeling; and (5) problems and solutions inherent in image handling and analysis.

ACCESSION for
NTIS
DDC
UNANNOUNCED
JUSTIFICATION
BY
DISTRIBUTION/AVAILABILITY CODES
A

PREFACE

This document was prepared by the Kansas Simulation Group, Remote Sensing Laboratory (RSL), The University of Kansas, Lawrence, Kansas, to report the results of a Radar Simulation Study performed under Contract DAAG53-76-C-0154, dated 15 May 1976, with the Engineer Topographic Laboratory (ETL), Fort Belvoir, Virginia. This Radar Simulation Study was performed to validate the point scattering radar image simulation method which had been developed previously, to investigate terrain feature extraction techniques for constructing category data bases for radar image simulation, and to use the point scattering radar image simulation method to generate radar reference scenes for terminal guidance applications.

The work and services to be provided under this contract were organized into two separate tasks. A summarization of these tasks, collecting and grouping activities according to their relationships to these tasks, would be:

Task 1 - Simulation Systems Approach

The point scattering radar image simulation model will be used to generate simulations of a test site centered around the Pickwick Landing Dam located in Tennessee. The purpose of these simulations will be to validate the simulation technique which has been developed. The subtasks to be performed under this task are:

(1) Perform a study to validate the point scattering radar image simulation model for Side-Looking Airborne Radar (SLAR) applications by making SLAR simulations of the Pickwick test site. The data base of Pickwick for the simulation (Subtask (3)) is to be made only from maps and optical imagery. After the SLAR simulation is complete, it will be compared to original furnished radar imagery and differences will be analyzed.

(2) Perform a study to validate the point scattering radar image simulation model for Plan-Position Indicator (PPI) radar applications by making PPI simulations of the Pickwick test site. The data base to use is the one constructed in Subtask (3), below. The PPI simulation model will be developed using a polar coordinate scan so that PPI radar simulations can be made in the correct geometry. After the PPI simulation is complete, it will be compared to the SLAR simulations produced in (1), above and differences will be analyzed. This approach is being used because real PPI imagery of the Pickwick site is not available.

(3) Construct data bases of the Pickwick Landing Dam test site. The appropriate planimetric terrain features will be extracted from optical photography and maps. Investigations will be conducted to develop automated terrain feature extraction techniques. The man hours for this feature extraction process will be recorded comparing human photo interpretation methods with interactive automated methods for the same area. Using digital elevation data and the planimetry extracted from photos only, a digital data base will be produced.

Task 2 - Advanced Simulation Methods

The point scattering radar image simulation model will be applied to generate reference scenes of a test site centered around the Pickwick Landing Dam located in Tennessee for testing by the Correlatron for terminal guidance applications. The purpose of these reference scenes will be to evaluate the simulation technique for reference scene generation and to measure quantitatively the simulation results. The subtasks to be performed under this task are:

(1) Produce PPI radar reference scenes appropriate for the terminal guidance. The final product will be digital tapes of the reference scene simulations.

(3) Construct data base of the Pickwick Landing Dam test site. The appropriate planimetric terrain features will be extracted from optical photography and maps. The man hours for this feature extraction process will be recorded, comparing photo interpretation methods with interactive automated methods for the same area. Using digital elevation data and the planimetry extracted from photos only, a digital data base will be produced. Repeat the process using existing radar imagery, maps, and digital elevation tapes.

Obviously, the work and services to be performed under this contract were very extensive, spanning many disciplines, and drawing upon the knowledge and experience of geographers, electrical engineers, botanists, and computer scientists. To report these diverse activities in a coherent fashion is difficult. The format of this document has been designed to simplify, as much as possible, reporting this work.

The document is divided into two volumes to reduce the bulk that must be handled at any one time. Volume I reports the work and results with technical details deferred to the appendices. Volume II is a collection of appendices containing the individual technical details of the work reported in the first volume. In addition to reducing the bulk which must be handled, dividing the document into two volumes adds flexibility; it is easier to turn to the appropriate appendix in Volume II for technical details while keeping the work and results description open for reference in Volume I.

The organization of Volume I is structured around the basic work being reported. As can be seen in the summary of the work and services specified under this contract, there are listed two tasks and five subtasks. Upon inspection, it can be seen that the five subtasks are really only four different activities: (1) SLAR Validation; (2) PPI Validation; (3) Reference Scene Generation; (4) Data Base Construction/Feature Extraction Techniques. Volume I is organized according to these major activities. The format of Volume I and the relationships of each section to the appropriate subtask of the work and services is as follows:

Document Section	Description of Work and Results	Statement of Work Task - Subtask	
1	Introduction and summary of results	All	
2	SLAR Validation	1	1
3	PPI Validation	1	2
4	Terminal Guidance Applications	2	1
5	Data Base Construction/Feature Extraction Techniques	(1 2)	(3 2)

Volume II has a simple organizational plan. It is an alphabetical listing of the appendices required to support the work and results reported in Volume I. These appendices represent the technical information necessary to support the discussions of work and results in Volume I. There are fourteen appendices provided in Volume II as follows:

Appendix	Technical Report No.	Title	Date
A	TR 319-1	"Construction of a Geometric Data Base for Radar Image Simulation Studies"	July, 1976
B	TR 319-2	"Baseline of Planimetric Data Base Construction: Pickwick Site"	July, 1976
C	TR 319-3	"Digital Elevation Data Base Construction: Pickwick Site"	July, 1976
D	TR 319-4	"Digitization of Pickwick Site Data Base"	February, 1977
E	TR 319-21	"Increased Resolution of Planimetric Data Base: Pickwick Site"	March, 1977
F	TR 319-5	"Medium Resolution Digital Ground Truth Data Base"	August, 1977

G	TR 319-7	"Backscatter Data for the Digital Radar Image Simulation of the Pickwick, Alabama Site"	February, 19
H	TR 319-9	"Medium Resolution Radar Image Simulation of Deciduous Forests: A Study of Candidate Techniques"	August, 1977
I	TR 319-8	"Digital Model for Radar Image Simulation and Results"	August, 1976
J	TR 319-19	"Digital PPI Model for Radar Image Simulation and Results"	August, 1977
K	TR 319-15	"Investigation of an Interactive Approach for Radar Image Simulation"	March, 1977
L	TR 319-24	"Automated Techniques in Feature Extraction"	June, 1977
M	TR 319-25	"Interactive Feature Extraction System Framework"	June, 1977
N	TR 319-26	"Image Handling and Processing"	June, 1977

PURPOSE

The purpose of this document is to report the results of a Radar Simulation Study conducted by the RSL (Remote Sensing Laboratory, University of Kansas) under contract with ETL (Engineer Topographic Laboratories, United States Army, Fort Belvoir, Virginia). The Radar Simulation Study was performed to test the Point Scattering Radar Image Simulation Model developed and reported in previous work[1,2]. The Point Scattering Model was applied to three specific problems in this study, and the technical details of the work performed are reported in this document. The three specific applications tested in this study are: (1) SLAR (Side-Looking Airborne Radar) Model Validation; (2) PPI (Plan-Position Indicator) Radar Model Validation; (3) Terminal Guidance Applications. In addition to the implementation and testing of these three applications of the simulation model, much effort was expended in peripheral activities required to support the main efforts. Principal of these was data base construction with emphasis on feature extraction methods and techniques. As these activities are of critical importance to successful implementation of radar simulation models and to successful utilization of these models, the purpose of this document is extended to supply detailed information about these support activities, also.

---

[1] Holtzman, J. C., V. H. Kaupp, R. L. Martin, E. E. Komp, and V. S. Frost, "Radar Image Simulation Project: Development of a General Simulation Model and Interactive Simulation Model, and Sample Results,: TR 234-13, Remote Sensing Laboratory, The University of Kansas, February, 1976.

[2] Holtzman, J. C., V. H. Kaupp, and J. L. Abbott, "Radar Image Simulation Project,: TR 234-15, Remote Sensing Laboratory, The University of Kansas, September, 1976.

SCOPE

The scope of the work performed in this Radar Simulation Study was limited to testing the Point Scattering Radar Image Simulation Model against one specific area. The three applications (SLAR, PPI, and Terminal Guidance) of the simulation model were each tested against this one area. The area selected for this test of the simulation model was the topographic region in the states of Tennessee, Alabama, and Mississippi, centered on the northwest corner of the powerhouse at the Pickwick Landing Dam, Tennessee. The SLAR and PPI validation work was limited to forming a sequence of radar image simulations from two different look directions of selected subregions of the Pickwick test site and comparing these simulated radar images to real images (of the same regions) having the same look directions. The terminal guidance work was limited to producing reference scenes of the Pickwick site from one altitude for running on the Correlatron. The data base construction/feature extraction work was limited to preparation of two data bases of the Pickwick site: (1) Data base for SLAR and PPI validation work, (2) Data base for Terminal Guidance work.

Table of Contents

Volume I ETL TR-0117

ABSTRACT i

PREFACE iii

PURPOSE viii

SCOPE ix

EXECUTIVE SUMMARY x

TABLE OF CONTENTS xiii

LIST OF FIGURES xxii

LIST OF TABLES xxv

LIST OF PROGRAMS xxvi

RADAR IMAGE SIMULATION: AN OVERVIEW xxvii

I. Radar Systems xxviii

II. Refinement of the Radar Model for Simulation: The Greytone Equation xl

III. Radar Simulation xlvi

1.0 SUMMARY AND RESULTS 1

1.1 Introduction 1

1.2 Summary of Work 6

1.2.1 Summary of Work - SLAR Validation 9

1.2.2 Summary of Work - PPI Validation 11

1.2.3 Summary of Work - Terrain Guidance Applications 12

1.2.4 Construction of Data Bases/Feature Extraction Techniques 17

1.3 Summary of Results 20

1.3.1 Summary of Results - SLAR Validation 21

1.3.2 Summary of Results - PPI Validation 22

1.3.3 Summary of Results - Reference Scene Simulation 22

1.3.4 Summary of Results - Data Base Construction/Feature Extraction 24

1.4 Qualifications, Limitations, and Constraints of Model 26

1.4.1. Validity of the Radar Equation for Simulation 36

1.4.1.1 Return Power Model 36

1.4.1.2 Resolution Cell Size 38

1.4.1.3 Resolution Cell Size: Alternate Method 44

1.4.2 Validity of Differential Scattering Cross-Section Concept 54
1.4.2.1 Explanation of Empirical σ° Data 57
1.4.2.2 Applications of Empirical σ° to Distributed Targets 57
1.4.2.3 Accuracy of Empirical σ° Data 59
1.4.3 Validity of the Data Base Concept 61
1.4.4 Validity of the Radar Receiver Model 64
2.0 SLAR SIMULATION AND VALIDATION 67
2.1 Work Plan 67
2.2 SLAR Validation Data Base 70
2.3 SLAR Simulation Model 71
2.4 Microwave Reflectivity Data 74
2.5 SLAR Validation Scenes 77
2.6 Results 77
2.7 Conclusions 84
3.0 PPI SIMULATION AND VALIDATION 86
3.1 Work Plan 89
3.2 Data Base for Simulations 91
3.3 PPI Simulation Model 92
3.4 Terrain Return Data 94
3.5 Simulated Radar Image Formation 96
3.6 Results and Validation 97
4.0 REFERENCE SCENE SIMULATION: QUANTITATIVE TEST 105
4.1 Work Plan 111
4.2 Reference Scene Data Base 114
4.3 Reference Scene Simulation Model 115
4.4 Terrain Return Data 124
4.5 Reference Scene Formation 126
4.6 Results: Reference Scenes 129
5.0 CONSTRUCTION OF DATA BASES/FEATURE EXTRACTION TECHNIQUES 131
5.1 Background and Description 131
5.1.1 Data Base Definition 131
5.1.2 Background 136
5.2 Description of Pickwick Data Bases 139
5.2.1 Reference Scene Data Base 139

5.2.2 Model Validation Data Base 144
5.3 Source Intelligence Analysis 147
5.3.1 Optical Intelligence Data 147
5.3.2 Radar Intelligence Data 148
5.3.3 Source Data Analysis Results 152
5.4 Interactive Feature Extraction 152
5.4.1 Background 152
5.4.2 Combining the Computer and the Human 154
5.4.3 Use of System in Data Base Update 156
5.4.4 General Specifications 157
5.4.5 Rationale for Interactive Data Base Construction 158
5.4.6 An Interactive Feature Extraction System: A Design Concept 160
5.4.7 Conclusions and Recommendations 162
6.0 CONCLUSIONS 164
6.1 Reference Scene Generation 164
6.2 SLAR and PPI Validation 165
6.3 Data Bases 166
6.4 Optimal Choice of Source Intelligence for Data Base Construction 166
6.5 Interactive Feature Extraction 166
6.6 Backscatter Data 167
6.7 Related Areas of Application 167
7.0 RECOMMENDATIONS 169
7.1 Long Range Goals 169
7.1.1 Operational Implementation for Image Simulation 170
7.1.2 Interactive Feature Extraction 171
7.1.3 Microwave Reflectivity Catalog 172
7.1.4 Advanced Modeling: Distributed Targets 174
7.1.5 Advanced Modeling: Cultural Targets 175
7.1.6 Temporal (Seasonal) Changes 176
7.1.7 Terminal Studies and Future Guidance Systems 177
7.1.8 Image Quality Measurements for General Application 179
7.1.9 Sensitivity Analysis 179

7.1.10 Data Compression Techniques 180
7.1.11 Theoretical Models 180
7.1.12 Real and Near Real Time Simulation Updating 18
7.1.13 Further Development of Radar Image Simulation Applications 182
7.1.14 Other Sensor Systems 182
7.2 Short Term Goals 183
7.2.1 Terminal Guidance Studies 183
7.2.1.1 Seasonal Variations 186
7.2.1.2 Alternate Approach to Model Ground Return 188
7.2.1.3 Step-Wise Resolution Degradation Study 189
7.2.1.4 Parameter Optimization Study 190
7.2.1.5 Evaluate Guidance Concepts 192
7.2.1.6 Field Reference Scene Update 194
7.2.1.7 Reference Scene Technology Evaluation 195
7.2.1.8 Development of the Area Spatial Filtering Technique for PPI 195
7.2.2 Data Base Studies 196
7.2.2.1 Interactive Feature Extraction 197
7.2.2.2 Rapid Data Base Update 198
7.2.2.3 Data Compression Techniques 199
7.2.2.4 Source Intelligence Analysis 199
7.2.3 Simulation Problems 200
7.2.3.1 Comprehensive Compilation of Backscatter Data 200
7.2.3.2 Evaluate "Optical" Simulation Technique 201
7.2.3.3 Hybrid Radar Simulation Study 201
7.2.3.4 Evaluate Digital Filtering Techniques for "Optical" Simulation 202
7.2.4 Extending the Simulation Frontiers 203
7.2.4.1 Application of Simulation Technology 203
7.2.4.2 Receiver Transfer Function 205
7.2.4.3 Automatic Gain Control (AGC) Effects 205
7.2.4.4 Synthetic Aperture Radar (SAR) Image Simulation Model 206
ABBREVIATIONS AND ACRONYMS 207
REFERENCES 208

Table of Contents

Volume II ETL TR-0118

ABSTRACT ii

PREFACE iii

PURPOSE viii

SCOPE ix

TABLE OF CONTENTS x

LIST OF FIGURES xix

LIST OF TABLES xii

COMPUTER PROGRAMS xxii

APPENDIX A: "Construction of a Geometric Data Base for Radar Image Simulation Studies,"(TR 319-1), E. Komp, V. Kaupp and J. Holtzman. July, 1976.

ABSTRACT A-2

INTRODUCTION A-3

METHODOLOGY A-8

FLOW CHART A-10

APPENDIX B: "Baseline of Planimetric Data Base Construction: Pickwick Site," (TR 319-2), E. Davison, V. Kaupp and J. Holtzman. July, 1976.

ABSTRACT B-2

INTRODUCTION B-3

SITE DESCRIPTION B-4

INPUT DATA SOURCES B-5

FEATURE EXTRACTION B-9

BASELINE CONSTRUCTION B-12

APPENDIX C: "Digital Elevation Data Base Construction: Pickwick Site," (TR 319-3), M. McNeil, V. Kaupp and J. Holtzman. July, 1976.

ABSTRACT C-2

INTRODUCTION C-3

SITE DESCRIPTION C-5

INPUT DATA SOURCE C-5

ELEVATION DATA MERGE C-11

VERIFICATION C-13

TIME AND RESOURCES USED C-14

CONCLUSIONS C-14

APPENDIX D: "Digitization of Pickwick Site Data Base," (TR 319-4), M. McNeil, E. Davison, V. H. Kaupp and J. C. Holtzman. February, 1977.

ABSTRACT D-2

INTRODUCTION D-3

SITE DESCRIPTION D-4

PLANIMETRY VALIDATION D-4

DIGITIZATION OF PLANIMETRY D-5

PROBLEMS D-15

VALUE OF THE FINISHED PRODUCT D-16

APPENDIX E: "Increased Resolution of Planimetric Data Base: Pickwick Site," (TR 319-21), E. Davison, V. H. Kaupp, and J. C. Holtzman. March, 1977.

ABSTRACT E-2

INTRODUCTION E-3

2.0 SITE DESCRIPTION E-6

2.1 Planimetry Categories E-7

2.2 Input Data Sources E-7

2.3 Feature Extraction Techniques E-8

3.0 CONCLUSION E-11

APPENDIX F: "Medium Resolution Digital Ground Truth Data Base: Pickwick Site," (TR 319-5), E. Komp, M. McNeil, V. H. Kaupp, and J. C. Holtzman. August, 1977.

ABSTRACT F-2

INTRODUCTION F-3

DATA PROCESSING PHILOSOPHY F-4

DIGITAL DATA MATRIX CONSTRUCTION F-10

DATA MATRIX QUALITY VERIFICATION F-25

CULTURAL TARGETS F-26

MERGE OF ELEVATION DATA WITH RADAR CATEGORY DATA F-27

APPENDIX G: "Backscatter Data for Digital Radar Image Simulations," (TR 319-7), J. L. Abbott, R. L. Martin, V. H. Kaupp, and J. C. Holtzman. February, 1977.

ABSTRACT G-2

1.0 INTRODUCTION G-3

2.0 DIELECTRIC BEHAVIOR G-5

2.1 Extension of Sigma Zero Data G-9

Soil G-10

Cultivated Vegetation G-11

Gravel, Sand, Grass G-11

2.2 Target Classification G-12

3.0 PICKWICK CATEGORIES G-13

4.0 RESULTS G-16

5.0 CONCLUSIONS G-19

BACKSCATTER DATA (TR 319-7) G-20

APPENDIX H: "Medium Resolution Radar Image Simulation of Deciduous Forests: A Study of Candidate Techniques," (TR 319-9), J. L. Abbott, V. H. Kaupp and J. C. Holtzman. August, 1977.

ABSTRACT H-2

1.0 INTRODUCTION H-3

2.0 ANALYSIS OF SIMULATED FOREST REGIONS H-4

3.0 RESULTS H-6

4.0 CONCLUSIONS H-7

APPENDIX I: "Digital Model for Radar Image Simulation and Results," (TR 319-8), R. L. Martin, J. L. Abbott, M. McNeil, V. H. Kaupp, J. C. Holtzman. August, 1976.

ABSTRACT I-2

1.0 INTRODUCTION I-3

2.0 RADAR IMAGE SIMULATION THEORY I-6

3.0 SLAR SIMULATION PROCESS I-14

3.1 Input Parameters I-15

3.2 Data Base I-15

4.0 DIGITAL SLAR SIMULATION RESULTS I-20

5.0 CONCLUSIONS AND RECOMMENDATIONS I-29

APPENDIX J: "Digital PPI Model for Radar Image Simulation and Results," (TR 319-19), E. Komp, V. H. Kaupp and J. C. Holtzman. August, 1977.

ABSTRACT J-2

1.0 INTRODUCTION J-3

2.0 RADAR SIMULATION THEORY J-4

3.0 PPI SIMULATION PROCESS J-7
3.1 Creation of Polar Data Base J-7
3.1.1 Ground Truth Data Base J-8
3.1.2 Radar Input Parameters J-8
3.1.3 Program Execution J-9
3.2 Greytone Calculation J-10
3.2.1 Calculation of Resolution Cell Size J-12
3.2.2 Calculation of Local Angle of Incidence J-13
3.2.3 Shadow J-13
3.2.4 Layover J-17
3.2.5 Backscatter Data J-18
3.2.6 Summary of Greytone Calculation J-18
3.3 Formation of Output Image J-18
4.0 SOFTWARE IMPLEMENTATION OF PPI SIMULATION MODEL J-20
4.1 Application of PPI Simulation Implementation to Terminal Guidance J-21
APPENDIX K: "Investigation of an Interactive Approach for Radar Image Simulation," (TR 319-15), V. S. Frost, J. L. Abbott, V. H. Kaupp, and J. C. Holtzman. March, 1977.
ABSTRACT K-2
1.0 BACKGROUND K-3
2.0 THE INTERACTIVE FEATURE EXTRACTION CAPABILITIES K-6
3.0 THE INTERACTIVE CONCEPT K-11
4.0 CONCLUSIONS K-12
APPENDIX L: "Automated Techniques in Feature Extraction," (TR 319-24), M. McNeil, V. H. Kaupp and J. C. Holtzman. March, 1977.
ABSTRACT L-2
1.0 INTRODUCTION L-3
2.0 TEMPLATE MATCHING L-5
3.0 EDGE DETECTION L-7
4.0 CLUSTERING AND REGION FINDING L-10
5.0 TRACKING AND CONTOUR FOLLOWING L-12
6.0 REPRESENTATION OF THE EXTRACTED FEATURES L-13
7.0 CONCLUSIONS AND RECOMMENDATIONS L-14

APPENDIX M: "Interactive Feature Extraction System Framework," TR 319-25, M. McNeil, V. H. Kaupp and J. C. Holtzman. March, 1977.

ABSTRACT M-2

1.0 MOTIVATION M-3

2.0 GENERAL REQUIREMENTS M-5

3.0 INPUT M-6

4.0 OUTPUT M-8

5.0 PHYSICAL REQUIREMENTS M-9

6.0 SUBSYSTEM M-10

7.0 SYSTEM BREAKDOWN M-11

8.0 CONCLUSIONS M-13

APPENDIX N: "Image Handling and Processing," (TR 319-26), M. McNeil, V. H. Kaupp and J. C. Holtzman. March, 1977.

ABSTRACT N-2

1.0 INTRODUCTION N-3

2.0 IMAGE STORATE N-4

2.1 Physical Storage, I/O Transfer Time N-4

2.2 Reconstructive Processing N-4

2.3 Tradeoff Between Compactness and Reconstruction N-4

2.3.1 No compaction N-5

2.3.2 Extreme Compaction N-5

2.3.3 Optimum Balance N-6

2.4 Sequential and Random File Structures N-9

2.5 Compatibility N-10

3.0 TRANSFER AND DISPLAY N-11

3.1 Display Devices N-11

3.2 Transfer Time N-11

4.0 IMAGE MANAGEMENT N-13

4.1 Image Control Blocks N-13

4.2 Image Description Records N-13

5.0 MANIPULATION OF IMAGES N-15

5.1 Whole Image Manipulation N-15

5.2 Image Editing and Modification N-15

6.0 CONCLUSIONS AND RECOMMENDATIONS N-17

VOLUME I
LIST OF FIGURES

		Page
Figure I.	Fundamental Block Diagram - Radar System	xxix
Figure II.	Conceptual Model of SLAR	xxx
Figure III.	Fundamental Block Diagram - Simulation	xlvii
Figure 1.	Conceptual Block Diagram and Flow of Work Performed	8
Figure 2.	Flat Earth Geometry	39
Figure 3.	Azimuth Geometry - Slope	40
Figure 4.	Pulse-Length Resolution	42
Figure 5.	Range Geometry - Slope	43
Figure 6.	Range Resolution	47
Figure 7.	Azimuth Resolution	49
Figure 8.	Resolution Cell with Arbitrary Slope (Plane Facet)	53
Figure 9.	297° Heading Data Base Swath	76
Figure 10.	116° Heading Data Base Swath	76
Figure 11.	Real + Simulated Radar Images @ 297° Heading	81
Figure 12.	Real + Simulated Radar Images @ 116° Heading	82
Figure 13.	Area of PPI Image	100
Figure 14.	297° Heading PPI Results	102
Figure 15.	116° Heading PPI Results	103
Figure 16.	Conceptual View of Correlatron	107
Figure 17.	Work Plan - Terminal Guidance Task	112
Figure 18.	Special PPI Image Format	118
Figure 19.	Reference Scene Image Format	119
Figure 20.	Effect of Look Direction Error	121
Figure 21.	Conceptual Model of Reference Scene Format	127
Figure 22.	Reference Scene: Pickwick Site, Test 1	132
Figure 23.	Reference Scene: Pickwick Site, Test 2	132
Figure 24.	Reference Scene: Pickwick Site, Test 3	133
Figure 25.	Reference Scene: Pickwick Site, Test 4	133
Figure 26.	Reference Scene: Pickwick Site, Test 5	134
Figure 27.	Elevation Data Base	143
Figure 28.	Planimetry Data Base	143

VOLUME II
LIST OF FIGURES

Figure A-1	Data Base of Geometric Solids	A-5
Figure A-2	Relative Position and Elevation	A-6
Figure A-3	Microwave Reflectivity Category Assignments	A-7
Figure A-4	Geometric Data Base Flow Chart	A-11
Figure B-1	Pickwick Dam Test Site in Relation to U.S.G.S. Topographic Sheets of the Area	B-8
Figure C-1	Elevation Data Orientation of Pickwick Dam Test Site in Relation to U.S.G.S. Topographic Sheets of the Area	C-4
Figure C-2	Orientation of U.S.G.S. Map in UTM at Pickwick	C-6
Figure C-3	Elevation Data Orientation	C-7
Figure C-4	Scan Line Orientation	C-9
Figure C-5	Constant E-W Line of Orientation	C-10
Figure C-6	Orientation of Merged Elevation Data Matrix for Pickwick Site	C-11
Figure D-1	Misregistration of Data	D-6
Figure D-2	Comparison of Boundaries in Two Coordinate System	D-10
Figure F-1	Category Map With Vertical Scan Line	F-6
Figure F-2	Example of Multiple Categories Enclosing a Homogeneous Region	F-7
Figure F-3	Example of "Vertex" Point	F-9
Figure F-4	Continuous Boundary Lines	F-9
Figure F-5	Horizontal Tangent Points	F-14
Figure F-6	Resolution of the Horizontal Tangent Problem Depicted in Fig. 5	F-16
Figure F-7	Example of "Double Tangent" at x=5, x=6	F-16
Figure F-8	Multiple Crossings of a Vertical Scan Line	F-19
Figure F-9	Vertical Boundary	F-19
Figure F-10	Anomalous Vertical Line Segments	F-21
Figure F-11	Data Packing Scheme	F-28
Figure G-1	Real Part and Imagery Part of Dielectric Constant of Water as a Function of Frequency for Various Temperatures	G-9
Figure G-2	Side by Side Comparison of APD -10 Imagery and PSM Simulated Imagery	G-17
Figure I-1	Macro-flow Chart of SLAR Simulation Process	I-14
Figure I-2	Simulation Parameters	I-16

Figure I-3 Data Base of Geometric Solids I-1
Figure I-4 Relative Position and Elevation I-15
Figure I-5 Microwave Reflectivity Category Assignments I-19
Figure I-6a 20° Depression Angle, 40,000 feet altitude I-21
Figure I-6b 40° Depression Angle, 40,000 feet altitude I-21
Figure I-7 Radar Simulation Study of Geometric Data Base I-22
Figure I-8 Flight Geometries for Equal Altitude Test I-26
Figure I-9 Shadow and Layover Difference I-26
Figure I-10 Radar Simulation of Artificial Data Base 20° Depression Angle 40,000 ft. SLAR Altitude I-27
Figure I-11 Radar Simulation of Artificial Data Base 24° Depression Angle 40,000 ft. SLAR Altitude I-27
Figure J-1 Tilting of a Resolution Cell J-14
Figure J-2 Radar Image Shadow J-16

VOLUME I
LIST OF TABLES

		Page
Table 1.	Radar System/Ground Return Interaction Parameters	56
Table 2.	PPI Parameters for 116° Flight Heading	98
Table 3.	PPI Parameters for 297° Flight Heading	98

VOLUME II
LIST OF TABLES

Table A-1	Geometric Solids	A-9
Table B-1	Pickwick Test Site Planimetry Categories Validation and Terminal Guidance Tasks	B-6
Table B-2	Planimetric Data Base Construction Time for the Pickwick Site	B-13
Table D-1	Four Corner Points of Map on Scan Line (x,y) Coordinates Taken at Two Different Digitizing Sessions	D-14
Table E-1	Pickwick Test Site Planimetry Categories[2]	E-5

COMPUTER PROGRAMS

APPENDIX	TR	TITLE	PURPOSE	
A	319-1	FAKE	Creates a geometric data base for radar image simulation	A-13
I	319-8	SLAR	Produces Side Looking Radar simulations	I-30
J	319-19	SLICE	Produces a polar data base	J-22
		VERIF	Generates simulations from polar data base	J-28
		STEP 1	Creates resolution cell size matrix in polar coordinates	J-35
		STEP 1B	Part two of data base creation	J-40

APPENDIX A

CONSTRUCTION OF A GEOMETRIC DATA BASE FOR RADAR IMAGE SIMULATION STUDIES

The following technical report (TR 319-1) prepared by the Center for Research, Inc., University of Kansas, is included in this volume to provide details in support of the technical discussions of Volume I.

THE UNIVERSITY OF KANSAS CENTER FOR RESEARCH, INC.

2291 Irving Hill Drive Campus West
Lawrence, Kansas 66045

Telephone: 913-8644832

CONSTRUCTION OF A GEOMETRIC DATA BASE FOR RADAR IMAGE SIMULATION STUDIES

Remote Sensing Laboratory

RSL Technical Report 319-1

E. Komp

V. Kaupp

J. Holtzman

July, 1976

Supported by:

U. S. Army Engineer Topographic Laboratories
Fort Belvoir, Virginia 22060
CONTRACT DAAG 53-76-c-0154

ABSTRACT

A computer program to produce a digital ground truth data base of a controlled scene of geometric solids has been developed for radar image simulation studies. The data bases which can be made by this program have well-defined geometric properties and have well-defined boundaries separating different microwave reflectivity categories. Since the geometric properties of the features in a scene are of paramount importance to radar, a test data base with known geometry is crucial in radar image simulation work. Also important to radar image simulation work is the capability to easily change the reflectivity categories of the various regions contained in a scene. All of these features are available in the data bases which can be produced by this program. These data bases are ideally suited, therefore, to radar image simulation studies; they provide a known input to the simulation program and produce a calculable response.

INTRODUCTION

A software package to produce a digital ground truth data base of a controlled scene of geometric shapes and solids has been developed for radar image simulation studies. The data base produced by this computer program provides to the simulation program a known input with well-defined interactions and relationships between the various features contained in the scene. Uncertainties in the geometric and dielectric properties of the features inherent in a radar scene and, thus, in digital data bases of real terrain, are removed. The data bases which can be developed from this program are, therefore, ideally suited to provide a known calibrated input to radar image simulation studies; they have a calculable response, and, hence, can be thought of as a standard scene or as a "test" scene.

In a broad sense, there are two classes of radar image simulations that can be produced: The first is the class of simulations where the simulated radar imagery is the goal, and the second is the class where the imagery is not the goal. The simulations of the first class are produced, typically, to reproduce faithfully the radar imagery characteristics of a particular scene based on the real properties of that scene. For this class of simulations it is necessary to generate the appropriate ground truth data base for each scene. The applications of the simulated imagery produced in the second class are independent of the scene itself. Some examples of simulation where the radar image is not the goal are found in troubleshooting software, in sensitivity studies, in parameter optimization studies, etc. A standard calibrated data base with known input characteristics would be ideal for this class of simulation. Typically, however, an existing data base produced for some other application is used. This approach has the problem that it is difficult (if not impossible) to determine the origin of the results produced in the simulation (e.g. anomalies might be caused either by the data base or the study). In fact, experience has shown that using existing data bases to troubleshoot new software, to determine thresholds of sensitivity for

various simulation parameters, to evaluate discrimination capabilities, or for other similar studies is inefficient, can mask results, and can lead to incorrect interpretations of results. A controlled data base of known geometry, dielectric categories and features, and having a calculable response is required for studies of this kind.

Among the most difficult aspects of the radar image simulation problem is building the digital data bases containing the geometry of the ground features and the microwave reflectivity properties of the ground scene. Some of the more common sources of data used to build these data bases are high resolution aerial photography, maps, and other radar imagery at, perhaps, a different wavelength and polarization. It is necessary to extract the desired information and convert these raw data into a suitable digital data base (ground truth data base) for input into a digital computer. Typically, a scene will require a large number of points in the ground truth data base. For example, if the radar system being modeled is capable of resolving features (objects) which are separated by 100 feet or more, a 10 mile by 10 mile scene would contain at least 279,000 independent elements. (50 foot resolution would require more than a million elements.) Production of simulated images of scenes of a reasonable size can require the development of a very large ground truth data base.

Several geometric data bases have been produced by the program described here. A specific example is shown in Figures 1, 2, and 3. Figure 1 shows a perspective line drawing of this data base. This figure illustrates the relative orientation of the various geometric shapes included. Figure 2 is a plan view of the same data base showing the relative heights and positions of the various solids. Figure 3 is also a plan view of this data base, but this figure illustrates the microwave reflectivity category assignments made for this one data base. This data base has been used to troubleshoot the image simulation software and to verify that layover, shadow, range compression, local angle of incidence, and fading were all properly simulated. It has also been used to demonstrate visually the effects of changing carrier frequency and polarization for a well-defined scene. It is expected that this data base will be invaluable in future studies.

Figure 1. Data Base of Geometric Solids.

Figure 2. Relative Position and Elevation.

LEGEND

Backscatter Category Number	Backscatter Category Description
1	Wheat Stubble
2	Alfalfa, Tall
3	Alfalfa, Short
4	Corn, Short
5	Corn, Tall
6	Milo, Short
7	Milo, Tall
8	Soybeans, Short
9	Soybeans, Tall
10	Green Wheat

Figure 3. Microwave Reflectivity Category Assignments

METHODOLOGY

The software design for the construction of the various geometric solids was made as simple as possible. The various geometric solids which can be selected for inclusion in a data base to be constructed by this program are available as subroutines. The objects available as subroutines and the options for user specification are listed in Table 1. This program creates a data base as a grid matrix of a specified size. (Present operation creates 1000 x 1000.) Cell size is user-specified, but must be square.

Within this grid matrix, 3-dimensional representations of regular surfaces are constructed by user specifications (details of available structures to follow). In addition, a specific microwave reflectivity category may be assigned to each object. The elevation and category assignment (i.e. reflectivity category) for each cell is packed into one word; the rightmost six bits are reserved for category number (giving possible range of values 0 → 63), and the remaining upper bits are reserved for the elevation. Those cells where no object is defined are implicitly defined with Ø elevation and Ø category.

This computer program will produce data bases as large (or as small) as is desired. To prevent problems caused by computer core limitations, the data base is constructed in horizontal strips. In the first pass, all the columns of the first N rows of the grid matrix are constructed and stored on an external device; pass 2 produces the next N rows; pass 3 the next N rows, etc., until the matrix is complete. N is chosen by the user so that (N ROWS * M COLUMNS) is a "reasonable" amount of core requirements. For the specific data base herein described (Figures 1, 2, and 3), a 1000 x 1000 matrix was produced in 20 passes. Each pass produced a section of 50 rows x 1000 points which required a single block of 50K words of core (compared to 1000K which would be required if the entire matrix were constructed in one pass). The multiple pass concept does not sacrifice execution speed since only one object can be constructed at a time. And since the output is performed on entire lines, the data base is built as a single unit, eliminating the complexities of mosaicking

TABLE I

Geometric Solids

Solid	Options
Cube	User supplies start row and start column, length and width of base, the angle of rotation in degrees, and a microwave reflectivity category number assignment.
Wedge	User specifies start row and start column, length of wedge, maximum height of the wedge, slope of the front face, width of the flat top, slope of the back face, angle of rotation, and a microwave reflectivity category number assignment.
Pyramid	Produces a pyramid of arbitrary height with a square base. User specifies start row and column, length of base, angle of rotation, and microwave reflectivity category number. (Slope of the sides is implicitly determined by the height and length of the side.)
Hemisphere	User specifies row and column of center, radius, and microwave reflectivity category number.
Tower	Produces a cylinder standing on end. User specifies start row and column of center, radius, length, and microwave reflectivity category number.
Cylinder	Produces half of a cylinder lying on its side. User specifies start row and column of center line, radius, length, and microwave reflectivity category number.
Patch	Assigns to a given number of matrix cells the desired microwave reflectivity category number with no elevation.
Checkerboard	Assigns to a series of matrix cells each of the different microwave reflectivity categories consecutively with no elevation.

Note:

Patch and checkerboard are available to compare changes in the return strength of the various categories at different incident angles. Patch may provide a reference greytone next to an object to analyze the effect of changing the local incident angle.

sub-images together. The one sacrifice that is made, however, is that the objects produced must lie completely within a single strip; an object cannot be constructed that begins someplace within one horizontal strip and overlaps to some point within another strip. Aside from this minor limitation, the computer program is general in design.

FLOW CHART

The computer program (geometric data base construction) has been designed to be as versatile as possible. The size of the data base and the size, location, orientation, and shape of the various geometric solids are easily controlled. Figure 4 presents the basic operations, in flow chart form, of this computer program. Digital data bases of geometric solids are constructed according to the user's design and specification. Also shown is a listing of the eight geometric shapes available as subroutines in this program. A copy of the program is included in this work.

To use this program, it is necessary first to design the desired data base. The size, location, orientation, and shape of each geometric shape must be specified. The computer program will produce a digital data base matrix constructed of 1000 elements by 50 rows at a time for as many repetitions as are desired. (A matrix of 1000 x 1000 data values would require 20 repetitions.) If it is desired to construct a data base of a different size, the dimension statements must be changed from the values specified (specified in the program listed in Appendix I) to the desired values. The only real constraints on data base size are imposed by the core size of the computer used to construct the data base. After the data base size has been defined and the dimension statements of the program have been specified correctly, the data base can be constructed. Reference to the flow chart shown in Figure 4 will make clearer the following discussion. For each geometric shape to be included in each horizontal strip, two data cards are required to be input to the program. The first card is a number between Ø and 9 that specifies the surface to be constructed, and the second card specifies

Geometric Solid	Variable Choose	Software Name
Patch	1.	Patch
Cube	2.	Cube
Wedge	3.	Wedge
Pyramid	4.	Pyramid
Hemisphere	5.	Hmisp
Cylinder	6.	Cyl
Tower	7.	Tower
Checkerboard	8.	Chkrbd

Figure 4. Geometric Data Base Flow Chart

the required construction data. (These requirements are listed both in Table 1 and in the various subroutine listings). Two cards are required for the program for each shape desired in a strip. A data card of Ø signifies that no more geometric solids are desired for a given strip and that construction is to begin for the next strip. In this way, the data base is constructed strip by strip until the data base is completed.

The data bases which can be constructed from this program are not likely to be encountered in actual radar use. But they can be very useful for testing radar simulation programs and problems since all aspects--height, local slope, category assignment--are precisely known. Furthermore, any or all of these parameters can be changed by the user in a very controlled manner to study the relative effects and importance of each, as well as to evaluate the simulation techniques. These changes can be accomplished either by constructing a new data base for each arrangement or by incorporating this program in the simulation model to produce the desired effect. One can also study the importance of the size of the data base cells by changing the appropriate parameter during the production of a data base. There are many more potential uses for a data base like this, and to increase the value of this program by making the output more generally applicable, facilities to rotate the data base to run radar simulations with different flight paths have been included.

Listing of Computer Program to Construct
Geometric Data Bases

PROGRAM TITLE: FAKE

AUTHOR: E. Komp

IMPLEMENTED ON: Honeywell 635

PURPOSE:

Creates a grid matrix of elevation, geometry, and reflectivity for input to a radar image simulation program.

```
C                          FAKE
C
C
C   THIS PROGRAM CREATES AN ARTIFICIAL DATA BASE CONSISTING
C   OF REGULAR 3-DIMENSIONAL GEOMETRICAL SHAPES AS SPECIFIED
C   BY THE USER
C
C   THE DATA BASE IS CONSTRUCTED IN RECTANGULAR FORMAT.
C   EACH CELL REPRESENTS A FIXED SQUARE AREA, THE LENGTH OF
C   THE SIDE OF EACH CELL IS DETERMINED BY THE VARIABLE --
C   CELSIZ
C   EACH CELL IS CHARACTERIZED BY AN ELEVATION AND A CATEGORY
C   THIS INFORMATION IS PACKED INTO ONE WORD.
C   THE LOW ORDER 6 BITS ARE RESERVED FOR THE CATEGORY,
C   THE HIGH ORDER BITS FOR THE ELEVATION
C
C   TO USE THIS PROGRAM THE USER SPECIFIES THE OBJECTS DESIRED
C   ALONG WITH THE NECESSARY DESCRIPTIVE PARAMETERS AND THE
C   COORDINATES OF THE LOWER LEFT CORNER FOR PLACEMENT IN THE
C   DATA BASE ( OR THE CENTER FOR CYLINDRICAL AND SPHERICAL
C   OBJECTS) PLUS THE DESIRED CATEGORY
C
C   ALL UNSPECIFIED CELLS HAVE IMPLICIT ELEVATION AND CATEGORY 0
C
C   TO MINIMIZE CORE REQUIREMENTS THE DATA BASE IS CONSTRUCTED
C   IN STRIPS 50 CELLS WIDE (THIS NUMBER MAY BE VARIED BY
C   ALTERING THE DIMENSION STMT).  THE FIRST FIFTY ROWS ARE
C   CONSTRUCTED AND PUT ON TAPE WHEN COMPLETE.  THEN THE NEXT
C   FIFTY ROWS ARE CONSTRUCTED AND PUT ON TAPE, AND SO ON
C   THE USER SPECIFIES THE COMPLETION OF A STRIP ON INPUT
C   BY INPUTTING 0 FOR CHOOSE
C   THEREFORE AN ARBITRARY NUMBER OF ROWS MAY BE
C   CONSTRUCTED.  THE NUMBER OF COLUMNS IS LIMITED BY THE
C   DIMENSIONS OF THE ARRAY MATRIX (IN THIS CASE 1000)
C
C   OJECTS MAY NOT CROSS OVER THE BOUNDARIES OF STRIPS
C   EACH OBJECT SPECIFIED MUST FIT COMPLETELY INSIDE THE
C   STRIP
C
C
      IMPLICIT INTEGER (A-W)
C
C   THE FOLLOWING VARIABLES ARE REQUIRED BY MOST OR ALL OF
C   THE SUBROUTINES AND SO ARE PLACED IN BLANK COMMON AND SO
C   DO NOT HAVE TO BE PASSED AS FORMAL PARAMETERS
C
C   BUF - SMALL I/O BUFFER FOR DEBUG WRITE STMT IN VARIOUS
C         SUBROUTINES
C   MATRIX - THAT PORTION OF THE DATA BASE BEING CONSTRUCTED
C         THAT IS PHYSICALLY IN CORE
C   CELSIZ - SIZE OF EACH CELL IN DATA BASE (IN FEET)
C   STROW,STCOL - STARTING ROW AND COLUMN OF THE SURFACE TO BE
```

```
C          CONSTRUCTED (FOR RECTANGULAR OBJECTS THIS IS LOWER LEFT
C          CORNER; FOR CYLINDRICAL OBJECTS THE CENTER)
C          ***NOTE*** STROW MUST BE GIVEN MODULO 50 (SINCE ONLY A
C          50 ROW STRIP OF THE ENTIRE DATA BASE IS IN CORE
C          AT ANY ONE TIME)
C   CAT - CATEGORY ASSIGNMENT FOR THE SURFACE
C   ANG = ANGLE OF ROTATION (APPLICABLE ONLY TO RECTANGULAR SHAPES)
C          THE OBJECT WILL BE ROTATED COUNTER-CLOCKWISE ABOUT ITS
C          LOWER LEFT CORNER THE SPECIFIED NUMBER OF DEGREES
C
C
        COMMON BUF(20), MATRIX(1000,50),CELSIZ,STCOL,STROW,CAT,ANG
        DIMENSION LINE(500)
C
        CELSIZ =20
        HALFCEL = CELSIZ/2
C
        DO 620 I=1,70
        DO 620 J=1,500
 620    MATRIX(J,I)=10
C
C
C   CONSTRUCTION OF EACH SHAPE REQUIRES TWO DATA CARDS
C   FIRST CARD IS A NUMBER BETWEEN 0 AND 9 (I2 FORMAT)
C   THAT SPECIFIES THE SURFACE TO BE CONSTRUCTED
C   C - BEGIN NEW STRIP OF DATA BASE
C   1 - PATCH
C   2 - CUBE
C   3 - WEDGE
C   4 - PRYAMID
C   5 - HEMISPHERE
C   6 - CYLINDER
C   7 - TOWER
C 8 8 - CHECKERBOARD
C
C   SECOND CARD NECESSARY DESCRIPTIVE PARAMETERS FOR THAT
C   OBJECT  -- SEE SPECIFIC ROUTINE FOR DETAILS
C
C   ***NOTE***   STROW AND STCOL ARE TO BE SPECIFIES IN TERMS
C                OF MATRIX CELLS (ROW,COL) - ROW MODULO 50 -
C   FOR EASY PLACEMENT OF OBJECTS RELATIVE TO ONE ANOTHER
C      ALL OTHER PARAMETERS (LENGTH,WIDTH, HEIGHT) ARE TO
C   BE GIVEN IN FEET SO THE USER CAN EASILY DESCRIBE THE
C   PHYSICAL SIZE DESIRED
C
C
  1     READ(05,05,END=900) CHOOSE
  5     FORMAT(I2)
C
        IF(CHOOSE .EQ. 0) GO TO 500
        GO TO (10,20,30,40,50,60,70,80), CHOOSE
C
```

```
C   BAD INPUT  OUTPUT ERROR MESSAGE AND READ NEXT CARD
C
      WRITE(6,810) CHOOSE
 810  FORMAT(//,' ***WARNING***',I5,' IMPROPER VALUE TO
     1 SPECIFY NEW OBJECT.  IT HAS BEEN IGNORED'//)
C
C   SKIP CARD WITH PARAMETERS FOR BAD OBJECT
C
      READ(05,05,END=900) CHOOSE
      GOTO 1
C
C
C                    PATCH
C
C   PATCH - SQUARE AREA OF DATA BASE ASSIGNED SPECIFIED CATEGORY
C         NO ELEVATION IS ASSIGNED
C
C   SIZE - DESIRED LENGTH OF SIDE FOR THE SQUARE AREA
C
C
  10  READ(05,11) STROW,STCOL,SIZE,CAT
  11  FORMAT(10I5)
      CALL PATCH(SIZE)
      CONTINUE
      GO TO 1
C
C                     CUBE
C
C   CUBE - RECTANGULAR SHAPE OF ARBTRAY CONSTANT ELEVATION
C
C   LEN - LENGTH IN X DIRECTION (ACROSS)
C   WID - WIDTH IN Y DIRECTION (VERTICAL)
C   HGT - HEIGHT CONSTANT ELEVATION TO BE ASSIGNED EACH CELL
C
C
  20  READ(05,11) STROW,STCOL,ANG,LEN,WID,HGT,CAT
      CALL CUBE(LEN,WID,HGT)
      CONTINUE
      GO TO 1
C
C                    WEDGE
C
C   WEDGE - OBJECT WITH SLOPING FRONT FACE, A FLAT TOP
C         OF CONSTANT ELEVATION AND ARBITRARY WIDTH AND A SLOPING BACK
C         FACE
C
C   LEN - LENGTH OF WEDGE IN X DIRECTION (ACROSS)
C   SLOPE - SLOPE OF FRONT FACE
C   HGT - MAXIMUM ELEVATION OF WEDGE
C   TOP - WIDTH OF FLAT TOP
C   BSLOPE - SLOPE OF BACK FACE
C
```

```
C
  3C    READ(05,11) STROW,STCOL,ANG,LEN,SLOPE,HGT,TOP,BSLOPE,CAT
        CALL WEDGE(LEN,SLOPE,HGT,TOP,BSLOPE)
        CONTINUE
        GO TO 1
C
C                    PYRAMID
C
C  PRYAMID - REGULAR FOUR SIDED PYRAMID
C
C  LEN - SIZE OF BASE OF PYRAMID (WILL HAVE SQUARE BASE)
C  HGT - HEIGHT OF PYRAMID
C
C
  4C    READ(05,11) STROW,STCOL,ANG,LEN,HGT,CAT
        CALL PRYAMID(LEN,HGT)
        CONTINUE
        GO TO 1
C
C                    HEMISPHERE
C
C  HEMISPHERE - HEMISPHERE WITH CENTER AT (STROW,STCOL)
C
C  RAD - RADIUS
C
C
  5C    READ(05,11)STROW,STCOL,RAD,CAT
        CALL HEMISP(RAD)
        CONTINUE
        GO TO 1
C
C                     CYLINDER
C
C  CYLINDER - CYLINDER LYING ON ITS SIDE
C
C  RAD - RADIUS OF CYLINDER
C  LEN - LENGTH OF CYLINDER (IN X DIRECTION)
C
C
  6C    READ(05,11) STROW,STCOL,RAD,LEN,CAT
        CALL CYL(RAD,LEN)
        CONTINUE
        GO TO 1
C
C                TOWER
C
C  TOWER - CYLINDER STANDING ON END
C
C  RAD - RADIUS OF CYLINDER
C  HGT - HEIGHT OF CYLINDER
C
C
```

```
C
 70     READ(05,11) STROW,STCOL,RAD,HGT,CAT
        CALL TOWER(RAD,HGT)
        CONTINUE
        GO TO 1
C
C                CHECKERBOARD
C
C   CHECKERBOARD - ROW AND COLUMN OF PATCHES REPRESENTING THE
C          VARIOUS CATEGORIES (FOR COMPARISON OF RELATIVE
C          GREYTONES FOR THE CATEGORIES)
C
C
C
  80    READ(05,11) STROW,STCOL
        CALL CHKRBD
        CONTINUE
        GO TO 1
C
C
C   OUTPUT COMPLETED STRIP OF DATA BASE TO TAPE
C   IN BINARY UNFORMATTED FORM
C
 500    DO 600 I=1,50
        WRITE(01) (MATRIX(J,I),J=1,1000)
 600    CONTINUE
C
C   RESET ALL MATRIX CELLS TO 0 BEFORE BEGINNING NEXT
C   STRIP OF DATA BASE
C
        DO 610 I=1,50
        DO 610 J=1,1000
 610    MATRIX(J,I)  = 0
        GO TO 1
 900    STOP
        END
C       SUBROUTINE PATCH(SIZE)
        SUBROUTINE PATCH(SIZE)
C
        IMPLICIT INTEGER (A-W)
        COMMON BUF(20), ARRAY(500,70),CELSIZ,STCOL,STROW,CAT,ANG
C
C
C   SIZE DIMENSION CONVERTED TO NUMBER OF MATRIX CELLS
C
        SIZE = SIZE/CELSIZ
C
        DO 100 II=1,SIZE
        I = II-1
        DO 100 JJ=1,SIZE
        J =JJ-1
C
```

```
C   BR, BC - BIGGEST ROW AND COLUMN INCLUDED
C          THESE PARAMETERS REQUIRED FOR THE SUBROUTINE FILLIN
C
        IF(COL .LT. LC) LC = COL
        IF(COL .GT. BC) BC = COL
        IF(ROW .LT. LR) LR = ROW
        IF(ROW .GT. BR) BR = ROW
C
C                DEBUG WRITE STATEMENT
C
         GO TO 90
        I=I+2
        BUF(I)= COL
        BUF(I+1)= ROW
        IF(I .LT. 15) GO TO 90
        WRITE(6,60) (BUF(M), M=1,16)
 60     FORMAT(8(5X,2I5))
        I=-1
 90     FLD(18,12,ARRAY(COL,ROW)) = FLD(24,12,HGT)
C
C   PROPER ELEVATION AND CATEGORY IS ASSIGNED TO EACH OF THESE
C   CELLS
C
 100    FLD(30,6,ARRAY(COL,ROW)) = FLD(30,6,CAT)
C       WRITE(6,60) (BUF(M),M=1,16)
        DO 10 I=1,16
 10     BUF(I)=0
        CATEG = CAT
C
C
C   SINCE ROTATION WAS DONE ON DISCRETE POINTS IT IS POSSIBLE
C   THAT THERE ARE SOME HOLES (CELLS INTO WHICH NO ROTATED
C   POINT WAS PLACED) IN THIS OBJECT
C
C   SUBROUTINE FILLIN CHECKS OBJECTS FOR SUCH 'HOLES'
C   AND FILLS THEM IN WITH THE APPROPRIATE ELEVATION AND
C   CATEGORY
C
        CALL FILLIN(LC,BC,LR,BR,CATEG)
        CONTINUE
        RETURN
        END
C       SUBROUTINE WEDGE(LEN,SLOPE,HGT,TOP,BSLOPE)
        SUBROUTINE WEDGE(LEN,SLOPE,HGT,TOP,BSLOPE)
C
        IMPLICIT INTEGER (A-W)
        COMMON BUF(20), ARRAY(500,70),CELSIZ,STCOL,STROW,CAT,ANG
        DATA LC,BC,LR,BR/500,0,500,0/
C
        WRITE(6,65) STCOL,STROW
 65     FORMAT(/,30X,'WEDGE',5X,2I6,//)
C
```

```
C   APPROPRIATE CELLS ARE ASSIGNED THE DESIGNATED CATEGORY
C
 100    ARRAY(STCOL+I,STROW+J) = CAT
        RETURN
        END
C       SUBROUTINE CUBE(LEN,WID,HGT)
        SUBROUTINE CUBE(LEN,WID,HGT)
C
        IMPLICIT INTEGER (A-W)
        COMMON BUF(20), ARRAY(500,70),CELSIZ,STCOL,STROW,CAT,ANG
        DATA LC,BC,LR,BR/500,0,500,0/
C
C
        I=-1
C
        HALFCEL = CELSIZ/2
C
C   ANGLE OF ROTATION OF BASE OF CUBE
C
        ZANG = FLOAT(ANG)/57.295
C   SINE OF ANGLE OF ROTATION
        ZSIN = SIN(ZANG)
C   COSINE OF ANGLE OF ROTATION
        ZCOS = COS(ZANG)
C
C
C
C   THESE DO LOOPS CALCULATE THE CELLS THAT WOULD BE INCLUDED
C   IN THE CUBE IF THERE WERE NO ROTATION
C   IF THE OBJECT IS TO BE ROTATED (ANG =/ C)
C   A ROTATION OF AXES IS EFFECTED BY THE TRANSFORMATION
C       XNEW = X * COS(ANG) - Y * SIN(ANG)
C       YNEW = X * SIN(ANG) + Y * COS(ANG)
C
C   THEN THE CELLS REPRESENTED BY XNEW AND YNEW ARE ASSIGNED
C   THE APPROPRIATE VALUES
C   NOTE IF ANG = C NO CHANGE IS MADE
C
C   THIS SAME TECHNIQUE IS USED FOR ROTATION OF ALL OBJECTS
C
        DO 100 IY = HALFCEL,WID,CELSIZ
        Y = IY
        YSIN = ZSIN * Y
        YCOS = ZCOS * Y
        DO 100 IX = HALFCEL,LEN,CELSIZ
        X = IX
        COL = ZCOS * X - YSIN
        ROW = ZSIN * X + YCOS
        COL = COL/CELSIZ + STCOL
        ROW = ROW/CELSIZ + STROW
C
C   LR, LC - LEAST ROW AND LEAST COLUMN INCLUDED IN THIS OBJECT
```

```
C
      PT =-2
      HALFCEL = CELSIZ/2
C
C  ANGLE OF ROTATION -  SEE CURE FOR DETAILS OF ROTATION
C
      ZANG = FLOAT(ANG)/57.295
      ZSIN = SIN(ZANG)
      ZCOS = COS(ZANG)
C
      ZZTOP = FLOAT(SLOPE)/57.295
C
C  TANGENT OF SOP FOR FRON FACE OF WEDGE
      ZTAN = SIN(ZZTOP)/COS(ZZTOP)
      CNT =0
C
C  ELEVATION INITIALIZED TO ZERO
      ELEV =0
C
C  WIDTH OF TOP IN NUMBER OF CELLS
      TOP = TOP/CELSIZ
      INCR =0
      IY = HALFCEL-CELSIZ
 100  ELEV = ELEV+INCR
      IF(ELEV .LT.0) GO TO 200
      IY = IY+ CELSIZ
      Y = IY
      YSIN = ZSIN * Y
      YCOS = ZCOS * Y
C
      DO 110 IX =HALFCEL,LEN,CELSIZ
      X = IX
      COL = ZCOS * X - YSIN
      ROW = ZSIN * X + YCOS
      COL = COL/CELSIZ + STCOL
      ROW = ROW/CELSIZ + STROW
C
C  PARAMETERS FOR FILLIN AS DESCRIBED IN CURE
C
      IF(COL .LT. LC) LC = COL
      IF(COL .GT. BC) BC = COL
      IF(ROW .LT. LR) LR = ROW
      IF(ROW .GT. BR) BR = ROW
      GO TO 90
      PT =PT+3
C
C  DEBUG PRINT STATEMENT
C
      BUF(PT)=COL
      BUF(PT+1)=ROW
      BUF(PT+2)=ELEV
      IF(PT .LT. 13) GO TO 90
```

```
      WRITE(6,60) (BUF(M), M=1,15)
 60   FORMAT(5(5X,2I5,2X,I5))
      FT =-2
  90  FLD(18,12,ARRAY(COL,ROW)) = FLD(24,12,ELEV)
      FLD(30,6,ARRAY(COL,ROW)) = FLD(30,6,CAT)
 110  CONTINUE
C
C
C  CHECK FOR REACHING TOP OF WEDGE
C
      IF (ELEV .GE. HGT) GO TO 150
C
C  HAVE NOT REACHED TOP SO INCREMENT ELEVATION FOR NEXT ROW
C  PROCEED TO NEXT ROW
C
C
      INCR = ZTAN * FLOAT(CELSIZ)
      GO TO 100
C
  150 CNT = CNT +1
C  HAS FLAT TOP OF WEDGE BEEN COMPLETED?
C
      IF(CNT .GT. TOP) GO TO 160
C
C  NO, SO NO CHANGE IN ELEVATION FOR NEXT ROW
C
      INCR =0
      GO TO 100
 160  IF(BSLOPE .GE.90) GO TO 200
      Z3 = FLOAT(BSLOPE)/57.295
C
C  BEGIN DESCENDING ALONG BACK FACE OF WEDGE
C  MODIFY SLOPE FACTOR
C  NOTE IT IS NEGATIVE SO THAT ELEVATION WILL BE
C  DECREMENTED
C
      ZTAN = -SIN(Z3)/COS(Z3)
      INCR = ZTAN * CELSIZ
      GO TO 100
 200  CONTINUE
C200  WRITE(6,60) (BUF(M),M=1,15)
      DO 11 I=1,15
 11   BUF(I)=0
      CATEG = CAT
      CALL FILLIN (LC,BC,LR,BR,CATEG)
      CONTINUE
      RETURN
      END
C     SUBROUTINE PRYAMID(LEN,HGT)
      SUBROUTINE PRYAMID(LEN,HGT)
C
      IMPLICIT INTEGER (A-W)
```

```
      COMMON BUF(20), ARRAY(500,70),CELSIZ,STCOL,STROW,CAT,ANG
      DATA LC,BC,LR,BR/500,0,500,0/
C
C  ANGLE OF ROTATION IF ANY (SEE CUBE FOR DETAILS)
C
      ZANG = FLOAT(ANG)/57.295
      ZSIN = SIN(ZANG)
      ZCOS = COS(ZANG)
C
      HALFCEL = CELSIZ/2
C
C  HALF THE LENGTH OF THE FINISHED PRYAMID
C
      HFLEN = LEN/2 + HALFCEL
C
C  TANGENT OF SLOP OF PRYAMID FACE
C
      ZTAN = FLOAT(HGT)/FLOAT(HFLEN)
C
C
C  DO LOOP FOR ROWS OF PRYAMID
C  BECAUSE OF SYMMETRY FOR EACH POINT BELOW THE CENTERLINE
C  THERE IS A CORRESSPONDING POINT EQUAL DISTANCE ABOVE IT
C  Y - IS Y VALUE FOR LOWER CELL
C  Y2 - CORRESSPONDING CELL ABOVE
C
      DO 100 IY=HALFCEL,HFLEN,CELSIZ
      Y = IY
      YSIN = ZSIN * Y
      YCOS = ZCOS * Y
      Y2 = LEN - IY
      Y2SIN = ZSIN * Y2
      Y2COS = ZCOS * Y2
C
C
C  X VALUE FROM LEFT TO RIGHT
C
      DO 150 IX = HALFCEL,LEN,CELSIZ
      X = IX
      COL = ZCOS * X - YSIN
      ROW = ZSIN * X + YCOS
      COL = COL/CELSIZ +STCOL
      ROW = ROW/CELSIZ + STROW
C
      COL2 = ZCOS * X - Y2SIN
      ROW2 = ZSIN * X + Y2COS
      COL2 = COL2/CELSIZ +STCOL
      ROW2= ROW2/CELSIZ + STROW
C
C  PARAMETERS FOR FILLIN
C
      IF(COL .LT.LC) LC=COL
```

```
      IF(COL2 .LT. LC) LC=COL2
      IF(COL .GT. BC) BC = COL
      IF(COL2 .GT. BC) BC = COL2
      IF(ROW .LT. LR) LR = ROW
      IF(ROW2 .LT. LR) LR = ROW2
      IF(ROW .GT. BR) BR = ROW
      IF(ROW2 .GT. BR) BR =ROW2
C
C   IF X < Y THEN ELEVATION AT THAT CELL = XDIS*SLOPE OF FACE
C   (REPRESENTS LEFT FACE OF PYRAMID)
      ELEV = ZTAN * X
      IF (IY .GE. IX) GO TO 70
C
C   IF X>Y AND Y< LENGTH-XDIS THEN ELEVATION = YDIS*SLOPE
C   (FRONT FACE)
C
      ELEV = ZTAN * Y
      IF(IY .GE. (LEN+CELSIZ-IX)) ELEV = ZTAN*(LEN+CELSIZ-IX)
C
C   ASSIGN THE TWO SYMMETRICAL POINTS THE APPROPRIATE
C   ELEVATION AS JUST CALCULATED
C
   70 FLD(18,12,ARRAY(COL,ROW))=  FLD(24,12,ELEV)
      FLD(18,12,ARRAY(COL2,ROW2)) = FLD(24,12,ELEV)
      FLD(30,6,ARRAY(COL,ROW)) = FLD(30,6,CAT)
      FLD(30,6,ARRAY(COL2,ROW2)) = FLD(30,6,CAT)
  150 CONTINUE
  100 CONTINUE
      CATEG = CAT
      CALL FILLIN(LC,BC,LR,BR,CATEG)
      CONTINUE
      RETURN
      END
C     SUBROUTINE HEMISP(RAD)
      SUBROUTINE HEMISP(RAD)
C
      IMPLICIT INTEGER (A-W)
      COMMON BUF(20), ARRAY(500,70),CELSIZ,STCOL,STROW,CAT,ANG
      HALFCEL = CELSIZ/2
      PRAD = RAD
C
C   RADIUS IN NUMBER OF MATRIX CELLS
C
      NCELL = FLOAT(PRAD)/FLOAT(CELSIZ) +.5
      ZRAD2= RAD**2
C
C   CENTER POINT
C
      ARRAY(STCOL,STROW) = RAD
C
C   UTILIZE SYMMETRY OF HEMISPHERE
C   CALCULATE THE ELEVATION FOR EACH CELL IN ONE QUADRANT
```

```
C   AND THE SAME VALUE CAN BE ASSIGNED TO THE CORRES-
C   SPONDING CELLS IN THE OTHER THREE QUADRANTS
C
        DO 100 I =1, (NCELL+1)
        ROW = (I-1) * CELSIZ
        ZR2 = ROW**2
        NCOL = SQRT(ZRAD2-ZR2) /FLOAT(CELSIZ)+1.
C
C
        DO 100 CL=1,NCOL
        COL=CL-1
        ZHYP2=(COL*CELSIZ)**2 + ZR2
C
C   CALCLLATE ELEVATION AT CELL USING RIGHT TRIANGLES
C
        ELEV =SQRT(ZRAD2 - ZHYP2)
C
C   ALGORITHM TO PLACE VALUE IN CORRESSPONDING CELLS
C   OF ALL FOUR QUADRANTS
C
        S = -1
  7C    S2= -1
  71    C = S*COL + STCOL
        R = S2*(I-1) + STROW
        FLD(18,12,ARRAY(C,R)) = FLD(24,12,ELEV)
        FLD(30,6,ARRAY(C,R)) = FLD(30,6,CAT)
        IF (S .EQ. 1) GO TO 80
        S =1
        GO TO 71
  8C    IF(S2 .EQ. S) GO TO 100
        T = S2
        S2 = S
        S = T
        GO TO 71
 1CC    CONTINUE
        RETURN
        END
C       SUBROUTINE CYL(RAD,LEN)
        SUBROUTINE CYL(RAD,LEN)
        IMPLICIT INTEGER (A-W)
        COMMON BUF(20), ARRAY(500,70),CELSIZ,STCOL,STROW,CAT,ANG
C
C
        HALFCEL = CELSIZ/2
        ZRAD2= RAD**2
        NCOL = LEN/CELSIZ
        NROW = RAD/CELSIZ
C
C
        DO 100 II=1,(NROW+1)
        I= II-1
        ROW = I*CELSIZ
```

```
      ELEV = SQRT(ZRAD2-ROW**2)
      DO 100 J =1,NCOL
      JJ = J-1
C
C  PLACE APPROPRIATE ELEVATION IN CELLS EQUIDISTANT ABOVE
C  AND BELOW THE CENTERLINE
C
      FLD(18,12,ARRAY(STCOL+JJ,STROW+I)) = FLD(24,12,ELEV)
      FLD(18,12,ARRAY(STCOL+JJ,STROW-I)) = FLD(24,12,ELEV)
      FLD(30, 6,ARRAY(STCOL+JJ,STROW-I)) = FLD(30,6,CAT)
      FLD(30, 6,ARRAY(STCOL+JJ,STROW+I)) = FLD(30,6,CAT)
 1CC  CONTINUE
      RETURN
      END
C     SUBROUTINE TOWER(RAD,ELEV)
      SUBROUTINE TOWER(RAD,ELEV)
      IMPLICIT INTEGER (A-W)
      COMMON BUF(20), ARRAY(500,70),CELSIZ,STCOL,STROW,CAT,ANG
C
C  THE SAME ALGORITHM AS USED FOR HEMISPHERE EXCEPT
C  THAT THE ELEVATION NEED NOT BE CALCULATED - IT IS
C  CONSTANT VALUE
C
      HALFCEL = CELSIZ/2
      NCELL = FLOAT (RAD)/FLOAT(CELSIZ) +.5
      ZRAD2= RAD**2
      ARRAY(STCOL,STROW) = ELEV
C
C
      DO 100 I =1, (NCELL+1)
      ROW = (I-1) * CELSIZ
      ZR2 = ROW**2
      NCOL = SQRT(ZRAD2-ZR2) /FLOAT(CELSIZ)+1.
C
C
      DO 100 CL=1,NCOL
      COL = CL-1
C
C
      S = -1
   70 S2= -1
   71 C = S*COL + STCOL
      R = S2*(I-1) + STROW
      FLD(18,12,ARRAY(C,R)) = FLD(24,12,ELEV)
      FLD(30,6,ARRAY(C,R)) = FLD(30,6,CAT)
      IF (S .EQ. 1) GO TO 80
      S =1
      GO TO 71
   80 IF(S2 .EQ. S) GO TO 100
      T = S2
      S2 = S
      S = T
```

```
      GO TO 71
 100  CONTINUE
      RETURN
      END
C     SUBROUTINE CHKRBD
      SUBROUTINE CHKRBD
      IMPLICIT INTEGER (A-W)
      COMMON BUF(20),ARRAY(500,70),CELSIZ,STCOL,STROW
      DO 100 I=1,20,4
      CATEG = I/4+1
      DO 100 J=1,4
      DO 100 K=1,4
      ROW =J + STROW
      COL = I+K+STCOL
      ROW2 = I+K+STROW
      COL2 = J+STCOL
      ARRAY(COL,ROW)=CATEG
 100  ARRAY(COL2,ROW2)=CATEG
      RETURN
      END
C     SUBROUTINE FILLIN(LC,BC,LR,BR,CATEG)
      SUBROUTINE FILLIN(LC,BC,LR,BR,CATEG)
C
C  THIS SUBROUTINE CHECKS ROTATED SURFACES FOR 'HOLES
C  AND FILLS THEM IN WITH APPROPRIATE ELEVATION AND
C  CATEGORY IF IT FINDS ANY
C
      IMPLICIT INTEGER(A-W)
      COMMON BUF(20),ARRAY(500,70)
      DATA HUN/0100/
C
C
C
C  BEGIN AT LEAST ROW AND COLUMN AND PROCEED TO BIGGEST
C  ROW AND COLUMN TO CHECK FOR HOLES
C
C
      DO 100 I=LR,BR
      DO 100 J = LC,BC
C
C  IF THE CELL IS NONZERO IT HAS BEEN ASSIGNED A VALUE
C  AND IS OK.  PROCEED
C
      IF (ARRAY(J+1,I) .NE. 0) GO TO 100
C
C  THIS CELL IS ZERO.  IF BOTH OF ITS NEIGHBORS (TO LEFT AND
C  RIGHT) ARE NON ZERO, THEN WE HAVE FOUND A HOLE
C  OTHERWISE THIS CELL IS NOT PART OF THE OBJECT
C  AND CAN BE IGNORED
C
      IF(ARRAY(J,I) .EQ.0 .OR. ARRAY(J+2,I) .EQ. 0) GO TO 100
C
```

```
C  FILL IN HOLE WITH AVERAGE OF ITS TWO NEIGHBORS
C
      ARRAY(J+1,I) = ((ARRAY(J,I)+ ARRAY(J+2,I))/HUN * HUN /2) +CATEG
 100  CONTINUE
      RETURN
      END
```

APPENDIX B

BASELINE OF PLANIMETRIC DATA BASE CONSTRUCTION: PICKWICK SITE

The following technical report (TR 319-2) prepared by the Center for Research, Inc., University of Kansas, is included in this volume to provide details in support of the technical discussions of Volume I.

THE UNIVERSITY OF KANSAS CENTER FOR RESEARCH, INC.

2291 Irving Hill Drive Campus West
Lawrence, Kansas 66045

Telephone: 913-8644832

BASELINE OF PLANIMETRIC DATA BASE CONSTRUCTION: PICKWICK SITE

Remote Sensing Laboratory
RSL Technical Report 319-2

E. Davison

V. Kaupp

J. Holtzman

July, 1976

Supported by:

U. S. Army Engineer Topographic Laboratories
Fort Belvoir, Virginia 22060
CONTRACT DAAG 53-76-C-0154

ABSTRACT

A ground truth data base was required to be made of a test site centered around the Pickwick Landing Dam in Tennessee for radar image simulation and terminal guidance studies. This report presents the first step in making such a data base. The product reported here is a hand-drawn feature map containing the boundaries separating regions which are homogeneous to a radar (homogeneous at radar wavelengths). This feature map containing the radar planimetry was constructed by a photo-interpreter using standard feature extraction techniques. Standard 7 1/2' quadrangle USGS maps and high-resolution photographs were used as the input intelligence source for construction of the feature map. To support the terminal guidance studies, the spatial resolution of the features built into the map was approximately 100 feet. That is, the smallest region which can be categorized as distinct from its surroundings is approximately 100 feet square.

Subsequent steps in making this ground truth data base required this feature map to be digitized and formed into a digital matrix[1], and this digital matrix of radar planimetry to be merged with a digital matrix of elevation data[2].

In addition to serving its primary purpose of constructing a ground truth data base of the Pickwick site for radar image simulation, the construction time and quality of this feature map will be used as a baseline study against which candidate automated/interactive feature extraction techniques will be tested and evaluated. For this purpose, the actual feature extraction time required to produce this map was 28 hours.

---

[1] McNeil, M., V. H. Kaupp, and J. C. Holtzman, "Digitization of Pickwick Site Data Base," TR 319-4, Remote Sensing Laboratory, The University of Kansas, February, 1977.

[2] Komp, E., M. McNeil, V. H. Kaupp, and J. C. Holtzman, "Medium Resolution Digital Ground Truth Data Base: Pickwick Site," TR 319-5, Remote Sensing Laboratory, The University of Kansas, August, 1977.

INTRODUCTION

A ground truth data base for both terminal guidance and validation tasks, encompassing approximately 144 square miles, has been prepared for a test site centered around the Pickwick Landing Dam area located in the states of Tennessee, Mississippi, and Alabama. This data base was developed by a photo interpreter working from optical photographs and maps using manual feature extraction techniques. The basic philosophy motivating the development of this data base of the Pickwick site was to use only manual feature extraction techniques. Manual feature extraction techniques were used because the use of these techniques by photographic interpreters is common wherever images are analyzed. They allow complete control over the level of detail and complexity desired for any given data base. This planimetric data base will be used both as a reference baseline for future studies of automated feature extraction techniques and as the input ground truth data base for radar image simulation and terminal guidance studies. Since this data base was to become a baseline of known accuracy and complexity, standard interpretation techniques were used. The accuracy of this data base, the time required to construct it, and the cost of the construction were, therefore, known and these parameters could be compared to data bases of this same site built using automated feature extraction techniques.

To accomplish the objectives motivating the development of this data base (comparison baseline for automatic feature extraction techniques and ground truth data base for radar image simulation and terminal guidance studies), the planimetry information which was extracted from a variety of input data sources and transferred by hand to the feature map has been digitized and placed on magnetic tape. Computer software were developed to convert these digital boundaries into a symbolic image in the form of a three-dimensional array of terrain categories[1]. This

[1] McNeil, M., V. H. Kaupp, and J. C. Holtzman, "Digitization of Pickwick Site Data Base," TR 319-4, Remote Sensing Laboratory, The University of Kansas, February, 1977.

digital matrix (the symbolic image) was then merged with a digital matrix of elevation data of the same site. This final operation produced the ground truth data base of the Pickwick site[2]; a new four-dimensional array containing both the terrain types and elevation data which are stored on magnetic tape. It is this final product which is the desired ground truth data base of the Pickwick site as well as the baseline ground truth data base against which to test and evaluate candidate automated feature extraction techniques. Only the manual feature extraction of terrain types is reported in this report. For additional information concerning other aspects of the development of this data base, references should be made to the other listed appendices.

SITE DESCRIPTION

The Pickwick Dam test site is located in the Tennessee River Valley near the junction of the states of Tennessee, Alabama, and Mississippi. The area consists of open rolling hills as well as a portion of the Tennessee River floodplain. The boundary of the test site was defined as a six-mile radius extending from the northwest corner of the powerhouse at Pickwick Dam. The circle described by the sweep of this radius is contained within a square, the edges of which measure twelve miles on a side. The geographic area contained within this square comprises the 144 square miles of terrain examined to construct the data base.

PLANIMETRY CATEGORIES

The Pickwick Dam test site presented a major problem in category selection since the ground truth available on the site was limited or nonexistent. The relatively short time frame within which the data base

---

[2] Komp, E., M. McNeil, V. H. Kaupp, and J. C. Holtzman, "Medium Resolution Digital Ground Truth Data Base: Pickwick Site," TR 319-5, Remote Sensing Laboratory, The University of Kansas, August, 1977.

was to be constructed prevented field checks, and no information was readily available concerning crop types or distribution of native plant communities. Consequently, categories for the assignment of backscatter values were constructed at relatively gross levels of classification.

The most readily discernible distinction of the terrain in the test site is between areas which were covered with water and areas which were not. In a dichotomous manner, of those areas which were not water-covered, some were forested and some were not. Those areas covered by the Oak-Hickory-Pine forest, typical of the region, comprised one category. In a similar manner, those areas which were not forested were broken up, examined and identified to be agricultural land or non-agricultural land. This latter group is categorized as consisting of areas of open grassland. Areas belonging to this category were primarily parks, pastures and similar expanses which were neither covered by trees nor under cultivation (as discernible from photography). Some areas were not conducive to identification by strictly dichotomous means. For instance, there were areas of marshland that were wooded and which had to be dealt with separately from forested areas. As each category was established, it was given a three-digit identification number (to facilitate data handling). These category number assignments are listed in Table I.

INPUT DATA SOURCES

A) Photographic Products

At the beginning of the analysis, 96 frames of high-altitude (30,000 feet) aircraft photography were received from the Engineering Topographic Laboratories (ETL). This imagery was supplied in the form of black and white positive prints at a scale of 1:100,000. Upon checking, it was noted that significant scale variation occurred in the imagery, both within given prints and between adjacent flight lines. It was concluded that this deviation was due to some combination of the hilly topography, instability in or turbulence effecting the remote sensing platform itself, and possible slight variation in altitude between flight lines. The overall resolution in this imagery was rated as good but with tonal

TABLE 1

Pickwick Test Site Planimetry Categories
Validation and Terminal Guidance Tasks

Category Identification Number	Category Description
100	Group Targets - Roads
110	Heavy duty improved roads (none present)
120	Medium duty improved roads
130	Light duty improved roads
140	Unimproved roads
200	Group targets - Railroads
230	Fish Pond Dikes
240	Water Plant Plumbing
300	Group Targets - Water Bodies
310	Small impoundments
320	Small streams and rivers
350	Large streams and rivers
360	Large impoundments (Pickwick Lake)
400	Group targets - Marshes (see subgroup)
450	Wooded Marshes
500	Group Targets - Forested areas
600	Group Targets - Agricultural Land
700	Group Targets - Grass-covered areas (parks, etc.)
800	Pickwick Dam
810	Blockhouse
820	Spillway
900	Small buildings
910	Large buildings

Note: The planimetry categories listed in this table represent the level of detail present in this data base. Finer detail will be added to this data base when it is desired to model a radar system with higher resolution and discrimination capabilities.

contrast rated as only fair. Because of the uncontrolled scale of these prints, they were deemed inappropriate for use as the major source of the geometry of the various features in the data base. They were used to construct an uncontrolled mosaic which has repeatedly been used for feature discrimination and identification, and for reference and cross-checks.

A check for additional imagery was made with the United States Geological Survey's (USGS) data bank at the EROS Data Center in Sioux Falls, South Dakota. Although several NASA U-2 flights have been made in the region of interest, they had passed well to one side of the test site and were of no value to this effort. A great deal of USGS mapping photography was available, but this would have required the acquisition of photo index sheets with selections being made and then ordered. The resulting turn-around time was unacceptable considering the time available to make this data base, and it was deemed more practical to avoid such delays and work with the photographs and maps at hand.

B) Radar Products

No radar imagery of the test site was available at the time of data base construction. It was felt that the presence of such imagery might contribute to prejudicial judgments in the construction of this data base. Radar imagery of the data base area was purchased from Goodyear Aerospace Corporation and arrived after the planimetry data extraction was completed.

C) Maps

Six topographic maps at 1:24,000 scale were obtained from the USGS with sections of the test site falling into each of the six. Figure 1 illustrates the position, center, and boundaries of the Pickwick test site relative to the six maps. The identification numbers of the maps used to construct this data base are included in this figure. Because the data base was to be converted to digital format, it was thought preferable that the planimetry be extracted at a relatively large scale.

FIGURE I. PICKWICK DAM TEST SITE IN RELATION TO U. S. G. S. TOPOGRAPHIC SHEETS OF THE AREA

Digital elevation data tapes provided by ETL for this site were made from the USGS topographic maps at a scale of 1:25,000. These facts, together with the uncontrolled scale on the aerial photography, contributed to the decision to use the USGS maps as the primary planimetric source thereby matching the location and geometry of category features in both the elevation and planimetry data.

The center point for the test area was pinpointed and the distance was measured for the six-mile working radius of this data base. Appropriate boundaries were then drawn on the topographic sheets with measurements being made to check for correct geometry. The individual sheets were not physically mosaicked, as the resulting large sheet would have proven quite unwieldy. Instead, a grid corresponding to the collective boundaries on the individual sheets was constructed on a large sheat of tracing paper. The delineated areas on each of the six sheets was then matched to this grid.

FEATURE EXTRACTION

A) Methodology

In keeping with the philosophy of developing this data base by manual feature extraction techniques, the data base was prepared by a photo interpreter (PI) who used only conventional photographic interpretation techniques. The PI used only hand drafting tools to transfer the planimetric information from the input data sources to a piece of drafting paper that was temperature and humidity stabilized. The scale of the final product was equal to the scale of the USGS topographic maps that were used as input data sources (1:24,000; see Figure 1 for map identification numbers). The USGS topographic maps were used as the prime data source for the location and geometry of gross features and frequent crosschecks were made with a mosaic of the site previously made from aerial photographs (scale was 1:100,000).

The PI used hierarchical feature extraction techniques to develop this data base. The planimetry boundaries present on the topographic

maps were regarded as the major authority for gross features, with cross-checks made to the mosaic, in extracting the cartographic information and transferring it to the data base. The first class of features the PI transferred was the road network of the test site. The location of roads as they appeared on the topographic maps was regarded as the most reliable authority given the scale problems in the aerial photography. No major discrepancies were found either in the location of roads or in the number of roads identified when the uncontrolled mosaic of aerial photographs was compared to the topographic maps. The road network obtained from the maps was, therefore, used as the control reference for the placement of all other targets in the data base. Not all classes of roads were transferred by the PI to the data base. For instance, in some cases, foot trails were deleted due to the nature of the vegetation canopy present in the area.

Following the drafting of the road network, the positions of bodies of water and the sole railroad present in the site were transferred to the data base. The boundaries of the various bodies of water were transferred directly from the topographic maps. The actual boundaries at any point in time of lakes, ponds, and rivers are very much determined by the season and the precipitation and use history. It was decided that the water boundaries would be taken from the maps (instead of from the mosaic of aerial photographs) to more closely match the elevation data taken from those same maps. All of the bodies of water were transferred as they appeared on the maps, with the exception of some small ponds which were deleted due to the nature of the vegetation canopy present in the area. The placement of the railroad in the data base presented no problem, as it was not close enough to the regular road network to have been moved in accordance with standard cartographic practices. (Where two or more roads or railroads lie in juxtaposition, standard cartographic practice is to place them no closer than some minimum distance which is set by the map scale.)

Boundaries between forested and non-forested areas were the next features to be transferred to the data base. Although these boundaries are clearly marked on topographic maps, they tend to be too generalized. For

this reason, the mosaic of aerial photographs was regarded as the major authority for forest/non-forest boundaries, and the topographical maps were consulted as a crosscheck. All forested areas were combined into a single data base category. Minor features, such as hedge rows and small groves of trees, were not included in the data base being constructed. (Fine detail can be added later as desired.)

Discrimination between all other areas (such as agricultural land, grassy areas, marshes, etc.) was made both according to commonly used interpretation identifiers (i.e., texture, tone, pattern, relationship to other objects) as well as the interpreter's knowledge and experience. This level of detail of features in the terrestrial envelope is typically not present on topographic maps. Therefore, the mosaic of aerial photographs was essentially the sole source available of information concerning boundaries and terrain types in these areas.

Last, the cultural (hard) targets were transferred to the data base. The transfer of these targets was done strictly on a "house-by-house" basis. These buildings were located according to the aerial photographic mosaic and were hand drafted in the data base with consideration given to the radar reflectivity potential of each. Only the location of each building was transferred to the data base, not the geometry. No major towns or cities exist in the test site. The greatest density of cultural targets consisted of small industrial areas and hamlets.

It should be pointed out that throughout the construction of this data base, only hand drafting tools were used. As the desire for detail (complexity and accuracy) of such construction increases so, of necessity, does the sophistication, accuracy and cost of the equipment and techniques increase. To minimize the unnecessary expenditure of resources (unnecessary with respect to the potential uses of this data base) the exact geometry of each building was not transferred.

B) Problems

The planimetric data base was constructed on a sheet of drafting paper that was nearly three feet square (nine square feet). It was found that the initial tracing medium, a one hundred percent rag paper, was swelling and contracting in the temperature and humidity conditions prevalent in the building.

C) Solutions

A sheet of stabilized tracing material with a three mil acetate base was obtained to eliminate the problem of distortion caused by swelling and contracting of the drafting paper. The construction of the data base was started over using this stabilized material.

BASELINE CONSTRUCTION

This planimetric data base will be used as a baseline for future studies of automated feature extraction techniques. The bases for comparison will be the accuracy with which the cartographic information is preserved by the various candidate automated feature extraction methods and the savings each candidate method represents in cost and in manhours of data base construction time. To facilitate these projected evaluations, the manhours expended in the construction of this data base are reported in Table 2. By reference to this table, it can be seen readily that actual planimetric data base construction time, the time required after initial site familiarization and preparation, was 28 hours. This is the baseline time to be used when comparing the automated feature extraction techniques to manual techniques.

TABLE 2

Planimetric Data Base Construction Time
for the Pickwick Site

Manhours	Description of Effort
30	Site Investigation (Preliminary work; mosaic preparation)
1	Research of Available Data Products
4	Preparation of Site Data Sources
18	Feature Extraction Time
10	Layout Time (Initial)
6	Layout Time (Stable Base Materials)
1	Checking

APPENDIX C

DIGITAL ELEVATION DATA BASE CONSTRUCTION: PICKWICK SITE

The following technical report (TR 319-3) prepared by the Center for Research, Inc., University of Kansas, is included in this volume to present additional details of the task of data base construction and to support the discussions of Volume I.

THE UNIVERSITY OF KANSAS CENTER FOR RESEARCH, INC.

2291 Irving Hill Drive Campus West
Lawrence, Kansas 66045

Telephone: 913-8644832

DIGITAL ELEVATION DATA BASE CONSTRUCTION: PICKWICK SITE

Remote Sensing Laboratory

RSL Technical Report 319-3

M. McNeil

V. Kaupp

J. Holtzman

July, 1976

Supported by:

U. S. Army Engineer Topographic Laboratories

Fort Belvoir, Virginia 22060

CONTRACT DAAG 53-76-C-0154

ABSTRACT

A digital elevation data base of a test site centered around Pickwick Dam was developed from six magnetic tapes, each containing the digital elevation data from one of the six standard United States Geological Survey 7-1/2 minute quadrangle maps of the site. The initial effort of digitizing the elevation data was done by the Defense Mapping Agency (DMA). DMA digitized 20 foot contours and used an interpolation computer program to calculate an elevation value for every 6.25 m increment in both directions. Approximate accuracy of the resultant digital elevation matrix was estimated to be 10 feet. The six maps were digitized independently of each other and, therefore, were not produced to form one cohesive, two-dimensional data base.

The elevation data from these six tapes have been structured into one very large two-dimensional orthogonal grid matrix. This matrix was merged with the digital planimetry data of the same site[1]. The result is a data base for radar image simulation and terminal guidance studies.

[1] Komp, E., M. McNeil, V. H. Kaupp, and J. C. Holtzman, "Medium Resolution Digital Ground Truth Data Base," TR 319-5, Remote Sensing Laboratory, The University of Kansas, August, 1977.

INTRODUCTION

A ground truth data base for radar image simulation and terminal guidance studies was constructed for the geographic area encompassing approximately 144 square miles (36 square miles for the terminal guidance study) of a test site centered around the Pickwick Landing Dam in the states of Tennessee, Alabama, and Mississippi. This appendix presents the work performed to develop a digital elevation matrix which accurately models the relief present in the topography of the site.

This digital elevation matrix was developed from six digital, computer-compatible, magnetic tapes. Each of the tapes contained the elevation data from one of the standard 7 1/2' quadrangle USGS (United States Geological Survey) maps of the site. Figure 1 illustrates the position, center, and boundaries of the Pickwick test site relative to the six maps. The identification numbers of the maps used to construct this data base are included in this figure.

The six digital elevation data tapes were provided to us by ETL (Engineering Topographic Laboratories). The Defense Mapping Agency (DMA) produced the digital elevation data stored on these tapes. DMA digitized 20 foot elevation contours for each of the USGS maps and ran an interpolation computer program on these digitized contours to produce a digital elevation matrix for each map. The interpolation program calculated an elevation value for each 6.25 m increment in either direction with an estimated accuracy of approximately ten feet.

These digital elevation tapes could not be used for our intended application as received. The six tapes were not produced to form one cohesive, two-dimensional array of elevation data for the Pickwick site. Instead, the maps were apparently digitized individually. This meant that it was necessary to merge into one orthogonal grid matrix the elevation data from different magnetic tapes each having data stored in a matrix with a coordinate system skewed slightly from all others. Constructing this large orthogonal grid matrix of elevation data required a significant amount of data manipulation by both man and machine.

Figure 1. Elevation Data Orientation of Pickwick Dam Test Site in Relation to USGS Topographic Sheets of the Area.

Upon completion of this task, the digital elevation data were merged[1] with the digital planimetry data produced in a separate effort[2] for the same site. The resulting data matrix was the desired ground truth data base of the Pickwick site for radar image simulation and terminal guidance studies.

SITE DESCRIPTION

The Pickwick Dam test site is located in the Tennessee River Valley near the junction of the states of Tennessee, Alabama, and Mississippi. The area consists of open rolling hills as well as a portion of the Tennessee River floodplain. The boundary of the test site was defined as a twelve mile square centered on the northwest corner of the powerhouse at Pickwick Dam. The geographic area contained within this square comprises the 144 square miles of this data base.

INPUT DATA SOURCE

The initial effort of digitizing the elevation data was done by the Defense Mapping Agency (DMA). DMA produced one magnetic tape of digital elevation data for each standard USGS (United States Geological Survey) 7-1/2 minute quadrangle map (Figure 1). These digital elevation data were stored on tape in a two-dimensional array, not as elevation contours. The two-dimensional array represented an orthogonal grid of elevation data at a specified metric interval. The orthogonal grid was oriented according

[1] Komp, E., M. McNeil, V. H. Kaupp, and J. C. Holtzman, "Medium Resolution Digital Ground Truth Data Base," TR 319-5, Remote Sensing Laboratory, The University of Kansas, August, 1977.

[2] McNeil, M., V. H. Kaupp, and J. C. Holtzman, "Digitization of Pickwick Site Data Base," TR 319-4, Remote Sensing Laboratory, The University of Kansas, February, 1977.

to the UTM (Universal Transverse Mercator) coordinate system. The specified metric interval was 6.25 meters in the area of interest. Thus, an elevation data value existed in the digital two-dimensional array stored on magnetic tape for each 6.25 meter increment in either the UTM Easting or Northing direction. The elevation data were interpolated values calculated between actual elevation contour lines from these maps by some computer interpolation routine.

The coordinate system into which these elevation data were placed (lines of constant UTM value) was skewed with respect to the coordinate system of the maps; the map coordinate system was defined by lines of constant latitude and longitude which, in a small enough region, form an orthogonal grid. The orientation of the maps in the UTM coordinate system is shown in Figure 2.

Figure 2. Orientation of USGS Map in UTM at Pickwick

The view shown in Figure 2 is exaggerated for clarity, but it illustrates the orientation of the map in the UTM coordinate system. The relative orientation of these coordinate systems was found according to the instructions which accompanied the digital tapes. The four corner points of each map were found to the nearest meter. From these four points, an origin (0,0 point) was found by taking the smallest Easting and Northing value.

The six tapes of elevation data, as we received them, were not produced to form one cohesive, two-dimensional array for the Pickwick area test site. Instead, the six maps were digitized and placed on tape, each apparently independent of the others, with one exception: the corner points of the six maps have the same value for shared corners. The elevation data appeared on tape as an array running in a South to North string with strings sweeping West to East. The situation is depicted in Figure 3.

Figure 3. Elevation Data Orientation.

Another way to interpret this is to say that scan lines run from South to North (each scan line represents a constant Easting value) and scan lines are ordered from West to East. Since the six maps were apparently digitized independently of each other, the scan lines of one map do not coincide with the scan lines of its neighbors. The scan lines for each map are established at 6.25 meter intervals from the origin of that map. Even though the shared corners of the various maps have exactly the same UTM values, the scan lines of the various maps will coincide only if the corners (or origins) are located by integer multiples of 6.25 meters in the Easting direction; this is the exception rather than the rule. The problem is illustrated in Figure 4.

Only the scan lines of Maps 1 and 2 coincide exactly. (See Figure 1 for relative orientation and identification of the six maps.) Map 3 is offset 2.75 meters to the left of Map 1, and Map 5 is offset 2.75 meters to the left of Map 3, making it 5.5 meters offset to the left of Map 1. Map 4 is offset 1.0 meters to the left of Map 2, and Map 6 is offset 2.0 meters to the left of Map 4, making it 3.0 meters offset to the left of Map 2. This is the situation graphically shown in Figure 4. It was also determined that Map 3 is offset 1.75 meters left of Map 4 and that Map 5 is offset 2.5 meters left of Map 6.

In addition to the problem of scan lines not coinciding at the boundaries of the six maps, East-West lines (which would be drawn through the centers of the elevation array values across scan lines) do not coincide. The problem is illustrated in Figure 5. Map 1 is offset 1.5 meters lower than Map 2. Map 3 is offset 0.5 meter lower than Map 1, and Map 5 is 0.5 meter higher than Map 3, making Map 5 coincide with Map 1. Map 4 is 0.5 meter higher than Map 2; Map 6 is 0.5 meter lower than Map 4, making Map 6 coincide with Map 2. It was also determined that Map 3 is 2.5 meters lower than Map 4, and Map 5 is 1.5 meters lower than Map 6.

Figure 4. Scan Line Orientation

Figure 5. Constant E-W Line Orientation.

ELEVATION DATA MERGE

The elevation data contained on the six tapes were produced by DMA by interpolating between constant elevation contour lines from the maps. The data on these tapes are no better than the interpolation technique used and the accuracy of the contour lines on the original maps. Since this is the case, we decided against remaking the six tapes to cause coincidence of the scan lines; a second interpolation would be necessary to shift the data. It was felt that the elevation error introduced by rounding off the metric interval between maps to force coincidence of the scan lines was less than the error introduced by interpolation (especially a second interpolation). This decision is subjectively justified by appeal to the terrain of the test site. It is basically rolling hills and water (floodplain) where the elevation doesn't change much in a 6.25 meter increment in any direction. Therefore, we shifted the scan lines slightly from tape to tape to force coincidence when we made the single two-dimensional elevation array.

Figure 6 illustrates the geometry of the resultant merged data base of elevation data. This figure also lists relevant information concerning how the merged data base was constructed. Reference to Figure 6 may add clarity to the following discussion. Only those data which fell within a 12 mile square were merged to form the final elevation data base. Data were registered for a 12 mile square instead of a six mile square required for the terminal guidance work to support other radar simulation studies. The Easting limits of this 12 mile square were found to be 376000 and 395800, and the Northing limits were 3871500 and 3891300. These limits produced an area of 19,800 square meters which, when translated to a 6.25 meter grid, produced 3169 scan lines by 3169 data point grid. From the corner points of Maps 1, 3, and 5, it was calculated that the westernmost scan line (the scan line labelled 376000) would consist of the merge of blocks 270, 239, and 209 from those maps, respectively. Since the first several hundred scan lines of Maps 1, 3, and 5 were not included in the merged data base, the problem of partial scan lines was not encountered until the junctions between Maps 1 and 2, 3 and 4, and 5 and 6. It was

Figure 6. Orientation of Merged Elevation Data Matrix for Pickwick Site.

found that instead of a one-point overlap (which is what was specified to occur) at these junctions, there was a gap of either 0, 1, or 2 data points. This problem was solved by straightline interpolation between bounding points. For the cases where the specified one-point overlap occurred (at any border) the two elevations were averaged, and that value was used; there were differences in elevation due to the different historical data apparently used in the interpolation routine--the historical data should be a function of the map.

The final elevation data base produced consists of 3169 physical records, with each record containing 3169 elevation values. This represents an elevation data base in excess of 10 million data points. These data are spread across three 7-Track, 800 BPI magnetic tapes in a compressed format.

VERIFICATION

As the merge of the six tapes was produced, a check was made to detect bad elevation values. Each new elevation value was decoded and compared both to upper and lower limits (800 and 300 feet, respectively, for this site) and to the previous elevation value (100 feet or less change from the previous elevation was allowed). Errors in the data were detected. The most serious error was found on the tape representing Map 3; the entire river and various bodies of water were missing. On the other maps, the bodies of water were apparently forced in at appropriate values, but on Tape 3, the interpolation routine was functioning. This problem was corrected after this elevation data matrix was merged with the digital planimetry data matrix[1]. Less serious problems showed up in the data. Among these were the absence of elevation data for the Pickwick Dam, other hard (cultural targets), and numerous ponds. These problems will be addressed during construction of the final, complete data base.

---

[1] Komp, E., M. McNeil, V. H. Kaupp, and J. C. Holtzman, "Medium Resolution Digital Ground Truth Data Base," TR 319-5, Remote Sensing Laboratory, The University of Kansas, August, 1977.

TIME AND RESOURCES USED

Approximately 5 manweeks and many dollars of computer time were required to get the data into a usable, merged format. Many reasons exist to account for this large utilization of resources. Some records were unreadable. Octal dumps of those records were produced; the octal data were examined by hand; correct values were determined; and the correct data were inserted by hand. Because of the very large amount of data involved, small problems created expensive reruns and extensive delays. A significant amount of time resources were utilized in gaining understanding of the problem and experience in handling very large data bases.

CONCLUSIONS

Radars do not typically image terrain according to USGS standard 7-1/2' quadrangle maps, and our radar simulation computer programs model the operation of radars. Since this is the case, when preparing digital data for radar image simulation it is advisable to choose a data structure compatible with the simulation operation. Doing this will minimize subsequent effort necessary to correct these problems.

Construction of this elevation data matrix clearly indicates that the optimum structure as well as the content of data bases for radar image simulation need to be determined. The larger the data base, the more important it is to optimize the structure and content to minimize the development cost and time.

APPENDIX D

DIGITIZATION OF PICKWICK SITE DATA BASE

The following technical report (TR 319-4) prepared by the Center for Research, Inc., University of Kansas, is included in this volume to provide additional details for appropriate discussions in Volume I.

THE UNIVERSITY OF KANSAS CENTER FOR RESEARCH, INC.

2291 Irving Hill Drive—Campus West
Lawrence, Kansas 66045

Telephone: 913-8644832

DIGITIZATION OF PICKWICK SITE DATA BASE

Remote Sensing Laboratory
RSL Technical Report 319-4

M. McNeil
E. Davison
V. H. Kaupp
J. C. Holtzman

February, 1977

Supported by:
U. S. Army Engineer Topographic Laboratories
Fort Belvoir, Virginia 22060

CONTRACT DAAG 53-76-C-0154

ABSTRACT

A data base was constructed of a test site centered on the Pickwick Landing Dam for radar simulation and terminal guidance studies. This report presents a portion of the work involved in making this data base. Specifically, the work reported here is the digitization of the radar planimetry map[1]. The map was digitized manually using a large-table digitizer which was electrically interfaced to a minicomputer. The total time required to digitize this data base was 23 hours.

Upon completion of this digitization task, the digital radar planimetry data were ordered into a symbolic image data matrix which was then merged with the digital elevation data[2] of the same site. The resultant digital matrix[3] is the desired Pickwick test site radar image simulation data base.

---

[1] Davision, E., V. H. Kaupp, and J. C. Holtzman, "Baseline of Planimetric Data Base Construction: Pickwick Site," TR 319-2, Remote Sensing Laboratory, The University of Kansas, July, 1976

[2] McNeil, M., V. H. Kaupp, and J. C. Holtzman, "Digital Elevation Data Base Construction: Pickwick Site," TR 319-3, Remote Sensing Laboratory, The University of Kansas, July, 1976.

[3] Komp, E., M. McNeil, V. H. Kaupp and J. C. Holtzman, "Medium Resolution Digital Ground Truth Data Base: Pickwick Site,"TR 319-5, Remote Sensing Laboratory, The University of Kansas, August, 1977.

INTRODUCTION

A ground truth data base encompassing approximately 144 square miles has been prepared for a test site centered around the Pickwick Dam area located in Tennessee. This data base was developed by a photo interpreter working from optical photographs and maps using manual feature extraction techniques[1]. The data base produced was in the form of a map of the geometric and dielectric properties of the terrain, water, and objects which are physically located within this 144 square mile area. This map was digitized and converted into a two-dimensional array of microwave backscatter category information. The digitization process was performed on a large-table digitizer.

The basic philosophy motivating the development of this data base of the Pickwick site was to use only manual feature extraction techniques. This planimetric data base will be used both as a reference baseline for future studies of automated feature extraction techniques and as the input ground truth data base for radar image simulation and terminal guidance studies. Manual feature extraction techniques were used because the use of these techniques by photographic interpreters is common wherever images are analyzed. They allow complete control over the level of detail and complexity desired for any given data base. Since this data base was to become a baseline of known accuracy and complexity, standard interpretation techniques were used. The accuracy of this data base, the time required to construct it, and the cost of the construction were, therefore, known and these parameters could be compared to data bases of this same site built using automated feature extraction techniques.

To accomplish the objectives motivating the development of this data base (comparison baseline for automatic feature extraction techniques and ground truth data base for radar image simulation studies), the planimetry information which was extracted from a variety of input data sources and transferred by hand to the data base has been digitized and placed on

---

[1] Davison, E., V. H. Kaupp, and J. C. Holtzman, "Baseline of Planimetric Data Base Construction: Pickwick Site," TR 319-2, Remote Sensing Laboratory, The University of Kansas, July, 1976.

magnetic tape. Specifically, the various boundaries drawn by hand on the data base were digitized. Computer software was developed to convert these digital boundaries into a symbolic image in the form of a two-dimensional array of terrain categories. This digital matrix (the symbolic image) was then merged with a digital matrix of elevation data[2] of the same site. This merging operation produced the ground truth data base of the Pickwick site; a new two-dimensional array which contains both the terrain types and elevation data which are stored on magnetic tape[3]. It is this final product which will become the baseline ground truth data base against which to test and evaluate candidate automated feature extraction techniques and the input to radar image simulation studies. Only the digitization of the planimetry data is reported here. For additional information concerning aspects of the development of this data base, reference should be made to the other listed technical reports.

SITE DESCRIPTION

The Pickwick Dam test site is located in the Tennessee River Valley near the junction of the states of Tennessee, Alabama, and Mississippi. The area consists of open rolling hills as well as a portion of the Tennessee River floodplain. The boundary of the test site was defined by a 12-mile square centered on the northwest corner of the powerhouse at Pickwick Dam. The geographic area contained within this square comprises the 144 square miles of terrain examined to construct the data base.

PLANIMETRY VALIDATION

Before the actual digitization was performed, it was necessary to make sure that the planimetry map[1] was valid. It was derived from portions

---

[2] McNeil, M., V. H. Kaupp, and J. C. Holtzman, "Digital Elevation Data Base Construction: Pickwick Site," TR 319-3, Remote Sensing Laboratory, the University of Kansas, July, 1976.

[3] Komp, E., M. McNeil, V. H. Kaupp and J. C. Holtzman, "Medium Resolution Digital Ground Truth Data Base; Pickwick Site," TR 319-5, Remote Sensing Laboratory, The University of Kansas, August, 1977.

[1] Davison, E., V. H. Kaupp, and J. C. Holtzman, "Baseline of Planimetric Data Base Construction: Pickwick Site," TR 319-2, Remote Sensing Laboratory, The University of Kansas, July, 1976.

of six different USGS (United States Geographic Survey) topographic maps plus other, supporting data. The map as drawn was approximately 31 inches square. There were many possible sources for error in the map. The dominating error source is believed to be misorientation and mis-location caused by the necessity to physically mosaic together the six different maps. Of lesser importance, but worthy of note, are such error sources as human factors and manual transference of data from source to planimetry map. Errors in judgment, minute shaking and slips of the hand, non-uniform pencil size, strain, scaling of data from source to data base, and, etc., are some examples of the kinds of errors which probably exist in the data base.

Evaluation of the planimetric data base showed that the errors which might be present were probably statistically independent; they didn't all add in phase. The map was judged to be of good quality. Good is defined here to mean that the map was acceptable for its ordinary purpose; it was to become the input data source for a terminal guidance reference scene formation program which was set up to model a reasonably coarse radar. The most sigrificant problems discovered in the data base were that maps 3 and 4 as well as 5 and 6 were not fit together to the same tolerance as the others; in the digital equivalent of the planimetry map these errors were as large as 11 scan lines. Figure 1 illustrates this problem. In addition, there was a noticeable distortion within map 4 amounting to three or four scan lines which was not a continuous distortion, but was very discrete. It seems reasonable to speculate that map 4 was moved a little with respect to the other maps during the tracing process. In the future, it would be advisable to develop a control grid (instead of just working outlines) for the planimetry map. If this is put down first before anything is traced, it would assist the interpreter in keeping all the various pieces properly oriented.

DIGITIZATION OF PLANIMETRY

The technique selected to convert the planimetry map into a digital format was manual digitization of the boundaries which separate the various categories on the map. This manual digitization was performed on a large digitizing table which was electrically connected to a NOVA/minicomputer

Figure 1. Misregistration of Data.

(Data General Corp.). The accuracy of this table was specified to be 3/1,000 inch. The system consists of the table, a hand-held cursor, the NOVA minicomputer, and a 9-track magnetic tape recording unit.

The digitizing system works in the following way. A two-dimensional representation of the candidate map to be digitized is placed on the table. An orthogonal coordinate system is defined by the operator using the hand-held cursor to locate any two points on each axis of this coordinate system. Although the coordinate system must be orthogonal and two-dimensional, considerable latitude is given to the operator when defining the orientation of the axes on the table. In addition, the scale of each axis may be different. After the coordinate system is specified, the operator uses the crosshairs on the cursor to find any spot or follow any boundary (contour) on the map. The digitizing system has capabilities to record data in either a "point" or a "line" mode. In the "point" mode (or, discrete mode) one point is recorded every time a button on the cursor is pushed. The data recorded are the x-y coordinates of each point relative to the specified coordinate system, plus a third, or z, value. This z value is entered from a teletypewriter by the operator. Each point is assigned a triple of values (x,y,z) and these data are then stored on 9-track magnetic tape; one triple of values per physical record. Each physical record corresponds to a logical card image (80 characters) in which the three values are recorded in a real format (FORTRAN 3F7.1). In "line" (or continuous) mode, the digitizer samples "continuously" from when a start button is pushed until a stop button is pushed. Of course, "continuously" does not mean truly continuous sampling. Actually, there is a timing mechanism and a parameter to control how many samples per second are to be taken. Thus, the digitizer will "follow" the cursor and record points so many times per second. In addition to this timing element, there is also an additional

---

[3] Davison, E., V. H. Kaupp, and J. C. Holtzman, "Baseline of Planimetric Data Base Construction: Pickwick Site," TR 319-2, Remote Sensing Laboratory, The University of Kansas, July, 1976.

distance check (which defines the minimum distance that the cursor must move from the last recorded point before it will record another point). Thus, with the timing parameter set high enough and the distance parameter set low enough, this mode simulates a continuous sampling mode.

This system, then, was used to transform the planimetric drawing (map) into a digital file. This digital file is not an end product. Rather, it is an intermediate step. This file consists of boundaries stored as one-dimensional arrays of triple values. The x and y values represent the position of each data point in the coordinate system which was previously established. The third value (z) can be anything which is a function of the two other dimensions; in this case, the z value specified the radar backscatter category. The backscatter category along a boundary signifies the outermost region of one scattering type and the start of a new category. When the category matrix is filled in, a boundary is used to flag the presence of a distinct scattering target.

Standard 7 1/2' quadrangle USGS maps are drawn in a modified polyconic map projection. The backscatter category data were drawn from USGS maps. The elevation data[2] which were to be merged[3] with this backscatter category map of Pickwick were produced in the Universal Transverse Mercator (UTM) mapping projection according to the standard practices of the agency (Defense Mapping Agency) which produced the digital elevation data. This difference in mapping projections between the planimetry and elevation data necessitated the conversion of one coordinate system to the other. Since the planimetry data were not yet digitized, it was advantageous to select the coordinates for the digitizing system to be the UTM system. This is not a simple, straight-forward conversion, therefore, several necessary assumptions were made. It was assumed that over "small" areas the UTM coordinate system is orthogonal. Second, over "small" areas there was negligible distortion caused by digitizing a

---

[2] Davison, E., V. H. Kaupp, and J. C. Holtzman, "Baseline of Planimetric Data Base Construction: Pickwick Site," TR 319-2, Remote Sensing Laboratory, The University of Kansas, July, 1976.

[3] Komp, E., V. H. Kaupp, and J. C. Holtzman, "Construction of a Geometric Data Base for Radar Image Simulation Studies," TR 319-1, Remote Sensing Laboratory, The University of Kansas, July, 1976.

modified polyconic in a UTM system. "Small" is here defined to be at least the size of our data base (12 miles square). The alternative to making these assumptions was to digitize the category map as a modified polyconic and then use the digital computer to transform one coordinate system into the other. Figure 2 compares the boundaries of the data base in the two-coordinate systems. Appeal to this figure will subjectively justify these assumptions. It can be seen that the coordinate systems are very close to being orthogonal (as measured by the ratios of the east to west and north to south sides of the data base). For the small corrections of this alternative considering the maximum inherent resolution of the data sources which comprise the final result, it was expedient to make the simplifying assumptions.

Minimizing the distortions involved in these assumptions was accomplished by setting the axes to coincide with the center of the map rather than the lower left corner, where it would have been placed normally to coincide with the elevation data map. Thus, there should be no distortion at the center of the map (data base), but increasing distortion toward the edges. Operationally, the 1,000 meter UTM tick marks present in the USGS maps were used to define the coordinate system.

Though the axes were set-up to cross in the middle, it was not necessarily the case that the point where the axes crossed (the origin) must be assigned coordinates (0.0, 0.0). It would be extremely useful to assign the initial parameters of the digitizing system so that the (x,y) coordinates as read from the digitizing table were exactly the correct UTM (x,y) coordinates, thus eliminating a coordinate transformation in a future step. This was in fact accomplished by assigning the origin (i.e., crossing of the x and y axes) to be (1600,1520) rather than (0,0). This maneuver in effect moved the origin down and to the left of where (0,0) should be, but still minimizes distortion. Scale in both the x and y directions were then set by appropriate calculations.

One last artifice was used in assigning the parameters for the digitizing system. As was noted before, the coordinates were registered as real numbers. In the data base, coordinates must be integers (i.e.,

Measurements Taken from
Coordinates of Elevation Data
(UTM Coordinate System)

NOTES:

1. Ratio of the Length of the East Side to the West Side is

$$\frac{b}{b+95.1} \approx 0.9985$$

North to South is

$$\frac{a}{a+78.7} \approx 0.9988$$

2. The Sides are $a = b = 12$ Miles

Measurements Taken from
Planimetry Map (Modified Polyconic)

NOTES:

1. Ratio of the Length of the East Side to the West Side is

$$\frac{b}{b+80.0} \approx 0.9987$$

North to South is

$$\frac{a}{a+65.6} \approx 0.9999$$

2. The Sides are $a = b = 12$ Miles

Figure 2. Comparison of Boundaries In Two Coordinate Systems (Exaggerated for Clarity)

subscripts in a two-dimensional matrix). Thus, there was required a real-to-integer conversion somewhere with numbers preferably rounded off. It is an easy process to use the computer to convert from real to integer, but it requires a little more effort to round off numbers. Since the digitizer must output real numbers regardless, we decided to have the computer do the real-to-integer conversion, but have the digitizer do the rounding. Since rounding can be accomplished by adding 0.5 and then truncating, and also since the computer would already be truncating in performing the real-to-integer conversion, all that was needed was to add 0.5 to both the x and y coordinates. The easiest way to accomplish this was to again shift the axes by setting the "origin" to (1600.5, 1520.5). So, the x and y coordinates registered by the digitizing system only needed to be truncated to integers and they were the actual matrix of coordinates needed.

After the digitizing system was set up, the accuracy of the location of the coordinate system was verified. The technique used was to extend lines of constant UTM coordinates across the map in both directions, as defined by the UTM tick marks on the USGS maps. Then the digitizer was used to determine the coordinates of various points. Since one could independently calculate what these coordinates should be, results could be compared to verify that the system was properly calibrated.

It was decided that the best way to digitize would be to completely digitize closed boundaries and only digitize each line once. This approach worked fine for what were called "island" boundaries, i.e., regions completely surrounded by the same category.

However, when regions shared a common border it was impossible to digitize the lines both as closed boundaries and also digitize each line only once. This occurred when two regions share a common border, and also all around the edges of the map.

To solve this problem, it was decided to digitize this problem case as three line segments with two common end points, or "triple points," rather than digitize the common line twice. This would have resulted in extra digitization and the possibility of more errors in duplicating the line.

After this decision was made, it was necessary to decide what to use for a "Z" value. Three digit categories had been previously assigned to all regions on the map[1] (000 for the border of the map). If there were only "island" type regions, the category inside the island could be easily assigned. But because of the problem case, the choice was made to assign to each line the concatenation of category values for the categories on either side of the line. Since each of the three-digit category numbers ended in zero it was decided to treat them as two-digit numbers to save space (i.e., 500 → 50, 350 → 35, etc.). Therefore, the concatenation yields a four-digit number for a Z-value for any line (i.e., 3050, 5060, 7000) and a six-digit number for a Z-value for all triple points, since each triple point is associated with three line segments.

In performing the digitization, it was realized that the output digitized data would have to be processed further. With this in mind,

---

[1] Davison, E., V. H. Kaupp, and J. C. Holtzman, "Baseline of Planimetric Data Base Construction: Pickwick Site," TR 319-2, Remote Sensing Laboratory, The University of Kansas, July, 1976.

every effort was made in the digitizing system to insure that it would not be necessary to perform an extra processing step later. This was the rationale behind carefully setting up the axes and assigning the origin to (1600.5, 1520.5). One more feature of the NOVA CECASCII digitizing system was used to advantage. The boundaries being digitized were those surrounding regions to be filled in later. It is required that the boundaries be connected, i.e., not have gaps or holes in them. Formally, this is represented by

$$1 \leq |x_i - x_{i+1}|^2 + |y_i - y_{i+1}|^2 \leq 2$$

and

$$z_i = z_{i+1}$$

Besides not having gaps in boundaries, it was desirable to eliminate duplicate points. This result can partially be accomplished by setting the digitizing system to "continuous" line mode with the distance check set to one scan line. This (supposedly) guarantees "connection" with a minimum of duplicate points.

The actual digitizing of the map took place in two distinct phases. It was decided that the information present on the planimetry map could be partitioned into two classes; regular category data and hard target (cultural) data. These hard targets consisted of all roads, railroads, houses, and the dam, blockhouse, and locks. It was decided prior to digitizing the map that it would be best to process the category data first and create a data base without hard targets, and then go back later and superimpose the hard targets on this data base. With this strategy in mind, it was decided to digitize the map twice, the first time to extract only categories and the second time to get the hard targets.

However, even though the digitizing was split up, it still required several sessions just to digitize the categories. It finally took over 14 hours of on-line digitization in more than three sessions to completely digitize the category data. At the end of every session, the operator

(of the digitizing system) was required to physically remove the map from the table and remount the map at the beginning of the next session. This could have been a source of critical error, however careful precautions taken in remounting the map and reassigning the axes kept any error to 0.2-0.3 scan lines average in all directions (see Table 1). When compared against the width of one pencil line (~0.5-2.0 scan lines wide), this error is negligible.

TABLE I

Four Corner Points of Map on Scan Line (x,y) Coordinates
Taken at Two Different Digitizing Sessions

	Coordinates				
	West Side of Map		East Side of Map		
	x	y	x	y	
Session 1	48.0	3159.6	3144.5	3104.1	North Side
Session 2	47.5	3159.8	3144.6	3104.1	
Error	(-0.5)	(+0.2)	(+0.1)	(+0.0)	
Session 1	10.5	66.4	3107.2	14.1	South Side
Session 2	10.3	66.7	3106.5	14.2	
Error	(-0.2)	(+0.3)	(-0.7)	(+0.1)	

When the digitization of the category data was completed, the information was contained on six tapes.

The last of the categories to be digitized were the hard targets. The digitization method was altered slightly to accommodate these data. The dam, blockhouse and large industrial building appear in the same form as category data, i.e. closed boundaries. Therefore, they were handled the same way as the category data. However, roads and railroads appear on the map only as lines. Therefore, that is exactly how they were digitized. The various house sizes presented a problem. There were mainly small houses, but there were also a few larger buildings. Since 20 foot square resolution points, were employed, it was decided to digitize all houses

as single points except in the case of very large buildings which were represented as line segments. This approach was taken with the idea that houses could "grow" or expand in the data processing step. The same reasoning also applied to the decision to make roads and railroads only one scan line wide. It required four hours in one session to digitize all the hard targets and the data was stored on one tape. The z value specifying the radar backscatter category was constant for all hard targets because all were treated as isotropic scatterers with σ° set to +20 dB.

PROBLEMS

To clarify any doubts as to cateogry identification or other questions pertaining to the construction of the data base, the assistance of the interpreter involved in its construction was made available to the digitizer operator. As the digitizing progressed, various questions arose as to resolution requirements and differentiation of quite similar targets which this close interaction proved useful in eliminating.

The overall accuracy of the NOVA system is approximately three one-thousandths of one inch. This is obviously much greater accuracy than can be duplicated by a human interpreter using hand-drafting techniques. Problems arose due to the fact that in a system with more than 100 scan lines per inch, an exceptionally wide pencil line might cover several scan lines. This did not however, produce any major effect as the operator was able to treat such things as a single scan line width at the direction of the interpreter.

Of greater consequence was a problem involving perception on the part of the digitizer operator. In many cases lines derived directly from a boundary on a topographic map (such as forest/non-forest boundaries) became confusing to the operator even though one would have had little trouble following such lines on the map itself and in fact often did so in the course of normal duties. It is apparent that some perceptual differences appear when information is converted from a very familiar format (colored topographic sheets) to one which is less familiar (black and white line drawings). This should be considered as an influencing factor whenever methods such as these are implemented.

To resolve the above problem, the operator requested that the planimetry be color-coded by category. The task was accomplished in approximately eight hours of working time. Subsequently, the operator experienced considerably fewer problems in completing the digitizing effort.

VALUE OF THE FINISHED PRODUCT

The utility of a data base produced by the methods described above must be judged according to the time required as well as the accuracy with which information is preserved. The time required to digitize the data base is as follows:

Definition of Coordinate System	1 hour
Grid Layout (UTM)	2 hours
Digitization Time	
Distributed Targets	14 hours
Cultural Targets	4 hours
Evaluation	2 hours
Total	23 hours

This total reflects the extra time spent due to the problems mentioned earlier as well as the actual time on the final version. If such problems were eliminated, the total time could be reduced by a factor of approximately eight hours for a site of equal area and complication.

The accuracy of such a data base is subject to the limitations placed upon it by the use of topographic maps as the base for planimetry, combined with the presence or lack of detailed ground truth for the area. When these limitations are considered it is apparent that this is a practical, if time consuming, method of data base construction. Provided that reasonable care is taken in drafting, human error factors can be held to a minimum. This is borne out by the fact that spot checks on the digital plotter revealed errors on the order of no more than ten scan lines in the geometry of the data base grid. This amounts to an error factor of one in

three hundred. As quality of imagery, ground truth and familiarity with the subject matter on the part of all persons involved improve, the accuracy of the data base content can be improved correspondingly. Concurrently any step in the set of procedures which can be further automated, while maintaining accuracy, would be a worthwhile effort.

APPENDIX E

INCREASED RESOLUTION OF PLANIMETRIC DATA BASE: PICKWICK SITE

The following technical report (TR 319-21) prepared by the Center for Research, Inc., University of Kansas, is included in this appendix to provide technical information concerning construction of an improved resolution category data base of the Pickwick site.

THE UNIVERSITY OF KANSAS CENTER FOR RESEARCH, INC.

2291 Irving Hill Drive — Campus West
Lawrence, Kansas 66045

Telephone: 913-8644832

INCREASED RESOLUTION OF PLANIMETRIC DATA BASE: PICKWICK SITE

Remote Sensing Laboratory
RSL Technical Report 319-21

E. Davison
V. H. Kaupp
J. C. Holtzman

March, 1977

Supported by:

U. S. Army Engineer Topographic Laboratories
Fort Belvoir, Virginia 22060
CONTRACT DAAG 53-76-C-0154

ABSTRACT

A planimetric data base of a test site centered around Pickwick Dam was constructed by a photointerpreter working from panchromatic photography and maps, using manual feature extraction techniques. This data base was constructed at a level of detail appropriate for its use as a baseline study for the evaluation of automated feature extraction techniques, and for the simulation of coarse resolution radar systems. To facilitate the simulation of finer resolution radar imagery, this baseline effort was improved by the addition of finer detail in land-use classification. The categories which were delineated consisted of targets with similar microwave backscatter response. The actual feature extraction time required in the original effort was 28 hours, with an equivalent period being devoted to the inclusion of the additional detail.

1.0 INTRODUCTION

A ground truth data base encompassing approximately 144 square miles was prepared earlier in this study, for a test site centered around the Pickwick Dam area, located in Tennessee[1,2,3,4]. This data base was developed by a photointerpreter working from panchromatic photography and topographic maps of the site, using manual feature extraction techniques[2]. This planimetric data base was subsequently digitized[4] with the resulting digital matrix being merged with a matrix of elevation[3] data of the site. The end product[5], containing both category and elevation data, has been used to produce a digital radar simulation. This data base has provided the baseline against which automated feature extraction techniques have been and are being tested, as well as providing input for other radar image simulation studies.

The ground truth data base as originally constructed contained 23 categories pertaining to the planimetry of the site[2]. These categories (Table 1) were selected according to the relatively gross level of discrimination capabilities of the system to be simulated (>100 feet). The purpose of this study is to further define and delineate these categories. This allows the simulation of a system exhibiting greater resolving capabilities (~60 feet). The major change in the original planimetry categories is the subdivision of the category identified as agricultural land (Table 1). In addition, numerous changes were made in the detail with which individual areas were delineated. In some cases a given area was expanded, reduced or eliminated. Other minor features (e.g., individual

---

[1] Komp, E., M. McNeil, V. H. Kaupp, and J. C. Holtzman, "Medium Resolution Digital Ground Truth Data Base," TR 319-5, Remote Sensing Laboratory, The University of Kansas, August, 1977

[2] Davison, E., V. H. Kaupp, and J. C. Holtzman, "Baseline of Planimetric Data Base Construction: Pickwick Site," TR 319-2, Remote Sensing Laboratory, The University of Kansas, July, 1976.

[3] McNeil, M., V. H. Kaupp, and J. C. Holtzman, "Digital Elevation Data Base Construction: Pickwick Site," TR 319-3, Remote Sensing Laboratory, The University of Kansas, July, 1976.

[4] McNeil, M., V. H. Kaupp, and J. C. Holtzman, Digitization of Pickwick Site Data Base," TR 319-4, Remote Sensing Laboratory, The University of Kansas, February, 1977.

treelines) were included where they had been ignored previously. The resulting planimetry matrix was digitally superimposed upon the original matrix. This combined matrix was merged with the elevation tape to produce a revised data base having a spatial category resolution of approximately 60 feet.

TABLE I

Pickwick Test Site Planimetry Categories[2]

Category Identification Number	Category Description
100	Group Targets - Roads
110	Heavy duty improved roads (none present)
120	Medium duty improved roads
130	Light duty improved roads
140	Unimproved roads
200	Group targets - Railroads
230	Fish Pond Dikes
240	Water Plant Plumbing
300	Group Targets - Water Bodies
310	Small Impoundments
320	Small streams and rivers
350	Large streams and rivers
360	Large Impoundments (Pickwick Lake)
400	Group targets - Marshes (see subgroup)
450	Wooded Marshes
500	Group Targets - Forested areas
600	Group Targets - Agricultural Land
700	Group Targets - Grass-covered areas (parks, etc.)
800	Pickwick Dam
810	Blockhouse
820	Spillway
900	Small buildings
910	Large buildings

Note: The planimetry categories listed in this table represent the level of detail (>100 feet for radar backscatter categories) present in this data base.

[2] Davison, E., V. H. Kaupp, and J. C. Holtzman, "Baseline of Planimetric Data Base Construction: Pickwick Site," TR 319-2, Remote Sensing Laboratory, The University of Kansas, July, 1976.

2.0 SITE DESCRIPTION

The Pickwick Dam test site is located in the Tennessee River Valley near the junction of the states of Tennessee, Alabama, and Mississippi. The area consists of a portion of the Tennessee River floodplain surrounded by open rolling hills. The boundary of the test site was defined as a twelve mile square centered on the northwest corner of the powerhouse at Pickwick Dam. The geographic area contained within this square comprises the 144 square miles of terrain examined to construct the data base.

Physiographically, the Pickwick site belongs to the Interior Low Plateau province. Within the site are smaller units belonging to the Highland Rim, the Tennessee Valley (including terraces of Recent age) and the slope of west Tennessee. Slopes in the area are diverse, varying from level to quite steep. Individual slopes are as little as 0 - 3 per cent in the floodplain and as severe as 25 - 45 per cent in upland sites east of the lake. Soil characteristics likewise vary, from poorly drained silty clay loams to excessively drained gravelly sandy loams.

Precipitation in the Pickwick area averages 54.5 inches annually. This is generally well distributed throughout the year, but with a slight late winter to early spring maximum, the peak month being March. Temperature trends show an average annual temperature of 61.6° F. The highest recorded monthly average is 80.4° F in July, with a record low average of 42.4° F in January. Individual extreme temperatures during the recording period (69 years) were 112° F and -12° F respectively. During the period, an average of 105 days per year received .01 inches or more of precipitation. The temperature fell below freezing an average of 75 days per year.

Activities in the Pickwick area include farming, lumbering, and paper processing, and some light commercial fishing. Transportation is well developed in the area with hard surface and gravel county, state and federal roads as well as water-borne traffic on the Tennessee River. River traffic is relatively heavy with more than 11,000 units passing through the locks at Pickwick annually. The area is crossed by one railroad spur which serves the papermill located near the town of Counce.

2.1 Planimetry Categories

The planimetric data base as originally constructed included a limited number of categories for several reasons. The principle reason is that the data base was constructed to serve as the ground model for simulation of a relatively coarse radar. Also the photointerpreter involved in the study was not familiar with the region, particularly its agriculture. Available ground truth information was, at the time, limited or nonexistent. Visits to the site with a detailed ground survey were not within the scope of the study. The result was a generalized range of categories (Table 1) as derived from topographic maps and supplementary small scale aerial photography.

The data base presented here represents the original planimetry with modifications and additions based upon more detailed examination and additional information. Categories for the assignment of backscatter values, which were originally constructed at relatively gross levels of classification were further defined. The most significant change was the subdivision of the category designated as undifferentiated agriculture. Specifically, the former category number (600) was redefined to discriminate areas of fallow or recently cultivated ground, and areas planted to soybeans, corn, milo, wheat or orchards (Table 2). In addition, the category used to delimit areas of grass, pastures, and similar ground cover was modified to include a local golf course. This was done to allow for the relative smoothness of the regularly maintained grass covered surface in such an area. Other modifications consisted of the assignment of identification numbers to structures associated with recreational facilities in the area.

2.2 Input Data Sources

Information was gathered from several sources to establish the characteristics of the ground scene by way of elevation and backscatter response. The data base, an enormous matrix of data points, was constructed with the aid of aerial photography, topographic maps, soil and crop surveys.

Aircraft imagery was acquired from Mark Hurd Aerial surveys by the Engineering Topographic Laboratories (ETL) and used by the photo-interpreter in the construction of the original data base. This was also used as input in the new data base. The imagery was supplied at a scale of 1:100,000 in a black and white contact print format. In addition, a set of negative transparencies of the same flight have been used to make enlargements of portions of some frames. The additional detail provided was useful in compiling information for the data base.

No radar imagery was used in the construction of the data base. This is an important aspect of the philosophy of simulating radar imagery, because part of the value of the simulation study at RSL is that it does not presuppose the existence of radar imagery. The six topographic maps used in the construction of the original data base were also used in the revised version. These remain in the sole map inputs.

In an effort to obtain additional information about the test site, especially its agriculture, several county extension agents and Soil Conservation Service (SCS) officers were contacted. The SCS supplied soil surveys for the counties in and near the test site. These surveys provided information pertaining to drainage attributes of the soils in the area and other geographical information. The County Extension Service offices in Hardin County, Tennessee and Tishimingo County, Mississippi were contacted by letter and by telephone concerning the general crop types in the area. Some information was also obtained regarding proportions and distribution when actual radar imagery was generated for the site.

2.3 Feature Extraction Techniques

As in the case of the original data base, only manual feature extraction techniques were employed in the creation of the planimetry map. The tracing medium was of an acetate base type. The scale of the planimetry map, was equal to that of the original data base and the six topographic maps used as input (1:24,000).

As stated, the primary objective in raising the level of detail in the planimetry map was to subdivide the general vegetation category. The assignment of backscatter categories was made difficult by several factors. No ground truth was taken when an actual radar mission was flown

(April, 1976). This eliminated the possibility of acquiring real time data pertaining to crop types on a field-by-field basis. The high-altitude photography used in the site analysis had been flown the previous October (October, 1975). Thus, a gap of nearly six months existed between the two flights. The photointerpreter was forced to deal not only with the identity of a crop for given fields, but the effects of seasonal change and such events as crop rotation. Identification of crops from panchromatic photography alone is extremely difficult at the time of year the imagery was flown. It was not possible to assign real-time identification to the crops in reference to the flight of the radar system. Nor was it possible to provide absolute identification for the fields at the time of the photographic mission.

Because of the inability to assign categories strictly on the basis of interpretation criteria, information was sought from the County Extension Services. Several readily identifiable areas in the Pickwick site were selected and their boundaries and locations noted. The extension agents were then contacted and this information was given to them. Based on their knowledge of their home counties and, in one case a field check by one of the agents, the fields were identified. The resulting maps were returned to the Remote Sensing Laboratory and were analyzed by the photo-interpreter in an attempt to ascertain crop distribution. Using the identity of the fields examined by the extension agents and the greytones associated with them, the interpreter constructed a qualitative key for the grouping of other fields in the test site. Such a method is subject to several sources of error. Crops at emergence or during harvest periods may have similar greytones. Differences in greytone may be a function of soil moisture or morphology and are not always reliable indicators of crop types.

The result of the effort described above is a system of field boundaries with assigned crop types. Due to the noted problem of temporal factors including crop rotation, the resulting patterns may not correspond exactly to those found at another given point in time. Instead, they represent a modeled distribution of crops as they are usually grown in the test site, within the actual boundaries of fields.

In addition to the subdivision of agricultural areas, new categories were created for such cultural features as boat docks, boat houses and other facilities in recreation areas. Identification parameters in the photography used in the study did not permit absolute identification of material types and other attributes of these facilities. It was noted that in one area, the fairways of a golf course were quite visible. The boundaries of these fairways were transferred to the planimetry map so that backscatter values could be assigned and their smooth, grassy surface simulated.

The transfer of additional information to the data base was accomplished by means of an overlay. The planimetry map used in the preparation of the coarse resolution data base was secured to a drafting table with a second acetate sheet placed over it. The boundaries of the original planimetry map were then traced onto the second sheet. Major physical features were retained for reference, including river boundaries. To introduce control for alignment purposes, the registration marks for the UTM grid were transferred to the overlay. In several successive sessions, the other planimetry details (agricultural boundaries, deletions or changes in cells, etc.) were transferred. The resulting map was digitized with the data and coordinate system being stored on magnetic tape. This tape has been merged with one containing the data from the original data base. A comparative check of boundaries on the planimetry map and the overlay revealed an error of .6 scan lines at the corners of the matrix. The physical dimensions of the map and overlay measured approximately 36 inches on a side, representing approximately 3,000 scan lines.

3.0 CONCLUSION

The planimetric data base which has been constructed is suitable for the simulation of a radar system with moderate resolving capabilities (>60 feet). Its construction illustrates that it is possible to represent the major terrain features of the test site as they appear, with relatively little direct input and without field survey. Although problems were encountered in the construction of such things as the agricultural categories, it is anticipated that increasing the level of ground truth would increase overall data base accuracy accordingly. Accuracy in actual position and boundary location has been maintained in spite of the lack of more sophisticated cartographic techniques. This had posed a potential problem since the construction of the data base involved the matching of several topographic maps, published at different times.

APPENDIX F

MEDIUM RESOLUTION DIGITAL GROUND TRUTH DATA BASE: PICKWICK SITE

The following technical report (TR 319-5) prepared by the Center for Research, Inc., University of Kansas, is included in this appendix to provide the technical details concerning construction of the data base for the Pickwick Landing Dam site.

THE UNIVERSITY OF KANSAS CENTER FOR RESEARCH, INC.

2291 Irving Hill Drive — Campus West
Lawrence, Kansas 66045

Telephone: 913-8644832

MEDIUM RESOLUTION DIGITAL GROUND TRUTH

DATA BASE: PICKWICK SITE

Remote Sensing Laboratory
RSL Technical Report 319-5

E. Komp

M. McNeil

V. H. Kaupp

J. C. Holtzman

August, 1977

Supported by:
U. S. Army Engineer Topographic Laboratories
Fort Belvoir, Virginia 22060
CONTRACT DAAG 53-76-C-0154

ABSTRACT

A digital ground truth data base for radar image simulation studies was constructed of the topographic area in the states of Tennessee, Alabama, and Mississippi, surrounding the Pickwick Landing Dam. The area comprising the data base consisted of a square 12 miles on a side and was centered on the northwest corner of the power house building at the dam. The completed data base consists of a digital matrix representing symbolically the radar backscatter properties of the various different radar echo categories in the target area together with the appropriate elevation values of the terrain at each point. The matrix consists of more than ten million entries. Each entry contains the radar category and elevation of a point on the ground. Points on the ground, entries in the matrix, are separated by 6.25 m in both directions in a rectangular, orthogonal grid coordinate system. The radar backscatter category data have a spatial resolution >100 feet, and the elevation data have an accuracy on the order of 10 feet in the final data base matrix. The backscatter category data were produced[1] and digitized[2] in previous work. The work reported here converted these raw data into a final, complete digital matrix. The elevation data were pre-processed in earlier work[3]. The category data matrix was combined with the elevation data matrix to produce a final, complete ground truth digital data base of the Pickwick Landing Dam Site.

---

[1] Davison, E., V. H. Kaupp, and J. C. Holtzman, "Baseline of Planimetric Data Base Construction: Pickwick Site," TR 319-2, Remote Sensing Laboratory, The University of Kansas, July, 1976.

[2] McNeil, M., V. H. Kaupp, and J. C. Holtzman, "Digitization of Pickwick Site Data Base," TR 319-4, Remote Sensing Laboratory, The University of Kansas, February, 1977.

[3] McNeil, M., V. H. Kaupp, and J. C. Holtzman, "Digital Elevation Data Base Construction: Pickwick Site," TR 319-3, Remote Sensing Laboratory, The University of Kansas, July, 1976.

INTRODUCTION

The Point Scattering Model, a radar image simulation model, has been implemented on a high-speed digital computer. One principle input requirement of this model is a ground truth data base of the target site for which a radar image is to be simulated. The ground truth data base is a symbolic representation of the planimetric features and topography of the terrain in the target site. As the model has been implemented on the digital computer, so must the ground truth data base be in a digital format. The ground truth data base is, then, a digital matrix containing position information, radar backscatter category, and elevation for every point on the ground.

This report presents the work and sequence of events to construct a digital ground truth data base for the topographic region in a 12 mile square (144 square miles) centered on the Pickwick Landing Dam area located in the states of Tennessee, Alabama, and Mississippi. The radar category planimetry data for this site had been extracted and reported previously[1]. The boundaries defining homogeneous radar backscatter terrain features had been digitized and reported[2]. The digitization of these boundaries was performed using a manually operated large-table digitizer and resulted in several computer-compatible magnetic tapes of digital boundary data. These data were not the final digital ground truth input data required by the simulation software. In fact, these boundary data needed a lot of correction and the data were extensively manipulated in the process of forming the required digital matrix. This work is reported here.

In addition to radar category planimetry data, elevation data are also required to be included in the ground truth data matrix. Elevation data had been acquired previously and had been manipulated into the desired format. This work with the elevation data has also been reported

---

[1] Davison, E. C., V. H. Kaupp, and J. C. Holtzman, "Baseline of Planimetric Data Base Construction: Pickwick Site," TR 319-2, Remote Sensing Laborabory, The University of Kansas, July, 1976.

[2] McNeil, M., V. H. Kaupp, and J. C. Holtzman, " Digitization of Pickwick Site Data Base," TR 319-4, Remote Sensing Laboratory, The University of Kansas, February, 1977.

previously[3]. It remains in the work reported here to combine the digital matrix of radar category data with the digital matrix of elevation data. Merge of these two matrices results in the final ground truth data matrix (data base) of the Pickwick Dam site.

The final Pickwick data base contains an entry in the matrix for each point on the ground at 6.25 m intervals. This means that the radar backscatter category and elevation of the terrain is specified every 6.25 m in both the range and azimuth directions. This resulted in the ground truth data matrix containing 3169 elements in each direction for a total number of entries in excess of ten million.

The final resolution of the Pickwick data base was constructed to be approximately 100 feet in both range and azimuth for radar category data, and approximately 10 feet in both range and azimuth for elevation data. This means that, even though there is an entry in the data matrix for every 6.25 m (20.5 feet) interval on the ground, the spatial resolution of the category data is estimated to be greater than 100 feet. Also, the accuracy of the elevation data is estimated to be half the contour interval (20 feet) over which the data were interpolated.

DATA PROCESSING PHILOSOPHY

Due to the extreme size of the ground truth data matrix (more than ten million points), it was impossible to place this matrix into computer core memory at once. Even if 100 k of core memory were available, one hundred subimages would have been required. This approach was rejected as being both time-consuming and expensive for forming the ground truth digital matrix from the input digital boundary tapes[2].

Therefore, another approach to the problem was developed. All of the useful information on the map was now contained in the digitized boundary

---

[3] McNeil, M., V. H. Kaupp, and J. C. Holtzman, "Digital Elevation Data Base Construction: Pickwick Site," TR 319-3, Remote Sensing Laboratory, The University of Kansas, July, 1976.

[2] McNeil, M., V. H. Kaupp, and J. C. Holtzman, "Digitization of Pickwick Site Data Base," TR 319-4, Remote Sensing Laboratory, The University of Kansas, February, 1977

lines. One could reconstruct any vertical line on the map (range direction, or lines of constant x) if he knew the y-value of each boundary line crossing it and the corresponding category of both sides of the boundary (see Figure 1). The general concept of this approach was to sort the input data points according to their x-coordinates (azimuth direction) and from there reconstruct each vertical line (range direction, or y-coordinates) independently. (This assumed that the lines would be continuous in the x direction.)

Before the boundaries of the radar category planimetry map were digitized, a general approach for the software package was outlined so that the boundaries could be digitized in the appropriate format. Each line on the map represented a division between two categories. One possibility was to digitize each line twice (once for each category). This could result in the lines crossing one another and, or gaps existing between the two lines. It was decided to digitize each line once and assign to it both categories. Notice, then, that the boundary for a homogeneous area may have more than one value depending on the surrounding categories (Figure 2).

In addition, some boundaries also represented discrete targets, such as roads. Furthermore, these discrete targets might be as small as one resolution cell wide or they might be larger. It was decided to handle all discrete targets in a separate digitizing pass and superimpose them on the category matrix after the homogeneous areas had been completed. This way separate software could be developed to handle the special problems of discrete targets. The superposition solution seemed valid because discrete targets should take precedence over local ground-cover in the data base. Where discrete targets also represented boundaries between natural category changes, they were digitized as natural boundaries on the category map.

In general, the beginning and ending points of digitized line segments were of crucial importance. If a boundary represented a completely enclosed area, the ending point should equal the starting point. This would be nearly impossible for the digitizer to accomplish, so these points were uniquely marked and it was left to the computer software to connect them properly. Other points were "vertex points" where three

F-1 Category Map With Vertical Scan Line

The category information for the line representing X = 60 can be represented with the following data:

1) the Y-value where a boundary crosses the vertical line
2) the category at that point.

Category	from Y-value	to Y-value
60	0	18
50	19	31
60	32	40
10	41	50
60	51	77
30	78	82
60	83	100
50	101	132
60	133	140

Figure 1.

F-2 Example of Multiple Categories Enclosing A Homogeneous Region

The closed curve enclosing the area designated by category 50 must be digitized as four separate line segments to reflect the changes in the surrounding categories:

from	to	Category
A	B	50-40
B	C	50-20
C	D	50-30
D	A	50-20

Figure 2.

categories met. This single point was part of two or more separate lines. It was important that these points coincide for all lines without overlapping. This was also left to the computer software. The digitizer simply labelled these as special reference points (Figure 3).

For the success of this approach for creating the digital matrix in which we sort the boundary data according to their x-values and then build the matrix across their y-values, it was essential that the digitized points form continuous boundaries (see Figure 4). Although the digitizing mechanism was operated in the continuous mode and sampled points 100 times per second[2], analysis of the boundary data stored on tape showed many discontinuities. To rectify this error, the computer software routine "FIX" was developed to process the raw data. It compared adjacent points on each boundary line and if the absolute value of the difference in the x-values or the y-values was greater than one, a software routine, "CONNECT", was called. This routine returned the coordinates of those intervening points interpolated to make the border continuous. These points were subsequently added to the raw data points. The first implementation of this solution failed to maintain the sequential order of the data points and we were forced to rerun this segment with some alterations so that the data were recorded in a sequential manner as required for later programs.

Another aspect of acquiring continuous boundaries was the problem of joining beginning and ending points of closed boundaries and causing common boundaries to meet each other properly. Each vertex point (where two or more boundaries met) was labeled as such by the digitizer with a unique reference number. If the starting or ending point of a line was one of these vertex points, the digitizer labeled it by a special reference number. The program FIX maintained a table of these points and then called CONNECT to connect the first and last points actually digitized on a line. Reference numbers of zero were used to signify that the following boundary formed a closed border (i.e., the first point equals

---

[2] McNeil, M., V. H. Kaupp, and J. C. Holtzman, "Digitization of Pickwick Site Data Base," TR 319-4, Remote Sensing Laboratory, The University of Kansas, February, 1977.

Point A must be part of 3 separate line segments:

1) AB Category 20 50
2) AC Category 20 60
3) AD Category 60 50

Figure 3.

Example of "Vertex" Point

If there was a gap in boundary AB (as shown at X = 30) the category would be incorrectly specified as 50 for all Y-values because no category change was indicated in that vertical line.

Figure 4.

Continuous Boundary Lines

the last point). In this case the first point was saved and, before leaving that boundary, CONNECT was called to connect this last point to the first point. To avoid problems caused by overlapping ends at the beginning and ending of a closed boundary, the digitizer stopped digitizing the boundary shortly before returning to the starting point and, thus, allowed the software to fill in the intervening gap.

This completes a brief description of the operating procedures, provided to the digitizer, that were designed to produce an output product compatible with the software used to convert these digital boundaries into a matrix of digital ground truth values.

DIGITAL DATA MATRIX CONSTRUCTION

If the input data were perfectly accurate, the software routines used to convert the digital boundaries into a digital matrix would be very simple. But considering the number of points collected and the detail of the map, a certain percentage of human errors had to be expected. Therefore, the software processing the raw data was designed to detect errors. Since the program had no information on the actual scene content, it could only check for inconsistencies in the data. These error detection checks included the following: (1) excessive number of points required to connect adjacent points on the input data; (2) x or y values or category out of range; (3) no common category when a line met a vertex point; and (4) reference point number greater than maximum value. In addition a complete summary of boundary lines was produced for comparison to the map. These diagnostics uncovered numerous errors. A number of times the reference point for the beginning or ending of a line was mistyped. This caused the program to connect two unrelated points on the map and left the proper border discontinuous. For closed boundaries, the digitizer forgot to use the reference number for the first point and the program interpreted the first data point as the reference point. None of these errors could be corrected automatically. The only solution was to resort to the original map and deduce the proper data from surrounding points. These data, then, had to be inserted in a rather ad hoc manner so that the remaining data would be properly interpreted by the program.

Once the digitized boundaries of the radar category planimetry map were transformed into an acceptable format, they were input into a program called "FIX." The input data consists of the x-y position and category of each point on each continuous boundary in sequential order with separators between different boundaries. The purposes of FIX were to reformat, check and correct, if needed, the data. Among its functions were:

(1) Check (x,y) coordinates of all points (i.e., $0 \leq x \leq 3169$ and $0 \leq y \leq 3169$);

(2) Truncate (x,y) coordinates from real format to integer format;

(3) Find and eliminate duplicate points;

(4) Keep track of starting and ending points of lines (put triple points into special array);

(5) Connect points;

   (a) Connect starting and ending points of closed boundaries;

   (b) Connect line segments to appropriate triple points;

(6) Output on paper information about each line and a lot of other "housekeeping" information;

(7) Split output onto three tapes according to x value

Tape Number	x Value
1	0 - 1055
2	1056 - 2111
3	2112 - 3169

This is to keep the number of points per tape down to a manageable level;

(8) Check and correct for tangent points. This implies that the direction of each line be known at all times.

FIX required 17 versions before it finally ran without error. Several of these versions were required to alter the tangent check and correct subroutine. However, many of the problems arose from faulty input data. Most errors were minor, but they caused errors in FIX and required that a new version be written with special checks. Also, the errors were found one at a time, not all at once. Some of the problems incurred were:

(1) Faulty, multiple, and missing separators between different boundaries;

(2) Incorrect labeling of categories (bad value);

(3) Incorrect labeling of starting and ending nodes of line segments;

(4) Incorrect labeling of seven triple points;

(5) One line segment completely missing.

An additional complication came up unexpectedly. The "continuous" line mode of the digitizer should not have allowed any disconnected lines. However, this was not the case. The "CONNECT" subroutine was called in excess of 26,500 times to repair discontinuities that should not have existed. There are two explanations for this behavior. First, if the digitizer operator moved the cursor too fast around a line it is possible that small gaps might occur. But this cannot explain all the gaps. After much search, it was finally discovered that the timing mechanism which controls the sampling time had an unexpected failure mode. After "long" periods of time ($\gtrsim$30 seconds), the timer lost synchronization resulting in (evidently) longer gaps between samples. This explanation agrees with our observations about the data. Many of the larger gaps occurred toward the end of long boundaries. This unfortunate circumstance didn't affect the data base very much because the connect subroutine effectively completed the lines. However, much computer and programming time went into identifying and fixing this problem.

FIX and the ad hoc manual work were very important parts of this overall effort to create a digital matrix from digital boundaries. A large amount of data were corrected by FIX. The final statistics are:

1. Number of points input to FIX ~90,000
2. Number of lines and line segments input to FIX 714
3. Number of triple points input to FIX 299
4. Number of points output by Fix 185,711
   - Number on tape 1 (left) 90,262
   - Number on tape 2 (center) 62,663
   - Number on tape 3 (right) 32,786
5. Number of duplicate points discovered by FIX 9,730
6. Number of calls to "CONNECT" by FIX 26,519
7. Number of points added by CONNECT ~95,000
8. Computer core memory requirements 11K

There are several points to note in these statistics. First, the number of final points was doubled by adding connecting points. Second, each call to CONNECT averaged inserting four points. Third, approximately one-tenth of all input points were duplicates. Last, the distribution of points is heavily weighted towards the leftmost third of the map, and lightly weighted toward the rightmost third which is evidenced by the number of points stored on each tape. This observation is corroborated by an examination of the map, which reveals much complexity in the west (left side of the map) and more simplicity of boundaries toward the east (right side of the map).

Once the program FIX ran without error messages, one still could not be certain that the data were error free. Plot routines were written and run to produce a visual output of the processed (fixed) data points. These plots were compared to the original radar category planimetry map and found to be in good agreement within the limits of resolution provided by the plotter. At this point we were confident of the accuracy of our data and moved on to the next step of the process.

Horizontal Tangent Points

Horizontal tangent points were another potential problem that was recognized and corrected at this point in the work. A horizontal tangent is a point of a boundary line such that there is no point of that same boundary above or below it (Figure 5). A vertical line (line of constant x, range direction) drawn through that point would intersect the enclosed area at only a single point. This would cause problems because the final software routine, "FILL", would be anticipating the top of the border it had just encountered to be somewhere above in that line, but would never find it. Because the data were sequential, tangent points could be rather easily recognized. A point is a tangent if its x-value is greater than (or less than) the x-value of both the point preceding it and the one following it.

To eliminate tangents, a software routine, "TANCON," altered by one the x-value of each tangent found so that it would belong to the previous point. This insured that there would be at least two points from

Horizontal tangent at X = 6 (the vertical line at X = 6 intersects the area of category 50 at only one point).

By standard techniques the category assignments for the line X = 6 would be

Category	from Y	to Y
60	0	4
50	5	8

when the category should be 60 from Y = 0 to 8.

Figure 5.

Horizontal Tangent Points

a boundary in a given vertical line (Figure 6). Altering the x-value in this way was justified since the point was a boundary point and the category for that cell was indeterminant. It belonged to one of the two categories associated with it, but one could not determine which one.

Further tests revealed that this method of detection and solution was not exhaustive. Tangents could still remain for a number of reasons. There was the possibility of a "double tangent" if the boundary moved left and right for two or more cells with the same y-value (Figure 7). The first order solution only moved the tangent back one cell but still left a problem. Occasionally, a tangent could be created by FIX when it closed a boundary which could go undetected because we did not have left and right information on leaving the starting point and returning to that point as the ending point.

Data Verification

Before proceeding any further, a test for accuracy of the data was needed. At this point, the test served two purposes. First, we needed a visual verification of the accuracy of the digitization work done. We specifically needed to check for three possible errors:

(1) Did the digitizer follow all borders with sufficient accuracy to provide the required accuracy and detail needed for the data base?

(2) Were any boundaries, or portions thereof, inadvertently omitted?

(3) Had any superfluous points been included in the input data?

Second, a test was needed for the performance of the software which had processed the raw data.

(1) Had the beginning and ending of boundaries been correctly sensed and connected?

(2) Had the CONNECT routine joined discontinuous points without adding unwanted points to the data stream?

(3) Had the modification of perceived errors in the raw data caused other errors of their own?

Resolution of the horizontal tangent problem depicted in Figure 5.

Figure 6.

Example of "double tangent" at X = 5, X = 6.

Figure 7.

A visual output of the entire data stream at this point seemed to be the only viable test available. Software routines were written to plot the stored boundary map at twice the original scale. We compared this plot to the original map and verified the accuracy of the data.

First Data Ordering

After verification of the accuracy of the data, the next processing step taken was sorting the data by their x-values. Each set of points with the same x-values represented one vertical line of the data base and constituted one range scan of the radar over the target site. From this information that line could be reconstructed to its full size of 3169 points.

At this point there were approximatly 180,000 data points that needed to be sorted into one of 3169 bins (one for each vertical line). Because of the large core requirements, this was a difficult sorting operation. The first step taken to make the data easier to handle was dividing the data into six segments and storing each on a different magnetic tape. In this way only about 30,000 points were handled at a time.

If each vertical line had the same number of points, we could make an array of the proper size for each x-bin and, thus, could easily do the job. But the points were not evenly distributed. One line contained as few as ten points and another had well over a hundred points. This presented the problem of memory management. If we used arrays of fixed size so that no bin overflowed, memory requirements would have been excessive. Therefore a version of heap storage memory management was implemented in the Fortran program. Each x-bin was given a fixed size array and, in addition, a reservoir of overflow arrays was allocated. When an x-bin was filled the next available overflow array was attached (via pointers) to that x-bin for the remainder of the points in that line. In a further effort to reduce the volume of data to be handled, the y-value and the two categories associated with each point were packed into a single word. Also the x-value was dropped since that information was now implicit in which x-bin each data point was stored[2].

[2] McNeil, M., V. H. Kaupp, and J. C. Holtzman, "Digitization of Pickwick Site Data Base," TR 319-4, Remote Sensing Laboratory, The University of Kansas, February, 1977.

There was, however, a problem to be solved more severe than one of size. Not all points in a given x-bin should be retained. If a portion of a boundary forms a vertical line segment by running straight up a vertical line (x-bin), we should only keep the bottom and top point of that segment. Intuitively, all that should be kept is the bottom and top point where the given line crosses an area. If all the points were kept, the "FILL" routine, which later expands these data into a complete matrix, would alternate categories all along this line because the way the FILL routine was designed would cause it to change categories everytime it encountered another point. It interprets each point as meaning the beginning of a new field.

Another problem occurred when a single boundary crossed a given vertical line several times. All these intermediate changes as well as the top and bottom points must be recorded and accounted for. See Figure 8 for an illustration of these difficulties.

A close observation of these problems reveals that it was not necessary to keep both the top and bottom points of a vertical boundary segment. In the case illustrated in Figure 9, we should not even keep the top point of the vertical boundary. If the routine has already changed to the category of the interior at the lower point as shown in the figure, then the top point of the vertical line segment does <u>not</u> represent a change in category.

Because of the serial nature of the data, it was possible to make rules guiding decisions on which points to keep and which to throw away in vertical lines. These rules were as follows:

(1) If the x-value of the current point differs from the point preceding it, then keep the current point. (First point of all vertical lines is saved. This is necessary to maintain continuity in the x-direction.

(2) If the x-value of the previous point and that of the following point are the same as the x-value of the current point, then throw away the current point. (This point is part of the interior of a vertical line segment and therefore is not needed.)

(3) If the x-value of the current point equals that of the previous point but differs from the next point, then this point is the end point of a vertical line segment. This point may or may not be

Example of a single boundary recrossing a vertical line (X = 7) several times.

Figure 8.

Look at boundary changes in this vertical line, X = 8. If all end points (Y = 2, 6, 8) are saved, the following error will be created by the fill routine:

Category	from Y	to Y	
60	0	2	
50	3	6	
(60	7	8)	ERROR
(50	9	9)	

Figure 9.

Vertical Boundary

kept; more information is needed to make a decision. If the direction of the boundary before the vertical line segment was from left to right and, on leaving the vertical line the boundary continues in the same direction, then do not keep this point. If the direction of movement is reversed on leaving the line segment, then keep this point. See Figure 10 for a pictorial representation of this rule.

Pre-processing according to these empirical rules eliminated the great majority of potential problems. Unfortunately, these rules did not cover all possible cases. Beginning and ending points of boundaries proved to be especially difficult to resolve. When approaching the end of a boundary there was no directional information available to use for rule 3, above. This problem was critical since a mistake on any one point in a line of constant x (x-bin) would cause the routine "FILL" to produce an error in that line. For closed boundaries this problem was overcome by retaining the first direction of movement from the beginning point. If a boundary was not closed, then the ending point was a special vertex point and the solution for these points was provided at a later stage in the processing.

At this point the data have been sorted by their x-values and only the significant points of the original data stream have been retained. Significant points are those marking a valid change in the categories along a vertical line (x-bin). Continuity has been maintained in the x-direction although there are often discontinuities in the y-direction (such as areas where there had been vertical line segments). Also, the data points are no longer serial, they have been sorted according to their x-values.

Final Sort

The next step was to order each group of the points having the same x-values (all those points belonging to one range scan line of the data base) according to their y-values. A modified bubble-sort was implemented to sort each x-bin of data by y-values, from smallest to largest.

F-10 Anomalous Vertical Line Segments

Rules for which points to keep in vertical line segments:

A) if the direction of the line on entering the vertical segment is the same as the direction on leaving the vertical segment, keep only the bottom point.

B) if the direction of the line changes on entering and leaving the vertical segment, save both the bottom and top points of the vertical segment.

Figure 10.

Special care had to be taken at this time to eliminate any duplicate points because their presence would cause the following "FILL" routine to create errors. If two points occupied the same cell but had different categories (likely in areas near vertex points where three categories met) a choice had to be made of which categories to keep. If one point was actually a vertex point (had three categories associated with it) it was kept and the other point eliminated since the vertex point contains more information. If both points were normal points with different categories there was no longer sufficient information to make a correct decision. This condition actually represents an error in digitizing because it means that two boundary lines have crossed each other. This happened occasionally near vertex points where two lines had to come together and meet at a single point. In these cases the first point obtained by the sort routine was arbitrarily saved.

All vertex points had been saved because of the special information they contained. Since they were exceptions to the general rules for keeping or throwing away data points, it was possible to have an extra point saved near these vertex points. Only after running the routine "FILL" a number of times was this error located. There was no information in the data stream to identify this problem. In a rather utilitarian move we decided to eliminate any point adjacent to a vertex point in a given vertical line. Review of the data showed this decision solved many more problems that it created and so was used. The only occasion when this move might cause an error was where a vertex point was at a tangent point, which, because of the digitizing technique utilized, was very unlikely.

Matrix Fill

The steps described thus far have been largely data compression techniques utilized to remove redundance and to place the remaining data into the proper format. However, the ultimate goal was data reconstruction. The purpose of this work was to produce a matrix of points

(approximately ten million points) to describe the radar category planimetry of the Pickwick site. These preliminary compression routines were employed to maximize the return for expenditure of computer resources.

At this point, all of the data received from the digitizer has been placed into one of the 3169 vectors, representing lines of constant x, of the data matrix to be constructed. Each element of these vectors represents the point where the boundary line crosses a given line of constant x. Stored at each point is the y-value where the line of constant x is intersected together with the two categories of the boundary line.

The routine, "FILL", was written to process these data and to produce the desired, completely filled data matrix. This routine was simple because the data had been pre-processed into the correct format by all the previous routines. At this point, the data in each vector representing a line of the matrix (constant x) is independent of the data in all other vectors. So, the memory requirements for actual construction of the matrix are rather small. Each 3169 element line can be constructed and written to permanent storage separately from all others.

The routine FILL accepts a data vector and generates from it the corresponding line of the data matrix. The elements of the data vector have already been placed in ascending order according to their y-values. Points up to the value of the first point in the vector were given a category of zero meaning no data. The first point in each vector should have only one category, since it forms part of the border. So the category between the first and second y-value of the data vector must be a single category. This category must also be one of the two categories of the second point, or there is an error. At the second point, the category changes. Since we know what the category was below this point, it must be the other of the two categories associated with a data point. In this manner, the entire line of the matrix is constructed by alternating categories.

The procedure is somewhat more complicated at vertex points (a point containing three categories). The category used up to the vertex point eliminates one of the three. More information is needed to make a choice between the remaining two categories for the continuation of the

present x-bin of data. The needed information can be acquired from the next entry in the data vector. If the two categories of the next point are not the same as the two remaining at the vertex point, the proper category can be determined. In case they are the same, the software routine looks ahead in the data until it has the needed information. If the next point in the data vector also represents a vertex point, even more information is required to resolve the dilemma of which is the next category. After much search we determined that there was a large enough potential for an error occurring in this situation that it was best to list all such occurrences and print a warning message to alert us to each instance. We used these data to determine the proper categories from the original map and manually added this information to the data vector.

The early runs of this routine, FILL, found inconsistencies in a large number of the lines (i.e., the procedure determined that from one point to the next the category should be a particular value, but that was not one of the category choices at the next point). In order to resolve these problems we were forced again to compare the data vectors to the original map. This effort was a very tedious and time-consuming task. However, in this way we discovered many of the subtle anomalous characteristics of the data which led to development of the methods of correcting the data described earlier. On a number of occasions we identified anomalies which forced us to alter the pre-processing routines and then repeat earlier steps of the procedure to correct them.

In some cases, we determined that it was expedient to modify the data vector or add information by hand. One example was where two vertex points occurred sequentially in a given line. Another example is where, in a few cases, we found an extra point in a data vector (both the top and bottom of vertical line segment had been retained when only one was required). These were fixed by hand.

At this point in our analysis we uncovered a very subtle error in the original digitizing work. The digitizer operator had digitized both sides of a very narrow stream and at one point had crossed the stream and followed the opposite bank. Since this was a very specific error in the data, we developed an ad hoc solution here and inserted the data manually without retracing our steps and repeating all the intermediate steps.

When all identified errors and problems had been removed from the data, the FILL routine was run one final time and, as each line was constructed, each completed line was written to magnetic tape to form the desired complete data matrix. At this point it was absolutely essential that we maintain the sequential order of the data and that the processing be done in the proper order to produce a single, contiguous file. In this run, if a line produced an inconsistency a warning message was output and the previously completed line was written to tape again in place of this line. This recourse was used in only a few lines of the entire data base. Since each line in the data base represented a width of 20 feet on the ground and since the accuracy of the construction process was no greater than this, we decided this was an acceptable contingency plan. It is important to notice that the error introduced by this step is not a cumulative one.

DATA MATRIX QUALITY VERIFICATION

Two programs were developed to verify the output data matrix. The first program, "MACRO", produced a symbolic map from the data matrix for comparison to the original map. Its purpose was to insure that, in general, the processing had been properly performed. This program allowed us to verify that fields were in their proper locations, that fields had not been eliminated, etc.

The second program, "MICRO," looked at the data file to a greater detail than was easily possible by human beings. It especially checked to be sure that building each line independently of all others had not produced errors. The method employed was to compare each point of every line to its neighbor on either side. If neither of them was of the same category as the point in question, then the point was suspected as being bad and a warning message was generated. This routine did not check to see if any errors were accumulated over a number of lines. MACRO checked for these kinds of errors. A manual evaluation of all warning messages generated by both check programs found that a condition of "ERROR" could occur which was not really bad data, so each case had to be individually checked.

The combined results of these tests verified that the digital data base which had been constructed was accurate within the limits of the input data.

CULTURAL TARGETS

As mentioned earlier the digital cultural target data had not been processed with the rest of the data but had been kept separate. Now, the data had to be added to the digital data base.

The cultural target data were considerably easier to process. These data fell into one of three major classes:

(1) Point Targets;

(2) Roads and railroads;

(3) Dam and blockhouse.

Most point targets in the Pickwick site were mainly private homes and cabins. They were assigned a single cell in the data matrix. No other information than location, such as orientation, was included for cultural targets, so no complex processing for such targets was needed.

Roads and railroads were epresented as lines. They were assigned a width of one cell in the data matrix. The main problem with this class of target was we had to insure continuity of the line. The routine CONNECT used on borders in an earlier section served this purpose.

The dam and blockhouse were the only cultural targets encompassing an area larger than a single cell. These targets were actually digitized as area extensive targets by following their outside borders. A modified version of the FILL routine was written to handle the dam and blockhouse. It accepted the border points as input, and output the coordinates of all those cells lying in these targets.

At this point, all points representing cultural targets have been collected. As in the processing done for area extensive targets these data were sorted into their respective x-bins, and then ordered according to their y-values. In all cases, cultural targets took priority over the planimetry data, so merging the two pieces of data was simple. A line of the output data matrix was read in together with the vector

representing all cultural targets for that line. The values of the cultural targets were written over the planimetry data of area extensive targets in their respective locations and the modified line of the data matrix was written out to tape for the final ground truth data matrix of the Pickwick site.

MERGE OF ELEVATION DATA WITH RADAR CATEGORY DATA

At this point all of the necessary information for the data base required in the radar simulation packages had been collected and compiled into an acceptable format (rectangular digital matrix). Unfortunately the data was physically separated with the elevation matrix residing on one set of magnetic tapes and the matrix of category values on another set. For the purposes of simulation both pieces of data (category and elevation) about a particular cell are accessed simultaneously, so having the data physically separated was a poor utilization of system resources.

In order to better utilize the available reosurces, a special program was written had executed to merge the two data matrices (elevation matrix and category matrix) into a single composite matrix. The category matrix had been constructed in an exact manner so that data point in the category matrix corresponded to the same area of ground as that matrix point in the elevation matrix. Therefore, there existed an exact match between the matrix cells in the two matrices and no rotation, translation, or scale change was required. Because of this fact, the combination process was very straight-forward.

There was, however, a question of the optimum means of combining the two matrices. Three alternatives were proposed. They were:

(1) On the merged output tape, simply alternate records (each record representing a column of the matrix) of elevation data and category data.

(2) Merge corresponding records of elevation and category data so that the first word represents the elevation of the first point; the second word the category of the first point; the third word the elevation of the second point, etc.

(3) Combine the elevation for each point with its corresponding category into a single computer word.

Methods (1) and (2) reduce the input data to a single input device (one tape drive) instead of two, but does nothing to reduce the volume of the data. The third method reduces the volume of input data from two matrices of the same size to a single matrix of that same size, by making more complete use of the computer word size available. The trade-off is that in order to use the data, the two pieces of information in each computer must be separated in the program using it for input.

After careful consideration of the alternatives, the third method, that of combining category and elevation data into a single word, was chosen for this application. By compressing the volume of the data both tape drive charges and memory requirements could be reduced. The unpacking of the information could be easily accomplished with a few shift and masking operations that incurred relatively little overhead.

In the actual implementation the low order six bits of each computer word were reserved for the category number for that cell. This provided for an allowable range of values from 0 to 63 which was more than adequate for the 20 categories identified in the data base. The next 12 bits were reserved for the elevation value, giving a range from 0 to 4096 (Figure 11). These boundaries within the word were arbitrarily chosen to fit the current data; however, they could easily be modified to allow for more categories or a wider elevation range.

In this scheme the upper half of each computer word remain unused. The data was not packed more closely for two reasons:

(1) The result would have required a special addressing scheme to locate the desired point within a scan, providing a greater chance for errors and causing more computer overhead.

(2) The flexibility would be largely decreased by reducing the freedom of changing the width of boundaries in each word and also by linking the data base to the 36 bit word length of the Honeywell computer.

0 18	30 35	
0 0	ELEVATION	CAT #

Figure 11. Data Packing Scheme

APPENDIX G

BACKSCATTER DATA FOR DIGITAL RADAR IMAGE SIMULATIONS

The following technical report (TR 319-7) prepared by the Center for Research, Inc., University of Kansas, is included in this volume to provide details in support of the technical discussions of Volume I.

THE UNIVERSITY OF KANSAS CENTER FOR RESEARCH, INC.

2291 Irving Hill Drive - Campus West
Lawrence, Kansas 66045

Telephone: (913) 864-4832

BACKSCATTER DATA FOR DIGITAL RADAR IMAGE SIMULATIONS

Remote Sensing Laboratory
RSL Technical Report 319-7

J. L. Abbott
R. L. Martin
V. H. Kaupp
J. C. Holtzman

February, 1977

Supported by:
U. S. Army Engineer Topographic Laboratories
Fort Belvoir, Virginia 22060
CONTRACT DAAG53-76-C-0154

ABSTRACT

A theoretical closed-system model has been developed at the Remote Sensing Laboratory for digital simulation of imagery of radar systems for both SLAR (Side-Looking Airborne Radar) and PPI (Plan-Position Indicator) in visual formats. Described herein is the assignment of microwave backscatter characteristics to the categories recognized in the construction of the Pickwick, Tennessee data base, which was used as an input to the simulation programs. The choice of accurate backscatter data to describe the electrical behavior of the ground scene aids the validation of the digital model and the production of realistic simulated radar imagery.

Background information on the frequency-dependent electrical characteristics of dielectric constants is included for the justification of interpolation and extrapolation of empirical backscatter data to categories or frequencies for which such data has not been gathered. This development only scratches the surface of frequency behavior and does not represent the solution to the problem, that is, the lack of empirical data for numerous categories at several frequency/polarization combinations. Much work remains to accomplish the extrapolation/interpolation task as methods have not been established to extend the usefulness of known microwave reflectivity data. Because the scattering mechanisms suited to the categories observed in terrestrial scenes are so highly diversified, it becomes necessary to treat individually the frequency variation of sigma zero curves for each general type of scatterer. It must be realized that the handicap of insufficient empirical σ^0 data may place limitations on the accuracy of relative image densities (greytones) within the simulated imagery. As an initial effort, however, empirical sigma zero data for lower frequencies and/or analogous categories were applied as input to the simulation software routines. Photographs of simulated imagery are presented with real radar images of the same site for comparisons to detect necessary changes in the backscatter category treatment. The actual curves of sigma zero versus angle of incidence which were supplied to the simulation package are shown.

1.0 INTRODUCTION

The radar image simulation technique which has evolved over the last several years at the Remote Sensing Laboratory is based upon a closed-system model for imaging radars, as opposed to the work of several other investigators who employ, for example, actual radar video signals as an input to their simulation programs. In bypassing the need to specify the geometrical and dielectric properties of the ground scene, researchers have avoided a troublesome yet challenging aspect of the radar image simulation problem, and severely limited the usefulness of radar simulation. Using actual radar video signals as the input presupposes the radar imagery exists, and the simulation, in fact, is only that of the radar processor. When the objective is to simulate various scenes for navigation matching or MGI comparison, this technique is not applicable. The lack of empirical backscatter information for a wide variety of reflectivity categories at many frequencies has compounded the difficulty of constructing data bases for the validation of the closed-system model. Rather than attempting to gather data for all the possible targets at all frequencies and polarizations of interest, it is more feasible to estimate the backscatter variations for categories by extension of known data with guidance provided by scattering models.

The need for σ° data arises in the roots of the digital radar image simulation theory. The starting point of the development[1] is the radar equation[2,3].

$$\overline{P}_r = \frac{\overline{P}_T \cdot \sigma^\circ(\theta_\ell) \cdot G^2(\theta) \cdot \Delta A \cdot \lambda^2}{(4\pi)^3 R^4} \qquad (1)$$

---

[1] Holtzman, J. C., V. H. Kaupp, R. L. Martin, E. E. Komp, and V. S. Frost, "Radar Image Simulation Project: Development of a General Simulation Model and an Interactive Simulation Model, and Sample Results," TR 234-13, Remote Sensing Laboratory, The University of Kansas, Feb., 1976

[2] Moore, R. K., Remote Sensing Manual, (Editor - Reeves), Chapter 9, American Society of Photogrammetry, 1975.

[3] Kell, R. E., and R. A. Ross. Radar Handbook, (Editor - Skolnik), Chapter 27, McGraw-Hill Book Company, 1970.

$\sigma^{\circ}(\theta_{\ell})$ = Scattering coefficient of the ground spot resolution cell (function of local angle of incidence, θ_{ℓ}, the incident wavelength, λ, transmit/receive polarization, and surface parameters);

$\overline{P}_r$ = Average return power received at the antenna terminals;

$\overline{P}_T$ = Average power transmitted;

θ = Radar incidence angle to surface;

$G(\theta)$ = Antenna gain in the direction of ΔA (a function of the radar system being modeled and assumed by reciprocity to be identical in the return direction;

λ = Wavelength of the incident microwave energy;

R = Distance from the radar platform to the ground spot resolution cell (a function of altitude and look angle);

ΔA = Area of the ground spot resolution cell;

θ_{ℓ} = Local angle of incidence accounting for tilt.

Thus the accurate specification of target sigma zero characteristics is important to the relative intensity of return radar signal, and subsequently to the imagery brightness. Further enlightenment on the definition and validity of application of the scattering coefficient σ° can be obtained in References [3] and [4].

The major source of empirical σ° data has been the agriculture/soil moisture data bank at the Remote Sensing Laboratory. This information falls short, however, in supplying data concerning targets which are more suited to general reconnaissance applications. The agriculture-related sigma zero data has been useful (in a limited sense) in confirming the variation of microwave response with changing frequency and polarization of simulated radar systems[5]. Most existing data represents VV

---

[3] Kell, R. E. and R. A. Ross, Radar Handbook, (Editor - Skolnik), Chapter 27, McGraw-Hill Book Company, 1970.

[4] Moore, R. K., Radar Handbook, (Editor - Skolnik), Chapter 25, McGraw-Hill Book Company, 1970.

[5] Martin, R. L., J. L. Abbott, M. McNeil, V. H. Kaupp, and J. C. Holtzman, "Digital Model for Radar Image Simulation and Results," TR 319-8, Remote Sensing Laboratory, The University of Kansas, August, 1976.

and HH polarizations, consequently circular polarization simulations must apply analytical expressions from scattering theories for the assignment of scattering coefficients to a very limited set of targets. Overall, the situation at present with respect to empirical sigma zero data is that not enough exists; however, estimation of data has produced good results for several types of scatterers in simulated imagery, the validation being provided by SAR (Synthetic Aperture Radar) images of one test site.

Before proceeding to the discussion of categories within the Pickwick, Tennessee, test site and the manner in which microwave reflectivity assignments were made, the frequency dependence of the dielectric characteristics of material will be briefly investigated. The impacts of altered electrical behavior for a surface (or volume) will be implicitly related to its sigma zero versus angle of incidence variation with frequency, moisture content, etc.

2.0 DIELECTRIC BEHAVIOR

Perhaps the best place to begin a discussion of electromagnetic behavior is to state the rules which govern the phenomena: Maxwell's Equations. These four relations are the foundation upon which rests the study of electromagnetic radiation, its sources and its effects. In vector notation they may be expressed as[6]:

$$\vec{\nabla} \times \vec{H} = \vec{J} + \frac{\partial \vec{D}}{\partial t} \tag{2}$$

$$\vec{\nabla} \times \vec{E} = -\frac{\partial \vec{B}}{\partial t} \tag{3}$$

$$\vec{\nabla} \cdot \vec{D} = \rho \tag{4}$$

$$\vec{\nabla} \cdot \vec{B} = 0 \tag{5}$$

---

[6] Stratton, J. A., Electromagnetic Theory, Chapter 1, McGraw-Hill Book Company, 1941.

where:

$\overline{H}$ = Magnetic field (units - amperes/meter);

$\overline{J}$ = Vector current density (units - volts/ohm - meter^2);

$\overline{D}$ = Electric displacement field (units - coulomb/meter^2);

$\overline{E}$ = Electric field (units - volts/meter);

$\overline{B}$ = Magnetic Induction field (units - weber/meter^2);

ρ = Volume charge density (units - coulombs/meter^3);

$(\overline{\nabla} \times \overline{F})$ = The curl of $\overline{F}$, denoting the vortex sources;

$(\overline{\nabla} \cdot \overline{F})$ = The divergence of $\overline{F}$, denoting the radial sources;

$(\frac{\partial}{\partial t})$ = The partial derivative with respect to time.

Let us assume harmonic time variation of the electric and magnetic fields, denoted by $e^{j\omega t}$. The explicit time dependence in Equations (2) and (3) disappears becasue:

$$\frac{\partial(\overline{E})}{\partial t} = j\omega\overline{E} \tag{6}$$

$\omega = \frac{2\pi}{T_o}$ = angular frequency in radians per second when T_o is the period, and similarly for $\overline{B}$. Thus we find that:

$$\nabla \times \overline{H} = \overline{J} + j\omega\overline{D} \tag{7}$$

$$\nabla \times \overline{E} = -j\omega\overline{B} \tag{8}$$

The constitutive relations which exist between $\overline{B}$ and $\overline{H}$, or $\overline{E}$, $\overline{J}$, and $\overline{D}$ can be written as

$$\overline{B} = \mu\overline{H} \tag{9}$$

$$\overline{J} = \sigma\overline{E} \tag{10}$$

$$\overline{D} = \epsilon\overline{E} \tag{11}$$

where:

$\mu = \mu_r\mu_o$ = the permeability of a material, equal to the permeability of a free space μ_o ($4\pi \times 10^{-7}$ henry/meter) times the relative permeability $\mu_r \approx 1.0$ for most nonmagnetic materials;

σ = The conductivity of a material in mhos/meter (not to be confused with sigma, the scattering cross-section);

$\varepsilon = \varepsilon_r \varepsilon_o$ = the permittivity of a material equal to the permittivity of free space ε_o (8.854×10^{-12} farad/meter) times the relative permittivity which varies widely over the categories found in the terrestrial envelope.

Rewriting Equation (7) to incorporate (10) and (11) we arrive at

$$\nabla \times \overline{H} = (\sigma + j\omega\varepsilon)\overline{E} \tag{12}$$

This relation has been interpreted to mean that the complex dielectric variation of a material may be written as

$$K_r(\lambda, T) = \frac{\sigma + j\omega\varepsilon}{j\omega\varepsilon_o} = \varepsilon_r - \frac{j\sigma}{\omega\varepsilon_o} \tag{13}$$

where:

λ = Wavelength of the radiation in meters;

T = Temperature of the material.

$\frac{\sigma + j\omega\varepsilon}{j\omega\varepsilon_o} = -\frac{j\sigma}{\omega\varepsilon_o} + \varepsilon_r$ when "normalized" by $j\omega\varepsilon_o$[7]. The real part is labeled K_r' and the imaginary part is denoted by K_r''. Highly conductive materials may show the strong frequency dependence in their dielectric behavior, as seen in the term on the right which contains ω and σ. In general K_r is a complex function with a real and imaginary part given by

$$K_r = K_r' - jK_r'' \tag{14}$$

As an example[8],

$$K_r'_{water} = K_\infty + \frac{(K_s - K_\infty)}{1 + (f/f_o)^2} \tag{15}$$

and

$$K_r''_{water} = (f/f_o)\frac{(K_s - K_\infty)}{1 + (f/f_o)^2} + \frac{\sigma_i}{2\pi f \varepsilon_o} \tag{16}$$

where:

K_∞ = relative permittivity (~5.5) at frequencies much higher than the relaxation frequency f_o;

---

[7] Janza, Frank J., Remote Sensing Manual, (Editor - Reeves), Chapter 4, Page 79, The American Society of Photogrammetry, 1975.

[8] Stogryn, A., "Equations for Calculating the Dielectric Constant of Saline Water," IEEE Trans. Microwave Theory Tech., MTT-19, 733-736, 1971.

K_s = static relative permittivity (~80);

σ_i = ionic conductivity of the salt solution in mhos per meter;

$f_o = 1/(2\pi\tau)$ where τ is the molecular relaxation time;

f = frequency of operation;

ε_o = free space permittivity.

It is seen that labeling K_r as a dielectric constant is in most cases a misnomer unless the frequency and temperature are fixed. The values of K_r' and K_r'' are also functions of λ and T in most materials, as opposed to constants.

An excellent example of the frequency dependence of these functions has been studied for pure and salt water[8]. The dielectric effects of water in saline soil at several frequencies produced another very interesting investigation which may aid in the understanding of moisture effects for many microwave reflectivity categories[9]. To add impetus to the example of K_r frequency dependence in water, consider the range of objects in a terrestrial scene that contain water. The changing dielectric characteristics of water among other factors often dominate the electrical behavior, for example, in vegetation and soil[10,11]. Figure 1[12] illustrates for water the wide range of values K_r' and K_r'' can take on, thus allowing moisture content to be a controlling factor in the electrical characteristics of many materials.

---

[8] Stogryn, A., "Equations for Calculating the Dielectric Constant of Saline Water," IEEE Trans. Microwave Theory Tech., MTT-19, 733-736, 1971.

[9] Ulaby, F. T., L. F. Dellwig, and T. Schmugge, "Satellite Microwave Observations of the Utah Great Salt Lake Desert," Radio Science, Vol. 10, No. 11, November, 1975.

[10] Ulaby, F. T., "Radar Response to Vegetation," IEEE Trans. on Antennas and Propagation," Vol. AP-23, No. 1, January, 1975.

[11] Batlivala, P. P. and C. Dobson, "Soil Moisture Experiment (Kansas)-Documentation of Radar Backscatter and Ground Truth Data," Remote Sensing Laboratory, The University of Kansas, TR-264-7, March, 1976.

[12] Paris, J. F., "Microwave Radiometry and Its Application to Marine Meteorology and Oceanography," Texas A & M University, Department of Oceanography, January, 1969.

Figure 1. (a) Real part and (b) imaginary part of dielectric constant of water as a function of frequency for various temperatures.

The objective of the previous development was to establish frequency variation in $K_r(\lambda, T)$. The succeeding sections attempt to connect in some logical manner (through the use of empirical data) the effects of K_r' and K_r'' in an implicit sense on sigma zero variation. A simple scaling of σ° curves versus frequency is not usually feasible because there is another factor unaccounted for thus far. If a vegetation canopy, for instance, is viewed as a type of screen which causes several phenomena (namely reflection, refraction, diffraction and attenuation) to affect the (plane) waves of incident microwave energy, it is realized that the irregular lattice (which forms the screen) changes its effective interelement spacing (and scattering cross-section) as frequency increases or decreases. In summarizing these ideas, two overriding factors are at work, variability of K_r and effective scattering cross-section changes.

2.1 Extension of Sigma Zero Data

With the realization that most objects in terrestrial scenes have a K_r' and K_r'' with functional dependence on frequency, temperature, etc. the next step desired in this report is the establishment of trends for sigma zero curves varying with frequency. As alluded to in the abstract and introduction, methods have not been developed to analytically operate on σ° versus θ plots for a frequency translation. What is possible at

this time is to demonstrate σ^o trends for several radar distributed targets common to a majority of existing imagery. To accomplish this goal, empirical backscatter data at multiple frequencies for soil, cultivated vegetation, deciduous trees, gravel, sand, and grass will be investigated to develop an intuitive feeling for σ^o behavior for each of these categories, and associate the physical mechanisms responsible for dielectric and scattering variations. When pertinent to the radar image simulation studies, alternate factors which have been shown to have marked effects upon the microwave return, from a category will be mentioned to give a more complete description of that material's electrical characteristics.

SOIL

The dielectric variation K_r' and K_r'' of three soil types, sand, loam and clay, have been studied by Batlivala and Ulaby[13]. In particular, at 4.0 GHz and 10.0 GHz K_r' and K_r'' have been plotted versus soil moisture. The referenced plots illustrate the inclination (for a given water content) of K_r' to decrease with higher frequency, as opposed to K_r'' which increases with frequency between 4.0 and 10.0 GHz. These changes track the dielectric variation of water to some degree. A relatively high scattering coefficient is measured at low incidence angles for smooth, wet soil (~+10dB). However, the characteristically flat sigma zero plot of a rough surface can be seen to counterbalance the soil moisture effects by lowering σ^o to ~0dB at low angles of incidence. Parameterized in Ulaby's data[14] are surface RMS height, soil moisture, frequency, etc. For low levels of moisture the RMS roughness has a less drastic effect in the empirical backscatter characteristics. The justification or interpolation of σ^o versus frequency is highly dependent upon the soil conditions. Again, it is emphasized that this is an area of future research and that an analytic solution has not been arrived at for σ^o vs. (θ, λ, RMS height, moisture, soil ionizable salts content, etc.).

---

[13] Baltivala, P. P. and F. T. Ulaby, "Effects of Roughness with Radar Response to Soil Moisture of Bare Ground," Remote Sensing Laboratory, University of Kansas, TR 264-5, September, 1975.

[14] Batlivala, P. P. and F. T. Ulaby, "Remotely Sensing Soil Moisture with Radar," Remote Sensing Laboratory, University of Kansas, TR 264-8, August, 1976.

CULTIVATED VEGETATION

The empirical backscatter data reported by Ulaby and Bush for several vegetation types are the best source of information for frequency extension of σ° curves[15]. It is seen that plant type, season, moisture content, canopy height soil moisture and roughness, plant spacing, frequency, etc., are all important parameters of the backscatter response[15].

Deciduous trees have been studied by Bush and Ulaby[16]. Careful comparison of these data reveal that the autumn sigma zero backscatter level drops with increasing frequency (to 14 GHz) for both HH and VV configurations, while the spring microwave return increases directly with frequency. Though the mechanisms for these changes have not been analyzed, significant information can be gleaned; the σ° data can be estimated at intermediate frequencies with little error because of the existence of sufficient backscatter data (linear polarizations) for the deciduous tree category.

GRAVEL, SAND, GRASS

Two references have reported X-band data for these categories: (1) The Response of Terrestrial Surfaces at Microwave Frequencies by W. H. Peake (1971)*, and (2) Radar Terrain Return Study Final Report: Measurements of Terrain Backscattering Coefficients with an Airborne X-Band Radar by Goodyear Aerospace Corporation (1959). There is not enough available empirical data to state trends of the frequency behavior of these categories; however, theoretical scattering models which apply to gravel and sand have been used to predict their behavior. The implementation of a theoretical two-scale surface model (provided by Mr. Richard Hevenor of the Engineer Topographic Laboratories, Fort Belvoir, Virginia) is being studied to determine its applicability to the frequency translation problem. To conclude the topic of grass scattering, theoretical modeling of this category as a collection of thin cylinders is expected to produce acceptable estimated data, though this is another future task.

[15] Bush, T. F. and F. T. Ulaby, "Radar Return from a Continuous Vegatalon Canopy," Remote Sensing Laboratory, University of Kansas, TR 177-56, August, 1975.

[16] Bush, T. F., F. T. Ulaby, T. Metzler, and H. Stiles, "Seasonal Variations of the Microwave Scattering Properites of the Deciduous Trees as Measured in the 1-18 GHz Spectral Range," Remote Sensing Laboratory, University of Kansas, TR 177-60, June, 1976.

* The data used in current radar image simulations have been corrected to eliminate previous bias errors.

Summarizing this section, it can be seen that some effort should be devoted to fulfilling the objective of collecting accurate sigma zero data at desired frequencies. Realizing that the error tolerance in backscatter data depends upon the application the radar image simulation, efforts will be devoted to refining data to allow flexibility in its later use. This will make possible a large number of greytones, if necessary, within the simulated images. The trade-off between resolution and dynamic range will first be examined to establish priorities. Of course the seasonal, i.e., climatological effects on the simulated imagery and the necessity of accurate representation impacts the thoroughness of the required studies.

A distinction has not been outlined between the types of commonly occurring objects in the terrestrial envelope. In the course of simulating the radar return corresponding to a test site, it is discovered that certain types of targets (cultural, or hard) must be treated differently than distributed targets such as a vegetation canopy. This need arises because the phenomena involved in the electromagnetic behavior of these targets are dissimilar.

2.2 Target Classification

Two general types of targets are recognized in the radar simulations, distributed and hard (or cultural). Microwave illumination scatters/reflects in diffuse and specular fashions from these targets, respectively, when viewed by a single radar system. The true distinction between target classifications depends on the resolution with which the system operates. For example, a radar with 10 x 10 foot resolution will produce bright returns by specular reflection from automobiles, and in this situation the use of a σ° value for simulating that target will not be valid. A coarse resolution system viewing a junk-yard of automobiles will produce a value of σ° which will be representative because there exist sufficient randomly located scattering centers within a single resolution cell such that the return signal is a contribution of many independent-phase signals[4].

---

[4] Moore, R. K., Radar Handbook, (Editor - Skolnik), Chapter 25, McGraw-Hill Book Company, 1970.

Difficulties naturally arise in the treatment of cultural targets within the ground scene, because (1) proper σ° data as outlined in the previous paragraph is not readily available, and (2) the detail of specification of geometry needed for hard target simulation is much greater than that needed for distributed targets. A full explanation of the possible simulation techniques for hard targets returns is found in Reference [17], whereas attention will be focused here on the distrubuted targets which can be described by coarser geometry, and sigma zero versus theta variation.

3.0 PICKWICK CATEGORIES

The general categories whose backscatter characteristics were needed in the radar image simulation of the Pickwick site (excluding hard targets) are:

Forest, hardwood	Soybeans
Grass	Corn
Golf Course	Milo
Marshes, open and woody	Wheat
Water	Orchards
Roads, various degrees of	Small Scattered Garden Plots
Improvement	Bare Ground

This information was collected by E. Davison of the Remote Sensing Laboratory from several sources including aerial photography and personal communication with local agriculture agents (County Extension Service and Soil Conservation Service).

---

17 Frost, V. S., J. L. Abbott, V. H. Kaupp, and J. C. Holtzman, "An Alternative Approach for the Simulation of Cultural Targets," Remote Sensing Laboratory, The University of Kansas, TR 319-12, February, 1977.

The forest category backscatter data was equated to empirical measurements of the return from spring deciduous trees (Appendix I) as reported in Reference [16]. In the initial simulation attempt, variation in the height of trees was not added to the elevation data, and thus, in a sense, the trees were assumed to have equal heights and to follow the undulations of the earth exactly. Radar imagery of the site indicated: (1) A terrain-smoothing effect due to variations in tree heights; and (2) Significant multipath effects causing greytone variability. The forest backscatter data was not adjusted as the average greytone of the forest category seemed reasonable in comparison to the greytones of surrounding categories in the simulated imagery. Reference [18] describes the methods which attempted to correct the evident geometry errors in the data base description of the forest.

Agriculturally developed land represented a very broad category in the Pickwick site and the backscatter characteristics from the agriculture/soil moisture data bank at RSL were applied to various crop types observed. Without entering into a discussion of ground truth gathering for crop identification, it is obvious that different σ° curves should be applied to fields in the river bed lowlands as opposed to hilly areas due to the high probability of moisture differences throughout most of the year. Reserving judgement of the adequacy of this data for agriculturally developed land to the presentation of results, it is important to take into account mission objectives. The trade-off which exists between resolution and dynamic range for radar systems assures one that crop discrimination (besides being generally irrelevant to the defense of the country) will usually not occur. An in-depth section discussing returns from grass is also precluded by an understanding of the large-scale objectives of the radar simulations, except to say that empirical backscatter data exists for grass in various seasonal stages as shown in the Appendix at the frequency of interest.

---

[16] Bush, T. F., F. T. Ulaby, T. Metzler, and H. Stiles, "Seasonal Variations of the Microwave Scattering Properties of the Deciduous Trees as Measured in the 1-18 GHz Spectral Range," Remote Sensing Laboratory, University of Kansas, TR 177-60, June, 1976.

[18] Abbott, J. L., M. McNeil, E. Davison, V. H. Kaupp, and J. C. Holtzman, "Treatment of a Forest Category for Radar Image Simulation," Remote Sensing Laboratory, University of Kansas, TR 319-9, February, 1977.

As alluded to throughout this report, the dielectric variation of terrestrial scenes due to water content is potentially large. The next Pickwick category, open and woody marshes, becomes more important because of the significance of land-water boundaries. In terms of area correlation of radar images with the simulated products, the correct specification of σ° for marsh lands becomes valuable. The open marsh can be represented as grass over water and the radar return becomes greater as frequency increases because of the smaller penetration due to the water and scattering by the grass coverage. This of course does not imply the averaging of grass and water sigma zero curves, which would falsely represent return power. Empirical data exists for swamps which are analogous in terms of active microwave sensing.

The single most important category (of non-cultural targets) in the Pickwick site is water because of the dominance of specular reflection and therefore, low return power. The value of water bodies in terms of providing image contrast is believed to have a significant effect on the correlation of reference scenes with actual imagery. An important question to consider is whether climatic conditions may mask lake boundaries, for instance, in the case of significant ice and snow cover. A study of the seasonal changes in weather and ground conditions and the corresponding effects on simulated radar imagery should be performed and empirical σ° data for snow and ice should be collected in this investigation.

The Pickwick scene contains roads consisting mainly of graded dirt and gravel, for which empirical data exists. The importance of road bed materials themselves is not overwhelming, but it is the corresponding edge brightening effects which reveal the road to be of major significance in the simulation effort.

After the backscatter data literature was searched, X-band data was found for the remaining categories specified at the beginning of this section. Simulations of radar imagery for the test site were produced; at that time the backscatter data were appropriate to an X-band radar. The imagery generated, along with real imagery for comparison, are described in the following section. The presentation format is that for SLAR imagery. This was accomplished by producing a mosaic of images photographed individually from the visual display monitor in the IDECS* system.

* IDECS - Image Discrimination, Enhancement, Combination and Sampling, an image processing station at the Remote Sensing Laboratory.

4.0 RESULTS

Radar image simulations were produced for an area approximately centered at the Pickwick dam, three miles from near-to far-range and twelve miles in length. Figure 2 is a side by side comparison of a simulated radar image (lower image) and fine resolution radar imagery. The resolution of the simulated scene is approximately 60 feet, whereas the resolution of the real radar imagery is approximately 10 feet. Despite the obvious differences between the real and simululated images caused by the differing resolution, the comparison between real and simulated images is for most purposes, exact. The resolution difference is easily seen by the decrease of fine detail in the simulated image. By increasing the inherent resolution of the data base, the Point Scattering Method of simulation would produce imagery precisely like the real system.

Shape, pattern, and size are the normal indicators for the interpretation of radar imagery. Collectively, these discriminants define (as used here) the geometric fidelity of the various features within the scene. As can be seen the geometric fidelity of simulated images compared to the actual images is very good. This fact is immediately evident and reflects, in part, the accurate treatment by the simulation model of the various radar phenomena and, in part, the precise construction of the ground truth data matrix.

Tone, texture, and shadow, critical in any type of interpretation, are highly dependent upon the radar system being employed and the flight parameters (platform location and look-direction), thus these must normally be employed with caution when comparing two radar images from different systems. Within reason, the conditions generating the images present here are similar. Thus, these critieria may be invoked as a further means to qualify the validity and quality of the simulated images Shadow areas, best illustrated in the forested highlands, are very similar in shape, orientation and length. Shadows indicate accuracy of the elevation data in the ground truth data matrix and the formation of

Figure 2. Side by Side Comparison of APD - 10 imagery and PSM simulated imagery (lower).

shadow by the digital implementation of the simulation model.

A textural analysis of the synthesized images also shows validity of the model. The rough texture of the deciduous forests in the simulated images compares well with the APD-10 images. The growth pattern of deciduous trees in the Pickwick area was modeled as a Gaussian distribution having a mean height of 70 feet with a standard deviation of 10 feet. The comparison illustrates the accuracy of both the simulation model, the ground truth data base, and the growth model for the trees.

Similarly, tone can be seen to compare favorable between the real and simulated images. Radar signals are normally returned from the terrain to the receiver by a scattering process (reradiation) with the intensity of return determining the relative degree of brightness in the radar image. As can be seen by comparing the simulated images to the APD-10 images, the relative degree of brightness from feature to feature is faithfully reproduced. This speaks eloquently for the accuracy of the empirical backscatter data used to model the radar return from the various categories present in the scene as well as the accuracy of prediction by the model of the relative intensity for each pixel in the images. There are some noticeable differences that appear, but, in each case, these are caused by errors in interpreting the source intelligence data.

Without appeal to an expert in radar interpretation it would be difficult to make further claims about the quality or the shortcomings of the simulated imagery. Since the real test of the simulated imagery's ability to perform successfully is non-human and subjective in a sense, that judgement will be reserved for the correlation device. The results of simulation with the improved data base will be reported, along with the results of its correlation test at a later date.

One further interesting note about the simulated imagery mosaic is that it was mistaken for real imagery by two experienced interpreters and visitors with a great deal of radar expertise. Hopefully this indicates progress in the right direction.

5.0 CONCLUSIONS

Radar image simulations have been produced for a real test site (Pickwick dam area), with the backscatter information taken from empirical X-band data. It is desired in the future to translate these sets of data to a specific frequency to suit certain applications. From the small amount of theoretical information provided about the likelihood of K_r and σ° varying with frequency, moisture, etc., and from the discussion of empirical data, it is known to be important to investigate and obtain improved backscatter data to produce accurate simulations.

At present, the techniques do not exist to make all-encompassing generalities about alteration of sigma zero curves to match the change in one or more controlling parameters. Those parameters range from the frequency to surface roughness, but moisture content is often seen to be the determining factor. A good deal of research in this area remains and the problems encountered must be handled. An alternative approach to theoretical treatment of backscatter data is the collection of empirical data. This would necessitate gathering sufficient equipment, either from the existing inventory or outright purchasing, to form a system similar to that operated for another agency by the Remote Sensing Laboratory. The problem still exists that data cannot be gathered for all types of targets. Either approach can be taken, but for the present the only alternative open to this team is to estimate data when not available.

BACKSCATTER DATA

TR 319-7

The following X-band sigma zero versus angle of incidence curves served as input data in the simulation of the test site. The category and the source of data are listed as headings for each plot.

ROAD - MED. DUTY
THETA IN DEGREES

PLOT OF CATEGORY 3 ROAD - LIGHT DUTY

PLOT OF LATERAL
ROAD - DIRT
+1000
0000
-1000
-2000
-3000
-4000
0
10
20
30
40
50
60
70
80
90
THETA IN DEGREES

RAILROAD TRACKS

PLOT OF CATEGORY 5

PLOT OF CATEGORY 6
FISH POND DIVIDERS
THETA IN DEGREES
0
10
20
30
40
50
60
70
80
90

PLOT OF C-THEORY 7 1

PAPER MILL

PLOT OF CATEGORY
POND
THETA IN DEGREES

PLOT OF DATE.DAT 9
CREEKS
+100
0.0
-100
-200
-300
-400
0
10
20
30
40
50
60
70
80
90
THETA IN DEGREES

RIVER
THETA IN DEGREES

RESERVOIR
THETA IN DEGREES

PLOT OF CATEGORY 12
MARSH
THETA IN DEGREES

PLOT OF LISTEGLPY 13

-0.3972[illegible]E-04 [illegible] -[illegible]E 00 0.1654100E 01

WOODED MARSH

WOODS
THETA IN DEGREES

GOLF COURSE
THETA IN DEGREES

LOW GRASS AND PASTURE
THETA IN DEGREES

PLOT OF CATESLPV 17
DAM
THETA IN DEGREES

BLOCK HOUSE

PLOT OF CATEGORY 13

0.2000100E 02

PLOT OF CATEGORY 19
LOCKS
THETA IN DEGREES

HOUSES - SMALL

PLOT OF DATA BY 21 HOUSES - LARGE

PLOT OF DATE.DAY 22
0.
0.
0.
0.23003108E 12
+1000
500
-1000
-2000
-3000
-4000
0
10
20
30
40
50
60
70
80
90
THETA IN DEGREES

BARE GROUND
THETA IN DEGREES

SOYBEANS
PLOT OF CATEGORY 24
THETA IN DEGREES

PLOT OF SITEWOPT 25
CORN
THETA IN DEGREES
G-44

MILD
THETA IN DEGREES

WHEAT
THETA IN DEGREES

PLOT OF DATEGLFY 2:

ORCHARDS

GARDEN PLOTS

APPENDIX H

MEDIUM RESOLUTION RADAR IMAGE SIMULATION OF DECIDUOUS FORESTS: A STUDY OF CANDIDATE TECHNIQUES

The following technical report (TR 319-9) prepared by the Center for Research, Inc., University of Kansas, is included in this volume to provide details in support of the technical discussions of Volume I.

THE UNIVERSITY OF KANSAS CENTER FOR RESEARCH, INC.

2291 Irving Hill Drive - Campus West
Lawrence, Kansas 66045

Telephone: 913 8644832

MEDIUM RESOLUTION RADAR IMAGE SIMULATION OF DECIDUOUS FORESTS: A STUDY OF CANDIDATE TECHNIQUES

Remote Sensing Laboratory
RSL Technical Report 319-9

J. L. Abbott
V. H. Kaupp
J. C. Holtzman

August, 1977

Supported by:

U.S. Army Engineer Topographic Laboratories
Fort Belvoir, Virginia 22060
CONTRACT DAAG 53-76-C-0154

ABSTRACT

The closed-system model for simulation of radar imagery, developed at the Remote Sensing Laboratory, requires as input a ground truth data base which is a sampled version of the terrain to be imaged. The individual locations of the ground truth data base contain information about the elevation and backscatter type of the corresponding terrain site. The vicinity of the Pickwick Landing Dam on the Tennessee River was chosen as the ground scene for simulation experiments, in part because it contained a wide variety of scattering categories and radar imagery of the site was in existence. Radar simulations were produced for two subareas within the test region, and the resulting images were compared on a qualitative basis with real radar imagery.

Excellent correspondence in texture, greytones, and geometric fidelity was noted between the real and simulated imagery except in areas where ground truth indicated the presence of deciduous forests. Here there was found a discrepancy in texture and greytones. Several candidate methods for the alleviation of this problem were investigated, and they are the subject of this report.

1.0 INTRODUCTION

The digital implementation of the Point Scattering Model, a closed-system model for radar image simulation, has produced a series of computer programs which assimilate radar system parameters and terrain information to form visual products[1]. The terrain information which serves as an input to the simulation programs exists as a large matrix of data points which contain the elevation and radar backscatter category for each of the matrix locations. SLAR (Side-Looking Airborne Radar) and PPI (Plan-Position Indicator) simulated imagery have been produced with considerable success. An objective measure of the SLAR simulation package was provided by a geometric data base[2] with assigned shapes, heights, and empirical backscatter curves[3].

During the quantitative test period a ground truth data base of a real site was being constructed. The source materials for this work were digital elevation tapes supplied by ETL and 1:100,000 scale aerial photography[4]. Upon completion of the first Pickwick ground truth data base (having 20.5 foot resolution) a preliminary radar image simulation of the site was produced at 60 foot resolution and the result was photographed.

---

[1] Holtzman, J. C., V. H. Kaupp, R. L. Martin, E. E. Komp and V. S. Frost, "Radar Image Simulation Project: Development of a General Simulation Model and an Interactive Model, and Sample Results," TR 234-13, Remote Sensing Laboratory, The University of Kansas, February, 1976.

[2] Komp, E. D., V. H. Kaupp, and J. C. Holtzman, "Construction of a Geometric Data Base for Radar Image Simulation Studies," TR 319-1, Remote Sensing Laboratory, The University of Kansas, July, 1976.

[3] Martin, R. L., J. L. Abbott, M. McNeil, V. H. Kaupp, and J. C. Holtzman, "Digital Model for Radar Image Simulation and Results," TR 319-8, Remote Sensing Laboratory, The University of Kansas, August, 1976.

[4] Davison, E., V. H. Kaupp, and J. C. Holtzman, "Baseline of Planimetric Data Base Construction: Pickwick Site," TR 319-2, Remote Sensing Laboratory, The University of Kansas, July, 1976.

2.0 ANALYSIS OF SIMULATED FOREST REGIONS

Within areas populated by deciduous trees a discrepancy was found between the two sets of imagery. The simulated forest regions followed the undulations of the terrain (thus allowing considerable shadowing) and the greytones were more consistent (rather than showing a large range of image brightness as in the real imagery). The first effect was due to the fact that no tree heights (and implicitly, no tree height variations) were accounted for in the data base. The digital elevation data provided by ETL represented only the elevation of the bare ground. The second effect in the simulation (i.e., pixel-to-pixel greytone constancy) was basically caused by the fact that the forests were treated as surface scatterers with no additional height above the ground. This is contrary to the known behavior of deciduous trees, however as a first attempt at simulation the results were reasonable. It is understandable that the tree should be treated as a volume scatterer with a variable penetration depth (which is a random process when we image an entire forested area).

Further investigation revealed that the scattering from the trees had the tendency to mask the undulations of the earth in real imagery (i.e., a terrain-smoothing effect yielding greytone consistency, on a large scale). However, the multipath effects caused a good deal of small scale greytone variability to be seen in the real imagery. Several mechanisms were proposed to be responsible for the terrain smoothing: (1) tree heights greater in valleys than on hillsides; (2) moisture differences between hillside and valley trees, possibly resulting in different backscatter responses; and (3) random tree heights (or random elevation scattering centers).

Several techniques were employed for manipulation of the forest category. First, a bias of 70 feet (representative of the average tree height for the Pickwick forest areas) was added to data points in the tree category. Then, four additional methods were suggested: (1) extra tree height could be added in the valleys to make the composite elevation more nearly constant; (2) a routine could be tacked onto the simulation package to perform running elevation averages in the across- and along-track directions; (3) a randomly weighted variance could be

superimposed on the average height within the tree category; or (4) backscatter properties could be adjusted to reflect valley/hillside moisture differences.

It was realized that neither (1), (2) nor (4) would reproduce the pixel-to-pixel greytone variability to make the simulated image more comparable to the real imagery. It was felt that only (3) could both mask terrain relief and yet mimic the variability of location (depth from the canopy surface) of scattering centers. Therefore, it was decided to implement the bias and randomly weighted variance to affect data point elevations. It is also thought that this is a more realistic approach for modeling actual tree growth patterns in the absence of extensive ground truth gathering capability.

An average height of 70 feet was assigned, and a standard deviation of 10 feet was weighted by a Gaussian random number (zero mean, unit variance, going both positive and negative). The standard deviation was arrived at by knowledge of general growth behavior of the particular varieties of deciduous trees in the Pickwick Dam vicinity.

3.0 RESULTS

This method was applied, and simulated radar images produced for qualitative comparison with real imagery. Excellent correspondence between the two sets of imagery for deciduous forests was attained. The imagery are presented in Reference [5]. The success of this solution is credited to the fact that we were trying not only to reproduce the desired visual effects, but also the random height treatment actually modeled the locations of scattering centers realistically. Both goals were achieved, i.e., shadow areas became less sharply defined, and the small scale (i.e., pixel-to-pixel) greytone variation increased. Just as in the real imagery, the more rough texture of the deciduous forests became clearly differentiable from that of dense, low, cultivated vegetation.

---

[5] Holtzman, J. C., V. H. Kaupp, J. L. Abbott, V. S. Frost, E. E. Komp, and E. C. Davison, "Radar Image Simulation: Validation of the Point Scattering Model, Volume I," ETL-0117, Engineer Topographic Laboratories (U.S. Army), Fort Belvoir, Virginia, June, 1977.

4.0 CONCLUSIONS

A special method of treatment has been necessary for deciduous forests in radar image simulation. This is due to several phenomena: random heights, volume scattering, heights possibly exceeding the canopy skin depth, and so on. The net effect of deciduous forest behavior has been seen in real imagery to be a rough texture yet a softening of shadows which would be present if the ground were bare. Similar difficulties were not encountered for other vegetation, namely low cultivated crops because they more faithfully follow the terrain, and in some cases the microwave penetration depth may exceed the canopy height. This study suggests that for certain targets special consideration must be given. That is, not only a backscatter curve and a canopy height are needed to characterize the visual radar response to the target, but also, the random nature of the location of scattering centers must be taken into account.

APPENDIX I

DIGITAL MODEL FOR RADAR IMAGE SIMULATION AND RESULTS

The following technical report (TR 319-8) prepared by the Center for Research, Inc., University of Kansas, is included in this volume to provide details in support of the technical discussions of Volume I.

THE UNIVERSITY OF KANSAS CENTER FOR RESEARCH, INC.

2291 Irving Hill Drive Campus West
Lawrence, Kansas 66045

Telephone: 913-864-[illegible]32

DIGITAL MODEL FOR RADAR IMAGE SIMULATION AND RESULTS

Remote Sensing Laboratory
RSL Technical Report 319-8

R. L. Martin

J. L. Abbott

M. McNeil

V. H. Kaupp

J. C. Holtzman

August, 1976

Supported by:

U. S. Army Engineer Topographic Laboratories

Fort Belvoir, Virginia 22060

CONTRACT DAAG 53-76-C-0154

ABSTRACT

The theory and implementation of a digital closed-system, radar image simulation model are reported. Visual results of several SLAR (Side-Looking Airborne Radar) simulations with a slant range mode are presented. The model and computer software, which were developed at Remote Sensing Laboratory, have the capability of producing SLAR or PPI (Plan-Position Indicator) imagery. A test of the integrity of the model and software implementation was conducted with a data base consisting of computer-generated geometric objects. This allowed the calculable shadow and layover response to be employed for validation of the operation of the SLAR model. Though the shapes of objects for the test site were artificially generated, the dielectric properties encompassed in the scene were derived from empirical backscatter data. This input reflectivity information originated from the agricultural/soil moisture data bank available at the Remote Sensing Laboratory.

Pictorial examples reveal that the simulated imagery realistically models radar effects seen in real imagery; for example, layover, shadow and fading. Examination of layover and shadow on the imagery produced for various flight tracks reveals that the simulation package is performing well. As an example of the application of radar image simulation to "real world" problems, a simple study of the visual effects caused by changing frequency or polarization or both is included. The frequency-polarization tests indicate the applicability of simulation for optimum discrimination studies, feature enhancement tasks, and general radar system design.

1.0 INTRODUCTION

A theoretical model has been developed at the Remote Sensing Laboratory in an effort to simulate radar imagery. The result is a closed-system model, that is, a mathematical formulation which encompasses the phenomena that affect the known, transmitted, pulsed energy and subsequently the final image product of a radar system. The implementation of the model with suitable input data and software routines for a digital computer has been accomplished. SLAR and PPI formatted imagery have since been successfully simulated. A description of the radar image simulation theory (in particular for the SLAR model), applications of the model, and experimental results are included in this document in addition to a copy of the digital simulation program and a brief outline of the necessary input data.

Radar image simulation (RIS) techniques have been applied to military tasks, for example, interpretation training, navigational aids and guidance systems for unmanned airborne vehicles[1]. The suitability of RIS for technologically advanced electronic guidance has been recognized in light of recent testing of tactical and strategic SLCM's (Sea-Launched Cruise Missiles) and ALCM's (Air-Launched Cruise Missiles) in the United States. The resources exist at RSL to develop simulation models for potential hybrid guidance schemes employing, for instance, multi-spectral sensors. The capability to predict sensor (and successful mission) performance must have an impact on the design of such future systems as well as, of course, use in providing the reference information.

RIS also has importance as an instrument for research in the areas of electromagnetics and scattering theory and it provides a method of predicting and optimizing system performance when information concerning the microwave response of terrestrial scenes is available. Although volume scattering theories and backscatter data for vegetation have existed for

[1] Holtzman, J. C., V. H. Kaupp, J. L. Abbott, V. S. Frost, E. E. Komp and E. C. Davison, "Radar Image Simulation: Validation of the Point Scattering Model, Volume 1," TR 319-27, Section 4.0, Remote Sensing Laboratory, The University of Kansas, June, 1977.

several years, it is doubtful that this knowledge has been previously assembled (that is, to visually predict a radar response) in the manner presented in this report. For example it would be very difficult for one to postulate or mentally envision the differences in radar images (for a particular complex ground scene in which the frequency of the radar was changed) without the aid of simulation of the imagery. Thus, the point of the frequency/polarization study is that the microwave response of a terrestrial scene varies rapidly with frequency and that optimum performance of a discrimination radar can be sought. Another valuable application of RIS is estimation of seasonal variations on the radar imagery, particularly for radar guidance systems in which the resolution and dynamic range of greytones (image density) are fixed. For example, it would be important to know before flight whether the presence of a heavy snow fall over a target area would mask the otherwise dominant characteristics of the scene.

This work is organized into three subtopics, (1) SLAR model theory, (2) the mechanics of producing radar image simulations, and (3) results which include simulated imagery from several experimental flight situations. The ground scene for this work is an artificial data base, in that the elevation data was assigned by a computer to describe three dimensional objects on a plane. The purpose of employing a deterministic scene was to allow the researchers to calculate known shadow and layover response to a SLAR at a known altitude and distance from the near range of the data base for verification of accuracy of the SLAR simulation program. Since radar return also depends on the backscatter characteristics of the ground, empirical sigma zero data were associated with specific areas within the scene. The significance of this may not be clear until it is realized that the simulated imagery will accurately represent the dynamic range of image density on the film that a real radar would produce from the same scene. What has been attempted here has been to provide a fine resolution data base which doubles as a test mechanism for SLAR simulation. Some of the results may be surprising to those persons not familiar with multi-frequency/polarization imagery; nevertheless, the imagery included reinforces the idea that radar image simulation or similar modeling techniques may have many interesting applications.

Before proceeding to the description of the mechanics of simulating SLAR images, the theory which is the basis of the closed system model will be presented to illustrate the determination of greytones (image density).

2.0 RADAR IMAGE SIMULATION THEORY

In an imaging SLAR, a short pulse of microwave energy is transmitted into space from an antenna whose boresight is orthogonal to the flight track. This energy, confined within the antenna beam solid angle (and sidelobes) strikes the ground. A portion is reradiated in the direction of the antenna (amount governed by backscatter characteristics and geometry of the scene) and is detected by the receiver. The video signal can be processed in various ways. For example, it can be recorded directly on signal film, it can be converted to digital data and stored on magnetic tape, it can be displayed by intensity modulating the beam of a CRT (Cathode Ray Tube) and photographed, or processed by other, more elaborate methods. The final image (the visual record of the radar return signal) depends on variations in the relative strengths of the signal returned from different parts of the area imaged to produce contrasts, edges, and the range of image brightness (greytones). The two primary factors determining the strength of the signal observed on the final image produced, and consequently the brightness of an image point, are geometry and dielectric properties.

Simulation of radar images may be accomplished by mathematical operations on the known parameters of the radar to be modeled and site to be imaged to form a visual display. The general technique utilized in this document is reported by Holtzman, et. al.[2] In this method the radar equation is used to relate target empirical backscatter coefficients to relative image greytones. A general form of the greytone expression is developed in this section. Also included in the simulation model are (1) the effects of relief and tilting of resolution cells, (2) layover and shadow, and (3) radar fading.[2]

The general model for mean power received is based upon the radar equation which may be expressed as[3]

---

2 Holtzman, J. C., V. H. Kaupp, R. L. Martin, E. E. Komp, and V. S. Frost, "Radar Image Simulation Project: Development of a General Simulation Model and an Interactive Simulation Model, and Sample Results," TR 234-13, Remote Sensing Laboratory, The University of Kansas, Feb., 1976.

3 Moore, R. K., Remote Sensing Manual, (Editor - Reeves), Chapter 9, American Society of Photogrammetry, 1975.

$$\overline{P}_r = \frac{P_T \cdot \sigma^\circ(\theta_\ell) \cdot \Delta A \cdot G^2(\theta) \cdot \lambda^2}{(4\pi)^3 R^4} \qquad (1)$$

where $\overline{P}_r$ = Average return power received at the antenna terminals;

P_T = Average transmitted power;

$\sigma^\circ(\theta_\ell)$ = Scattering coefficient of the ground spot resolution cell (function of local angle of incidence, θ_ℓ, the incident wavelength (λ), transmit/receive polarization, and surface parameters);

θ = Radar incidence angle to surface;

$G(\theta)$ = Antenna gain in the direction of ΔA (function of the radar system modeled and assumed to be identical in the return direction);

λ = Wavelength of the incident wave;

R = Distance to the ground spot resolution cell (A function of altitude and look angle);

ΔA = Area of the ground spot resolution cell.

The area ΔA of a resolution cell for a pulsed radar can be modeled as[2]

$$\Delta A = w\ell = \left[\frac{h'\phi}{\cos\theta \cos\theta_A}\right] \cdot \left[\frac{c\tau'}{2\sin(\theta - \theta_c)}\right] \qquad (2)$$

where: w = Size of resolution cell in the along-track direction;

ℓ = Size of resolution cell in the cross-track direction;

h' = Height difference between the cell and the radar;

ϕ = Antenna beam width;

θ = Radar incidence angle;

τ' = Signal pulse width;

θ_A = Local slope of resolution cell in the along-track;

θ_c = Local slope of resolution cell in the across-track.

---

[2] Holtzman, J. C., V. H. Kaupp, R. L. Martin, E. E. Komp, and V. S. Frost, "Radar Image Simulation Project: Development of a General Simulation Model and an Interactive Simulation Model, and Sample Results," TR 234-13, Remote Sensing Laboratory, The University of Kansas, February, 1976.

If the average return power $\overline{P}_{r_{c_1}}$ from a particular category of scatterer (C) at a particular angle of incidence (θ_1) for a calibration system is known, then equation (1) can be rewritten:

$$\overline{P}_{r_{c_1}} = \frac{P_{T_1}\sigma^\circ_c(\theta_{\ell_1}) \cdot \Delta A_1 \cdot G_1^2(\theta_1)\lambda_1^2}{(4\pi)^3 \cdot R_1^4} \tag{3}$$

Equation (1) can be used to define the average return power $\overline{P}_{r_{c_2}}$ from the same category of scatterer (C) at a different angle of incidence (θ_2) for the radar system being modeled.

$$\overline{P}_{r_{c_2}} = \frac{P_{T_2}\sigma^\circ_c(\theta_{\ell_2}) \cdot \Delta A_2 \cdot G_2^2(\theta_2)\lambda_2^2}{(4\pi)^3 R_2^4} \tag{4}$$

Dividing Equation (4) by Equation (3) gives:

$$\overline{P}_{r_{c_2}} = \overline{P}_{r_{c_1}} \left[\frac{P_{T_2}\sigma^\circ_c(\theta_{\ell_2})\Delta A_2 G_2^2(\theta_2)\lambda_2^2 R_1^4}{P_{T_1}\sigma^\circ_c(\theta_{\ell_1})\Delta A_1 G_1^2(\theta_1)\lambda_1^2 R_2^4} \right] \tag{5}$$

Thus the average return power for each point in an image (for any particular category of scatterer) can be found if the average return power for one point belonging to a category is known (σ°_c absolute) in addition to backscattering coefficients and angles of incidence for both points.

The visual presentation medium modeled was photographic film. An image density which is related to the average return power is displayed on real SLAR images. This image density is often called a greytone, which is a relative measure, that is, it was produced with respect to some calibration reference. In a digitized image, the greytone represents a specific image density level within the possible dynamic range on the image. The return power (related to video intensity), properties of the film and photographic processing methods determine the photographic density (D) on the film by[4]

---

[4] Goodman, J. W., Introduction to Fourier Optics, Chapter 7, McGraw-Hill, 1968.

$$D = \gamma \log_{10} I + \log_{10} k \tag{6}$$

where: γ = Gamma of the film;

k = A positive constant which depends upon the exposure time and on the film processing and development;

I = Image intensity;

D = Photographic density.

This relationship holds true in the linear portion of the film dynamic range. If a linear radar receiver is assumed, the the intensity (I) is directly related to the average return power ($\overline{P}_r$).

$$I = M \cdot \overline{P}_r \tag{7}$$

where: M = A proportionality constant.

Rewriting (6) to incorporate (7) gives

$$D = \gamma \log_{10} \overline{P}_r + \log_{10} k + \gamma \log_{10} M \tag{8}$$

This result will be used for the final greytone expression.

The photographic density (D) is defined by[4]

$$D = \log_{10}\left(\frac{1}{\tau}\right) \tag{9}$$

where the film transmittance τ is given by[4]

$$\tau = \frac{I_t}{I_o} \tag{10}$$

with: I_t = Transmitted intensity;

I_o = Incident intensity.

Rewriting Equation (9) produces

$$D = \log_{10}\left(\frac{I_o}{I_t}\right) \tag{11}$$

---

[4] Goodman, J. W., Introduction to Fourier Optics, Chapter 7, McGraw-Hill, 1968.

The normalized value of the density (D) is the greytone level. The image simulations were digitized into 8-bit binary words for display, allowing 256 distinct greytones. If the optical density used in a simulation is in the range from 0 to X, the linear portion of the film dynamic range is required to be of sufficient size that

$$I_{t_{min}} = 10^{-X} I_o \tag{12}$$

In the simulated imagery produced at RSL, a greytone of 255 was associated with white and zero with black. This is the expression for a positive. This signal is then used to intensity modulate a CRT (a positive image). This image is then photographed (a negative). The negative is then used to produce a photo (a positive) which is the final product. The greytone level G_r corresponding to a density D was given by

$$G_r = \frac{255}{X} \cdot D \tag{13}$$

The general relationship between the average return power and greytone level for each resolution cell in the simulated image is obtained by rewriting Equation (8), incorporating Equations (5) and (13):

$$G_{r_{c_2}} = G_{r_{c_1}} + \frac{255}{X} \left[\gamma \log_{10} \left\{ \frac{P_{T_2} \sigma^\circ_{c_2}(\theta_{\ell_2}) \Delta A_2 G_2^2(\theta_2) \lambda_2^2 R_1^4}{P_{T_1} \sigma^\circ_{c_1}(\theta_{\ell_1}) \Delta A_1 G_1^2(\theta_1) \lambda_1^2 R_2^4} \right\} + \log_{10}\left[\frac{k_2}{k_1}\right] + \gamma \log_{10}\left[\frac{M_2}{M_1}\right] \right] \tag{14}$$

This equation states that the relative greytone values for each point in an image corresponding to any angle of incidence of a category of scatterers can be obtained if the greytone and angle of incidence is known for one discrete point in any category. This equation is the general result which establishes the relationship between the average return power and relative greytone levels for a simulated radar image. It should be pointed out that on a radar image and, thereby, on a simulated radar image, the important parameter is the relative greytone level between points on the image and not the absolute values of the greytone. Since this is true, Equation (14) can be used to establish the relative greytone levels between all categories of scatterers included in an image provided that the absolute

values of the scattering coefficient, σ°, and the appropriate angles of incidence for all points are known and provided that, in addition, the greytone of one point in the image is known. The more general result is, then:

$$G_{r_{c_2}} = G_{r_{a_1}} + \frac{255}{x}\left[\gamma \log_{10}\left\{\frac{P_{T_2}\sigma^\circ_{c_2}(\theta_{\ell_2})\Delta A_2 G_2^{\,2}(\theta_2)\lambda_2^{\,2} R_1^{\,4}}{P_{T_1}\sigma^\circ_{a_1}(\theta_{\ell_1})\Delta A_1 G_1^{\,2}(\theta_1)\lambda_1^{\,2} R_2^{\,4}}\right\} + \log_{10}\left(\frac{k_2}{k_1}\right) + \gamma \log_{10}\left(\frac{M_2}{M_1}\right)\right] \qquad (15)$$

where: $G_{R_{c_2}}$ = Greytone level of some point belonging to category 'c' and having the scattering coefficient, $\sigma_{c2}^\circ(\theta_{\ell 2})$, measured at the appropriate angle of incidence, $\theta_{\ell 2}$, by a radar system;

$G_{R_{a_1}}$ = Known greytone level for a particular point belonging to category 'a' and having the scattering coefficient, $\sigma_{a1}^\circ(\theta_{\ell 1})$, measured at the appropriate angle of incidence $\theta_{\ell 1}$, by a calibrated system.

Equation (15) is the general theoretical result which has been implemented in a computer simulation package. Implicit in this equation are the effects of geometry which cause the phenomena of shadow, layover and reflection of microwave energy (in accordance with local tilt of resolution cells). Reference [2] contains a more complete treatment of these effects.

2.1 Fading

Equation (15) gives the impression that a deterministic process is occurring. The interaction between the radar and a surface is a random process to some extent, and the mean value of the greytone level for a given set of parameters is described by Equation (15).

If one assumes that the process can be modelled as additive, i.e. signal = mean + noise, a model for the true statistical properties of the phenomena can be derived. The return amplitude from a single scattering region ha a noise-like characteristic that follows a Rayleigh distribution[5]. This

[5] Moore, R. K., Radar for Geoscience Instrumentation, Chapter 5.6, Geoscience Instrumentation, (E.A. Wolfes, Editor), John Wiley and Sons.

distribution is characterized by equal mean and standard deviation. The probability for N' such scatterers follows a chi-squared distribution with 2N' degrees of freedom. The mean and variance of chi-squared distribution are given by:

mean = degrees of freedom = $2N' = \mu$

variance = 2 (mean) = $4N'$

standard deviation = $\frac{\text{mean}}{\sqrt{N'}}$

For radar, the number of independent samples in a given resolution cell is given by[5]:

$$N = \frac{2w}{L} = \frac{2h'\phi^2}{\lambda\cos\theta} \tag{16}$$

where: θ = Radar incidence angle;

w = Azimuth resolution;

L = Horizontal aperture length of antenna ($L \approx \frac{\lambda}{\phi}$);

h' = Effective altitude;

ϕ = Antenna beamwidth;

λ = Wavelength of microwave energy.

Therefore, the standard deviation due to fading can be represented by:

$$\frac{\text{mean}}{\sqrt{N}} = \bar{S}_x = \frac{\mu}{\sqrt{N}} \tag{17}$$

For a particular cell, the return power (P_r) can be represented as [2]

$$P_r = \mu + \bar{S}_x \cdot RN \tag{18}$$

when the number of independent samples is large. RN is a Gaussian random variable (bipolar).

For computational purposes the random variable (RN) can be normalized to zero mean and variance of one leading to:

$$P_r = \mu + \frac{\mu}{\sqrt{N}} \times RN = \mu\left(1 + \frac{RN}{\sqrt{N}}\right) \tag{19}$$

---

[5] Moore, R. K., Radar for Geoscience Instrumentation, Chapter 5.6, Geoscience Instrumentation, (E. A. Wolfes, Editor), John Wiley and Sons.

This can be shown in terms of greytones[2] as

$$G_r = G_{r_{c_2}} + \log_{10} \left(1 + \frac{RN}{\sqrt{N}}\right) \tag{20}$$

where: $G_{r_{c_2}}$ = Equation (15)

The term, due to fading, is described by:

$$G_{r_{FAD}} = \frac{255}{x} \gamma \log_{10} \left(1 + \frac{RN}{\sqrt{N}}\right) \tag{21}$$

where x is base 10 logarithm of the linear dynamic range of the film.

This equation used in conjunction with Equation (15) defines the image intensity a specified SLAR would produce corresponding to a particular target area. Geometrical effects that enter the determination of target area (resolution cell size and local orientation) are treated rigorously in the digital simulation model and are thoroughly covered in Reference [2]. The SLAR simulation programs are able to treat both the case of little or no and of significant relief. Second-order effects such as multipath which may occur due to local geometry have not yet been implemented in the SLAR program.

The succeeding sections discuss the macro-flow chart of the SLAR programs, the input data which may be adjusted by the user according to the application, and the data base that has been an experimental tool for validation of the closed system model for digital simulation of radar images. A copy of the SLAR programs is included at the conclusion of this appendix.

---

[2] Holtzman, J. C., V. H. Kaupp, R. L. Martin, E. E. Komp, and V. S. Frost, "Radar Image Simulation Project: Development of a General Simulation Model, and Sample Results," TR 234-13, Remote Sensing Laboratory, The University of Kansas, February, 1976.

3.0 SLAR SIMULATION PROCESS

The SLAR simulation package consists of four separate programs and associated subroutines as given in Figure 1. They can be run separately so that (1) the failure of one module does not affect successful completion of the previous programs, and (2) experimentation can be conducted in one program for a particular radar being modeled, thus avoiding the cost of rerunning the entire package. For example, it might be desired to change a reference greytone or the dynamic range in the simulated image. This structure for the software would allow SLAR Geometry, SLAR Slope, and SLAR Shadow (the first three programs) to be run once, and the output tape to be used for several successive experimentations of SLAR Greytone (the fourth program).

MACRO-FLOW CHART OF SLAR SIMULATION PROCESS

Figure 1

3.1 Input Parameters

Four arrays, RADAR, GROUND, REFER and PROCES contain locations for data which vary with the desired application of the SLAR simulation and may be adjusted by the user. Descriptions of the arrays are contained in Figure 2 which is a summary of all parameters which are changeable. The asterisks denote exceptions as explained at the conclusion of the listing. The user must have a minimal knowledge of the interdependence of the radar and ground parameters before attempting to alter the values in these matrices. For example, a change in the parameter RADAR(6) may necessitate compensation in GROUND (1), (2), (5), (10) and RADAR (7), (8), and (9) parameters.

3.2 Data Base

An artificial data base (scene consisting of geometrical solid shapes) was used to test the operation of both the simulation model and the software. A data base of geometrical shapes was used for these purposes because of the known input/output response characteristics of such a data base. The input was known exactly and the output was deterministic and calculable. The artificial data base described by Komp[6] is a composite of three dimensional objects The matrix of elevation and category data (stored in a computer word for each position) which forms the data base is 700 by 1000 points in extent, representing 20 by 20 foot resolution. Thus, the data base appears to be 14,000 by 20,000 feet. Known empirical backscatter characteristics, in the form of sigma zero versus angle of incidence, for ten separate vegetation categories, have been assigned to the ground scene. The value of such a data base with completely specified dielectric and geometric properties is that it provides us with a means to validate the SLAR simulation. The calculable response (eg., extent of shadow and layover due to each object) of the geometric solids to various flight tracks, frequency/polarization combinations, near and far range depression angles, etc., allows the final image product to be judged objectively. Naturally, a high resolution (or large matrix) data base can be generated by a computer; however, the number of data points used will be limited by the specific application and computer resources.

---

[6] Komp, E. D., V. H. Kaupp, and J. C. Holtzman, "Construction of a Geometric Data Base for Radar Image Simulation Studies," TR 319-1, Remote Sensing Laboratory, The University of Kansas, July, 1976.

SIMULATION PARAMETERS

SIMULATED RADAR PARAMETERS

*RADAR (1)	ORIENTATION (# OF DEGREES)
*RADAR (2)	LATITUDE OF START (DEGREES)
*RADAR (3)	LONG. OF START OF FLIGHT TRACK
**RADAR (4)	LATIT. OF END OF FLIGHT TRACK
**RADAR (5)	LONG. OF END OF FLIGHT TRACK
RADAR (6)	SWEEP DELAY TIMES (SECONDS)
RADAR (7)	SWEEP LENGTH TIME (SECONDS)
RADAR (8)	PULSEWIDTH (SECONDS)
RADAR (9)	AZIMUTH BEAMWIDTH (RADIANS)
**PRF (10)	PRF (HERTZ)
**RADAR (11)	AIRSPEED (FT/SEC)
RADAR (12)	ALTITUDE (MSD. IN FT.)
RADAR (13)	FREQUENCY (GHz)
RADAR (14)	POLARIZATION (1,HH;2,VV;3,HV;4,VH)
*RADAR (15)	NEAR RANGE DIST. (FT) (COMPLETED)
RADAR (16)	POWER TRANSMITTED (WATTS)
RADAR (17)	# OF INDEPENDENT SAMPLES (REFERENCE 2)
RADAR (18-20)	NOT PRESENTLY USED

GROUND TRUTH PARAMETERS

GROUND (1)	DISTANCE FROM FLIGHT TRACK
GROUND (2)	DATAPT SIZE (ALONG TRACK)
GROUND (3)	DATAPT SIZE (ACROSS TRACK)
GROUND (4)	IS F AT EARTH REASONABLE (1=YES)
GROUND (5)	SIZE OF MATRIX (ALONG TRACK,#DATA PTS)
GROUND (6)	SIZE OF MATRIX (ACROSS TRACK,#DATA PTS)
GROUND (7)	AVERAGE HEIGHT OF GROUND-IF FLAT EARTH
GROUND (8)	NO CATEGORIES PRESENT (1=TRUE)
*GROUND (9)	RESOLUTION SIZE IN CROSS-TRACK (COMP)
*GROUND (10)	RESOLUTION SIZE IN ALONG-TRACK(COMP)
*GROUND (11)	# OF DATAPT IN ALONG DIRECTION (COMP)
*GROUND (12)	# OF CELLS ALONG IN GEO. MATRIX (COMP)
*GROUND (13)	# OF CELLS ACROSS IN GEO. MATRIX (COMP)
GROUND (14-20)	PRESENTLY NOT USED

***REFERENCE CELL PARAMETERS

REFER HOLDS INFORMATION ABOUT THE REFERENCE CELL AS FOLLOWS

REFER (1)	REFERENCE GREYTONE
REFER (2)	PT. POWER TRANSMITTED
REFER (3)	S.GMA *-(FUNCT., ANG.) REFER = [dB/10]
REFER (4)	AREA OF CELL
REFER (5)	ANTENNA GAIN - F(THETA), REFER = $GAIN_2$
REFER (6)	REFERENCE RADAR WAVELENGTH, REFER - λ^2
REFER (7)	RANGE TO CELL SQUARED
REFER (8-10)	PRESENTLY NOT USED

PROCESSING PARAMETERS

- PROCES HOLDS INFORMATION PERTAINING TO THE PROCESSING OF THE MATRIX. PROCES (5) THRU PROCES (13) HOLD SWITCHES WHICH TELL THE PROGRAM TO EXCLUDE (SW 0) OR INCLUDE SW=0), THE TERM SPECIFIED IN THE GREYTONE FORMULA.

PROCES (1)	X, OPTICAL DENSITY RANGE
PROCES (2)	GAMMA, A PHOTOGRAPHIC CONSTANT
PROCES (3)	KR, PHOTO CONSTANT OF REFERENCE RADAR
PROCES (4)	AR, PHOTO CONSTANT OF REFERENCE RADAR
PROCES (5)	SW - K TERM
PROCES (6)	SW - M TERM
PROCES (7)	SW - PT TERM
PROCES (8)	SW - WAVELENGTH TERM
PROCES (9)	SW - ANTENNA GAIN TERM
PROCES (10)	SW - BACKSCATTER TERM
PROCES (11)	SW - AREA TERM
PROCES (12)	SW - RANGE TERM
PROCES (13)	SW - FADING TERM
PROCES (14)	K, PHOTO CONSTANT OF TEST RADAR
PROCES (15)	M, PHOTO CONSTANT OF TEST RADAR

*THESE PARAMETERS ARE COMPUTED BY THE PROGRAM - INPUT A ZERO
**THESE PARAMETERS ARE PRESENTLY NOT USED - INPUT A ZERO
***IF NOT INCLUDING A PARAMETER AS INDICATED IN THE PROCESS ARRAY - INPUT A ZERO

Figure 2

The data base of geometric solids is shown in Figure 3. The flat checkerboard conceptually represents areas of several vegetation types. The objects consist of hemispheres, cylinders, pyramids and rectangular parallelopipeds, with heights and dimensions drawn to scale. The variety of shapes present many different local tilts to the radar and will cause greytone changes at the sides of objects. By making the data base large and with resolution much finer than the simulated radar image, it was possible to study many properties of the simulation software (e.g., the formation of resolution cells by the SLAR package and the layover effects).

Figure 4 shows the relative position and elevation of the many objects on the data base. The maximum variation of relief is 2000 feet, therefore, a wide range in the amounts of shadow and layover are expected to be evident in simulated imagery of this scene.

Figure 5 illustrates the microwave reflectivity category assignments for the objects. Comparing Figures 4 and 5, it is seen that the same vegetation covers several objects of differing height. The value of empirical backscatter data for simulation is readily apparent, for if it were not available, complicated analytical expressions derived from volume scattering theories would have been necessary as input data. Backscatter data at increments of one degree between zero and ninety degrees (incidence angle) were stored (for each vegetation type) on input file tapes.

The SLAR simulation produces two types of output: (1) a "shadow map" (hard-copy output of the SLAR package) and (2) a magnetic tape compatible with a digital computer controlling a display medium. The shadow map consists of a large matrix of characters, zeroes representing shadow areas, nines indicating no data, and all other numbers the sum of the data points falling within a resolution cell bin. Both types of output are illustrated in the following sections for a radar system operating at 8.6 GHz with HH polarization (with the exception of the frequency/polarization study results). Thus, the shadow map indicates the shadow as well as layover, and sums the number of transposed cells.

Figure 3 Data Base of Geometric Solids

Figure 4. Relative Position and Elevation.

Figure 5. Microwave Reflectivity Category Assignments.

4.0 DIGITAL SLAR SIMULATION RESULTS

The data base information (data point location in the array, height, and category) contained on magnetic tape[7] generated by the method of Reference [6] was used as the ground scene for the SLAR simulation programs. Two sets of visual displays were produced, shadow maps and radar image simulations.

The map shown in Figure 6a illustrates the effects of layover and shadow that would occur for a SLAR operating at 8.6 GHz, HH polarization flown at 40,000' altitude and 20° near range depression angle. Comparing this map to Figures 3 and 4 it is seen that the level checkerboard pattern lies in the upper portion of Figure 6a. The symbols on the map indicate the sum of the resolution cells whose radar returns simultaneously reach the SLAR receiver. Zeroes represent shadow, or lack of appreciable return, and nines are employed to show no data. Shadow and layover effects do not occur on the flat terrain, just as would be predicted. Careful study of Figures 3, 4, and 6a reveal that the SLAR package is accurate in handling radar geometrical phenomena. This particular version of the simulation limited the number of resolution cells laying over into one bin to a sum of six. Examination of the map shows that the maximum number of points to enter one bin was five, thus the above restriction did not limit the performance of the programs.

Figure 6b simulates the shadow and layover effects for a SLAR (8.6 GHz HH) at 40,000 feet and a near range depression angle of 40°. Shadows are shorter and layover more pronounced for this case than for the previous one. The centrally located pyramid in 6b shows a larger area of layover. However each bin contains a maximum of three resolution cell radar returns. The smaller area of layover in Figure 6a has bins with up to five resolution cell returns summed. The more distinct outlines of

---

[7] Martin, R. L, Martin's Thesis, "SLAR Simulation and Applications," University of Kansas, 1976. (This document addresses problems of handling input and output data, tape compatibilities of machines, and system routines.)

[6] Komp, E. D., V. H. Kaupp, and J. C. Holtzman, "Construction of a Geometric Data Base for Radar Image Simulation Studies," TR 319-2, Remote Sensing Laboratory, The University of Kansas, July, 1976.

FIGURE 6a. 20° DEPRESSION ANGLE, 40,000 FT. ALTITUDE
9's INDICATE NO DATA, 1's-NO LAYOVER, 0's-NO SHADOW

FIGURE 6b. 40° DEPRESSION ANGLE, 40,000 FT. ALTITUDE

RADAR SIMULATION STUDY OF GEOMETRIC DATA BASE

Original

Orginal 3-D

8.6 HH

8.6 VV

13.8 HH

13.8 VV

16.2 HH

16.2 VV

FIGURE 7.

6b allow each object to be identified in conjunction with Figures 3 and 4.

It can be seen that radar interpretation for military personnel could be facilitated by the use of shadow-layover maps similar to those presented. The shadow map allows one to understand differences in image formation between radar and conventional photographic processes. The distinctions arise because the radar emits pulsed energy such that the return energy can be plotted versus time, the lack of return power causing shadow, and the propagation time determining the placement of relative greytones due to return signals. However, in the case of optical photographic imaging, the source of illumination (the sun) emits energy continuously, the reflected energy giving rise to a video signal which cannot be charted versus time without considerable ambiguity as to the time origin of the transmitted signal. Thus the continuous superposition of reflected energy masks shadow formation (in the sense of lack of return signal) when the photographic imaging platform is in the path between the illuminator and the object to be sensed. The handicap imposed by familiarity solely with photographic images is difficult to overcome, and this can only be accomplished through practice. Experience with shadow and layover for various flight situations simulated with the aid of an artificial data base could be extremely useful for overcoming the urge to think in terms of photographic phenomena. The shadow map, by the very fact that it contains a limited amount of quantified information about the processes occurring in radar terrain imaging, provides the necessary elements for understanding shadow and layover. Despite the fact that the geometrical data base is not representative of common landscapes, the corresponding maps generated by simulation are valuable.

Simulated radar images for a SLAR altitude of 4000 feet and a near range depression angle of 40 degrees (far range ten degree depression angle) were produced for a frequency/polarization study. The accentuation of shadow and layover was caused by the presence of objects on the data base up to 2000 feet high and the extremely low flight altitude. The results are shown in Figure 7 for six different frequency/polarization combinations. Several interesting radar effects can be spotted by an untrained eye: (1)

near-range compression; (2) backscatter category discrimination and variability; (3) same-category blocks of the checkerboard changing greytone across the map because of changes in sigma zero data with progressing angle of incidence, theta; (4) selective brightening of objects with polarization adjustment; (5) obviously, layover and shadow; and (6) loss of signal due to local tilt of resolution cell on sides of cylinders.

It is possible that a fine resolution radar (and correspondingly finer data base resolution) would indicate the presence of hard targets on the geometrical data base due to some of the corners and edges of the shapes involved, although the categories strictly represent vegetation. However, this would not be likely if the real SLAR produced 60 foot resolution as simulated, for the random averaging processes across a resolution cell would somewhat mask the effects of return power from the hard targets (eg., top edges of cylinders and parallelepipeds). Treatment of this data base (with computer generated resolution as fine as desired) as a composite of distributed targets is accurate as long as coarse (eg., 60 x 60 foot) resolution is employed in the simulated imagery.

Assume that the flight mission of a real SLAR was to produce category information over flat terrain and that a low altitude constrained the near and far range depression angles such that only the middle third of the images in Figure 7 were sensed. Such a mission would be fruitful if the particular radar system was operating at 8.6 GHz with HH polarization. With any other combination of frequency/polarization shown, the information gathered would not justify the expense. Radar parameters have already been studied in this manner for the Earth Resources Shuttle Imaging Radar to predetermine discrimination ability[8]. Whether the imaging system is a SAR or SLAR, waste and inefficiency in related missions can be avoided by forethought; a helpful aid would be the simulation of radar images from data base with relief and category (hopefully, empirical backscatter) information, be it a real site or the construct of a geographer's imagination.

Rather than simulating flight at 4000 feet in the midst of terrain varying in height from 0-2000 feet, an altitude of 40,000 feet might be more reasonable for a SLAR. Experiments have been conducted with two

---

[8] Bush, T. F., "Cropland Inventories Using a Satellite Altitude Imaging Radar," Ph.D. Dissertation, University of Kansas, Spring, 1977.

flight geometries as shown in Figure 8. The extent of shadow and layover will differ in the cases of 20 and 24 degree near range depression angles. Figure 9 illustrates that shadows will be longer in the 20° example, and that data points layover greater distances toward the flight track in the 24° depression angle case, especially in the near range (this assumes ground range presentation). However, more brightening will occur in the layover areas of the 20° case due to the greater density of cells entering bins on the near range side of tall objects.

The resulting simulated radar images are shown in Figures 10 and 11. Close examination of lengths of shadows and layover are aided by employing the checkerboard as a distance marker. Allowing for the distortion in photographs of the IDECS* visual display, the shadows are longer in Figure 10 and data points lay over greater distances toward the flight track in Figure 11. Near range compression due to the slant range presentation partially masks the difference in layover between the two images. Unfortunately, the scan lines in the IDECS display interfere with the study of the two simulations, so they must be viewed at a distance. However, differences can still be distinguished in the category discrimination ability, fading, object brightening, shadow and layover. The center pyramid in the Figure 10 has many cells located in bins along a line parallel to the flight track, whereas Figure 11 has a larger area of brightening (lower intensity) constituting a triangular section.

In conclusion, it has been shown that the SLAR simulations correctly represent layover and shadow. The use of the artificial data base makes possible the quantification of these radar phenomena for validation of the SLAR model. Empirical backscatter data at the frequencies and polarizations of Figure 7 lends credence to the idea that the SLAR simulation is very valuable for system optimization tests, especially

---

* IDECS - Acronym for Image Discrimination, Enhancement, Combination and Sampling, (an image processing station), Remote Sensing Laboratory.

Figure 8. Flight Geometries for Equal Altitude Test.

Figure 9. Shadow and Layover Difference.

RADAR SIMULATION OF ARTIFICIAL DATA BASE
20° DEPRESSION ANGLE 40,000 FT. SLAR ALTITUDE

FIGURE 10

RADAR SIMULATION OF ARTIFICIAL DATA BASE
24° DEPRESSION ANGLE 40,000 FT. SLAR ALTITUDE

FIGURE 11

in the area of backscatter category discrimination. It is not possible to make statements about the overall adequacy of the closed system model for it is not yet possible to objectively measure the quality of the simulated images and their suitability for specific applications. Future studies are being aimed toward the goal of defining image quality factors for very specific applications, but until these factors can be quantified, the value and quality of the simulation is left to the subjective opinion of engineers and radar interpreters. It should be noted, however, that simulations of real scenes produced at RSL have consistently been mistaken for real imagery by staff radar interpreters.

5. Conclusions and Recommendations

The calculable response of an artificial data base has been used to explore the accuracy of the SLAR simulation programs which embody the theory briefly described in this document. The closed-system model which has been developed to imitate ground-radar interactions has been shown to have considerable flexibility by the application of the results presented. The simulated images shown herein illustrate effects such as (1) layover; (2) shadow; (3) frequency/polarization; (4) near range compression; and (5) category discrimination. The value of image simulation for pre-evaluation of future radar systems has been stressed because recent developments since the publishing of Reference [2] have shown the utility of a geometrical data base (with empirical backscatter characteristics and computer generated shapes) in conjunction with the SLAR simulation programs of the Remote Sensing Laboratory.

The lack of a large catalogue of empirical backscatter data (VV, VH, HH, HV, circular direct and cross polarizations) and computer resources, have hampered the simulation efforts to some degree. The apparent advantages of accurately predicting system performance prior to construction as well as use in guidance systems would seem to be incentive to apply what has been learned about radar-ground interactions through use of this particular closed-system model.

The radar image simulation methods developed will next be applied to a real data base (Pickwick, test site) to measure its performance and to validate the model by comparison of simulations with real imagery of the same site. Subsequently, experiments will be performed at the Engineering Topographic Laboratories to determine the suitability of the simulated imagery for missile navigation systems in the developmental stage.

---

[2] Holtzman, J.C., V.H. Kaupp, R.L. Martin, E.E. Komp, and V.S. Frost, "Radar Image Simulation Project: Development of a General Simulation Model and an Interaction Simulation Model, and Sample Results," TR 234-13, Remote Sensing Laboratory, The University of Kansas, February 1976.

The SLAR simulation program is contained in this section. The data which must be supplied to the program were previously listed in Table 1. Due to interdependence of some of the Radar and Ground matrix values, the following formulas are useful:

$$*Gr(9) = R(8) \cdot \frac{c}{2}$$

$$*Gr(10) = R(6) \cdot \frac{c}{2} \cdot R(9)$$

$$*Gr(11) = Gr(10)/Gr(2) + .5 \qquad 1 \leq Gr(11) \leq 5$$

$$*Gr(12) = G(5)/Gr(11)$$

$$*Gr(13) = R(7)/R(8)$$

$$*R(15) = R(16) \cdot \frac{c}{2}$$

The asterisked quantities are computed within the body of the program. When changing R(6), for example, R(9) must be altered to compensate such that Gr(10) has the desired value.

Parameters for Figures 4-5 and 4-6 are given below for a more complete description of the radar systems being simulated by the SLAR programs.

	Figure 4-5	Figure 4-6
Gr(1)	109,900	89,840 feet
Gr(2)	20	20 feet
Gr(3)	20	20 feet
Gr(5)	700	700 feet
Gr(6)	1000	1000 feet
Gr(10)	80	80 feet
R(6)	2.38×10^{-4}	2.00×10^{-4} seconds
R(7)	3.93×10^{-5}	3.77×10^{-5} seconds
R(8)	1.63×10^{-7}	1.63×10^{-7} seconds
R(9)	6.80×10^{-4}	8.10×10^{-4} radians
R(12)	40,000	40,000 feet

```
$       IDENT   46093595$STILSTINGY+D,CHEUVRONT      MCVEIL2 BIG - 116.4
$       OPTION  FORTRAN
$       FORTRAN
C       SLAR GEOMETRY
C
C
C
C     PARAM1 IN LABELED COMMON FOR THE GEOMETRY PARAMETERS
C
C           GEOM(1)       ALONG TRACK RESOLUTION CELL SIZE (FEET)
C           GEOM(2)       ACROSS TRACK RESOLUTION CELL SIZE (FEET)
C           GEOM(3)       ALONG TRACK DATA POINT SIZE (FEET)
C           GEOM(4)       ACROSS TRACK DATA POINT SIZE (FEET)
C           GEOM(5)       ALONG TRACK DIMENSION SIZE (NLIN)
C           GEOM(6)       ACROSS TRACK DIMENSION SIZE (NCOL)
C           GEOM(7)       DISTANCE TO NEAR RANGE (FEET)
C           GEOM(8)       ALTITUDE - MEAN SEA LEVEL (FEET)
C           GEOM(9)       AVERAGE HEIGHT OF GROUND (FEET)
C           GEOM(10)      ALTSQ - REAL ALTITUDE SQUARED (FEET)
C           GEOM(11)      NCELAL - NUMBER OF CELLS IN ALONG TRACK
C           GEOM(12)      NCELAC - NUMBER OF CELLS IN ACROSS TRACK
C           GEOM(13)      IALONG - NUMBER OF DATA POINTS PER ALONG
C                                      TRACK RESOLUTION CELL
C           GEOM(14)      NEARNG - NEAR RANGE DISTANCE (FEET)
C           GEOM(15)      FARRNG - FAR RANGE DISTANCE (FEET)
C
C     NOTE - GEOM 10-15 ARE CALCULATED FROM GEOM 1-9
C
C
C
      IMPLICIT INTEGER (O)
      COMMON /IO/ IN,OUT,INTAP,OUTTAP
      COMMON /PARAM1/ GEOM(15)
      COMMON /IDATA/ IDATPT(3,905)
      DATA IACRDIM /905/,IALGDIM /3/
      COMMON /ICATEG/ ICATTP(64),ICATOC(63)
      COMMON /WORK/ IBIN(320)
      DATA IACRSIZ /320/
      COMMON /NAMES/ IALONG,NEARNG,FARRNG,CELSIZ,PTSIZ,NCELAC,
     2  NPTSAC,ICELIN,ICELOT,ALTSQ
      DATA ICELIN,ICELOT /5,5/
      DATA IN,OUT,INTAP,OUTTAP /5,6,9,10/
      DATA ELBIAS,RMSFAC /70.,10./
C
C
C     SET RANDOM NUMBER SEED
C
      IS=ISEED(IDUMB)
C
C     READ THE FILE LOCATION ON THE INPUT TAPE
C
      READ(IN,5) INFIL
    5 FORMAT(I2)
C
      WRITE(6,6) IN,OUT,INTAP,INFIL,OUTTAP
    6 FORMAT(/,' INPUT IS FROM DEVICE #',I5,/,
     1 ' OUTPUT IS TO DEVICE #',I5,/,
     2 ' INPUT TAPE IS FROM DEVICE #',I5,' FILE-',I5,/,
```

```
0059          3 ' OUTPUT FILE IS DEVICE #',I5,//)
0060   C
0061   C    CALL POST TO POSITION THE INTAP TAPE TO THE PROPER FILE
0062   C
0063            IF(INFIL.LE.1) GOTO 7
0064            CALL POST(INTAP,0,INFIL,1,N)
0065            IF(N.NE.0) STOP
0066   C
0067   C        READ IN THE GEOMETRY PARAMETERS
0068   C
0069    7       READ(IN,10) GEOM
0070   10       FORMAT(F20.5)
0071   C
0072   C     MISCELLANEOUS CONSTANTS
0073   C
0074            PTSIZ=GEOM(4)
0075            CELSIZ=GEOM(2)
0076            NPTSAC=GEOM(6)
0077   C
0078   C    CHECK THAT NPTSAC DOESN'T EXCEED THE DIMENSIONED LIMIT
0079   C
0080            IF(NPTSAC.GT.IACRDIM) CALL ERROR(2)
0081   C
0082   C    FIND REAL ALTITUDE
0083   C
0084            ALTSQ=(GEOM(8)-GEOM(9))**2
0085            GEOM(10)=ALTSQ
0086   C
0087   C    FIND NUMBER OF DATA POINTS PER RESOLUTION CELL AND CHECK
0088   C    THAT IT DOESN'T EXCEED THE DIMENSIONED LIMIT.
0089   C
0090            IALONG=GEOM(1)/GEOM(3) + 0.5
0091            GEOM(13)=IALONG
0092            IF(IALONG.LT.1 .OR. IALONG.GT.IALGDIM) CALL ERROR(1)
0093   C
0094   C    FIND NUMBER OF CELLS IN THE ALONG TRACK DIRECTION
0095   C
0096            NCELAL=GEOM(5)/IALONG
0097            GEOM(11)=NCELAL
0098   C
0099   C    FIND NUMBER OF CELLS IN THE ACROSS TRACK DIRECTION
0100   C
0101   C    FIRST FIND THE NEAR RANGE DISTANCE AND FAR RANGE DISTANCE
0102   C
0103            NEARNG=SQRT(ALTSQ+GEOM(7)**2)
0104            NEARNG=NEARNG-CELSIZ*ICELIN
0105            GEOM(14)=NEARNG
0106   C
0107            FARRNG=SQRT(ALTSQ+(GEOM(7)+NPTSAC*PTSIZ)**2)
0108            FARRNG=FARRNG+CELSIZ*ICELOT
0109            GEOM(15)=FARRNG
0110   C
0111            NCELAC=(FARRNG-NEARNG)/CELSIZ+0.9999
0112            GEOM(12)=NCELAC
0113            IF(NCELAC.GT.IACRSIZ) CALL ERROR(3)
0114   C
0115   C    CALL FINDCELL TO DETERMINE THE CELLS IN THE ACROSS TRACK
0116   C
0117            CALL FINDCELL
0118   C
```

```
C       WRITE OUT RESULTS SO FAR
C
          WRITE(OUT,10) GEOM
          WRITE(OUTTAP) GEOM
C
C
C
C  NOW THAT THE PRELIMINARIES ARE DONE, DO THE WORK
C
C
C
C     MAIN DO LOOP, ONCE FOR EACH SCAN LINE
C
          DO 20 I1=1,NCELAL
C
C
C
C     READ IN IALONG COLUMNS OF DATA
C
          DO 30 I2=1,IALONG
          READ(INTAP) IX,IGARBG,(IDATPT(I2,J),J=1,NPTSAC)
   30     CONTINUE
C
C
C
C     NOW TAKE CARE OF THE AVERAGING OF DATA PTS
C
C
C
C     ROW COUNTER
C
   76     IROW=1
C
C     NOW FOR THESE COLUMNS OF DATAPTS--AVERAGE FOR EACH CELL
C     THE CORRECT NUMBER.
C
          DO 70 I6=1,NCELAC
C
C     INITIALIZE THE OUTPUT WORD
C
          IOUT=0
C
C   IF NO DATA IN THIS BIN, WRITE IOUT ( IOUT = 0 )
C
          IF(IBIN(I6) .EQ. 0) GO TO 75
C
C     SET POINTERS
          IZERO=0
          HIFLAG=0
          TREEHI=0
          TREFLG=0
          IHIGH=0
          ITEMP=1
C
C     AVERAGE THE CORRECT NUMBER OF ROWS
C
          DO 80 I7=1,IBIN(I6)
C
C     AVERAGE THE CORRECT NUMBER OF COLUMNS
C
```

```
      DO 90 I8=1,IALONG
C
C   UNPACK THE WORD.
C
      FLD(30,6,ICATTP(ITEMP))=FLD(30,6,IDATPT(I8,IROW))
      FLD(24,12,IHIGHT)=FLD(18,12,IDATPT(I8,IROW))
C
C ***********************************************************************
C
C   THIS CHANGES THE ELEVATION OF TREES ( CAT = 13 ).
C
      IF(ICATTP(ITEMP).NE.13) GOTO 94
      IHIGHT=IHIGHT+FLBIAS
      TREEHI=TREEHI+IHIGHT
      TREFLG=TREFLG+1
      HIFLAG=HIFLAG+RMSFAC
C
C ***********************************************************************
C
C
C     CHECK FOR DATA
C
C
 94   IF(ICATTP(ITEMP).NE.0) GOTO 92
C
      IZERO=IZERO+1
      GOTO 90
C
C   IF THERE WAS DATA IN THE POINT, UPDATE "ITEMP"
C
 92   ITEMP=ITEMP+1
C
C     ADD THE HEIGHTS FOR THE RESOLUTION CELL.
C
      IHIGH=IHIGH + IHIGHT
C
C
 90   CONTINUE
C
C
C     INCREMENT ROW COUNTER
C
      IROW=IROW+1
C
C
 80   CONTINUE
C
C
C     NUMBER OF NON-ZERO POINTS IN THAT CELL
C
      NLEFT=IALONG*IBIN(16)-IZERO
      IF(NLEFT.LE.0) GOTO 75
C
C     AVERAGE THE HEIGHTS FOR EACH CELL.
C
      IHIGH=(IHIGH + HIFLAG*RMS(IS))/NLEFT
C
C     IF THERE ARE 3 OR MORE POINTS OF TREES (CAT=13)
C     IN A RESOLUTION CELL, MAKE THAT CELL ALL TREES TO
C     ENHANCE FAR-SHORE BRIGHTENING.
```

```
C
      IF(TREFLG.LT.3) GOTO 88
C
      ICATTP(1)=13
      ICATOC(1)=ITEMP-1
      IHIGH=(TREEHI+RMS(IS)*HIFLAG)/TREFLG
      GOTO 100
C
C     SORT THE CATEGORIES IN THE RESOLUTION CELL
C
 88   ICATTP(ITEMP)=0
      CALL SORT(NUMB)
C
C     NUMB WAS THE NUMBER OF CATEGORIES ENCOUNTERED IN THIS CELL
C
      IF(NUMB.EQ.1) GO TO 100
C
C     IF THERE ARE MORE THAN ONE CATEGORY, THEY
C     MUST BE PACKED INTO THE RESOLUTION MATRIX
C     ALSO,  THIS IS DONE BY ADDING THE SIGN BIT
C     (IONE) TO EACH OF THE MULTIPLE CATEGORIES
C     IN ADDITION TO THE CATEGORIES, THE
C     NUMBER OF OCCURENCES IS ALSO PACKED.
C     THE NUMBER OF OCCURENCES IS KEPT IN AN ARRAY
C     CALLED ICATOC.
C
      FLD(0,1,IOUT)=1
      FLD(30,6,IOUT)=ICATTP(1)
      FLD(18,12,IOUT)=IHIGH
      FLD(12,6,IOUT)=IBIN(I6)
      FLD(6,6,IOUT)=ICATOC(1)
      WRITE(OUTTAP) IOUT
C
C     NOW FOR THE REST OF THE MULTIPLE CATEGORIES
C
      IF(NUMB .EQ. 2) GO TO 99
C
      DO 85 I9=2,NUMB-1
      FLD(30,6,IOUT)=ICATTP(I9)
      FLD(6,6,IOUT)=ICATOC(I9)
      WRITE(OUTTAP) IOUT
 85   CONTINUE
C
C     MAKE THE LAST WORD POSITIVE
C
 99   IOUT=ICATTP(NUMB)
      FLD(6,6,IOUT)=ICATOC(NUMB)
      GO TO 75
C
 100  IOUT=ICATTP(1)
      FLD(18,12,IOUT)=IHIGH
      FLD(12,6,IOUT)=IBIN(I6)
      FLD(6,6,IOUT)=ICATOC(1)
C
C
C     WRITE THE RESULTS TO TAPE
C
 75   WRITE(OUTTAP) IOUT
C
C
```

```
C
C
 70     CONTINUE
C
C
C
C    NOW WRITE OUT A MESSAGE TO INDICATE THIS ROW IS DONE
C
C
 73     WRITE(OUT,74) I1
 74     FORMAT(' SCANLINE ',I5,' DONE.')
C
C
C
C
 20     CONTINUE
C
C
C
C
C
C     THE GEOMETRY MATRIX IS COMPLETE
C
C
C
        ENDFILE OUTTAP
        REWIND OUTTAP
C
        STOP
C
        END
        SUBROUTINE FINDCELL
C
C       THIS ROUTINE FINDS THE PROPER PLACE TO START SEEING
C       WHAT THE RADAR SEES. THIS IS WITH RESPECT TO THE GEOMETRY
C       OF THE SITUATION.  IT ALSO FILLS IN THE RANGE BINS, I.E.
C       NUMBER OF DATA POINTS PER CELL.
C
C
        IMPLICIT INTEGER (O)
        COMMON /IO/ IN,OUT,INTAP,OUTTAP
        COMMON /PARAM1/ GEOM(15)
        COMMON /WORK/ IBIN(320)
        COMMON /NAMES/ IALONG,NEARNG,FARRNG,CELSIZ,PTSIZ,NCELAC,
     &  NPTSAC,ICELIN,ICELOT,ALTSQ
C
C     DEFINE CONSTANTS
C
        IPT=0
C
C
C    FILL IN IBIN WITH "ICELIN" ZEROES
C
        DO 10 I=1,ICELIN
        IBIN(I)=0
 10     CONTINUE
C
C    SET UP INITIAL PARAMETERS
C
        X=NEARNG+ICELIN*CELSIZ
        DIST=SQRT(X**2-ALTSQ)
```

```
0359.          ISUM=DIST/PTSIZ
0360   C
0361   C
0362   C
0363   C     NOW FILL IN THE REST OF THE IBINS
0364   C
0365   C
0366   C
0367    200    DO 300 I=ICELIN+1,NCELAC
0368           X=X+CELSIZ
0369           DIST=SQRT(X**2 - ALTSQ)
0370           OLDSUM=ISUM
0371           ISUM=DIST/PTSIZ
0372           IBIN(I)=ISUM-OLDSUM
0373   C
0374           IF(IPT+IBIN(I).GT.NPTSAC) GOTO 500
0375           IPT=IPT+IBIN(I)
0376    300    CONTINUE
0377   C
0378           CALL ERROR(4)
0379   C
0380   C
0381   C     COME HERE WHEN YOU RUN OUT OF POINTS
0382   C
0383   C
0384    500    IBIN(I)=NPTSAC-IPT
0385           IF(I.GE.NCELAC) GOTO 600
0386           DO 590 I=I+1,NCELAC
0387           IBIN(I)=0
0388    590    CONTINUE
0389   C
0390   C     NOW WRITE OUT IBIN TO LOOK AT IT
0391   C
0392    600    WRITE(OUT,601) NCELAC,(IBIN(I),I=1,NCELAC)
0393    601    FORMAT(/,' IBIN( 1 - ',
0394         & I4,' ) IS ',/,100(10(1X,O12),/),/)
0395           RETURN
0396   C
0397           END
0398   C       SUBROUTINE SORT
0399   C
0400   C     THIS ROUTINE SORTS AN ARRAY OF NUMBERS (UP TO
0401   C     SIXTY-THREE) AND RETURNS THE NUMBERS THAT
0402   C     WERE IN THE ARRAY,THE NUMBER OF OCCURENCES
0403   C     AND THE NUMBER OF DIFFERENT NUMBERS PRESENT(NUMB).
0404   C     THE FIRST TWO THINGS MENTIONED ARE PASSED THRU
0405   C     BY LABELED COMMON.
0406   C
0407   C     A ZERO(0) IN ICATTP(I) MEANS END OF DATA, IT ALSO
0408   C     ASSUMES THAT AT LEAST ONE NUMBER IS IN THE ARRAY
0409   C     PASSED TO IT.
0410   C
0411           SUBROUTINE SORT(NUMB)
0412           COMMON /ICATEG/ ICATTP(64),ICATOC(63)
0413   C
0414           DO 10 I=2,63
0415    10     ICATOC(I)=0
0416           ICATOC(1)=1
0417           NUMB=1
0418   C
```

```
      DO 30 J=2,63
      IF(ICATTP(J) .EQ. 0) RETURN
C
      DO 20 K=1,NUMB
      IF(ICATTP(J) .NE. ICATTP(K)) GO TO 20
      ICATOC(K)=ICATOC(K) + 1
      GO TO 30
   20 CONTINUE
C
      NUMB=NUMB + 1
      ICATTP(NUMB)=ICATTP(J)
      ICATOC(NUMB)=1
C
   30 CONTINUE
C
      RETURN
C
      END
      SUBROUTINE ERROR(I)
C
      IMPLICIT INTEGER (O)
      COMMON /NAMES/ IALONG,NEARNG,FARRNG,CELSIZ,PTSIZ,NCELAC,
     & NPTSAC,ICELIN,ICELOT,ALTSQ
      COMMON /PARAM1/ GEOM(15)
      COMMON /IO/ IN,OUT,INTAP,OUTTAP
C
      WRITE(OUT,10) I,GEOM
   10 FORMAT(I5,/,15(F20.5,/))
C
      GEOM(99999)=0
C
      RETURN
C
      END
$     EXECUTE
$     LIMITS   25,15K
$     TAPE     09,A1DD,,62009,,AAA333
$     FFILE    09,BUFSIZ/910
$     FILE     10,A2SR,170L
 1
 60.
 60.
 20.505
 20.505
 3000.
 905.
 67579.
 29100.
 362.
 0.
 0.
 0.
 0.
 0.
 0.
$     IF       ABORT,ENDJOB
$     OPTION   FORTRAN
$     FORTRAN
C     SLAR SLOPE
C
```

```
C     MAINLINE TO SLAR SLOPE
C
C
C
      IMPLICIT INTEGER (O)
      COMMON /IO/ IN,OUT,INTAP,OUTTAP,OUTEM(2)
      COMMON /PARAM1/ GEOM(15)
      COMMON /DATA/ ICAT(2,320),IHIGH(2,320),ANGLES(3,320)
      COMMON /NAMES/ NCELAC,ALGSIZ,PTSIZ,CELSIZ,NEARNG,ALTSQ,
     & SIZE,THETA,THETAL,ALTMSL
C
      DATA ITEMPO,IZERO /2*0/
      DATA IN,OUT,INTAP,OUTTAP,OUTEM /5,6,9,10,11,12/
C
C   WRITE /IO/ PARAMETERS
C
      WRITE(OUT,5) IN,OUT,INTAP,OUTTAP,OUTEM
    5 FORMAT(/,' INPUT  IS FROM DEVICE #',I5,/,
     1 ' OUTPUT IS FROM DEVICE #',I5,/,
     2 ' INPUT  FILE IS DEVICE #',I5,/,
     3 ' OUTPUT FILE IS DEVICE #',I5,/,
     4 ' TEMPORARY OUTPUT FILE NUMBER 1 DEVICE CODE IS',I5,/,
     5 ' TEMPORARY OUTPUT FILE NUMBER 2 DEVICE CODE IS',I5,///)
C
C
C
C     START PROCESSING
C
C
C     READ IN MATRIX PARAMETERS
C
      READ(INTAP) GEOM
C
C     WRITE OUT THE MATRIX PARAMETERS TO THE OUTPUT FILE
C
      WRITE(OUT,10) GEOM
      WRITE(OUTTAP) GEOM
   10 FORMAT(F20.5)
C
C   SET UP MISCELLANEOUS CONSTANTS
C
      NCELAC=GEOM(12)
      NCELAL=GEOM(11)
      NHAFAL=NCELAL/2
      PTSIZ=GEOM(4)
      CELSIZ=GEOM(2)
      NEARNG=GEOM(14)
      ALTSQ=GEOM(10)
      ALGSIZ=GEOM(1)
      ALTMSL=GEOM(8)
C
C   CALCULATE THE RADAR INCIDENCE ANGLE FOR ALL BINS.
C
      ALT=SQRT(ALTSQ)
      RANGE=NEARNG+CELSIZ/2.0
C
      DO 15 I=1,NCELAC
      DIST=SQRT(RANGE**2-ALTSQ)
      ANGLES(1,I)=DIST/ALT
      RANGE=RANGE+CELSIZ
```

```
 15     CONTINUE
C
C
C     UNPACK THE FIRST LINE
C
       CALL UNPAC2(1)
C
C     SET FIRST COLUMN POINTER
C
       NCOL1=1
C
C
C
C
C
C     NOW FOR EACH SCAN LINE OF RESOLUTION CELLS,
C     FIND THE LOCAL SLOPE AND THE RADAR ANGLE
C
       DO 30 I2=1,NCELAL-1
C
C     RATHER THAN PHYSICALLY SWITCHING THE TWO COLUMNS AROUND
C     EACH PASS THRU FOR EACH SCAN LINE, THE INDEX NAMING THE
C     FIRST COLUMN IS SWITCHED
C
       IF(NCOL1 .EQ. 1) NCOL2=2
       IF(NCOL1 .EQ. 2) NCOL2=1
C
C     SET ICURF TO THE CURRENT TEMPORARY FILE TO
C     AVOID 66/60 COMPILER ERROR
C
       ICURF=OUTEM(NCOL1)
C
C     UNPACK THE NEW LINE
C
       CALL UNPAC2(NCOL2)
C
C     FIND THE LOCAL CROSS TRACK SLOPE FOR EACH CELL OF THE COL.
C
       CALL ANGLE1(NCOL1)
C
C
C
C
C     NOW FOR EACH CELL FIND THE LOCAL ALONG TRACK SLOPE
C     AND THEN OUTPUT THE DESIRED PARAMETERS.
C
       DO 40 I3=2,NCELAC
C
C     STORE THE DO LOOP INDEX TO AVOID WARNINGS FROM THE COMPILER
C
       ITEMP2=I3
C
C     IF THE VALUE OF THE CELL IS ZERO, NO DATA IS ASSUMED,
C     PROCESSING OF THAT CELL IS TERMINATED, AND A ZERO
C     IS WRITTEN ON THE OUTPUT FILE.
C
       IF(ANGLES(2,I3).EQ.-1013.7) GOTO 50
C
C     FOR THIS CELL FIND THE LOCAL ALONG-TRACK SLOPE
C
```

```
0599.          CALL ANGLE2(NCOL1,NCOL2,ITEMP2)
0600           IF(ANGLES(3,I3).EQ.-1013.7) GOTO 50
0601   C
0602   C     NOW THAT ALL THREE ANGLES ARE IN THE LABELED COMMON,
0603   C     FIND THE OUTPUT ANGLES
0604   C
0605           CALL ANGOUT(NCOL1,ITEMP2)
0606   C
0607   C     NOW WRITE THE PACKED WORD CONTAINING THE CATEGORY AND HEIGHT
0608   C
0609           WRITE(OUTTAP) ICAT(NCOL1,I3)
0610   C
0611   C     IF THE CELL WAS MULTI-CATEGORY (A NEGATIVE NUMBER) GET THE
0612   C     OTHER CATEGORIES FROM THE APPROIATE TEXP FILE.
0613   C
0614           IF(ICAT(NCOL1,I3) .GT. 0) GO TO 60
0615   C
0616   C     THE LAST WORD OF THE GIVEN CELL WAS POSITIVE--I.E.
0617   C     IF THIS WORD IS POSITIVE GO TO THE NEXT STEP.
0618   C     OTHERWISE HANDLE ALL WORDS DESCRIBING THIS CELL.
0619   C
0620    70     READ(ICURF) IWORD
0621           WRITE(OUTTAP) IWORD
0622           IF(IWORD .LT. 0) GO TO 70
0623   C
0624   C     WRITE TO OUTPUT THE PARAMETERS COMPUTED FOR THIS CELL.
0625   C                1)THETA IS THE RADAR ANGLE
0626   C                2)THETAL IS THE LOCAL INCIDENCE ANGLE
0627   C                3)SIZE IS THE RELATIVE SIZE OF THE CELL
0628   C
0629    60     WRITE(OUTTAP) THETA,THETAL,SIZE
0630   C
0631           IF(NHAFAL.EQ.I2) WRITE(OUT,9998) ICAT(NCOL1,I3),THETA,
0632         &  THETAL,SIZE
0633   9998    FORMAT(5X,O12,3(3X,F10.4))
0634   C
0635           GO TO 40
0636   C
0637   C
0638   C     IF THE GIVEN CELL WAS ZERO, WRITE OUT A ZERO
0639   C     AND SKIP THE MULTICATEGORY WORDS ASSOCIATED
0640   C     WITH THIS CELL, IF ANY.
0641   C
0642    50     WRITE(OUTTAP) IZERO
0643           IF(NHAFAL.EQ.I2) WRITE(OUT,9998) IZERO
0644   C
0645           IF(ICAT(NCOL1,I3).GE.0) GOTO 40
0646   C
0647    51     READ(ICURF) IWORD
0648           IF(IWORD.LT.0) GOTO 51
0649   C
0650   C
0651    40     CONTINUE
0652   C
0653   C
0654   C
0655   C
0656   C     HALF WAY THRU WRITE OUT SOME RESULTS
0657    80     IF(NHAFAL .NE. I2) GO TO 85
0658           WRITE(OUT,101) (IHIGH(NCOL1,I10),I10=1,NCELAC)
```

```
      WRITE(OUT,103) NHAFAL
      WRITE(OUT,101) (IHIGH(NCOL2,I10),I10=1,NCELAC)
      WRITE(OUT,103) NHAFAL
      WRITE(OUT,102) (ANGLES(1,I10),I10=1,NCELAC)
      WRITE(OUT,103) NHAFAL
      WRITE(OUT,102) (ANGLES(2,I10),I10=1,NCELAC)
      WRITE(OUT,103) NHAFAL
      WRITE(OUT,102) (ANGLES(3,I10),I10=1,NCELAC)
      WRITE(OUT,103) NHAFAL
  101 FORMAT(20I6)
  102 FORMAT(20F6.2)
  103 FORMAT(/,(' DONE WITH THAT PARAMETER OF COLUMN #-',I5,//))
C
C     SWITCH THE FIRST COLUMN INDICATOR FOR THE NEXT SCAN LINE
C
   85 NCOL1=NCOL2
C
C    WRITE OUT A MESSAGE TO INDICATE THIS SCANLINE DONE
      WRITE(OUT,86) I2
   86 FORMAT(' SCANLINE ',I5,' IS DONE.')
C
C
   30 CONTINUE
C
C     MATRIX IS DONE
C
  100 ENDFILE OUTTAP
      REWIND OUTTAP
C
      STOP
C
      END
C     SUBROUTINE UNPACK2(IFILE)
C
C     THIS ROUTINE UNPACKS ONE COLUMN OF INFORMATION
C     PREPARED BY THE SLAR GEOMETRY PROGRAM.
C     MULTICATEGOREIS FOR ONE OF THE TWO
C     COLUMNS BEING PROCESSED ARE PUT ON
C     TEMPORARY FILES.
C
C     IFILE IS THE PRESENT FIRST COLUMN
C     ALL OTHER DATA IS PASSED THRU COMMON STATEMENTS
C
C
      SUBROUTINE UNPAC2(IFILE)
C
      IMPLICIT INTEGER (O)
      COMMON /IO/ IN,OUT,INTAP,OUTTAP,OUTEM(2)
      COMMON /DATA/ ICAT(2,320),IHIGH(2,320),ANGLES(3,320)
      COMMON /NAMES/ NCELAC
C
C
C     REWIND THE TEMPORARY FILE WHERE THE MULTICATEGORIES
C     WILL BE PLACED.
C
      ICURF=OUTEM(IFILE)
      REWIND ICURF
C
C     UNPACK THE GIVEN ARRAY
C
```

```
0719          DO 10 I1=1,NCELAC
0720    C
0721    C     READ IN THE NEXT WORD FROM THE INPUT FILE
0722    C     IF IT IS ZERO SKIP THE UNPACKING
0723    C
0724          READ(INTAP) IWORD
0725          IF(IWORD .EQ. 0) GO TO 20
0726    C
0727    C     STORE THE WORD WITH THE CATEGORY AND THE HEIGHT IN
0728    C     THE CATEGORY ARRAY--NO PROCESSING IS DONE ON THIS NUMBER
0729    C
0730          ICAT(IFILE,I1)=IWORD
0731    C
0732    C
0733    C     UNPACK THE HEIGHT FROM THE NORMAL WORD
0734    C
0735          FLD(24,12,IHIGH(IFILE,I1))=FLD(18,12,IWORD)
0736    C
0737    C     IF THE GIVEN WORD IS NEGATIVE, MULTICATEGORIES
0738    C     ARE INDICATED--IN THIS CASE READ IN THE CATEGORY WORDS
0739    C     AND STORE THEM ON A TEMPORARY FILE, WHICH WAS SPECIFIED.
0740    C     A POSITIVE WORD AFTER THE NEGATIVE WORD INDICATES
0741    C     THE LAST CATEGORY WORD FOR THIS CELL.
0742    C
0743          IF(IWORD .GE. 0) GO TO 10
0744    C
0745    40    READ(INTAP) IWORD2
0746          WRITE(ICURF) IWORD2
0747          IF(IWORD2 .LT. 0) GO TO 40
0748          GO TO 10
0749    C
0750    C     IF THE CELL HAS NO DATA SET THE HEIGHT OF THE CELL TO ZERO
0751    C
0752    20    IHIGH(IFILE,I1)=0
0753          ICAT(IFILE,I1)=0
0754    C
0755    C
0756    10    CONTINUE
0757    C
0758    C
0759    C     DONE WITH THE COLUMN
0760    C
0761    C
0762          REWIND ICURF
0763    C
0764          RETURN
0765    C
0766          END
0767    C     SUBROUTINE ANGLEAC(IFILE)
0768    C
0769    C
0770    C     THIS SUBROUTINE FINDS THE SLOPE BETWEEN A
0771    C     GIVEN POINT AND THE POINT BELOW IT IN A
0772    C     COLUMN.  IF THAT POINT IS ZERO, THE GIVEN POINT
0773    C     IS SET TO ZERO.
0774    C
0775    C     IFILE IS THE CODE FOR THE PRESENT FIRST COLUMN
0776    C     ALL OTHER DATA IS PASSED THRU LABELED COMMON
0777    C
0778    C
```

```
      SUBROUTINE ANGLE1(IFILE)
C
C
      IMPLICIT INTEGER (O)
      COMMON /IO/ IN,OUT,INTAP,OUTTAP,OUTEM(2)
      COMMON /DATA/ ICAT(2,320),IHIGH(2,320),ANGLES(3,320)
      COMMON /NAMES/ NCELAC,ALGSIZ,PTSIZ,CELSIZ,NEARNG,ALTSQ
C
C
C     FIND THE FIRST NON-ZERO CELL
C
      ITEMP=1
C
   15 ANGLES(2,ITEMP)=-1013.7
      IF(ICAT(IFILE,ITEMP) .NE. 0) GO TO 25
      ITEMP=ITEMP + 1
      IF(ITEMP .LT. NCELAC-1) GOTO 15
C
      WRITE(OUT,16)
   16 FORMAT(/,' ANGLEAC FOUND A ROW OF ZEROES',//)
      ANGLES(2,NCELAC-1)=-1013.7
      ANGLES(2,NCELAC)=-1013.7
      RETURN
C
C     NOW PROCESS THE ARRAY
C
   25 DO 10 I1=ITEMP+1,NCELAC
C
C     IF THE NEXT CELL HAS NO HEIGHT(NO DATA)
C     SKIP FURTHER PROCESSING OF THAT CELL
C
      IF(ICAT(IFILE,I1) .NE. 0) GO TO 20
C
   19 ANGLES(2,I1)=-1013.7
      GO TO 10
C
C     IF THE GIVEN POINT HAS DATA, CHECK THAT THE POINT BELOW
C     HAS DATA.  IF IT DOES, FIND THE LOCAL ACROSS TRACK SLOPE.
C
   20 IF(ICAT(IFILE,I1-1).EQ.0) GOTO 19
C
      HDELTA=IHIGH(IFILE,I1)-IHIGH(IFILE,I1-1)
      ANGLES(2,I1)=HDELTA/CELSIZ
C
C
   10 CONTINUE
C
C
      RETURN
C
      END
C     SUBROUTINE ANGLEAL(NCOL1,NCOL2,I1)
C
C
C     THIS SUBROUTINE FINDS THE ALONG TRACK SLOPE BETWEEN
C     TWO ADJACENT POINTS IN ROWS "NCOL1" AND "NCOL2".
C     IF EITHER OF THE TWO POINTS UNDER CONSIDERATION HAVE
C     NO DATA, THEN THE ALONG TRACK SLOPE IS SET TO ZERO.
C
C     NCOL1 IS THE PRESENT FIRST COLUMN
```

```
C     NCOL2 IS THE PRESENT SECOND COLUMN
C     I1 IS THE PRESENT CELL OF COLUMN OF INTEREST
C     ALL OTHER DATA IS PASSED THRU LABELED COMMON
C
C
C
C
      SUBROUTINE ANGLE2(NCOL1,NCOL2,I1)
C
C
      IMPLICIT INTEGER (O)
      COMMON /IO/ IN,OUT,INTAP,OUTTAP,OUTE[illegible]
      COMMON /DATA/ ICAT(2,320),IHIGH(2,320),ANGLES(3,320)
      COMMON /NAMES/ NCELAC,ALGSIZ
C
C
C     IF THERE IS NO DATA SKIP PROCESSING OF THIS CELL
C
      IF(ICAT(NCOL1,I1) .EQ. 0) GOTO 40
C
C     IF THE CELL TO THE RIGHT OF THE GIVEN CELL IS NOT ZERO, PROCES
C     THIS IS THE NORMAL OCCURENCE
C
      IF(ICAT(NCOL2,I1) .EQ. 0) GO TO 20
C
C     FIRST FIND THE RELATIVE HEIGHT BETWEEN THE CELLS
C
      HDELTA=IHIGH(NCOL2,I1) - IHIGH(NCOL1,I1)
C
C     NOW FIND THE LOCAL ALONG-TRACK ANGLE
C
10    ANGLES(3,I1)=HDELTA/ALGSIZ
      RETURN
C
C     IF THE CELL TO THE RIGHT OF THE GIVEN CELL WAS ZERO, CHECK
C     CHOICE NUMBER 2--IF IT IS ZERO GO TO LAST CHOICE
C
20    IF(I1 .EQ. 1) GO TO 30
      IF(ICAT(NCOL2,I1-1) .EQ. 0) GO TO 30
      HDELTA=IHIGH(NCOL2,I1-1) - IHIGH(NCOL1,I1)
      GO TO 10
C
C     CHECK TO SEE IF THE THIRD AND LAST CHOICE IS ZERO
C     IF NOT PROCESS IT
C
30    IF(I1.GE.NCELAC) GOTO 40
      IF(ICAT(NCOL2,I1+1) .EQ. 0) GO TO 40
      HDELTA=IHIGH(NCOL2,I1+1) - IHIGH(NCOL1,I1)
      GO TO 10
C
C     IF THE LAST CHOICE WAS ZERO, WE CANNOT FIND THE DESIRED ANGLE
C     SO STOP PROCESSING OF THIS CELL
C
40    ANGLES(3,I1)=-1013.7
      RETURN
C
C
      END
C     SUBROUTINE ANGLEOUT(NCOL1,ICOL)
C
```

```
C
C     THIS SUBROUTINE COMPUTES THE OUTPUT ANGLES OF
C     THE PROGRAM SLOPE.  THE INPUT IS THE THREE
C     ANGLES; THE RADAR INCIDENCE ANGLE, THE LOCAL
C     SLOPE OF THE GIVEN CELL IN THE CROSS-TRACK,
C     AND THE LOCAL SLOPE OF THE GIVEN CELL IN THE
C     ALONG TRACK.
C
C     THE OUTPUT OF THIS ROUTINE IS THE EQUIVALENT
C     LOCAL ANGLE OF INCIDENCE WITH RESPECT TO THE RADAR,
C     AND THE RELATIVE SIZE OF THE CELL DUE TO TILTING
C     OF THE CELL.
C
C
C     NCOL1 IS THE PRESENT FIRST COLUMN
C     THETAL IS THE RETURNED LOCAL ANGLE
C     THETA IS THE RETURNED RADAR ANGLE
C     SIZE IS THE RETURNED LOCAL RELATIVE SIZE
C     ICOL IS THE CELL IN THE GIVEN COLUMN OF INTEREST
C     ALL OTHER DATA IS PASSED THRU LABLED COMMON
C
C
      SUBROUTINE ANGOUT(NCOL1,ICOL)
C
C
      IMPLICIT INTEGER (O)
      COMMON /IO/ IN,OUT,INTAP,OUTTAP,OUTEN(2)
      COMMON /DATA/ ICAT(2,320),IHIGH(2,320),ANGLES(3,320)
      COMMON /NAMES/ NCELAC,ALGSIZ,PTSIZ,CELSIZ,NEARNG,ALTSQ,
     R  SIZE,THETA,THETAL,ALTMSL
C
      D   A SMALL/1.0E-03/
C
C
C     DEFINE  ANGULAR FUNCTION
C
      ARCOS(X)=1.5708 - ATAN(X/SQRT(1.-X**2))
C
C     IF THE GIVEN CELL IS ZERO SKIP PROCESSING
C
      IF(ICAT(NCOL1,ICOL) .EQ. 0) GO TO 10
C
C     GET THE THREE GIVEN ANGLES IN A MORE WORKABLE FORM
C
      TANTHE=ANGLES(1,ICOL)
      TANCRS=ANGLES(2,ICOL)
      TANALG=ANGLES(3,ICOL)
C
C     FIND THE ARC TANGENT FOR LATER PROCESSING
C
      THETA=ATAN(TANTHE)
      ACROS=ATAN(TANCRS)
      ALONG=ATAN(TANALG)
C
C     CHECK TO SEE IF THE LOCAL ANGLES ARE ZERO I.E. NO SLOPE
C
      IF(ABS(ACROS).LT.SMALL .AND. ABS(ALONG).LT.SMALL) GO TO 1
C
C     FIND THE COSINE OF THE EQUIVALENT LOCAL ANGLE FOR THE CELL
C
```

```
      THETAL=(TANCRS * SIN(THETA) + COS(THETA))/
     1  SQRT(1. + TANCRS**2 + TANALG**2)
C
C     IF THE NUMBER IS GREATER THAN 1.0 SOMEWHERE ALONG
C     THE LINE SOMETHING IS WRONG
C
      IF(ABS(THETAL) .GT. 1.0) GO TO 20
C
C     IF THERE WAS NO PROBLEMS FIND THE ARCCOSINE OF THE ANGLE
C
      THETAL=ARCOS(THETAL)
      GO TO 17
C
C     IF THE LOCAL ANGLES ARE ZERO THE CELL IS FLAT
C
 15   THETAL=THETA
C
C     NOW FIND THE EQUIVALENT RADAR CELL SIZE
C
 17   HEIGHT=IHIGH(NCOL1,ICOL)
      SIZE=(ALTMSL - HEIGHT)/
     1 (COS(THETA) * COS(ALONG) * SIN(THETA-ACROS))
      SIZE=SIZE/10.**3
      IF(SIZE .LT. 0.) SIZE=0.0
C
C     THE NORMAL PROCESSING OF THE CELL IS COMPLETED
C
 10   RETURN
C
C
C     ERROR MESSAGES
C
C     IF THE VALUE OF THE COSINE OF THE ANGLE WAS GREATER
C     THAN ONE-AN ERROR EXISTS.  SO WRITE THE CELLS LOCATION
C     AND SET AN ERROR FLAG THEN STOP PROCESSING
C
 20   NCOL2=1
      IF(NCOL1.EQ.1) NCOL2=2
C
      WRITE(OUT,30) ANGLES(1,ICOL),ANGLES(2,ICOL),ANGLES(3,ICOL)
     1  ,THETAL
 30   FORMAT(' FOR THE ANGLES--THETA-',F7.3,', ACROSS-',F7.3,
     '  ', ALONG-',F7.3,/,' THE VALUE OF THE LOCAL ANGLE IS',
     2  F7.3,',  A NUMBER .GT. 1.0 IS UNREALISTIC.')
C
      WRITE(OUT,40) IHIGH(NCOL1,ICOL),IHIGH(NCOL1,ICOL+1),
     1  IHIGH(NCOL2,ICOL+1),IHIGH(NCOL2,ICOL)
 40   FORMAT(' THE HEIGHT OF THE CELL IS',I5,' THE CELLS SURROUND
     1  ' IT ARE--',/,'      DIRECTLY ABOVE-',I5,/,
     2  '      ABOVE & TO THE RIGHT-',I5,/,
     3  '      DIRECTLY TO THE RIGHT-',I5)
      IF(ICOL .EQ. 1) GO TO 50
C
      WRITE(OUT,60) IHIGH(NCOL2,ICOL-1),IHIGH(NCOL1,ICOL-1)
 60   FORMAT('      TO THE RIGHT & DOWN-',I5,/,
     1  '      DIRECTLY BELOW-',I5)
C
 50   STOP
C
      END
```

```
$        EXECUTE
$        LIMITS   25,15K
$        FILE     09,A2RR
$        FILE     10,A3SR,500L
$        FILE     11
$        FILE     12
$        IF       ABORT,ENDJOB
$        OPTION   FORTRAN
$        FORTRAN
C        SLAR SHADOW & LAYOVER
C
C     SLAR SHADOW MAINLINE
C
         IMPLICIT INTEGER (O)
         COMMON /IO/ IN,OUT,INTAP,OUTTAP,ITEMFL,NCELAC
         COMMON /PARAM1/ GEOM(15)
         COMMON /DATA/ OVRLAY(7,320),IHIGH(320),IBIN(320),
       1   ICAT(320),ANGLE(3,320),RANGE(320)
         DATA IN,OUT,INTAP,OUTTAP,ITEMFL,IMAP,ITEM /5,6,9,10,11,.
C
C
C     READ IN FILE LOCATION -- DEFAULT IS ONE
C
         READ(IN,10,END=30) OUTFIL
  10     FORMAT(I2)
C
C     NOW POSITION THE TAPE TO THE CORRECT FILE
C
         IF(OUTFIL.LE.1) GOTO 30
         CALL POST(OUTTAP,1,OUTFIL,1,N)
         IF(N.NE.0) STOP
C
C     READ IN PARAMETERS AND ADJUST FOR THE SLOPE ROUTINE
C
  30     READ(INTAP) GEOM
         NHAFAL=GEOM(11)/2
         GEOM(11)=GEOM(11) - 1
         GEOM(12)=GEOM(12) - 1
         NCELAC=GEOM(12)
         NCELAL=GEOM(11)
C
C     WRITE OUT THE PARAMETERS
C
         WRITE(OUT,35) GEOM
         WRITE(OUTTAP) GEOM
  35     FORMAT(F20.5)
C
C     SET THE FLAGS & POINTERS
C
         CELSIZ=GEOM(2)
         IONE=1
         IZERO=0
C
C
C
C     NOW DETERMINE THE RANGE SQUARED TO EACH CELL.
C
         RANGE(1)=GEOM(14)+CELSIZ/2.0
C
         DO 36 I=2,NCELAC
```

```
      RANGE(I)=RANGE(I-1)+CELSIZ
   36 CONTINUE
C
      DO 37 I=1,NCELAC
      RANGE(I)=RANGE(I)**2
   37 CONTINUE
C
   31 ALTSQ=GEOM(10)
C
      DO 38 I=1,NCELAC
      IBIN(I)=SQRT(RANGE(I) - ALTSQ)
      RANGE(I)=RANGE(I)/1000000.0
   38 CONTINUE
C
C
C
C
C     NOW DO THE WORK
C
C
C
      DO 40 I1=1,NCELAL
C
C
C     UNPACK ONE SCAN LINE, PUTTING THE MULTICATEGORIES
C     ON A TEMPFILE
C
      CALL UNPAC3
C
C     COMPUTE THE SHADOW AND LAYOVER
C
      CALL LAYOVR
C
C     NOW WRITE OUT THE CELLS IN THE PROPER ORDER
C
   75 DO 50 I2=1,NCELAC
C
C     NO DATA IN THIS CELL
C
      IF(OVRLAY(7,I2) .EQ. -1) GO TO 200
C
C     CHECK FOR SHADOW
C
      IF(OVRLAY(7,I2) .EQ. 0) GO TO 100
C
C     NORMAL DATA IS PRESENT FOR PROCESSING
C     SO SET THE THIRD BIT OF THE FIRST WORD
      INUM=OVRLAY(1,I2)
      ISAVE=INUM
      FLD(2,1,ICAT(INUM))=FLD(35,1,IONE)
C
C     NOW PROCESS THIS CELL
      DO 60 I3=1,OVRLAY(7,I2)
C
C     FIND OUT WHICH CELL TO WORK ON
      INUM=OVRLAY(I3,I2)
C
C     WRITE OUT THE NORMAL INFORMATION PRESENT FOR EACH CELL
      WRITE(OUTTAP) ICAT(INUM)
      WRITE(OUTTAP) (ANGLE(I,INUM),I=1,3),RANGE(I2)
```

```
C
C     CHECK FOR MULTICATEGORIES,IF NONE THEN GO TO THE NEXT CELL.
        IF(ICAT(INUM) .GE. 0) GO TO 60
C
C     THIS CELL CONTAINS MULTICATEGORIES, SO NEED TO SEARCH THE
C     TEMP FILE FOR THE PROPER SET OF NUMBERS.
C     THE CELL WILL BE IDENTIFIED BY THE INPUT CELL NUMBER.
C
 65     IFLAG=0
 70     READ(ITEMFL,END=90) IWORD
        IF(IWORD .NE. INUM) GO TO 70
 80     READ(ITEMFL) IWORD
        WRITE(OUTTAP) IWORD
        IF(IWORD .LT. 0) GO TO 80
        GO TO 60
 90     IF(IFLAG .GT. 1) STOP
        IFLAG=IFLAG + 1
        REWIND ITEMFL
        GO TO 70
C
C
 60     CONTINUE
C
C
        FLD(2,1,ICAT(ISAVE))=0
        GO TO 50
C
C
C
C     SHADOW
 100    WRITE(OUTTAP) IONE
        GO TO 50
C
C     NO DATA
 200    WRITE(OUTTAP) IZERO
C
C
C
 50     CONTINUE
C
C     DONE WITH ONE SCAN LINE
C
C
C     HALF WAY THRU THE MATRIX WRITE OUT SOME RESULTS
        IF(NHAFAL .NE. I1) GO TO 52
        WRITE(OUT,45) (IHIGH(I5),I5=1,NCELAC)
        WRITE(OUT,45) (IBIN(I5),I5=1,NCELAC)
        WRITE(OUT,46) ((OVRLAY(I6,I7),I6=1,7),I7=1,NCELAC)
        WRITE(OUT,47) (ICAT(I5),I5=1,NCELAC)
 45     FORMAT(1X,20I6)
 46     FORMAT(7I5)
 47     FORMAT(10(1X,O12))
C
C     CHECK FOR MAPPING
 52     IF(IMAP .NE. 1) GO TO 40
        DO 51 I=1,NCELAC
        IF(OVRLAY(7,I) .EQ. -1) OVRLAY(7,I)=9
 51     CONTINUE
        WRITE(ITEM) (OVRLAY(7,J),J=1,NCELAC)
C
```

```
C
 40     CONTINUE
C
C
C
C     DONE WITH THE MATRIX
C
C
C
C     IF A MAP WAS DESIRED PRINT IT
        IF(IMAP .NE. 1) GO TO 5000
        CALL MAPPER(ITEM,OUT,NCELAC,NCELAL)
C
C
C
 5000   ENDFILE OUTTAP
        WRITE(OUTTAP) OUTTAP
        ENDFILE OUTTAP
        REWIND OUTTAP
        STOP
C
        END
C       SUBROUTINE UNPACK3
C
C     THIS ROUTINE UNPACKS ONE COLUMN OF INFORMATION
C     PREPARED BY THE SLAR SLOPE PROGRAM.
C     MULTICATEGORIES FOR THE COLUMN-IF PRESENT ARE
C     PLACED ON A TEMPORARY FILE.
C
C     ALL DATA IS PASSED THRU COMMON STATEMENTS
C
C
        SUBROUTINE UNPAC3
C
        IMPLICIT INTEGER (O)
        COMMON /IO/ IN,OUT,INTAP,OUTTAP,OUTEM,NCELAC
        COMMON /DATA/ OVRLAY(7,320),IHIGH(320),IBIN(320),
     1    ICAT(320),ANGLE(3,320),RANGE(320)
C
        NUMPTS=0
C
C     REWIND THE TEMPORARY FILE WHERE THE MULTICATEGORIES
C     WILL BE PLACED.
C
         REWIND OUTEM
C
C     UNPACK THE GIVEN ARRAY
C
        DO 10 I1=1,NCELAC
C
C     READ IN THE NEXT WORD FROM THE INPUT FILE
C     IF IT IS ZERO SKIP THE UNPACKING
C
        READ(INTAP) IWORD
        IF(IWORD .EQ. 0) GO TO 20
C
C     STORE THE WORD WITH THE CATEGORY AND THE HEIGHT IN
C     THE CATEGORY ARRAY--NO PROCESSING IS DONE ON THIS NUMBER
C
        ICAT(I1)=IWORD
```

```
C
C     UNPACK THE HEIGHT FROM THE NORMAL WORD
C
 25     FLD(24,12,IHIGH(I1))=FLD(19,12,IWORD)
C
C     IF THE GIVEN WORD IS NEGATIVE MULTICATEGORIES
C     ARE INDICATED--IN THIS CASE READ IN THE CATEGORY WORDS
C     AND STORE THEM ON A TEMPORARY FILE WHICH WAS SPECIFIED.
C     A POSITIVE WORD AFTER THE NEGATIVE WORD INDICATES
C     THE LAST CATEGORY WORD FOR THIS CELL.
C     ALSO WRITE OUT A RECOGNITION WORD FOR EACH CELL SO THESE
C     NUMBERS CAN BE FOUND AGAIN.
C
 35     IF(IWORD .GT. 0) GO TO 70
        WRITE(OUTEM) I1
 40     READ(INTAP) IWORD2
        WRITE(OUTEM) IWORD2
        IF(IWORD2 .LT. 0) GO TO 40
C
C     READ IN THE ASSOCIATED ANGLE TERMS
C
 70     READ(INTAP) (ANGLE(I2,I1),I2=1,3)
        GO TO 10
C
C     IF THE CELL HAS NO DATA SET THE HEIGHT OF THE CELL TO ZERO
C
 20     IHIGH(I1)=0
        ICAT(I1)=0
C
C
 10     CONTINUE
C
C
C     DONE WITH THE COLUMN
C
 90      REWIND OUTEM
C
        RETURN
C
        END
C       SUBROUTINE LAYOVR
C
C
C
C     THIS SUBROUTINE COMPUTES THE SHADOW AND LAYOVER OF ONE
C     SCAN LINE.  IT DOES THIS BY COMPUTING THE RANGE TO EACH
C     RESOLUTION CELL AND PLACING THAT CELL IN THE PROPER RANGE
C     BIN.  THE PROGRAM ALLOWS UP TO SIX CELLS TO COINCIDE
C     IN THE SAME RANGE BIN.  IF MORE THAN SIX TRY TO ENTER THE
C     SAME BIN, AN ERROR WARNING IS PRINTED ON THE GIVEN OUTPUT
C     DEVICE AND THE CELL IS DISCARDED.
C
C     SHADOW IS HANDLED BY THE FORMING OF AN EQUATION OF THE FORM
C               Y=ZX + B
C     THIS METHOD IS THE SAME ONE THAT HAS BEEN USED BEFORE IN
C     PREVIOUS SIMULATION WORK.
C
C     IN THE LAYOVER COMPUTATION, SOME ATTEMPT IS MADE TO ACCOUNT
C     FOR THE EFFECTS OF COMPRESSING A THREE DIMESIONAL SURFACE
C     INTO A TWO DIMENSIONAL SURFACE WITH A PROJECTED HEIGHT.
```

```
C     THIS IS DONE BY FIRST COMPUTING THE RANGE TO A CELL AND
C     THE PROPER BIN TO LAY IT INTO.  THEN THE RANGE IS CALCULATED
C     TO THE CELL AGAIN BUT THIS TIME USING THE HEIGHT OF THE
C     PREVIOUS CELL AND THE RANGE BIN IT WOULD FALL INTO.
C     ALL THE BINS BETWEEN THESE TWO NUMBERS ARE FILLED
C     IN.
C
C
C
      SUBROUTINE LAYOVR
      IMPLICIT INTEGER (O)
      COMMON /IO/ IN,OUT,INTAP,OUTTAP,ITEMFL,NCELAC
      COMMON /PARAM1/ GEOM(15)
      COMMON /DATA/ OVRLAY(7,320),IHIGH(320),IBIN(320),
     1  ICAT(320),ANGLE(3,320),RANGE(320)
C
C
C     DEFINE CONSTANTS
C
      ALT=GEOM(9)
      NRANGE=GEOM(14)
      Z=-99999
      ORANGE=1
C
C     INITIALIZE THE ARRAY TO ALL SHADOW, IF NOTHING FALLS
C     INTO A PARTICULAR BIN, THEN A SHADOW IS ASSUMED.
C
      DO 10 I2=1,NCELAC
      OVRLAY(7,I2)=0
   10 CONTINUE
C
C
C     NOW COMPUTE THE LAYOVER AND SHADOW FOR EACH CELL.
C
C
C
C
      DO 100 I3=1,NCELAC
C
C
C     CHECK FOR NODATA
C
      IF(ICAT(I3).NE.0) GOTO 20
C
C     HANDLE NODATA CASE - ASSUME HEIGHT = 0.
C
      IF(OVRLAY(7,I3).EQ.0) OVRLAY(7,I3)=-1
      IY=Z*IBIN(I3)+ALT
      IF(IY.GE.0) GOTO 70
      ORANGE=I3+1
      GOTO 100
C
C     COME HERE IF THERE IS DATA.
C
C
C     DETERMINE IF CELL IS IN SHADOW
C
   20 IY=Z*IBIN(I3)+ALT
      IF(IY.GE.IHIGH(I3)) GOTO 80
C
```

```
C     COME HERE IF THERE IS NO SHADOW.
C
C     RECOMPUTE THE NEW SLOPE Z AND FIND WHICH IBIN THIS
C     CELL GETS LAID OVER INTO.
C
        Z=(FLOAT(IHIGH(I3))-ALT)/IBIN(I3)
        R=SQRT(IBIN(I3)**2+(ALT-IHIGH(I3))**2)
        JR=(R-NRANGE)/GEOM(2)+1.0
C
C     NOW FILL IN THE BINS FROM THE IBIN WHERE THE CURRENT CELL
C     LAID OVER INTO BACK TO THE IBIN BEFORE WHERE THE PREVIOUS CE
C     LAID OVER INTO.  BUT FIRST CHECK FOR CELL OUT-OF-RANGE
C     OR NO FILL-IN NEEDED.
C
C
        IF(JR.GT.NCELAC) GOTO 100
        IF(JR.LE.1) JR=1
        IF(JR.GE.ORANGE) ORANGE=JR
C
C     PERFORM THE FILL-IN
C
        DO 30 I4=JR,ORANGE
C
        IF(OVRLAY(7,I4).GE.6) GOTO 90
        IF(OVRLAY(7,I4).LT.0) OVRLAY(7,I4)=0
        OVRLAY(7,I4)=OVRLAY(7,I4)+1
        OVRLAY(OVRLAY(7,I4),I4)=I3
 30     CONTINUE
C
C     UPDATE THE PREVIOUS CELL POINTER
C
        ORANGE=JR-1
        GOTO 100
C
C     COME HERE IF CELL IS IN SHADOW TO COMPUTE NEW ORANGE.
C
 70     IF(OVRLAY(7,I3).LT.0) OVRLAY(7,I3)=0
C
 80     R=SQRT(IBIN(I3)**2+(ALT-IY)**2)
        ORANGE=(R-NRANGE)/GEOM(2)+0.5
        GOTO 100
C
C     COME HERE IF MORE THAN SIX CELLS ARE PUT IN ONE BIN.
C
 90     WRITE(OUT,99) I4
 99     FORMAT(//,' *** WARNING - BIN ',I4,' EXCEEDS SIX CELLS',//
C
C
C
 100    CONTINUE
C
C     DONE WITH THE LINE
C
        RETURN
C
C
C
C
        END
C       SUBROUTINE MAPPER(IN,OT,NPTS,NROWS)
```

```
C
C     THIS ROUTINE PRINTS OUT AN IMAGE.
C
C
      SUBROUTINE MAPPER(IN,OT,NPTS,NROWS)
      IMPLICIT INTEGER(A-Y)
      COMMON /DATA/ LINE(500)
C
      DO 100 START=1,NPTS,130
      END=START + 129
      IF(END .GT. NPTS) END=NPTS
      REWIND IN
      WRITE(OT,50)
   50 FORMAT(1H1)
      DO 90 ROW=1,NROWS
      READ(IN) (LINE(I),I=1,NPTS)
      WRITE(OT,10) (LINE(K),K=START,END)
   10 FORMAT(1X,130I1)
   90 CONTINUE
  100 CONTINUE
C
      RETURN
      END
$     EXECUTE
$     LIMITS   25,17K
$     FILE     09,A3RR
$     TAPE     10,A4CD,,S1014,,WORK14,OUT
$     FILE     11
$     FILE     13,,25L
1
$     IF       ABORT,ENDJOB
$     OPTION   FORTRAN
$     FORTRAN
C                NEWGT
C
C                ****   NOTE   ****
C
C  THIS IS THE NEW GREYTONE ROUTINE WRITTEN BY EDK FOR THE SLAR P
C  TO CORRECT PROBLEMS IN THE GREYTONE OUTPUT
C  THIS ROUTINE WAS USED TO PRODUCE THE OUTPUT IMAGES OF THE PICK
C  SITE FOR 116 AND 300 HEADINGS FOR REPORT OF JUNE 1977.
C
C
C
      IMPLICIT INTEGER (A-Q)
      REAL ALOG10
      DIMENSION GT(300)
      COMMON SIGO(28,91),RGEOM(15)
      DATA PRMFL,INTAP,OTTAP/11,9,10/
C
      ISTRT = ISEED(DUMY)
C
      READ(5,10) INFIL,OTFIL
   10 FORMAT(2I5)
      IF(INFIL .GT. 1) CALL POST(INTAP,0,INFIL,1,ERROR)
      IF(ERROR .NE. 0) GOTO 800
      IF(OTFIL .GT. 1)  CALL POST(OTTAP,1,OTFIL,1,ERROR)
      IF(ERROR .NE. 0) GOTO 800
C
      READ(5,15) ROPDEN,REFGT,REFR,REFAREA,REFG,REFSIG,SAMPLE
```

```
15      FORMAT(7F10.4)
C
        S1 = 255./ROPDEN
        RCONST = REFR**2 * 1000. / (REFAREA*REFG**2)
        RCONST = S1*(ALOG10(RCONST) - REFSIG) + REFGT
        SSAMP = SQRT (SAMPLE)
C
        READ(PRMFL,20) SIGO
20      FORMAT(12F6.3)
C
        READ(INTAP) RGEOM
C
        NALONG = RGEOM(11)
        NACROSS = RGEOM(12)
        WRITE(6,23) NACROSS,NALONG,RCONST
23      FORMAT('    POINTS PER LINE',I5,' LINES=',I5,' CONSTANT=
       $ ,F12.4)
        READ(INTAP) IWORD
C
        DO 200 LINE = 1,NALONG
        DO 180 PT = 1,NACROSS
        GT(PT)=0
        TOTPWR = 0.
C
        IF(IWORD .EQ. 0 .OR. IWORD .EQ. 1) GOTO 170
C
90      CALL POWER(IWORD,ZPWR,INTAP)
        TOTPWR = TOTPWR + ZPWR
        READ(INTAP) IWORD
        IF(IWORD .EQ. 0 .OR. IWORD .EQ. 1) GOTO 101
        IF(FLD(2,1,IWORD) .EQ. 0) GOTO 90
C
101     ZFADE = 1. + RMS(ISTRT)/SSAMP
        IF(ZFADE .LT. 1.E-5) ZFADE = 1.E-5
        ZFADE = S1*ALOG10(ZFADE)
C
        GT(PT) = S1*ALOG10(TOTPWR) + RCONST + ZFADE
        IF(GT(PT) .LT. 0) GT(PT) = 0
        IF(GT(PT) .GT. 255) GT(PT)= 255
        GOTO 180
170     READ(INTAP) IWORD
180     CONTINUE
        WRITE(10) (GT(J),J=1,NACROSS)
        WRITE(6,626) (GT(J)/26,J=1,240,2)
626     FORMAT(1X,12I1)
200     CONTINUE
        STOP
800     PRINT, 'ERROR WITH POST'
        STOP
        END
        SUBROUTINE POWER(IWORD,ZPWR,INTAP)
        COMMON SIGO(28,91)
C
        ICNT = 0
        ZPWR = 0.
        READ(INTAP) THINC,THLOC,SIZE,R2
C
50      ICAT = FLD(30,6,IWORD)
        CALL RTPWR(ICAT,THINC,THLOC,SIZE,R2,ZOUT)
        ZPWR = ZPWR + ZOUT
```

```
      IF(IWORD .GT. 0) GOTO 100
      NUMPT = FLD(6,6,IWORD)
      ZPWR = ZPWR + (NUMPT-1)*ZOUT
      ICNT = ICNT + NUMPT
      READ(INTAP) IWORD
      GOTO 50
C
  100 IF(ICNT .GT. 0) ZPWR = ZPWR/ICNT
      RETURN
      END
      SUBROUTINE PTPWR(ICAT,THINC,THLOC,SIZE,R2,PWR)
      COMMON SIGO(28,91)
C
      IF(ICAT .GT. 28) GOTO 800
      ITHETA = 57.295*THLOC + 1.
      IF(ITHETA .LT. 1 .OR. ITHETA .GT. 91) GOTO 800
C
      GAIN2 = 1./COS(THINC)**4
      SIG = SIGO(ICAT,ITHETA)
C
      PWR = 10.**SIG * SIZE  * GAIN2 / (R2**2)
      RETURN
C
  800 CONTINUE
      RETURN
      END
$     EXECUTE
$     LIMITS   20,14K,,5K
$     PRMFL    11,R,S,4944MARTIN/NEWSIG
  1,1,
  2.,90.,.4614.,.57435.,.4.,-1.33,9.,
$     TAPE     09,A400,,51014,,WORK14
$     TAPE2    10,A500,,60711,,HELPR1,OUT,DEN5
$     ENDJOB
```

APPENDIX J

DIGITAL PPI MODEL FOR RADAR IMAGE SIMULATION AND RESULTS

The following technical report (TR 319-19) prepared by the Center for Research, Inc., University of Kansas, is included in this volume to provide details in support of the technical discussions of Volume I.

THE UNIVERSITY OF KANSAS CENTER FOR RESEARCH, INC.

2291 Irving Hill Drive Campus West
Lawrence, Kansas 66045

Telephone: 913-8644832

DIGITAL PPI MODEL FOR RADAR IMAGE SIMULATION AND RESULTS

Remote Sensing Laboratory

RSL Technical Report 319-19

E. Komp

V. H. Kaupp

J. C. Holtzman

August, 1977

Supported by:

U. S. Army Engineer Topographic Laboratories

Fort Belvoir, Virginia 22060

CONTRACT DAAG 53-76-C-0154

ABSTRACT

The implementation of a digital closed system radar simulation model developed to simulate PPI (Plan Position Indicator) radar imagery is described. Basically, the Point Scattering Method was adapted to conform to the polar format needed to produce PPI images. A theoretical development of the model and a complete description of the computer software implementation are presented.

1.0 INTRODUCTION

A general theoretical model for the simulation of radar imagery has been developed. Two separate applications of this model have been implemented through computer software routines at the Remote Sensing Laboratories (RSL). One application simulates the output of Plan Position Indicator (PPI) radar systems and the other produces simulated Side-Looking Airborne Radar (SLAR) imagery.

This report describes the details of the computer implementation for simulation of PPI imagery. Each of the three major phases in the simulation sequence are presented separately. Section 3.1 presents the motivation and technique for converting the rectangular input data matrix into polar coordinates. Section 3.2 describes the calculation of greytone for each resolution cell. The handling of geometry and propagation phenomena such as shadowing and layover are also discussed. Section 3.3 describes the conversion of the polar greytone matrix into appropriate format for output display. Section 4 contains a set of computer programs for implementation of the PPI simulation model and another set of programs demonstrating the application of the model to a terminal guidance problem.

Before describing the computer software implementation a brief presentation of the theoretical model on which it is based will be given.

2.0 RADAR SIMULATION THEORY

The theoretical model for the operation of real radar systems on which this PPI implementation is based, is a closed system model which accounts for all those phenomena which affect the returned power in the operation of a real radar.

For simulation purposes the five major factors that affect the final output of a radar image are:

(1) Radar system parameters (frequency, polarization, beamwidth, resolution, etc.);

(2) Flight parameters (platform location, look direction, etc.);

(3) Local geometry of the ground spot illuminated;

(4) Dielectric properties of the ground scene;

(5) Conversion of returned power to an image medium.

These factors are treated by the radar simulation model[1]. The geometry and propagation phenomena are all properly handled. The dielectric properties of the target spot are summarized through the coefficient of backscatter (σ°), made available from a large data bank of empirical values. The final image medium is photographic film so the value of power calculated to exit the receiver for each ground spot is converted to an image density, or greytone value, for final display.

A full development and justification of the model can be found in an earlier report[1], and therefore will not be reproduced at this point. Instead, only a statement of the final equation less fading for the relative greytone value for a point in the output image will be quoted here.

$$G_{R_{c_2}} = G_{R_{a_1}} + \frac{255}{x}\left\{\gamma \log_{10}\left[\frac{P_{T_2}\sigma^\circ_{c_2}(\theta_{\ell_2})\Delta A_2 G_2^2(\theta_2)\lambda_2^2 R_1^4}{P_{T_1}\sigma^\circ_{a_1}(\theta_{\ell_1})\Delta A_1 G_1^2(\theta_1)\lambda_1^2 R_2^4}\right] + \log_{10}\left(\frac{k_2}{k_1}\right) + \gamma \log_{10}\left(\frac{M_2}{M_1}\right)\right\} \qquad (1)$$

---

[1] Holtzman, J. C., V. H. Kaupp, R. L. Martin, E. E. Komp, and V. S. Frost, "Radar Image Simulation Project: Development of a General Simulation Model and an Interactive Simulation Model, and Sample Results," TR 234-13, Remote Sensing Laboratory, The University of Kansas, February, 1976.

$G_{R_{c_2}}$ = The instantaneous greytone to be calculated for each discrete point for scatterers belonging to backscatter category 'c';

$G_{R_{a_1}}$ = The greytone value added to the value computed for each point to calibrate the range of brightness in an image according to a known reference;

γ = A property of the image medium (in this case, film, i.e. image medium transfer function);

P_{T_2} = The transmitter output power of the radar to be simulated;

P_{T_1} = The transmitter output power of the calibrator;

$\sigma^{\circ}_{c_1}(\theta_{\ell_2})$ = The scattering coefficient per unit area for each ground point; corresponding to the local angle of incidence θ_{ℓ_2};

$\sigma^{\circ}_{a_1}(\theta_{\ell_1})$ = The scattering coefficient per unit area for the reference backscatter category; corresponding to the local angle of incidence θ_{ℓ_1};

ΔA_2 = Area of the ground spot resolution cell illuminated by the radar to be simulated;

ΔA_1 = Area of the ground spot resolution cell illuminated by the calibrator;

G_2 = One-way gain of the antenna of the radar to be simulated (in direction of ΔA_2);

G_1 = One-way gain of the antenna of the calibrator (in direction of ΔA_1);

λ_2 = Wavelength of the electromagnetic energy transmitted by the radar to be simulated;

λ_1 = Wavelength of the electromagnetic energy transmitted by the calibrator;

R_2 = The distance from the antenna of the radar to be simulated to each ground resolution cell;

R_1 = The distance from the antenna of the calibration system to the reference ground spot;

k_2, k_1 = Constants which depend upon the exposure time and on the film processing and development;

M_2, M_1 = The receiver transfer functions of the radar to be simulated and the calibrator , respectively.

This equation provides the basic theoretical result on which the computer implementation of the PPI radar simulation, to be described in the following sections is based.

3.0 PPI SIMULATION PROCESS

In very general terms the PPI simulation package accepts as input a description of the target site and the system parameters of the radar to be modeled and produces a matrix of greytone values representing a radar image for output. There are three major steps in this process. First the information about the ground scene is transformed into a digital polar ground truth data base to correctly account for the geometric properties of the ground scene and radarsystem being modeled. Second, Equation (1) is applied to each cell in the internal polar ground truth matrix to compute a greytone value for each pixel(picture element) the scene. Finally, this matrix of greytone values is transformed from polar coordinates to a rectangular matrix of greytones and formatted for an appropriate display device for observation of the final output results. Each of these steps will be discussed separately in the following sections.

3.1 Creation of Polar Data Base

The PPI radar is a forward-scanning, imaging device operating naturally in polar coordinates. In typical operation the antenna rotates to image an area 45° either side of the line of flight and from almost directly under the radar platform (0° incident angle) to near grazing (>80° incident angle). Becausc of these properties the size and orientation of the resolution cells varies greatly over the area of interest. The orientation of the resolution cell is critical for computing the local slope and also areas of shadow in the output image. The size of the resolution cell affects the amount of power returned from the ground spot to the antenna because of the area encompassed and the determination of the category types to be included. A rectangular matrix format is not well-suited to model these changes in the resolution cell. A polar coordinate system with its center at the radar platform location provides a much closer model for the geometry of the operation of a real PPI radar system. Therefore, the first step in the PPI simulation process is to transform the original rectangular data base into a resolution cell matrix in polar coordinates.

This step in the simulation process requires two types of input:

(1) A ground truth data base in rectangular matrix form;

(2) Specification of those radar parameters affecting the resolution cell size.

3.1.1 Ground Truth Data Base

The two major factors at the ground spot which affect the amount of the transmitted energy that is re-radiated in the direction of the antenna are local geometry and the dielectric propertie of the area. Therefore, the points in the data base must provide this information about the target scene. Each point in the data base represents a fixed area of ground and provides a category assignment and an average elevation of that area. The elevation information combined with the elevation tion data of neighboring cells is sufficient to calculate the local geometry, including local angle of incidence, local slopes, and area.

The area represented by each point in the original data base defines an upper limit on the resolution of the radar system which can be accurately simulated. The resolution cell size can be no smaller than the area represented by the points in the input data base. On the other hand, any radar with poorer resolution can be simulated by appropriately averaging data cells.

3.1.2 Radar Input Parameters

Those parameters which affect the size and position of the resolution cells for the radar being modeled must be specified at this step in the simulation package. Other system parameters will not be specified until a later point to enhance the flexibility of the total package.

First the radar platform position relative to the rectangular data base must be described, through (x,y) coordinates relative to the lower left corner of the data base. The elevation of the platform must also be provided. Minimum and maximum incident angle values are read in to describe the area to be imaged. The azimuth angular range is assumed to be ±45° as in normal PPI operation although this can easily be modified to any desired scan limits.

Also the pulse width (τ) and beamwidth (β) of the radar to be simulated must be provided to determine the resolution in both the range and azimuth directions, respectively.

3.1.3 Program Execution

The conversion from rectangular to polar coordinates is well-known to be:

$$R = (x^2 + y^2)^{1/2}$$
$$\theta = \arctan (x/y)$$

The creation of the polar resolution cell matrix requires mapping a rectangular matrix where each point represents an area into another symbolic matrix in polar coordinates. The mapping of areas is not simple because a rectangular area frequently lies partially inside two or more polar cells. For this particular application, however, the points in the original rectangular data base represent an area significantly smaller than a resolution cell in the desired polar data base. Therefore, several data points are combined to form a single resolution cell. With this fact in mind, the following approximation was made. Only the center point of each area in the original data base was mapped into polar coordinates. Some portion of the area represented by a data cell may be in another resolution cell but that overlap will be ignored since the area of overlap will be something less than half its area (the midpoint lies in another cell). Since several rectangular data points are combined for each polar resolution cell, this area of overlap is of little consequence and can be ignored.

The only other point to be resolved in this program was how to combine the rectangular data points that formed a single polar resolution cell. Each data point in the rectangular ground truth data matrix contains an elevation and a category value. The elevation values or the various rectangular points being combined into a single polar point were averaged. The category information was a more difficult conceptual problem. A cell containing one point of Category 4 and one point of Category 6 does not average to be Category 5. The categories represented one of two sorts of targets: (1) distributed targets, or (2) hard targets(cultural features). In general, distributed targets represent large areas of homogeneous make-up and so there will be few cases of multi-categorized cells except along field boundaries. Hard targets are

usually isolated features such as roads and houses which are superimposed on a distributed target area. In this case multi-category cells are frequent occurrences. Because of their make-up such cultural targets usually produce such high returns that the effect of the other contributing category is washed out. For these reasons a simple priority scheme was devised for the categories in the target site. This priority scheme is presented in Table 1. Whenever multiple categories appear in a resolution cell only the category with the higher priority is retained.

The output of this program then is a matrix somewhat similar to the original input rectangular data base, but each point represents an area on the ground in polar coordinates which it describes with both a backscatter category and an elevation value. The matrix is in polar coordinates so that a point is represented by a range (R) component and an azimuth angle (θ) value. As a consequence, each point does not represent a fixed area on the ground, but an area the size of the resolution cell which varies according to its position in the matrix.

3.2 Greytone Calculation

Equation 1 provides a means for calculating the relative greytone of any resolution cell or pixel, in an image. This formula represents calibration of the average return power from a cell by the average return power from a known, reference system. There is no requirement that the reference system be identical to the radar system being modeled.

However, if we choose the reference system to be the same as the modeled system the equation can be simplified considerably and thus, enhance the execution of the computer program implementation. If the systems are assumed to be the same, the following equalities are valid:

$$P_{T_2} = P_{T_1}$$

$$\lambda_2^2 = \lambda_1^2$$

$$K_2 = K_1$$

$$M_2 = M_1$$

$$G_2^2(\theta) = G_1^2(\theta)$$

Making use of these equalities allows us to reduce Equation (1) to the following:

$$G_{R_{c_2}} = G_{R_{a_1}} + \frac{255}{x}\left\{\gamma \log_{10}\left[\frac{\sigma^{\circ}_{c2}(\theta_\ell)\Delta A_2 G_1^{\,2}(\theta) R_1^{\,4}}{\sigma^{\circ}_{a1}(\theta_\ell)\Delta A_1 G_1^{\,2}(\theta) R_2^{\,4}}\right]\right\} \tag{2}$$

The return power from a single scatterer has noise-like characteristics. This noise in the return signal of a real imaging radar causes fluctuations in the greytone levels of discrete points within a homogeneous region. This effect is called fading. Equation (2) provides only a mean value for the category being imaged. In order to include the effects of fading in the model this noise-like characteristic which follows a Rayleigh distribution[2] must be included. The change in greytone due to fading can be represented by the following equation[3]:

$$G_R = G_{R_{c_2}} + G_{R_{FAD}} \tag{3}$$

where

$$G_{R_{FAD}} = \frac{255}{x}\ \gamma \log_{10}\left(1 + \frac{RN}{\sqrt{N}}\right) \tag{4}$$

where RN is a normally distributed random variable with mean 0 and standard deviation 1 and N is the number of independent samples. A full theoretical development of this formula has been previously reported[4]. The information included in the data base and the equations above can be used to calculate the greytone value for each resolution cell in the data base. There are, however, numerous radar effects only implicit in the greytone equation which must be handled and implementation details to be solved in the computer software. These problems and their solutions will be described in the following sections.

---

[4] Holtzman, J. C., V. H. Kaupp, R. L. Martin, E. E. Komp and V. S. Frost, "Radar Image Simulation Project: Development of a General Simulation Model and an Interactive Simulation Model, and Sample Results," RSL Technical Report 234-13, February, 1976.

3.2.1 Calculation of Resolution Cell Size

The size and shape of the resolution cell varies greatly over the range of the data base. In addition, the area of each cell, represented by ΔA in the greytone equation, varies with the local slopes of the ground spot and must be calculated for each cell in the data base.

Creation of the polar resolution cell matrix accounted for the changing shape of the resolution cell by collecting the appropriate data points from the rectangular input matrix. This accounts for the proper simulation of the widening of the resolution cell in the far range, etc. Size of the resolution cell was only implicit at this step, however, and the effect of local slopes on the area was not treated. The calculation of the ΔA term was reserved until the actual calculation of greytone for each resolution cell since it was necessary to determine the local slopes of the ground and these local slope data were used elsewhere.

The area, ΔA, of a resolution cell for a pulsed radar can be modeled by the following equation to account for slopes in the local terrain:

$$\Delta A = w\ell = \left[\frac{h'\phi}{\cos\theta\cos\theta_A}\right] \cdot \left[\frac{c\tau'}{2\sin(\theta - \theta_c)}\right] \tag{4}$$

where: w = Size of resolution cell in the azimuth direction;

ℓ = Size of resolution cell in the range direction;

h' = Height difference between the cell and the radar;

ϕ = Antenna beam width;

θ = Radar incidence angle;

τ' = Signal pulse width;

θ_A = Angle of local slope of resolution cell in the range direction;

θ_c = Angle of local slope of resolution cell in the azimuth direction.

The local slopes in the range and azimuth directions were easily obtainable from the elevation information in adjacent resolution cells (in the appropriate direction) within the polar data base.

$$\text{SLOPE} = \frac{\Delta\text{ELEV}}{\text{DIST}} \tag{5}$$

where: $\Delta ELEV$ = Change in elevation between two adjacent resolution cells;

DIST = Ground distance represented by the width of a resolution cell at that point.

The angle of the slope was obtained by calling a system provided ARCTANGENT routine.

3.2.2 Calculation of Local Angle of Incidence

The local angle of incidence, θ_ℓ, is another parameter in the greytone equation that depends on the slope of the local terrain. Therefore, this value must also be calculated independently for each resolution cell in the data base. The local incident angle is defined to be the angle between the local normal vector at the ground spot and the vector from the radar platform to the groundspot.

The information about local slopes in the azimuth and range directions used for calculation of the area term as described is the previous section, is also utilized to determine the local angle of incidence. These two slope vectors lie in the local plane, so when the cross-product is computed one produces an equation for the local normal vector. The dot product of this vector and the vector from the platform to the ground spot is formed to yield the cosine of the angle between them. This angle is the local incident angle which one wished to compute. A more formal mathematical derivation of this formula is presented here now. Refer to Figure 1 for a pictorial view of the various vectors.

3.2.3 Shadow

Particularly in target areas with high local relief, radar shadow is a very significant effect in the output image of imaging radar systems. The radar receiver will not receive any return power from an area in the target scene if a straight line from the radar platform to the ground spot (the radar transmitted energy) intersects some other portion of the terrain. Because no energy from the transmitter reaches this area there is no energy to be re-radiated. See Figure 2 which causes a dark area on the image signifying a gap in the return signal. The length and shape of the shadow depends on the position of the transmitter relative to the object casting the shadow.

Figure 1. Tilting of a Resolution Cell

$$\theta_\ell = \cos^{-1}(\hat{P} \cdot \hat{N}_{xyz}) \qquad \text{(dot product)} \qquad (6)$$

where: θ_ℓ = Local angle of incidence of the resolution cell;

$\hat{P}$ = Vector pointing from center of the resolution cell to the antenna boresight (see Equation (7);

$\hat{N}_{xyz}$ = Local normal vector to the resolution cell (see Equation (8).

The pointing vector is:

$$\hat{P} = -(\sin\theta\cos\gamma)\,\hat{x} - (\sin\theta\sin\gamma)\hat{y} + (\cos\theta)\,\hat{z} = \frac{\vec{P}}{|\vec{P}|} \qquad (7)$$

where: θ = Angle from z-local vertical of antenna boresight;

γ = Rotational scan angle from in-track axis (γ = 90° for all SLAR simulations);

$\hat{x},\hat{y},\hat{z}$ = Unit pointing vectors in the x,y, and z directions, respectively.

The unit vector normal to the resolution cell is:

$$\hat{N}_{xyz} = \frac{\vec{N}_{xyz}}{|\vec{N}_{xyz}|} = \frac{\vec{S}_{IT} \times \vec{S}_{CT}}{|\vec{S}_{IT} \times S_{CT}|} \qquad \text{(cross product)} \qquad (8)$$

where: $\vec{S}_{IT} = \hat{x} + (\tan\alpha)\hat{z}$ = Slope in the in-track direction;

α = Angle of the slope in the in-track plane;

$\vec{S}_{CT} = \hat{y} + (\tan\beta)\hat{z}$ = Slope in the across-track direction;

β = Angle of the slope in the across-track plane.

The effects of shadow are not directly accounted for in the greytone equation, and so must be modeled by other constructs in the software implementation. As described above a resolution cell will produce a return value in the output image only if it lies in direct line of sight from the radar platform. A simple test for this condition is provided in the software simulation package. An equation is developed for the line through the radar platform and the last non-shadowed resolution cell (in the range direction) in a coordinate system set at sea level, directly beneath the radar platform. The elevation information of the resolution cells from the ground truth data matrix corresponds to this coordinate system. The elevation of the next resolution cell is compared to the value of the "non-shadow" line at that point. If the elevation is greater than the y-value of the "non-shadow" line (see Figure 2) then the cell is in direct line of sight and a greytone is calcualted for the cell. Also the equation for the "non-shadow" line is updated to pass through this new cell. On the other hand, if the elevation of the resolution cell is less than the "non-shadow" line no return need be calculated for this cell because it lies in the "shadow" of a previous cell. See Figure 2 for a clarification of these conditions.

Accounting for radar shadow was another compelling reason to convert to a polar coordinate system for the resolution cell matrix in the PPI simulation implementation. The previous discussion on computation of shadowed areas implicitly assumed that the resolution cells were arranged in an order so that all cells imaged by a single transmitted pulse could be consecutively accessed. The algorithm depends on the elevations of the cells prior to the cells being tested which may block illumination of the present cell. For side-looking radar (SLAR) in which the antenna is fixed perpendicular to the flight path of the aircraft, a rectangular data base format is appropriate[4] since a column of cells in the data base correspond to the path of a transmitted pulse. PPI radar, however, is a forward-looking device and the antenna rotates as the aircraft proceeds to scan an area ahead of the flight track

[4] Martin, R. L, J. L. Abbott, M. McNeil, V. H. Kaupp, and J. C. Holtzman, "Digital Model for Radar Image Simulation and Results," TR 319-8, Remote Sensing Laboratory, The University of Kansas, August, 1976.

Figure 2. Radar Image Shadow.

(normally a 90° sector). The pulse path no longer corresponds to a single column in a rectangular system. If one converts to a polar coordinate system, the beam path then corresponds to a fixed angle (θ) value while the radius (R) value takes on all possible values in its domain. In the polar data base the cells are stored in a matrix of (R, θ) values, so one dimension represents a constant angle θ value and the other dimension represents a fixed radius (R) value. So the polar resolution cell matrix is ideal for the shadow algorithm to work in the PPI radar simulation model.

3.2.4 Layover

Layover is another radar phenomenon not explicitly accounted for in the greytone equation. Very briefly, the position of the return from a particular ground spot in the output image depends upon the range distance to the ground spot rather than the effective ground distance from beneath the radar platform. As a consequence, the top of a tall, vertical object will appear closer to the radar position than the base of the object in the output image because the slant range distance of the top of the object is less than the slant range distance to its base.

The technique to model this effect is suggested by the description of layover. After calculating the level of return at the antenna for a resolution cell, its position in the output matrix was determined by calculating the slant range distance to the cell, taking into account the local elevation of the cell. Depending upon relative elevations and the angles of incidence, the returns from several points in the ground swath may occur at the same time; thus laying-over into an earlier slot. It was important to calculate the right quantity at this point and reserve the conversion to greytone values until all the returns of a single scan line had been placed in the appropriate output cells. In case returns from two (or more) resolution cells mapped into a single output cell the effect at the antenna in a real system would be to add them. After layover has been accounted for, the combined return values are converted to greytone values. The simulation implementation models this aspect of the real radar operation.

For the purpose of simulating PPI imagery it is important that the output matrix be in polar coordinates as well as the resolution cell matrix. This allows those cells corresponding to a single pulse path to be accessed as a group.

3.2.5 Backscatter Data

This simulation implementation uses empirical backscatter data ($\sigma°$) to account for the effects of differing microwave reflectivities for the various target categories in the scene. For a given frequency and polarization, the $\sigma°$ value depends on the category and local angle of incidence. For each category of $\sigma°$ data, a third-order polynomial equation as a function of incident angle is fit to empirical values of $\sigma°$ at specific angles. Each time the computer program requires a $\sigma°$ value to calculate the return from a ground spot, the program computes the value of this polynomial for the particular conditions in existence thereby providing a continuous estimate of the $\sigma°$ value.

If the frequency and/or polarization of the transmitter of the radar to be simulated are changed, all that is required for the simulation program to properly account for this change is to change this set of input data.

3.2.6 Summary of Greytone Calculation

The software program to implement the simulation model brings all of the algorithms and calculations discussed above together to produce the image output. It requires a data base of resolution cells in polar coordinates, specification of all the radar parameters, and $\sigma°$ data for the categories included in the data base. The output is a matrix of greytone values still in polar coordinates representing the simulated radar image. Another program accepts this matrix as input and converts it to a rectangular format for display.

3.3. Formation of Output Image

The greytone matrix which formed the output of the previous step remains in polar coordinates. A single line of that matrix contains image density values corresponding to the returned power along a fixed azimuth angle sweep. In the normal operation of a real PPI imaging system the returned powers would be displayed by intensity modulating the

beam of a cathode ray tube (CRT). The output device available for this simulation program, however, is a raster scanning device and so the image must be modified somewhat for the display device.

The polar matrix of greytone values is converted to a rectangular array of values which can be displayed on the raster scanning facilities of IDECS*. The problem at this stage is to convert points representing areas in a polar coordinate system to points in a rectangular array without introducing geometric distortion. The reverse of this problem has already been encountered earlier when it was necessary to convert the rectangular ground truth data base into a polar matrix of resolution cells. In order to assure an accurate mapping and return all of the information in the polar matrix to a rectangular matrix, the resolution of the rectangular output matrix must be fine relative to the resolution of the polar matrix. With this criterion satisfied, the mapping into rectangular coordinates can be accomplished by repeating the algorithm of step one, which is to map the center point of each polar cell into the appropriate rectangular cell. The crucial difference between this step and the earlier one is that, at this point, information is being extracted from a large polar cell to provide the greytone value for a number of smaller rectangular cells while in the creation of the resolution cell matrix the information is passed from the smaller rectangular cell to the larger polar cell. However, in both cases the principle of the mapping algorithm is the same: Locate the relatively small rectangular area within the area of the relatively large polar cell area. In the earlier case the mapping was a function since there is exactly one value for the distinction of each rectangular cell. The mapping in the present case (from polar cells to rectangular cells) does not have this useful functional property: each polar cell maps to several smaller rectangular cells. The calculations required for this mapping are much more complex and have been avoided. The results produced by the mapping just described verify that it does maintain excellent geometric fidelity.

---

* IDECS (Image Discrimination, Enhancement, Combination, and Sampling) is an analog image processing device which is electrically interfaced with a digital computer (PDP-15) and is located at the Remote Sensing Laboratory.

4.0 SOFTWARE IMPLEMENTATION OF PPI SIMULATION MODEL

The software package to produce PPI radar simulations has been divided into two separate units. The first unit converts the rectangular input data base into a polar resolution cell matrix. The second unit produces a simulated radar image using the polar data base as input. This division within the package allows the user considerably more flexibility. Numerous changes can be made in the greytone calculation so that the effect on the output image can be analyzed without re-creating the polar data base for each run. This allows for considerable savings of computer resources because in the total simulation process the major expense is incurred when the polar data base is constructed.

The program SLICE creates the polar data base. For input it requires the rectangular data base and system radar parameters. Internally the program is divided into two separate activities which communicate via temporary file space. This step requires extensive computation and a large amount of memory space. Because of its expense the computation was separated from the memory requirements as much as possible, thus minimizing the (time-system resources) product.

The second program unit, labelled VERIF, accepts the polar resolution matrix for input, implements the actual simulation algorithms and converts the output to a suitable format for display. This unit actually combines two of the major steps described in the report: (1) Greytone calculation, and (2) conversion to output format. These steps were combined for user convenience. In almost all cases when the user produced a simulation he wanted the output converted to a suitable visual output form and seldom more than one output format was required. Combining the two steps reduced the user work involved in producing the output. The two steps are physically independent activities within the program VERIF which communicate via an intermediate tape. If the user has special purposes in mind the two activities can easily be separated.

4.1 Application of PPI Simulation Implementation to Terminal Guidance

The following sequence of programs represents the modification of the general PPI simulation programs to apply of the specific problem of terminal guidance.

Because of the increased size of the data base and some special calculations the program units were further subdivided to enhance error recovery if an error occurred during the sequence.

Programs STEPI and STEPIA perform the actions required to convert the data base to polar coordinates. For this application the radar platform was assumed to be directly over the center of the data base and a full 360° scan was simulated rather than only a forward-looking sector.

POLSIM calculated only the return power from each resolution cell rather than immediately converting this value to a greytone. This intermediate step was required because several resolution cells were averaged to simulate the return from each area.

ECRIT is the final program unit which converts the image output into a format suitable for input into DICOMED to produce a high resolution image output.

```
0001  C                SLICE
0002  C
C003  C   THIS PROGRAM PRODUCES A POLAR DATA BASE FOR PPI SIMULATIONS
C004  C   FROM A RECTANGULAR DATA BASE.
0005  C   THE INPUT IS A RELATIVELY SMALL RAECTANGULAR DATA BASE SO THE
C006  C   PPI SIMULATION WILL NOT BE A FULL 90 DEGREE SECTOR NOR WILL IT
C007  C   EXTEND TO ZERO DEGREE INCIDENCE.
C008  C   THE PURPOSE HERE IS VALIDATION BY COMPARISON TO SLAR SIMULATIONS
C009  C   OF THE SAME AREA.
C010  C   TO BE USED AS DATA BASE FOR THE SIMULATION PROGRAM
C011  C   THE RESULTS FORMED FROM THESE DATA BASES WILL BE
C012  C   USED ESPECIALY FOR THE FINAL REPORT 319-27 (6/77)
0013  C
C014  C
C015          IMPLICIT INTEGER (A-Y)
0016          REAL ARCOS,FLOAT,ATAN,SQRT
0017          DIMENSION PRIOR(29,29),RECORD(450),TABLE(1000),OT(3,250)
C018          DATA NFILE,FLIP /4,1/
0019          DATA HALF,STRT/0,1/
C020          DATA CNT,NUMB/1,0/
0021          DATA OCTB,OCT2/01000000000,0100/
C022  C
0023  C
C024  C   THIS PROGRAM COMPUTES THE POLAR RESOLUTION CELL THAT EACH
C025  C   POINT IN THE RECTANGULAR INPUT DATA BASE FALLS INTO.
0026  C   THIS IS COMPUTATIONALLY EXPENSIVE SO THIS INFORMATION
C027  C   IS WRITTEN SERIALLY TO A TEMP FILE (INSTEAD OF INTO A LARGE
0028  C   POLAR MATRIX RESIDING IN MEMORY).
0029  C   A SUBSEQUENT PROGRAM TAKES THIS INFO FROM THE TEMP
0030  C   FILE AND CONSTRUCTS THE ACTUAL POLAR MATRIX.  SINCE THE
C031  C   COMPUTATION HAS ALREADY BEEN DONE THE LARGE MATRIX RESIDES
C032  C   IN MEMORY FOR MUCH LESS TIME AND SO THE TOTAL JOB
C033  C   COST IS LESS IN THIS 2 STEP PROCESS.
0034  C
C035  C
C036          DO 95 I=1,29
C037          READ(05,5) (PRIOR(I,J),J=1,29)
C038          PRINT 5, (PRIOR(I,J),J=1,29)
0039   95     CONTINUE
0040    5     FORMAT(21I4)
C041  C
C042  C   ADJY = DISTANCE (IN FEET) FROM PLATFORM POSITION TO BEGINNING
C043  C       OF DATA BASE (DISTANCE TO NEAR RANGE)
C044  C   CELSIZ = SIZE (IN FEET) REPRESENTED BY DATA POINTS - ASSUMED SQUARE
0045  C   NUMPT = NUMBER OF DATA POINTS PER RECORD ON INPUT TAPE
0046  C   NUMREC = NUMBER OF RECORDS ON INPUT TAPE
0047  C          EACH RECORD GOES FROM SOUTH TO NORTH.  RECORDS ON TAPE IN
C048  C          A WEST TO EAST ORDER
C049  C   WIDTH = FIXED SIZE FOR RANGE RESOLUTION
C050  C   ZBMWD = BEAMWIDTH (IN RADIANS)
0051  C
C052  C
0053  C
C054  C
0055          READ(05,10) ADJY,CELSIZ,NUMPT,NUMREC,WIDTH,
C056        & ALT,ZBMWD,ZPULS
C057   10     FORMAT(6I8,2E12.4)
0058  C
```

```
C059          WRITE(6,121)ADJY,CELSIZ,NUMPT,NUMREC,WIDTH,ALT,ZBMWD,ZPULS,NFILE
C060   121    FORMAT(' PARAMETERS PASSED TO SIMULATION TAPE',/,
C061       &    ' ADJY,CELSIZ,NUMPT,NUMREC,WIDTH,ALT,ZBMWD,ZPULS =    ',
C062       &    /,6I8,2E12.4,/,'READ FROM FILE NUMBER ',I4,'    ON TAPE',//)
C063  C
C064  C
C065  C       CALL POST TO POSITION TAPE TO PROPER DATA BASE ON INPUT TAPE
C066  C
C067          IF(NFILE .NE. 1)CALL POST(01,0,NFILE,1,ERROR)
C068          IF(ERROR .NE. 0)WRITE(6,55)ERROR
C069   55     FORMAT(' ERROR IN POST ROUTINE, FIRST HALF OF PROGRAM')
C070  C
C071  C
C072          ALT2 = ALT*ALT
C073  C
C074  C
C075  C   NUMR - MAXIMUM NUMBER OF CELLS IN RANGE DIRECTION IN
C076  C          RESOLUTION CELL MATRIX BEING CONSTRUCTED
C077  C
C078          ZXMID = NUMREC/2 + .5
C079          ADJR = ADJY/CELSIZ
C080          ZMAXR = (NUMPT + ADJR)**2 + ZXMID**2
C081          ZNUMR = SQRT(ZMAXR) - ADJR
C082          ZCW = FLOAT(CELSIZ)/FLOAT(WIDTH)
C083          NUMR = ZNUMR*ZCW + 2.
C084  C
C085          ZCTAU = 983.57*ZPULS/2.
C086          ZHAFANG = ATAN(ZXMID/ADJR)
C087          HAFANG = ZHAFANG/ZBMWD + 1.
C088          NUMANG = HAFANG * 2
C089  C
C090  C
C091  C   TABLE = TABLE LOOK UP FOR ANGLE
C092  C           USE 1000 TIMES COSINE OF ANGLE AS INDEX
C093  C           RESULT IS PROPER ANGLE BIN FOR THE POINT
C094  C
C095          DO 100 I=1,1000
C096   100    TABLE(I)=ARCOS(FLOAT(I)/1000.)/ZBMWD + 1
C097  C
C098  C   WRITE PARAMETERS TO TAPE FOR DATA TO NEXT STEP
C099  C
C100          WRITE(03) NUMR,NUMANG,WIDTH,ALT,ZCTAU
C101  C
C102  C
C103  C
C104  C
C105          DO 200 I=1,NUMREC
C106          READ(01,END=800) LINE,(RECORD(M),M=1,NUMPT)
C107  C
C108  C       ALGORITHM TO PRODUCE 300 DEGREE DATA BASES
C109  C       USING 116 DEGREE BASES AS INPUT
C110  C
C111          IF(FLIP .NE. 1)GOTO 202
C112  C
C113  C
C114          DO 202 II=1,(NUMPT+1)/2
C115          TT = RECORD(II)
C116          RECORD(II) = RECORD(NUMPT+2-II)
C117          RECORD(NUMPT+2-II) = TT
C118   202    CONTINUE
```

```
C
C
      ZX = 1-ZXMID
      IF(OTF .EQ. 0 .AND. ZX .GT. 0) GOTO 600
 105  ZX = ABS(ZX)
      ZX2 = ZX**2
C
      DO 180 J=2,NUMPT
      ZY = J+ADJR
      ZY2 = ZY**2
C
      ZR = SQRT(ZX2+ZY2)
      ZNORM=ZR-ADJR
      R = ZNORM*ZCW + 1.
      COSANG = ZY/ZR * 1000.
C
      IF(COSANG .LT. 0) WRITE(6,69) COSANG,I,J,R
      IF(COSANG .GT. 1000) WRITE(6,69) COSANG,I,J,R
 69   FORMAT(//' **ERROR - COS > 1',5I8/)
      IF(COSANG .GT. 10 .AND. COSANG .LT. 990) GOTO 117
      ANG= ARCOS(ZY/ZR)/ZBMWD + 1
      GOTO 118
C
C ANG - APPROPRIATE ANGLE BIN FOR THE CURRENT POINT
C
 117  ANG = TABLE(COSANG)
C
C
 118  CONTINUE
C
 120  INDEX =  J
      IF(OLDR .NE. R .OR. OLDANG .NE. ANG) GOTO 150
      CNT = CNT + 1
      CAT=FLD(30,6,RECORD(INDEX)) +1
      IF(CAT .GT. 29 .OR. TCAT .GT. 29)WRITE(6,542) I,J,CAT,TCAT
 542  FORMAT(' I,J,CAT,TCAT ',4I6)
      IF(TCAT .EQ. 0) TCAT = 1
      TCAT = PRIOR(TCAT,CAT)
      TELV = TELV + RECORD(INDEX)/OCT2
C
      GOTO 180
C
C
 150  TELV = TELV/CNT * OCT2 + TCAT
      NUMB=NUMB+1
      OT(1,NUMB)=OLDR
      OT(2,NUMB)=OLDANG
      OT(3,NUMB)=TELV
      OLDR = R
      OLDANG = ANG
      IF(NUMB .LT. 250) GOTO 157
      WRITE(03) NUMB,OT
      NUMB=0
 157  SUM = SUM+1
      TELV = RECORD(J)/OCT2
      TCAT = FLD(30,6,RECORD(J)) + 1
      IF(TCAT .GT. 29)TCAT=1
      TOTCNT = TOTCNT + CNT
      CNT = 1
C
```

```
C179   180    CONTINUE
C180   200    CONTINUE
C181          GOTO 900
C182  C
C183  C
C184  C  ERROR MESSAGES
C185  C
C186   800    WRITE(6,801) I
C187   801    FORMAT(//'  UNEXPECTED END OF FILE AT RECORD ',I6)
C188          GOTO 900
C189  C
C190  C
C191   600    OTF = 1
C192          WRITE(03) NUMB,OT
C193          WRITE(03) -1,OT
C194          NUMB = 0
C195          GOTO 105
C196   900     WRITE(03) NUMB,OT
C197          WRITE(6,901) SUM,TOTCNT
C198   901    FORMAT(10X,'***DONE***',I8,'  RECORDS WRITTEN',
C199       &     10X,I8,' POINTS PROCESSED')
C200  C
C201          WRITE(6,911) NUMR,NUMANG
C202   911    FORMAT(' NUMR AND NUMANG =',2I8)
C203          STOP
C204          END
C205  $       EXECUTE
C206  $       LIMITS   10,14K,,1K
C207  $       TAPE     01,,,60808,,SL4-02
C208  $       FFILE    01,BUFSIZ/500
C209  $       FILE     03,X3SS,50L
C210  $       OPTION   FORTRAN
C211  $       FORTRAN
C212  $       LIMITS   ,28K
C213  C
C214  C  THIS IS THE FOLLOW UP ROUTINE TO SLICE.
C215  C  IT USES THE DATA SUPPLIED BY SLICE TO PRODUCE A DATA MATRIX
C216  C  IN POLAR COORDINATES.
C217  C
C218  C  SLICE AND PIE WERE DEVELOPED ESPECIALLY FOR PRODUCING
C219  C  VERIFICATION IMAGES FOR FINAL REPORT 319-27 (6/77)
C220  C  THIS PROGRAM TAKES THE DATA ON TEMP FILE PRODUCED BY
C221  C  'SLICE' TO PRODUCE THE ACTUAL POLAR DATA BASE.
C222  C
C223  C
C224          IMPLICIT INTEGER (A-Y)
C225          DIMENSION A(170,125),PRIOR(29,29),DAT(3,250)
C226          DIMENSION C1(155,20)
C227          DATA NFILE/6/
C228          DATA OCT2,SHIFT,CTR/0100,0100000,010000000000/
C229  C
C230          OTF = 0
C231  C
C232          WRITE(6,595) NFILE
C233   595    FORMAT('   OUTPUT WRITTEN TO FILE NUMBER',I3)
C234  C
C235  C
C236  C       POSITION TAPE TO WRITE TO PROPER FILE
C237  C
C238          IF(NFILE .NE. 1)CALL POST(02,1,NFILE,1,ERROR)
```

```
C239          IF(ERROR .NE. 0)WRITE(6,54)ERROR
C240   54     FORMAT(' TROUBLE IN POST ROUTINE, PIE SECTION')
C241  C
C242          REWIND(03)
C243          READ(03) NUMR,NUMANG,WIDTH,ALT,ZCTAU
C244          WRITE(02) NUMR,NUMANG,WIDTH,ALT,ZCTAU
C245          HAFANG = NUMANG/2
C246  C
C247          DO 110 I=1,29
C248           READ(05,321) (PRIOR(I,J),J=1,29)
C249   110    PRINT 321, (PRIOR(I,J),J=1,29)
C250   321    FORMAT(21I4)
C251  C
C252  C
C253   5      READ(03,END=500) NUMB,DAT
C254          IF(NUMB .LT. 0) GOTO 500
C255  C
C256          DO 200 I=1,NUMB
C257          R = DAT(1,I)
C258          ANG = DAT(2,I)
C259          IF(OTF .EQ. 0) ANG = IABS(ANG-HAFANG)+1
C260          CAT1=FLD(30,6,DAT(3,I))
C261          ELV1=DAT(3,I)/OCT2
C262          A(R,ANG)= ELV1 + CTR + A(R,ANG)
C263          WORD=ANG/6 + 1
C264          BIT = MOD(ANG,6)*6
C265          TAG = FLD(BIT,6,C1(R,WORD))
C266         IF(TAG .EQ. 0) TAG = 1
C267          IF(PRIOR(TAG,CAT1) .EQ. CAT1) FLD(BIT,6,C1(R,WORD))=CAT1
C268   200     CONTINUE
C269          GOTO 5
C270  C
C271  C
C272  C
C273  C
C274   500    CONTINUE
C275          DO 400 I=1,HAFANG
C276          WORD=I/6 + 1
C277          BIT = MOD(I,6)*6
C278  C
C279          DO 300 J=1,NUMR
C280          MULT = FLD(0,6,A(J,I))
C281          IF(MULT .EQ. 0) GOTO 250
C282          ELV1 = FLD(21,15,A(J,I))/MULT
C283         GOTO 251
C284   250     ELV1=0
C285          ELV2=0
C286  C
C287   251    A(J,I)=ELV1 * OCT2 + FLD(BIT,6,C1(J,WORD))
C288   300     CONTINUE
C289          WRITE(02) (A(K,I),K=1,NUMR)
C290          WRITE(06,23) (A(K,I),K=1,NUMR,2)
C291   23     FORMAT(1X,10O01)
C292          DO 410 K=1,NUMR
C293   410    A(K,I)=0
C294    21    FORMAT(17(1X,O6))
C295   400    CONTINUE
C296  C
C297  C
C298          DO 220 L=1,NUMR
```

```
0299          DO 220 L2=1,20
C300    220   C1(L,L2)=0
C301          OTF = 1 + OTF
0302          IF(OTF .LT. 2) GOTO 5
C303          ENDFILE(02)
0304          WRITE(02)A
0305          STOP
C306          END
0307  $       EXECUTE
0308  $       LIMITS   05,36K,,5K
C309  $       FILE     03,X3DD
C310  $       TAPE     02,X2DD,,61083,,DIC013,OUT
C311  $       ENDJOB
```

```
C                    VERIF
C
C   PROGRAM ACCEPTS DATA MATRIX IN POLAR COORDINATES FROM
C   FILECODE 01 (CREATED BY STEP1) AND PRODUCES A
C   SIMULATION.  THIS IS THE VERSION USED TO PRODUCE RESULTS
C   OF PPI VALIDATION.  (6 SQUARE IMAGES FOR FINAL REPORT -
C   319-27  DATED 6/77).
C
C
          IMPLICIT INTEGER (A-Y)
          REAL RMS
          REAL ALOG10,FLOAT,SIN,COS,ARCOS
          COMMON ZTAB(1000),ZCF(29,4),ZS(21),LEN(275)
          COMMON /IO/ BASE(275,3),CAT(275,3),NUMR,NUMANG,ISEED
          COMMON /OT/ GT(275),ZGT(275,3),ZOT(275),ZSTRT(275,2)
C
          DATA NFILE/5/
          DATA L1,L2,L3/1,2,3/
          DATA OCT2/0100/
          DATA ADJR,ZLICE/55519,.16/
          DATA GSCALE,GTREF/256,140/
C
          ISEED = 1234567893
C
C         CALL TO POST TO POSITION INPUT TAPE TO PROPER DATA BASE
C
          IF(NFILE .NE. 1)CALL POST(01,0,NFILE,1,ERROR)
          IF(ERROR .NE. 0)WRITE(6,1234)ERROR
 1234     FORMAT(' TROUBLE WITH POST ON INPUT TAPE')
C
C
          READ(01) NUMR,NUMANG,WIDTH,ALT,ZCTAU
  10      FORMAT(4I8)
C
          DO 100 I=1,29
 100      READ(05,15) (ZCF(I,J),J=1,4)
  15      FORMAT(4E14.7)
C
          READ(05,2) ZS
   2      FORMAT(10(F6.2))
C
C
C         WRITE OUT PARAMETERS
C
          WRITE(06,16)(ZCF(I,J),J=1,4)
  16      FORMAT(' Z C FY',/,4E14.7)
          WRITE(06,17)ZS
  17      FORMAT(' PARAMETERS ZS ',/,10(F6.2))
C  LENGTH OF RESOLUTION CELL IN AZIMUTH INCREASES WITH RANGE
C  THE ARRAY - LEN - CONTAINS THE RESOLTION CELL LENGTH
C  (TIMES 2) AT EACH RANGE BIN.  USED TO CALCULATE LOCAL
C  ACROSS TRACK SLOPE
C
          R = ADJR -WIDTH/2
          DO 105 I=1,NUMR
          R = R + WIDTH
 105      LEN(I)= ZLICE *R/FLOAT(NUMANG) + 1
C
```

```
C   WID2 = 2 TIMES WIDTH OF RESOLUTION CELL IN TRACK DIRECTION
C        (A CONSTANT VALUE FOR THIS SIMULATION)
C
         WID2 = WIDTH + WIDTH
C
C   TRANSFER PARAMETERS TO TAPE FOR OUTPUT ROUTINE
C
         WRITE(02) NUMR,NUMANG,WIDTH
         WRITE(6,707) NUMR,NUMANG,WIDTH
 707     FORMAT(' NUMR,NUMANG,WIDTH=',4I8)
C
C
         DO 110 I=1,1000
 110     ZTAB(I)= ARCOS(FLOAT(I)/1000.)
C
         CALL NEXT(2,IEV)
         IF(IEV .GT. 0) GOTO 800
         DO 120 I=1,NUMR
         BASE(I,1)=BASE(I,2)
 120     CAT(I,1)=CAT(I,2)
C
         DO 300 ANG = 1,NUMANG
         IF(ANG .LT. NUMANG) GOTO 210
         DO 205 I=1,NUMR
         BASE(I,L3)=BASE(I,L2)
 205     CAT(I,L3)=CAT(I,L2)
         GOTO 211
C
 210     CALL NEXT(L3,IEV)
         IF(IEV .GT. 0) GOTO 800
 40      FORMAT(1X,30I4)
 211     CONTINUE
C
         ZM = FLOAT(BASE(1,L2)-ALT)/FLOAT(ADJR)
C
         DO 270 ROW=1,NUMR
         ROW1 = ROW -1
         IF(ROW1 .LE. 0) ROW1 = 1
         ROW2 = ROW + 1
         IF(ROW2 .GT. NUMR) ROW2 = NUMR
C
         IF(CAT(ROW,L2) .EQ. 0) GOTO 270
C
 254     ZDELT = FLOAT(BASE(ROW,L2)-BASE(ROW1,L2))/FLOAT(WIDTH)
         IF(BASE(ROW1,L2) .EQ. 0) ZDELT = 0.
         ZY = ABS(BASE(ROW,L3)-BASE(ROW,L1))
         ZHYP = SQRT(ZY*ZY + LEN(ROW)**2)
         ZRHO = ZY/FLOAT(LEN(ROW))
         ZCOSRHO = FLOAT(LEN(ROW))/ZHYP
C
         NALT = ALT - BASE(ROW,L2)
C
         ZGDIS = ROW*WIDTH + ADJR
         Y = ZM*ZGDIS + ALT
         IF( Y .GT. BASE(ROW,L2)) GOTO 270
         ZM = FLOAT(BASE(ROW,L2)-ALT)/ZGDIS
         ZSR = SQRT(ZGDIS**2 + NALT**2)
         ZSINTH = ZGDIS/ZSR
         ZCOSTH = FLOAT(NALT)/ZSR
C
```

```
C119   250    CONTINUE
C120           IF(ZCF(CAT(ROW,L2),1) .LT.100.) GOTO 251
C121           ZOT(ROW)=10.
0122           GOTO 270
C123   251    CALL RTPWR(ZRHO,ZCOSRHO,ZDELT,NALT,CAT(ROW,L2),ZCOSTH,
C124        1  ZSINTH,ZPWR)
0125           ZOT(ROW)= ZOT(ROW)+ZPWR
0126   270    CONTINUE
0127   C
0128           DO 410 J=1,NUMR
0129           IF(ZOT(J) .LT. .00001) GOTO 409
C130           ZFADE = 1. + RMS(ISEED)/3.
C131           IF(ZFADE .LT. 10E-5) ZFADE = 10E-5
0132           FADE = GSCALE/2 * ALOG10(ZFADE)
0133           GT(J) = GSCALE/2 * ALOG10(ZOT(J)) + FADE + GTREF
C134           IF(GT(J) .GT. (GSCALE-1)) GT(J)=(GSCALE-1)
C135           IF(GT(J) .LT. 0) GT(J)=0
C136           GOTO 410
C137   409    GT(J)=0
C138   410    CONTINUE
0139           WRITE(02) (GT(J),J=1,NUMR)
C140           WRITE(6,13) (GT(J)/26,J=1,NUMR,2)
C141   13     FORMAT(1X,121I1)
0142   C
0143           T = L1
0144           L1=L2
C145           L2=L3
C146           L3=T
0147   C
C148           DO 290 K=1,NUMR
C149           ZGT(K,L3)=0
C150   290    ZOT(K)=0
C151   300    CONTINUE
0152           STOP
C153   800    WRITE(6,801) ANG
C154   801    FORMAT(' RAN OUT OF DATA AT RECORD ',I5)
C155           STOP
C156           END
C157           SUBROUTINE NEXT(LINE,IEV)
0158           IMPLICIT INTEGER (A-Y)
C159           REAL RMS
0160           COMMON /I0/ BASE(275,3),CAT(275,3),NUMR,NUMANG,ISEED
C161           DATA OCT2/0100/
C162   C
C163           READ(01,END=900) (BASE(I,LINE),I=1,NUMR)
0164   C
C165           DO 100 I=1,NUMR
C166           CAT(I,LINE) = FLD(30,6,BASE(I,LINE))
0167           BASE(I,LINE)=BASE(I,LINE)/OCT2
C168   C  MODIFICATION FOR TREE CATAGORY (CAT 14)
C169   C  ELEVATION IS MODIFIED TO REFLECT ADDITIONAL HEIGHT OF TREES
C170   C
0171           IF(CAT(I,LINE) .NE. 14) GOTO 100
C172           BASE(I,LINE)=BASE(I,LINE)+70 + RMS(ISEED)*10
0173   100    CONTINUE
C174           RETURN
C175   900     IEV = 1
C176           RETURN
0177           END
C178           SUBROUTINE RTPWR(RHO,COSRHO,DELT,NALT,ICAT,COSTH,SINTH,PWR)
```

```
C
      COMMON TABLE(1000),CF(29,4),S(21)
C
      ITRACE = 1
C
      DATA BASEALT,SIGREF/28500.,-1.2/
      IF(ICAT .EQ. 0) ICAT = 1
      IF(ICAT .GT. 1) GOTO 505
      PWR = 0
      RETURN
C
C  CALCULATE LOCAL ANGLE OF INCIDENCE
C
 505  ACOS = (COSTH + SINTH*DELT)/SQRT(1.+DELT**2+RHO**2)
      IF(ACOS .LT. 0.) GOTO 800
      NLOC = ACOS*1000.
      ALOC = TABLE(NLOC)*57.295
C
      IF(NLOC .LT. 6 .OR. NLOC .GT. 995) ALOC = ARCOS(ACOS)*57.295
C
C  CALCULATE SIGMA ZERO FOR GIVEN CATAGORY ,T THE LOCAL ANGLE
C  OF INCIDENCE JUST CALCULATED
C
      SIGO = ALOC*(ALOC*(ALOC*CF(ICAT,1) + CF(ICAT,2)) + CF(ICAT,3))
     &   + CF(ICAT,4)
C
      SIGO = SIGO/10. - SIGREF
C
C  THDELT = SINE OF ANGLE THETA-DELT
C  WHICH IS NEEDED FOR THE POWER FORMULA
C
      IF(DELT .LT. .05) GOTO 210
      THDELT = ABS((SINTH-COSTH*DELT)/SQRT(1.+DELT**2))
      GOTO 211
 210  THDELT = SINTH
 211  IF(THDELT .LT. .001) GOTO 810
C
C
      ALT = (BASEALT)/NALT
C
C  POWER EQUATION
C
      PWR = (10**SIGO)*SINTH*SINTH*(ALT**3)/(2*COSTH*COSRHO*THDELT)
      GOTO 900
C
C
 800  IF(ITRACE .GT. 3) WRITE(6,801) COSTH,DELT,ACOS
 801  FORMAT(' DELTA IS > THETA ',3F10.4)
      PWR = 0
      GOTO 900
 810  IF(ITRACE .GT. 3) WRITE(6,811) SINTH,COSTH,DELT,RHO,THDELT
 811  FORMAT(' DELTA = THETA',5F12.4)
      PWR = 10.
 900  RETURN
      END
$     EXECUTE
$     LIMITS   05,17K,,4K
$     TAPE     01,X1DD,,61083,,DIC013
$     TAPE     02,X2DD,,60724,,ABFAL1,OUT
$     OPTION   FORTRAN
```

```
$       FORTRAN
$       LIMITS  ,28K
C
C  PROGRAM TO TRANSLATE BACK FROM POLAR COORDINATES TO
C  RECTANGULAR TO DISPLAY IMAGE.  (THIS IS SPECIAL
C  VERSION FOR THE PPI VERIFICATION IMAGES OF 6/77).
        IMPLICIT INTEGER (A-Y)
        REAL FLOAT,ARCOS
        DIMENSION BUF(600),TABLE(1000)
        DATA ZSIZE,ZBMWD/60.,0.9631E-3/
        DATA ADJR,RES/55519,156/
C
C
C
        READ(05,15) MAXDIS,MAXANG,WIDTH
 15     FORMAT(4I6)
C
        DO 100 I=1,1000
 100    TABLE(I)= ARCOS(FLOAT(I)/1000.)/ZBMWD
C
C
C
        HFRES = RES/2
        IF(2*HFRES .LT. RES) HFRES = HFRES +1
C
 555    CONTINUE
C
        ZY = ADJR + ZSIZE*(RES+.5)
        DO 600 I=1,RES
        ZY = ZY - ZSIZE
        ZY2 = ZY*ZY
C
 200    DO 575 J=1,RES/2
        NX = J*ZSIZE -ZSIZE/2
        ZX2 = NX * NX
        ZRDIS = SQRT(ZX2+ZY2)
        R = (ZRDIS-ADJR)/WIDTH
        IF( R. GT. MAXDIS) GOTO 580
        COSANG = ZY/ZRDIS * 1000.
        ANG = TABLE(COSANG)
        IF(COSANG .LT. 6 .OR. COSANG .GT. 994)
     &   ANG = ARCOS(ZY/ZRDIS)/ZBMWD
        IF(ANG .EQ. 0) ANG = 1
C
C
        IF(R .EQ. OLDR .AND. ANG .EQ. OLDANG) GOTO 575
        CNT = CNT+1
        IF(CNT .GT. 1) GOTO 585
        OLDR = R
        OLDANG = ANG
 585    IF(CNT .GT. 600) GOTO 800
        FLD(0,10,BUF(CNT))=J-1
        FLD(10,10,BUF(CNT))=OLDR
        FLD(20,10,BUF(CNT))=OLDANG
        OLDR = R
        OLDANG = ANG
 575    CONTINUE
 580    CNT = CNT + 1
        FLD(0,10,BUF(CNT))=J-1
        FLD(10,10,BUF(CNT))=OLDR
```

```
C299          FLD(20,10,BUF(CNT))=OLDANG
C300          OLDR = R
C301          OLDANG = ANG
C302          WRITE(03) CNT
C303          WRITE(03) (BUF(I9),I9=1,CNT)
C304          TOT = TOT + CNT
C305          MOST = MAX(MOST,CNT)
C306          CNT = 0
C307   600    CONTINUE
C308          GOTO 950
C309   800    WRITE(6,805) I
C310   805    FORMAT(' BUFFER OVERFLOW AT LINE',I5)
C311  C
C312  C
C313   950    WRITE(6,951) TOT,MOST
C314   951    FORMAT(' WE ARE DONE ',2I8)
C315          STOP
C316          END
C317  $       EXECUTE
C318  $       LIMITS  10,11K,,2K
C319   154,216,60,
C320  $       FILE    03,X3SD,50L
C321  $       OPTION  FORTRAN
C322  $       FORTRAN
C323  $       LIMITS  ,28K
C324  C
C325  C
C326          IMPLICIT INTEGER (A-Y)
C327          DIMENSION BASE(156,56),BUF(600),RECORD(156)
C328          DIMENSION IN(300)
C329          DATA RES/156/
C330          HFRES = RES/2
C331  C
C332          REWIND(01)
C333          REWIND(03)
C334  C
C335          READ(01) NUMR,NUMANG,WIDTH
C336  C
C337          HAFANG = NUMANG/2
C338  C
C339          DO 110 I=1,NUMANG
C340          READ(01) (IN(J),J=1,NUMR)
C341          ANG = I
C342          WORD= (ANG-1)/4 + 1
C343          BIT = MOD((ANG-1),4)*8
C344  C
C345          DO 105 J=1,NUMR
C346   105    FLD(BIT,8,BASE(J,WORD))= IN(J)
C347   110    CONTINUE
C348  C
C349   5      DO 500 I=1,RES
C350          NUM = 1
C351          READ(03) CNT
C352          IF(CNT .GT. 0) GOTO 7
C353          OTF = 1
C354          READ(03) CNT
C355   7      READ(03) (BUF(I9),I9=1,CNT)
C356          IF(CNT .EQ. 1) GOTO 400
C357          STRT = 1
C358          END = FLD(0,10,BUF(1))
```

```
C359          R = FLD(10,10,BUF(1))
C360          ANG = FLD(20,10,BUF(1))
C361   C
C362   C
C363    120   TH = HAFANG-ANG + 1
C364          WORD = (TH-1)/4 + 1
C365          BIT = MOD((TH-1),4)*8
C366          LEFT = FLD(BIT,8,BASE(R,WORD))
C367          TH = HAFANG + ANG
C368          WORD = (TH-1)/4 + 1
C369          BIT = MOD((TH-1),4)*8
C370          RGT = FLD(BIT,8,BASE(R,WORD))
C371   C
C372    121   IF(NUM .EQ. 1) RECORD(HFRES)=RGT
C373          DO 200 J=STRT,END
C374          J1=HFRES+J
C375          J2 = HFRES-J
C376          RECORD(J1)=RGT
C377    200   RECORD(J2)=LEFT
C378   C
C379          NUM = NUM+1
C380          IF(NUM .GT. CNT) GOTO 400
C381          STRT = END+1
C382          END = FLD(0,10,BUF(NUM))
C383          R = FLD(10,10,BUF(NUM))
C384          ANG = FLD(20,10,BUF(NUM))
C385          GOTO 120
C386    400   CONTINUE
C387   C
C388   C      **********  THIS SECTION WRITES TO IDECS  ********
C389   C
C390   C
C391   C
C392   C
C393          WRITE(02)(RECORD(I9),I9=1,RES)
C394          WRITE(6,23)(RECORD(I9)/26,I9=1,RES,2)
C395    23    FORMAT(1X,120I1)
C396   C
C397    450   DO 490 M=1,RES
C398    490   RECORD(M)=0
C399    500   CONTINUE
C400          WRITE(6,501)
C401    501   FORMAT(//,' THAT IS ALL FOLKS')
C402          GOTO 900
C403    900   STOP
C404          END
C405   $      EXECUTE
C406   $      LIMITS   10,20K,,2K
C407   $      TAPE     01,X1DD,,60724,,ABFAL1
C408   $      TAPE7    02,X2DD,,60710,,HELPRO,OUT,DEN5
C409   $      FILE     03,X3DD,50L
C410   $      ENDJOB
```

```
0001  C                    STEP1
0002  C
0003  C  THIS PROGRAM ACCEPTS DATA POINTS (RECTANGULAR FORMAT) FOR INPUT
0004  C  AND CREATES THE RESOLUTION CELL SIZE MATRIX IN POLAR COORDINAT
0005  C  TO BE USED AS DATA BASE FOR THE SIMULATION PROGRAM
0006  C
0007  C
0008          IMPLICIT INTEGER (A-Y)
0009          REAL ARCOS,FLOAT
0010          DIMENSION PRIOR(21,21),RECORD(3169),TABLE(1000),OT(4,250)
0011          DATA HALF,STRT/0,1/
0012          DATA CNT,NUMB/1,0/
0013          DATA OCT8,OCT2/01000000000,0100/
0014          DO 95 I=1,21
0015   95     READ(05,5) (PRIOR(I,J),J=1,21)
0016    5     FORMAT(21I3)
0017  C
0018  C  MIDX = DISTANCE (FEET) FROM LEFT EDGE OF DATA BASE TO TARGET CE
0019  C  MIDY = DISTANCE (FEET) FROM BOTTOM EDGE OF DATA BASE TO TARGET
0020  C  RADIUS = RADIUS (IN FEET) OF SIMULATION DESIRED <= MIN(MIDX,MID
0021  C  CELSIZ = SIZE (IN FEET) REPRESENTED BY DATA POINTS - ASSUMED SQ
0022  C  NUMPT = NUMBER OF DATA POINTS PER RECORD ON INPUT TAPE
0023  C  NUMREC = NUMBER OF RECORDS ON INPUT TAPE
0024  C         EACH RECORD GOES FROM SOUTH TO NORTH.  RECORDS ON TAPE IN
0025  C         A WEST TO EAST ORDER
0026  C  WIDTH = FIXED SIZE FOR RANGE RESOLUTION
0027  C  ZBMWD = BEAMWIDTH (IN RADIANS)
0028  C
0029  C
0030  C  INPUT PARAMETERS USED FOR PICKWICK (9/14/76)
0031  C  31690  31690  26400  20  3169  31 9  200  .0175
0032  C
0033  C  ************MODIFICATION**************(10/14/76)
0034  C
0035  C  ONE HAS THE OPTION TO MAKE RANGE RESOLUTION FIXED
0036  C  LET WIDTH = DESIRED RESOLUTION
0037  C  OR TO HAVE RANGE RESOLUTION VARY WITH RANGE
0038  C  LET WIDTH = 0 AND INPUT VALUE FOR PULSEWIDTH
0039  C  (FACTOR OF E-06 ASSUMED -- SO INPUT PARAMETER WILL LIKELY
0040  C  BE BETWEEN .1 AND 2.)
0041  C
0042  C  INPUT PARAMETERS FOR PICKWICK (10/14/76)
0043  C  31690  31690  26400  20  3169  3169 0  4000  .0175  .25
0044  C
0045  C
0046          READ(05,10) MIDX,MIDY,RADIUS,CELSIZ,NUMPT,NUMREC,WIDTH,
0047        &  ALT,ZBMWD,ZPULS
0048   10     FORMAT(8I6,2F8.5)
0049  C
0050  C  QUARTER = FLAG TO DO ONLY ONE QUADRANT OF DATA BASE >0 YES
0051  C         <=C DO FULL 360
0052  C
0053          READ(05,15) QUARTER
0054   15     FORMAT(I2)
0055  C
0056          ALT2 = ALT*ALT
0057  C
0058  C
```

```
C   NUMR - MAXIMUM NUMBER OF CELLS IN RANGE DIRECTION IN
C          RESOLUTION CELL MATRIX BEING CONSTRUCTED
C
          IF(MIDX .GT. RADIUS .AND. MIDY .GT. RADIUS) GOTO 80
          WRITE(6,62) RADIUS,MIDX,MIDY
 62       FORMAT('  WARNING - DATA BASE TOO SMALL FOR DESIRED
     &      SCENE, LARGEST CIRCLE POSSIBLE WILL BE SIMULATED',//,
     &      ' INPUT PARAMETERS WERE RADIUS,MIDX,MIDY=',3I8)
C
          IF(MIDX .LT. RADIUS) RADIUS = MIDX
          IF(MIDY .LT. RADIUS) RADIUS = MIDY
C
 80       IF(WIDTH .EQ. 0) GOTO 85
          NUMR = RADIUS/WIDTH
          ZCW = FLOAT(CELSIZ)/FLOAT(WIDTH)
          GOTO 86
 85       ZCTAU = 983.57*ZPULS/2.
          MXSR = SQRT(RADIUS**2 + ALT**2)
          NUMR = FLOAT(MXSR - ALT)/ZCTAU
C
C   NUMANG - NUMBER OF ANGLE BINS TO BE CREATED IN RESOLUTION
C            CELL MATRIX
C
 86       N90 = 1.57/ZBMWD + 1
          N180 = N90*2 + 1
          NUMANG = N90*4
          IF(NUMANG .GT. 720) GOTO 815
C
C   X AND Y COORDINATES OF CENTER OF DATA BASE
C
          CENTRX = MIDX/CELSIZ
          CENTRY = MIDY/CELSIZ
C
C   RAD - NUMBER OF DATA CELLS FROM CENTER TO EDGE OF
C         SIMULATION AREA
C
          RAD = RADIUS/CELSIZ
          WRITE(6,73) NUMR,NUMANG,CENTRX,CENTRY,RAD
 73       FORMAT(//' INITIAL PARAMETERS ',5I8//)
          IF(RAD .GT. CENTRX .OR. RAD .GT. CENTRY) GOTO 810
C
C   TABLE = TABLE LOOK UP FOR ANGLE
C           USE 1000 TIMES COSINE OF ANGLE AS INDEX
C           RESULT IS PROPER ANGLE BIN FOR THE POINT
C
          DO 100 I=1,1000
 100      TABLE(I)=ARCOS(FLOAT(I)/1000.)/ZBMWD + 1
C
C   WRITE PARAMETERS TO TAPE FOR DATA TO NEXT STEP
C
          WRITE(02) NUMR,NUMANG,RADIUS,WIDTH,N90,ALT,ZCTAU
C
C
C   BEGIN - FIRST LINE OF DATA BASE TO BE USED IN THIS
C           SIMULATION
C   IN CASE ONE WISHES TO SIMULATE ONLY A SEGMENT OF THE
C   ENTIRE DATA BASE SOME LINES OF THE DATA BASE WILL BE
C   UNUSED.  THIS LOOP POSITIONS THE USER AT THE FIRST LINE
C   OF THE INPUT WHICH IS TO BE USED
C
```

```
C119          BEGIN = CENTRX - RAD
0120          IF(BEGIN .EQ. 0) GOTO 115
0121          DO 110 I=1,BEGIN
C122   110    READ(01)
C123   115    BEGIN = BEGIN+1
0124          WRITE(6,67) BEGIN
0125   67     FORMAT(' BEGIN =',I5//)
0126  C
0127  C
C128  C
0129          DO 200 I=BEGIN,NUMREC
0130  C
0131  C  READ IN NEW LINE OF INPUT
0132  C
0133          READ(01,END=800) XX,(RECORD(N),N=1,NUMPT)
0134  C
0135  C  NX - DISTANCE (IN NUMBER OF CELLS) IN X DIRECTION FROM
C136  C  THE CENTER TO THE CURRENT LINE OF INPUT
0137  C
C138          NX = I-CENTRX
0139          IF( NX .EQ. 0) NX = 1
C140          IF(NX .GE. 0) HALF = 1
0141          IF(NX .GE. 0 .AND. OTF .EQ. 0) GOTO 600
0142   105    IF(QUARTER .GT. 0 .AND. NX .GE. 0) GOTO 500
C143          IF(NX .GT. CENTRX) GOTO 500
0144          NX2 = NX * NX
C145          ZX = ABS(NX)
0146  C
C147  C
0148  C  PLACE EACH POINT OF THE CURRENT INPUT LINE INTO THE
C149  C  APPROPRIATE CELL OF THE RESOLUTION CELL MATRIX BEING
0150  C  CREATED
0151  C  EACH PASS THROUGH THIS LOOP PROCESSES TWO POINTS OF THE
0152  C  LINE -- THE ONE J CELLS ABOVE THE CENTER LINE
0153  C  AND THE ONE J CELLS BELOW THE CENTER LINE
0154  C
0155          DO 180 JJ=1,RAD+1
0156          J = JJ-1
0157  C       IF( QUARTER .GT. 0 .AND. J .GT. IABS(NX)) GOTO 200
C158          NY2 = J*J
0159          ZR = SQRT(NX2 +NY2)
C160  C
C161  C  R - DISTANCE (IN NUMBER OF CELLS) FROM CENTER POINT TO
0162  C       THE CURRENT POINT
0163  C
C164          IF(WIDTH .EQ. 0) GOTO 113
0165          R = ZR*ZCW + 1.
0166          GOTO 313
C167   113    SR = SQRT(ALT2 + (ZR*CELSIZ)**2) - ALT
0168          IF(SR .LT. 0) GOTO 180
0169          R = SR/ZCTAU + 1.
C170   313    IF(R .GT. NUMR) GOTO 200
0171          COSANG = ZX/ZR * 1000.
0172  C
0173          IF(COSANG .LT. 0) WRITE(6,69) COSANG,I,NX,J,R
C174          IF(COSANG .GT. 1000) WRITE(6,69) COSANG,I,NX,J,R
C175   69     FORMAT(//' **ERROR - COS > 1',5I8/)
0176          IF(COSANG .GT. 10 .AND. COSANG .LT. 990) GOTO 117
0177          ANG= ARCOS(ZX/ZR)/ZBMWD + 1
C178          GOTO 118
```

```
C
C ANG - APPROPRIATE ANGLE BIN FOR THE CURRENT POINT
C
 117    ANG = TABLE(COSANG)
C
C  HALF=1  IMPLIES RIGHT HALF OF THE SCENE IS BEING PROCESSED
C  SO THE VARIABLE ANG IS MODIFIED APPROPRIATELY
C
 118    CONTINUE
C
 120    INDEX = CENTRY + J
        IF(OLDR .NE. R .OR. OLDANG .NE. ANG) GOTO 150
        CNT = CNT + 1
        CAT=FLD(30,6,RECORD(INDEX)) +1
        TCAT = PRIOR(TCAT,CAT)
        TELV = TELV + RECORD(INDEX)/OCT2
C
        INDEX = CENTRY - J
        CAT=FLD(30,6,RECORD(INDEX)) +1
        BCAT = PRIOR(BCAT,CAT)
        BELV = BELV + RECORD(INDEX)/OCT2
        GOTO 180
C
C
 150    TELV = TELV/CNT * OCT2 + TCAT
        BELV = BELV/CNT * OCT2 + BCAT
        NUMB=NUMB+1
        OT(1,NUMB)=OLDR
        OT(2,NUMB)=OLDANG
        OT(3,NUMB)=TELV
        OT(4,NUMB)=BELV
        OLDR = R
        OLDANG = ANG
        IF(NUMB .LT. 250) GOTO 157
        WRITE(02) NUMB,OT
        NUMB=0
 157    SUM = SUM+1
        TELV = RECORD(CENTRY +J)/OCT2
        BELV = RECORD(CENTRY -J)/OCT2
        TCAT = FLD(30,6,RECORD(CENTRY+J)) + 1
        BCAT = FLD(30,6,RECORD(CENTRY-J)) + 1
        TOTCNT = TOTCNT + CNT
        CNT = 1
C
 180    CONTINUE
 200    CONTINUE
        GOTO 900
C
C
C  ERROR MESSAGES
C
 800    WRITE(6,801) I
 801    FORMAT(//'  UNEXPECTED END OF FILE AT RECORD ',I6)
        GOTO 900
 810    WRITE(6,811) RADIUS,MIDX,MIDY
 811    FORMAT(//,'  ***ERROR   RADIUS EXCEEDS DATA BASE
     1    SIZE - RAD=',I8,'  X=',I8,'  Y=',I8)
        GOTO 900
 815    WRITE(6,816) NUMR,RADIUS,WIDTH
 816    FORMAT(//,' ***ERROR - SIZE EXCEEDS DIMENSIONS OF ARRAY
```

```
      1  (MAX=160) RANGE CELLS=',I5,'RADIUS=',I8,' WID',I8)
         GOTO 900
C
C  WRITE OUT DATA BASE MATRIX FOR
C  VERIFICATION
C
 600     OTF = 1
         WRITE(02) NUMB,OT
         ENDFILE(02)
         NUMB = 0
         GOTO 105
 500     CONTINUE
 900      WRITE(02) NUMB,OT
         WRITE(6,901) SUM,TOTCNT
 901     FORMAT(10X,'***DONE***',I8,'  RECORDS WRITTEN',
       &    10X,I8,' POINTS PROCESSED')
         STOP
         END
$        EXECUTE
$        MSG2     2,MOUNT 61062,,SORTX2 AS CONT REEL FOR 61061
$        LIMITS   40,20K,,5K
$        TAPE     01,,,61061,,SORTX1
$        FFILE    01,BUFSIZ/3175
$        TAPE     02,,,61084,,DIC014,OUT
$        FFILE    02,BUFSIZ/2100
$        ENDJOB
```

```
0001  C                          STEP10
0002  C
0003  C   PART TWO OF  DATA BASE CREATION
0004  C
0005          IMPLICIT INTEGER (A-Y)
0006          DIMENSION A(265,181),PRIOR(21,21),DAT(4,250)
0007          DIMENSION OT(265),C1(265,31),C2(265,31)
0008          DATA OCT2,SHIFT,CTR/0100,0100000,010000000000/
0009  C
0010          OTF = 0
0011  C
0012          READ(01) NUMR,NUMANG,RADIUS,WIDTH,N90,ALT,ZCTAU
0013          WRITE(02) NUMR,NUMANG,RADIUS,WIDTH,ALT,ZCTAU
0014  C
0015          DO 110 I=1,21
0016   110    READ(05,321) (PRIOR(I,J),J=1,21)
0017   321    FORMAT(21I3)
0018  C
0019  C
0020   5      READ(01,END=500) NUMB,DAT
0021  C
0022          DO 200 I=1,NUMB
0023          R = DAT(1,I)
0024          ANG = DAT(2,I)
0025          CAT1=FLD(30,6,DAT(3,I))
0026          ELV1=DAT(3,I)/OCT2
0027          CAT2=FLD(30,6,DAT(4,I))
0028          ELV2=DAT(4,I)/OCT2
0029          A(R,ANG)=ELV1*SHIFT + ELV2 + CTR + A(R,ANG)
0030          WORD=ANG/6 + 1
0031          BIT = MOD(ANG,6)*6
0032          TAG = FLD(BIT,6,C1(R,WORD))
0033         IF(TAG .EQ. 0) TAG = 1
0034          IF(PRIOR(TAG,CAT1) .EQ. CAT1) FLD(BIT,6,C1(R,WORD))=CAT1
0035          TAG = FLD(BIT,6,C2(R,WORD))
0036          IF(TAG .EQ. 0) TAG = 1
0037          IF(PRIOR(TAG,CAT2) .EQ. CAT2) FLD(BIT,6,C2(R,WORD))=CAT2
0038   200      CONTINUE
0039          GOTO 5
0040  C
0041  C
0042   5      READ(01,END=500) NUMB,DAT
0043  C
0044          DO 292 I=1,NUMB
0045          R = DAT(1,I)
0046          ANG = DAT(2,I)
0047          CAT2=FLD(30,6,DAT(3,I))
0048          ELV2=DAT(3,I)/OCT2
0049          CAT1=FLD(30,6,DAT(4,I))
0050          ELV1=DAT(4,I)/OCT2
0051          A(R,ANG)=ELV1*SHIFT + ELV2 + CTR + A(R,ANG)
0052          WORD=ANG/6 + 1
0053          BIT = MOD(ANG,6)*6
0054          TAG = FLD(BIT,6,C1(R,WORD))
0055         IF(TAG .EQ. 0) TAG = 1
0056          IF(PRIOR(TAG,CAT1) .EQ. CAT1) FLD(BIT,6,C1(R,WORD))=CAT1
0057          TAG = FLD(BIT,6,C2(R,WORD))
0058          IF(TAG .EQ. 0) TAG = 1
```

```
0059          IF(PRIOR(TAC,CAT2) .EQ. CAT2) FLD(BIT,6,C2(R,WORD))=CAT2
0060    292     CONTINUE
0061          GOTO 6
0062  C
0063  C
0064    500   CONTINUE
0065          DO 400 II=1,N90
0066          I= N90 + 1 - II
0067          WORD=I/6 + 1
0068          BIT = MOD(I,6)*6
0069  C
0070          DO 300 J=1,NUMR
0071          MULT = FLD(0,6,A(J,I))
0072          IF(MULT .EQ. 0) GOTO 250
0073          ELV1 = FLD(6,15,A(J,I))/MULT
0074          ELV2 = FLD(21,15,A(J,I))/MULT
0075          GOTO 251
0076    250     ELV1=0
0077          ELV2=0
0078  C
0079    251   OT(J)=ELV2*OCT2 + FLD(BIT,6,C2(J,WORD))
0080          A(J,I)=ELV1 * OCT2 + FLD(BIT,6,C1(J,WORD))
0081    300   CONTINUE
0082          WRITE(02) OT
0083          IF(I .LT.4) WRITE(6,21) (OT(K),K=1,NUMR,3)
0084     21   FORMAT(17(1X,O6))
0085    400   CONTINUE
0086  C
0087          DO 420 I=1,N90
0088          WRITE(02) (A(J,I),J=1,NUMR)
0089          IF(I .LT. 4) WRITE(6,21) (A(J,I),J=1,NUMR,3)
0090          DO 410 J=1,NUMR
0091    410     A(J,I)=0
0092    420   CONTINUE
0093  C
0094          DO 220 L=1,NUMR
0095          DO 220 L2=1,31
0096          C1(L,L2)=0
0097    220   C2(L,L2)=0
0098          OTF = 1 + OTF
0099          CALL FCLOSE(01)
0100          IF(OTF .LT. 2) GOTO 6
0101          STOP
0102          END
0103  $       XEQUTE
0104  $       LIMITS   05,77K,,5K
0105  $       FILE     01,BUFSIZ/2100
0106  $       FILE     02,BUFSIZ/1600
0107  $       TAPE     01,X100,,61084,,DIC014
0108  $       TAPE     02,X200,,62012,,AAA444,OUT
0109  $       NOJOB
```

APPENDIX K

INVESTIGATION OF AN INTERACTIVE APPROACH FOR RADAR IMAGE SIMULATION

The following technical report (TR 319-15) prepared by the Center for Research, Inc., University of Kansas, is included in this volume to provide details in support of the technical discussions of Volume I.

THE UNIVERSITY OF KANSAS CENTER FOR RESEARCH, INC.

2291 Irving Hill Drive Campus West
Lawrence, Kansas 66045

Telephone: 913-8644832

INVESTIGATION OF AN INTERACTIVE APPROACH FOR RADAR IMAGE SIMULATION

Remote Sensing Laboratory
RSL Technical Report 319-15

V. S. Frost
J. L. Abbott
V. H. Kaupp
J. C. Holtzman

March, 1977

Supported by:

U. S. Army Engineer Topographic Laboratories

Fort Belvoir, Virginia 22060

CONTRACT DAAG 53-76-C-0154

ABSTRACT

The rationale for developing interactive feature extraction techniques is presented. These techniques will have a great impact on three major problem areas, data base construction, hybrid simulation, and the updating of the simulated products, encountered in the simulation process. The concept of interactive feature extraction is also presented. The essential ingredient of this interactive concept is the optimization of man-machine communications, i.e., the operational system would be designed so that an image interpreter, a non-computer expert, would use the computer as another interpretation tool, and thus become as familiar with it as he is with the stereoscope and proportional dividers.

1.0 BACKGROUND

The demand for a cost effective radar image simulation has been steadily increasing over the past several years. The impetus for developing such a capability lies in the exorbitant costs involved in building and operating modern imaging radars and in the growing interest in using simulated radar images as reference scenes in remotely piloted vehicles (RPV's). The simulation of imaging radar systems involves transforming the known geometric and microwave scattering characteristics of the target scene into the desired final product. The terrestrial envelope can generally be divided into two basic classes of microwave scattering categories, cultural and distributed, each class of targets presenting particular simulation problems. Thus, a different model has been developed to efficiently process each class of scatterers. The closed system approach, i.e., the target scene, radar transceiver and image medium are treated as a single closed system, was used to develop the point scattering model for distributed targets[1] and the area spatial filtering model for cultural targets[2]. The resulting simulation techniques can be combined to form a hybrid simulation[3] which is capable of efficiently simulating all types of target scenes.

A radar image simulation requires as input data the geometric characteristics, planimetric (category), and elevation data of the target scene, in addition to the microwave scattering properties for each category. This information is called the data base and its construction represents the single most expensive component of the simulation process. The data bases for the point scattering method are typically constructed from aerial photography, topographic maps, and other available information concerning the target scene[4]. The microwave scattering properties for the

---

[1] Holtzman, J. C., V. H. Kaupp, R. L. Martin, E. E. Komp, and V. S. Frost, "Radar Image Simulation Project: Development of a General Simulation Model and an Interactive Simulation Model, and Sample Results," TR 234-13, Remote Sensing Laboratory, The Univeristy of Kansas, February, 1976.

[2] Frost, V. S., J. L. Abbott, V. H. Kaupp, and J. C. Holtzman, "A Mathematical Model for a Terrain-Imaging Radar and Its Potential Application to Radar Image Simulation," TR 319-6, Remote Sensing Laboratory, The University of Kansas, November, 1976.

[3] Frost, V. S., J. L. Abbott, V. H. Kaupp, and J. C. Holtzman, "Hybrid Approach to Radar Image Simulation," TR 319-13, Remote Sensing Laboratory, The University of Kansas, March, 1977.

[4] Davison, E., V. H. Kaupp, and J. C. Holtzman, "Baseline of Planimetric Data Base Construction: Pickwick Site," TR 319-2, Remote Sensing Laboratory, The University of Kansas, July, 1976.

categories contained in the target scene are derived from either empirical or theoretical studies. A data base for one particular site, the Pickwick Dam area, has been constructed by a photointerpreter using manual feature extraction techniques. Even though a relatively small area (28 square miles) was considered, the data base still contained over 1.8 million points. The many problems encountered and the effort involved in constructing this single data base are reported in (5). This experience has taught us that manned feature extraction techniques for data base construction are not cost-effective. The effort and time involved in transforming the raw data, the aerial photographs, maps, etc., into a suitable radar simulation data base proved to be excessive. To illustrate this, the process will be briefly restated; the process started with preliminary data sources being gathered for analysis - maps, aerial photography, radar imagery, etc. Then all the relevant information would be transferred from the primary sources to an intermediate map, using only manual feature extraction techniques to extract the useful information. Next the map would be digitized, again using only manual digitizing techniques. Finally, the digitized output would undergo a series of manipulations in several computer programs which would finally create the required data base. This process commonly exceeded one man month of effort and even more waiting and delay time. Thus, it can be easily seen that if a large number of simulations for a wide variety of sites were needed on short notice, the construction of the simulation data bases presently would prove prohibitive to the simulation process. Then to enable these kinds of operations to become feasible a new approach is needed for data base construction.

For many important simulation applications, especially for RPV's, the simulated products need to be updated according to seasonal and meteorological variations, and other natural and man-made occurrences. These factors will cause the radar return from the terrain to change appreciably and therefore must be included if the simulation is to be successful as a reference scene in guiding RPV's. Obviously it is not practical to totally regenerate the data base each time the conditions of the target scene change, therefore, some efficient technique needs to be developed to enable the simulation products to be updated rapidly as conditions of the target scene vary.

---

[5] Komp, E., M. McNeil, V. H. Kaupp, and J. C. Holtzman, "Medium Resolution Digital Ground Truth Data Base," TR 319-5, Remote Sensing Laboratory, The University of Kansas, August, 1977.

If a wide variety of target scenes are to be simulated the hybrid technique is seen as the most effective route[3]. But recall to produce a hybrid simulation the image products of two different simulation techniques (the point scattering and area spatial filtering) need to be combined. This combination is a nontrivial task because it requires congruencing the image products from topographic maps and aerial photography, in addition to the removal of unwanted noise[6]. The difficulties associated with producing hybrid simulations will require the development of new processes.

Thus, there are three basic areas of radar image simulation in which there exist significant problems: (1) The creation of a hybrid radar image simulation; (2) The updating of the simulation products factoring in seasonal and meteorological changes; (3) The cost-effective construction of radar image simulation data bases. In each of these problem areas either totally manual or totally automated techniques can be employed as a solution. But as in the case with data base construction, manual methods are not cost-effective because of the lag time, extensive effort and the numerous errors involved. It has also been found that the state of the art in totally automated (computer) methods currently prohibits their feasibility even though they will play an ever increasing role as the technology advances[7]. The computer is a very useful tool for rapid manipulation and storage of vast amounts of data but its decision-making capabilities are very limited, i.e., the computer is not able to guide itself through the entire process of feature extraction. On the other hand, a human being is very well suited to make the necessary decisions while being very inefficient in dealing with large quantities of data. Therefore, the optimum solution in all three problem areas would be to combine the strengths of both the human and computer in an automated interactive feature extraction system.

---

[3] Frost, V. S., J. L. Abbott, V. H. Kaupp, and J. C. Holtzman, "Hybrid Approach to Radar Image Simulation," TR 319-13, Remote Sensing Laboratory, The University of Kansas, March, 1977.

[6] Frost, V. S., J. L. Abbott, V. H. Kaupp, and J. C. Holtzman, "An Alternative Approach for the Simulation of Cultural Targets in the Pickwick Area," TR 319-12, Remote Sensing Laboratory, The University of Kansas, March, 1977.

[7] Janza, Frank J., Remote Sensing Manual, (Editor - Reeves), Chapter 12, Page 79, The American Society of Photogrammetry, 1975.

2.0 THE INTERACTIVE FEATURE EXTRACTION CAPABILITIES

An automated interactive feature extraction system would use the computer for data manipulation, storage, pre-processing, image enhancement, image display, and otherwise aid or provide a tool for the interpreter as he performs his function. Viewed another way, the human is utilized to make decisions and guide the computer in a real time interaction. This interaction can be accomplished by providing the interpreter/operator with a few basic tools with which to communicate his decisions to the computer; a custom keyboard for the most common commands (e.g., display an image, store an image, image transforms) and a joystick for direct specification of image coordinates are probably the minimum to be provided. In effect, the computer would be performing several very important functions to assist the operator in achieving the desired goal whether that goal is data base construction, update or hybrid simulation. The computer's first function would be to pre-process the raw data (aerial photographs, maps, etc.), that is, transform the input information into a form suitable for digital and interpreter analysis. Pre-processing could consist of simply performing an analog to digital conversion and displaying the result of complex operations designed to separate the input image into its homogeneous regions. Second, the interpreter designated operations would be performed and the results stored in a format to allow for easy access. For example, suppose a gradient operation was valuable to identify boundaries in aerial photography; the interpreter would select this operation either through a simple command or a standard keyboard or pressing a simple switch on the custom keyboard, the gradient operation would then be applied and the results stored possibly in an image stack similar to that found in pocket calculators. The final task for the computer would be the bookkeeping, i.e., keeping track of all images created, processed, or altered with a complete record of the past operations performed on that image or file.

The routine library and accounting functions required by the interpreter could be handled by a computer. All of the standard report forms for interpretation could be accessed on a CRT terminal where the operator

would select the required form and begin answering questions. When finished, he would indicate it on the keyboard and the report would be printed. In an operational setup the report might be printed in a different location via a data link.

Image accounting functions would be handled in much the same fashion. With a light pen, joystick, or framer, etc., the coordinates on an image would be identified with respect to longitude and latitude or in terms of key features such as mountain peaks, river forks, bridges or cross roads. The machine would then file the coordinates, or statute miles from the key features, of any item tagged with a light pen or cursor.

Several purposes are served by this scheme: feature extraction for report purposes is simplified; and, since all the key elements in an image can be quickly fixed, an entire base map can be rapidly established. A coded legend of mapping terms would permit the operator to not only trace out the outline features, but also distinguish between roads, trails, streams, etc.

Huge files of base data of geographic areas actually can be structured to occupy a relatively small amount of memory. A six bit word can specify any of 64 categories and the high spatial correlation of the data in most maps makes them very conducive to compression techniques for economical storage. Higher levels of information could be contained in a cross referenced file scheme where specific information would be accessed several ways. The accumulated statistics about a city, for example, could be retrieved by giving the city name, its geographic coordinates, or the intersection of two highways. Large cities would have multiple levels of information available while smaller areas might contain all the facts in one table. Emphasis could be placed on the speed of operation and minimizing the effort so as to not distract the interpreter from his primary task. When multiple levels or classes of information are available, the analyst could be presented a list of categories and select with a light pen those he wishes to see.

The data files would be open ended so the analyst can update their contents to account for seasonal and meteorological variations. In addition, other physical changes in the target scene due to man-made

occurences (e.g., building of small dams, movement of large encampments of troops, or movement of munitions) could be factored into updated simulations easily through the use of this structure. The interpreter would also be able to develop his own private "scratch pad" associated with a specific target area. This digital notebook would contain his thoughts and observations which might be used to refresh his memory as to specific items to be examined later.

Written aids which help guide an interpreter through specific identification and extraction tasks are called interpretation keys and the entire subject can be handily managed by a computer. The most time efficient method of implementing interpretation keys, especially dichotomous keys, is with a computer. It is envisioned that a key word index would access the right key and an interactive mode with the computer would speed the interpreter through it. A file key is a private collection of representative imagery gathered by the interpreter to aid in identifying specific items. A typical example is a collection of photographs of different forest types. File keys tend to be personal and it is envisioned that the interpreter would have his own magnetic tapes or disks for such information. Since this information might be in the form of digitized high resolution photography, a large file might dictate a mechanical handling scheme for accessing and projecting actual photographs. General purpose image index keys which would be employed by many users warrant the cost of memory overhead to store them. A disk pack or drum memory could be devoted to an image index. This interactive approach is very different from others because it would be designed to be operated by a radar image interpreter, who is not necessarily a computer programmer.

An interactive system as described above would have a significant impact on the construction of data bases for radar image simulation. Many of the time consuming, routine, manual feature extraction tasks could be automated such that the interaction between the man and computer would result in the optimum cost-effective product. For example, there are several computer routines available which identify homogeneous

regions within a scene.[1] These programs are more or less effective depending on the scene. The interpreter would thus be saved from outlining the homogeneous regions of interest though he would still be required to categorize each of the identified regions because the computer still lacks the capability. He would also be required to add any regions not detected by the computer routines. A further savings is realized in that the result of this interactive processing is directly a digital terrain map which is also the desired final product of the manual techniques[5]. These kinds of interactive techniques could also be applied to the construction of the elevation data base.

Another important area in which interactive techniques could provide an appropriate solution is the updating of the simulation products to reflect environmental changes, e.g., both seasonal and meteorological variations over the scenes of interest. An updated version of an image simulation can be produced interactively without a second simulation of the entire area. This can be done by allowing the interpreter to select and communicate to the computer the regions in the target scene that are to be changed through the use of a joystick for example. A preliminary attempt at such a system is reported in Reference [1]. The map of homogeneous areas would be used in updating the simulation by the recatorization of the regions affected by the environmental changes. Only these specific areas would go through another simulation process. Thus a significant savings can be realized by not having to (1) totally regenerate the data base, nor to (2) resimulate the entire area. This application for an interactive system is essential to the success of an RPV because the microwave reflectivity may be radically affected by certain environmental factors, and thus, these effects must be incorporated into the reference scenes.

An interactive approach will also prove to be a valuable part of a hybrid simulation package. The advantages of the hybrid simulation

---

[1] Holtzman, J. C., V. H. Kaupp, R. L. Martin, E. E. Komp, and V. S. Frost, "Radar Image Simulation Project: Development of a General Simulation Model and an Interactive Simulation Model, and Sample Results," TR 234-13, Remote Sensing Laboratory, The University of Kansas, February, 1976.

[5] Komp, E., M. McNeil, V. H. Kaupp, and J. C. Holtzman, "Medium Resolution Digital Ground Truth Data Base," TR 319-5, Remote Sensing Laboratory, The University of Kansas, August, 1977.

are (1) the capability of handling all types of targets; (2) reducing the cost of data base construction; and (3) greatly increasing the flexibility of the simulation. The interactive route would provide a solution to several of the problems which will be created by congruencing the products of two different simulation methods, this operation would be greatly simplified by the man-machine communication which is the essential ingredient in an interactive concept. Another problem which will be faced is the removal of unwanted "noise" from the intermediate products. The noise could consist of both redundant and unwanted information, in addition to other types of noise introduced in the simulation and display process. The decisions involved in this activity, especially the removal of the unwanted information, will have to be made by the interpreter, while the results of the decision will necessitate the manipulation of large quantities of data. This must be handled by the computer. The interactive concept will provide an eloquent solution.

3.0 THE INTERACTIVE CONCEPT

The interactive concept centers around the optimization of man-machine communication. That is the interpreter using his many years of experience would guide the computer processing of the raw data and intermediate results to produce the desired product. However, most interpreters do not speak the particular jargon or have the necessary expertise to perform easily in the computer domain. Therefore it is essential that the interactive concept be implemented in a way that would enable a typical image interpreter to function comfortably without extensive retraining, thus, maximizing the system performance. Both the hardware and software of an operational system must be designed with this goal in mind. This naturally leads us to a special purpose station as the one described by Currier[8].

The major obstacle which must be overcome before this concept can be transformed into a reality is the apportioning between the man and machine the desired tasks. This is further complicated by the fact that digital technology is steadily improving, reducing to practice operations and functions which are not presently feasible. This realization will be factored into the design by forcing its structure to be very flexible. The optimum distribution of tasks between man and computer is not known. But, it is known that the interpreter operates efficiently on a macro-level, i.e., identifying macro-phenomena within the context of the required operations like classifying terrain types, and thus he would be required to make the higher level decisions. On the other hand, the computer operates very efficiently on a micro-level, e.g., calculating image statistics and image manipulations, and would be called upon to perform those tasks. Each of these observations are important concentrations in specifying the final form of an interactive system.

---

[8] Currier, P. L., "Mithra Specifications: A Unified Approach to Image Processing Hardware," University of Kansas, D. E. Thesis, 1976.

4.0 CONCLUSIONS

The interactive concept is viewed as the most promising general solution to the problems associated with radar image simulation. This approach would be invaluable in reducing the cost of data base construction which is presently exorbitant when manual techniques are used. The interactive concept would provide an appropriate mechanism for updating the radar image simulation data bases to account for environmental changes. Also the interactive concept would assist in the production of hybrid radar image simulations.

The interactive concept puts the interpreter at the center of the decision making processes, using his many years of experience. While the computer would provide the interpreter with the necessary information, along with storage and manipulation of the results. The computer would provide the background information through preprocessing the raw data; this preprocessing would transform the raw data into a form which is suitable for analysis. Also the computer would be the bookkeeper for the interpreter, storing interim results, interpretation keys, etc. Further the machine would act on the decisions of the interpreter by performing the desired processing and displaying and storing the results in a convenient format. All of these operations would be accomplished within the framework of a interpreter-oriented processing station.

There are several problems which need to be answered before the interactive concept can become a reality. Among these, the most important theoretical question is the determination of the optimum relationship between man and machine. There are many engineering problems involved, such as the choice of the computer hardware, whether analog or optical processing would be as cost-effective in performing some of the desired functions. There is necessary the design of a software structure which would be strongly interpreter oriented. There are many image processing software packages available, but few, if any, exhibit the necessary characteristics. Thus, there are many problems to be solved before the interactive concept is to be implemented. But these difficulties do not outweigh the many advantages which would be realized

by an interactive approach. The approach would optimize the available capabilities, expertise, and resources, yielding a cost-effective solution to many of the problems encountered in radar image simulation.

APPENDIX L

AUTOMATED TECHNIQUES IN FEATURE EXTRACTION

The following technical report (TR 319-24) prepared by the Center for Research, Inc., University of Kansas, is included in this report to provide details for appropriate discussions in Volume I.

THE UNIVERSITY OF KANSAS CENTER FOR RESEARCH, INC.

2291 Irving Hill Drive--Campus West
Lawrence, Kansas 66045

Telephone: 913-8644832

AUTOMATED TECHNIQUES IN FEATURE EXTRACTION

Remote Sensing Laboratory
RSL Technical Report 319-15

M. McNeil

V. H. Kaupp

J. C. Holtzman

March, 1977

Supported by:

U. S. Army Engineer Topographic Laboratories

Fort Belvoir, Virginia 22060

CONTRACT DAAG 53-76-C-0154

ABSTRACT

A study of automated techniques for use in an interactive feature extraction system has been initiated. The purpose of this study was to survey the present automated techniques for feature extraction and assess their value as possible inclusions into the interactive feature extraction system. Particular attention was paid to techniques that lend themselves to interactive capability and the extraction of features in an image or aid in the interpretation of images. Several techniques are anticipated to prove especially useful in the actual extraction of features automatically. The most favorable at present seem to be the clustering algorithms and edge-finding routines.

1.0 INTRODUCTION

Feature extraction and image analysis covers a broad spectrum of applications, but in general the problem is one of simplification. The general problem in image analysis is to transform an image, represented by tens of thousands of bits at least, to a simplified version of the image, represented by only tens or hundreds of bits. Thus, the techniques which are of interest are those that simplify the image by suppressing irrelevant detail, integrating portions of an image into meaningful entities, describing regions and features, and in general reducing the complexity of the data. A definition by Patrick[1] is: "Feature extraction is the reduction of a set of measurements containing a relatively large amount of data but a smaller amount of useful information to a set containing a relatively small amount of data (features)." Clearly, the problem to be faced is data simplification in an image.

What is an image? An image is a set of data that represents a picture. Images are normally perceived visually. But how do we select a representation of an image or a picture so that it is suitable to the computer, which is where we want to automate the feature extraction process? Almost always this is done by visualizing the physical picture as occupying a plane defined by orthogonal x and y components. Then a picture function[2] $f(x,y)$ is defined as proportional to the light intensity on the picture at a point (x,y). This intensity is also called the brightness or grey level. Thus a picture can be represented by a real-valued continuous function of two variables in the x-y plane. In order to fit this representation to the computer, the picture function must be discretely sampled, or quantized. This is usually accomplished by partitioning the x-y picture plane into a quadruled grid and sampling

---

[1] Patrick, Edward A., Fundamentals of Pattern Recognition, Prentice-Hall, Inc., Englewood Cliffs, N. J., 1972.

[2] Duda, Richard O. and Peter E. Hart, Pattern Classification and Scene Analysis, John Wiley and Sons, Inc., New York, 1973.

the picture function at each intersected point, although sometimes a hexagonal grid is used because of the property that any two cells sharing a common vertex also share a common side, a property useful in some picture processing operations which does not hold in a quaduled grid. Also, since the computer can only record a finite number of variations in grey level, the range of picture intensities must also be quantized to reduce the infinite spectrum of black and white to a finite partition representing the grey level from dark (minimum) to light (maximum). In answer to the question of what is an image, it is a picture which is represented in the computer by a matrix whose entries represent the grey level of the picture at the specified coordinates. Now we are interested in automated techniques that operate on images that will be useful in either automatically extracting features or aiding an image interpreter in the extraction of features.

2.0 TEMPLATE MATCHING

Before a feature or object can be extracted, it must first be detected. The classical pattern recognition technique used to detect the presence of a previously specified object of a known shape is called template matching. Given a priori knowledge of the shape and size of a particular object to be detected, a template is chosen and matched against the picture for every possible position and orientation of the object. If a perfect match occurs, then the object (or one occurrence of the object) has been found. However, in practice it is hardly ever the case that a perfect match will occur, so what is used is a relative measure of how well the object being tested matches the template. If that measure is high enough the object is tagged as being a possible match. Several mathematically rigorous measures have been defined for this use, and are described below:

$f(x,y)$ is the grey level of the picture at point (x,y)
$t(x,y)$ is the grey level of the template at point (x,y)
T is the set of all points (x,y) which lie within the template

$$\max_{x,y \text{ in } T} |f(x,y) - t(x,y)| \tag{1}$$

$$\sum_T \sum_T |f(x,y) - t(x,y)| \tag{2}$$

$$\left\{ \sum_T \sum_T \Big(f(x,y) - t(x,y) \Big)^2 \right\}^{1/2} \tag{3}$$

$$\frac{\sum_T \sum_T |f(x,y) \cdot t(x,y)|}{\left\{ \sum_T \sum_T f^2(x,y) \cdot \sum_T \sum_T t^2(x,y) \right\}^{1/2}} \tag{4}$$

The first three measures of goodness of template matching are all matrices based on the L^∞ norm, L^1 norm, and standard Euclidean distance respectively.

The last measure is basically a normalized cross-correlation measure, which corrects for some of the deficiencies the first three matrices have. Notice that a high measure of correlation between a template and an object has a value of zero, while larger values are progressively lower, except for the last measure.

Trying to match the template to every possible size and position of an object can be very computationally costly. What is sometimes done in order to reduce this cost is to break a template into several local templates, and then try and match them to portions of the object. One particular application that might prove useful in feature extraction system is a simple straight line template, or edge detector which will be discussed later. Some of the latest developments in the field of template matching are in the use of non-final, or variable templates. These techniques take into account the fact that objects are not always perfect, but are sometimes distorted and many times contain noise and other non-essential data. It is important to notice that template matching schemes use only local information, and in general are very restricted and data-dependent.

3.0 EDGE DETECTION

Given that the process of image analysis and feature extraction is one of simplification, a commonly used method to simplify an image is to reduce it to a set of objects, or features. As we have seen, there are techniques (i.e., template matching) for detecting a priori known objects. In the case where the objects in the image are not regular or do not lend themselves readily to template matching, this reduction of an image to its constituent features can be facilitated by converting the image to an outline drawing. Hopefully, this conversion will not result in the loss of any information, since many objects can be well specified by their outlines (or borders or edges). Under the assumption that the detection of edges of objects is a valid method for the extraction of features (which is an accepted assumption backed by experience), then automated edge-detection techniques will be an important class of schemes to analyze.

An outline drawing is produced by emphasizing regions of sharp light-dark transitions, and de-emphasizing regions of homogeneous intensity. In other words, it is the result of enhancing the edges or borders of objects. One way to detect edges it to use an edge-detecting template, as described earlier. However, edges exist in many variations, so a single template might not find all edges. One also wants to find edges in all directions. Therefore any edge-detection scheme should be isotropic, or direction independent. In a template matching scheme, this is accomplished by shifting the template to all possible angles and directions.

The most common technique for edge detecting is the taking of a derivative or gradient over the entire picture. On a continuous two-valued function $f(x,y)$ the gradient $\nabla f(x,y)$ is a vector valued function where:

$$\text{magnitude} = [(\partial f/\partial x)^2 + (\partial f/\partial y)^2]^{1/2}$$

and

$$\text{direction} = \tan^{-1}[(\partial f/\partial y)/(\partial f/\partial x)]$$

Another isotropic function used is the Laplacian:

$$L(f) = \partial^2 f/\partial x^2 + \partial^2 f/\partial y^2$$

A third example of historical and practical interest which operates on discrete functions, such as an image, and approximates the gradient, is the Roberts cross operator, $R(x,y)$, where $||\nabla f(x,y)|| \approx R(x,y)$ and

$$R(x,y) = \{f(x,y) - f(x+1,y+1)]^2 + [f(x,y+1)-f(x+1,y)^2\}^{1/2}$$

For computational purposes, the Roberts cross operator is often simplified to

$$R'(x,y) = |f(x,y) - f(x+1,y+1)| + |f(x,y+1) - f(x+1,y)|$$

We can see that $R'(x,y)$ takes a gradient by subtracting diagonal elements as indicated in Figure 1. Thus it will find differences (edges) in both the x and y direction. High values of $R'(x,y)$ indicate a marked difference in grey levels around the point (x,y) indicating an edge. If the resulting gradient function is viewed as an image, it will highlight the edges and downplay the homogeneous regions. A common practice is to threshold or re-scale the gradient image for easier interpretation. The process of taking the gradient of an image is also known as edge enhancement, sharpening, and spatial differentiation.

Figure 1.

One problem with using a gradient is that it is sensitive to noise in an image. In almost any real-world image, there is going to be noise. Besides noise, many times irrelevant data needs to be ignored. The gradient will then register the noise as an edge. In order to solve this problem, various methods have evolved for getting rid of noise. These techniques are known as smoothing or averaging techniques. One method of getting around the problem of noise and irrelevant data is to use weighted averaging windows around a point (x,y) to compute the gradient. A particular example from [2] is:

$$S = [S_x^2 + S_y^2]^{1/2} \text{ or } S' = |S_x| + |S_y|$$

where

$$S_x = [(x+1,y-1) + 2(x+1,y) + (x+1,y+1)] - [(x-1,y-1)+2(x-1,y)+(x-1,y+1)]$$

and

$$Sy = [(x-1,y+1)+2(x,y+1)+(x+1,y+1)] - [(x-1,y-1)+2(x,y-1)+(x+1,y-1)]$$

Besides being useful in the computation of the gradient, it is clear that smoothing and averaging operations would have great utility elsewhere with regards to cleaning up noise and getting rid of or toning down capricious data. A regularized image (also called a smoothed image) is given by the following moving or running average function acting on the image:

$$g_\omega(x,y) = \frac{1}{A_\omega} \cdot \sum_{(x,y)} \sum_{\varepsilon\ \omega} f(x,y)$$

where ω is a window on the image centered at (x,y) and A_ω is the area of that window.

---

[2] Duda, Richard O. and Peter E. Hart, Pattern Classification and Scene Analysis, John Wiley and Sons, Inc., New York, 1973.

4.0 CLUSTERING AND REGION FINDING

We have seen one class of feature extraction techniques that attempt to detect features by enhancing or extracting edges. In those methods, areas on the image where there were large transitions in grey scale were extracted. Now let us examine another class of feature extraction techniques that are complementary in a sense to the edge-detection techniques. These are the clustering or region analysis techniques. They attempt to simplify an image by segmenting it into disjoint homogeneous regions. So actually these routines are also detecting edges implicitly by failing to recognize them as homogeneous regions, and likewise the edge-detection routines are implicitly defining clusters of homogeneous regions as the area within the edges.

Clustering is usually performed by combining two types of algorithms-thresholding and connectedness. Regions are defined by having similar grey scale values and being connected. The problem of how to define connectedness depends on whether you choose 4- or 8- connectedness. In 4- connectedness a cell is conn-cted to another cell if and only if they share a common side. Thus, every cell is connected immediately to four other cells. In 8- connectedness, a cell is connected to another if and only if they share a common vertex. Thus, all diagonal cells are also immediately connected to a cell in addition to the four adjacent cells, giving any cell eight neighbors. A connected region is usually defined as all cells which are connected to each other in some fashion. Connectedness is where a hexagonal grid has advantages over a quadruled grid. A homogeneous region is the largest connected region such that all cells in the regions possess an intensity within some given range. This range is defined by a threshold or some quantization scheme. A pure thresholding algorithm to create a binary image is:

$$B(x,y) = \begin{cases} 1 & \text{if } f(x,y) \geq \theta, \\ 0 & \text{otherwise} \end{cases}$$

where θ is some pre-determined threshold value.

A semi-thresholding algorithm useful for deleting information below a given threshold θ is

$$T(x,y) = \begin{cases} f(x,y) & \text{if } f(x,y) \geq \theta, \\ 0 & \text{otherwise} \end{cases}$$

An example of a type of thresholding scheme used to quantize an image is

$$Q(x,y) = \begin{cases} q_1 & \text{if } f(x,y) \leq Q_1 \\ q_2 & \text{if } Q_1 < f(x,y) \leq Q_2 \\ q_n & \text{if } Q_{n-1} < f(x,y) \leq Q_n \end{cases}$$

where $Q_1, \ldots, Q_n$ is a set of strictly monotonic increasing grey scale keys and $q_1, \ldots, q_n$ is a set of new values.

One interesting result from using thresholding is that although it finds regions of homogeneous content, if a narrow band of values is chosen, thresholding can also produce edges. Once an image is thresholded or quantized, regions can be grown by judicious combinations of closely related thresholded levels. This method is called region growing, and looks particularly adaptable to an interactive environment.

5.0 TRACKING AND CONTOUR FOLLOWING

A major problem with both edge detection and region finding techniques is their inflexibility. In particular, edge detection techniques are stopped by gaps in lines. Region finding routines fail to handle small chunks of noise and irrelevant data. These shortcomings can be helped somewhat by pre-processing the image to smooth out unwanted data. A common method used to facilitate edge detection in images where gradients won't work well or where there are gaps in lines is called tracking or contour following. In this method a routine tracks or follows an edge. Gaps can be handled using such a scheme, but the problem of deciding what is an edge and what isn't still exists, as the need for backtracking occurs. Further pre-processing such as thresholding or quantizing the image can help in this circumstance. The contour following routine is mentioned here because it is anticipated that this type of routine might be very naturally implemented in such an interactive environment of the proposed interactive feature extraction system. Such a routine could run in real-time with the operator providing supervisory control.

6.0 REPRESENTATION OF THE EXTRACTED FEATURES

Once an image is simplifed to a set of extracted features, it is most natural to represent this new extracted image as an overlay to the original image. Since the object of extracting the features was to simplify the image, we should now look for a way to describe this simplified data. The most common method used to represent an extracted image is to describe the borders or edges of the objects in the image. There are many ways to implement this representation, some of which are run length coding, chain coding, and compressed matrix form. These methods are straight foward and are amply described in the literature.

Another method used to describe an extracted object is by specifying its skeleton, which is defined as follows: At each point p in the object S, there is a largest square Q_p centered at p suchthat Q_p is completely contained in S. If Q_p is contained in some other Q_p', then Q_p can be discarded. Clearly, S is the union of all such squares. Thus large objects can be specified by the union of a few squares. The skeleton turns out to be a very economical way to store extracted objects. See [3], pp. 163-164 for further details and examples.

Another method for describing extracted objects from an image is by constructing relational descriptions of the objects. Thus, we describe an object by describing it. Some examples of descriptions we might use are

(1) Geometrical representations of the object - location, shape, size, etc.

(2) Properties of the object - color, texture, etc.

(3) Relations among the objects - size, geometrical relations, etc.

From these descriptions, a labeled directed graph can be constructed which holds the descriptions and represents the interrelations among different objects in the image.

---

[3] Rosenfeld, A. and J. S. Weszka, "Picture Recognition," in Fu, K.S. (Editor), Digital Pattern Recognition, Springer-Verlag, New York, 1976.

7.0 CONCLUSIONS AND RECOMMENDATIONS

Several general classes of automated techniques are available for use in the interactive feature extraction system. Template matching is a relatively specialized method. Edge detection by means of edge template has some promise. Specialized application of feature extraction involving fixed shapes might make good use of template matching. For example, it could be used successfully to find aircraft parked at an airfield, and even differentiating between types of aircraft. The possibility of using interactive techniques to assist in general template matching is a very good prospect. Here, for example, an operator could specify the general location and orientation of an object, and the template matching routine could pinpoint the exact location.

The two general classes of edge detection techniques and clustering or region finding techniques would provide the operator with excellent preliminary extracted and enhanced features, and he could then expand upon those initial guidelines. In particular combining both methods to delineate both edges and homogeneous regions would provide an excellent basis for further image analysis and feature extraction. A gradient or pseudo-gradient routine like the simplified Roberts cross-operator is very effective in finding edges. Smoothing routines help find borders by smoothing out irrelevant data. In addition, smoothing and averaging routines can prove very useful in "clearing" noise from images. Clustering, or region finding routines complement edge-detection routines by finding homogeneous regions between the edges. In addition to its role within clustering, thresholding also looks like a very nice routine to implement and use in an interactive environment.

Tracking and contour following routines also look favorable with respect to interactive use. In particular, they might be used as a tool to expedite the hand-digitization and line-following chores the operator would normally perform. So instead of the operator having to position the cursor to a line and then manually follow the line, a routine could automatically follow a line once the initial starting point has been defined. The operator could interactively direct the line following routines' progress and specify decisions at intersecting lines and dead end lines.

Besides the techniques listed here, several other techniques exist for enhancing an image for the purpose of providing additional analysis clues. These routines do not actually perform feature extraction, but rather feature enhancement. Techniques in this area would provide additional information to the image interpreter and include taking the Fourier transform of an image, convoluting an image, filtering an image, linearly combining multiple images, coordinate transforms, and other transformational processes. In addition to these transformational routines, there are several statistical routines that could provide useful additional information to the image interpreter, such as a histogram plotter, averaging routines, quantization contours, and other statistical operations.

A combination of all of the automatic routines mentioned here, and others as they become available, will provide a very solid foundation on which to build a feature extraction system. They will aid the user through their interaction, and the image interpreter's interaction will open many new doors for semi-automated feature extraction techniques in return.

APPENDIX M

INTERACTIVE FEATURE EXTRACTION SYSTEM FRAMEWORK

The following technical report (TR 319-25) prepared by the Center for Research, Inc., University of Kansas, is included in this volume to provide details in support of the technical discussions of Volume 1.

THE UNIVERSITY OF KANSAS CENTER FOR RESEARCH, INC.

2291 Irving Hill Drive Campus West
Lawrence, Kansas 66045

Telephone: 913-8644832

INTERACTIVE FEATURE EXTRACTION SYSTEM FRAMEWORK

Remote Sensing Laboratory
RSL Technical Report 319-25

M. McNeil

V. H. Kaupp

J. C. Holtzman

March, 1977

Supported by:

U. S. Army Engineer Topographic Laboratories

Fort Belvoir, Virginia 22060

CONTRACT DAAG 53-76-C-0154

ABSTRACT

The production of digital ground truth data bases for radar image simulation is a complicated and costly venture, both in terms of time and expense. Part of the costliness is due to: (1) the methods by which features are extracted from source data, such as aerial photography; and (2) the digitization of feature boundaries. Alternate techniques are being sought to interactively extract features from source imagery while generating or updating the ground truth data base. A specially tailored interactive feature extraction system is proposed. The input to the system will be the maps and photographs the image interpreter would normally start with in the process of creating a ground truth data base for radar image simulation, but the computer will take this intelligence data as it is, digitize it, and allow the user to interactively make a data base without having to go through many intermediate steps. Various apsects of the proposed system are discussed, and a general high-level design is set forth.

1.0 MOTIVATION

In the past, it has been a slow, time-consuming job to identify and extract features from imagery. This imagery may be of several types - maps, photographs, pictures, radar and IR imagery - and may be presented in several formats, film (negative and positives), prints, analog or digital. The usual procedure used to accomplish the task of feature extraction is to take a skilled human operator (either a map specialist, radar interpreter, or photo interpreter), present him with the imagery, and then, using his knowledge and experience, have him manually extract the information needed. With the advent of the modern digital computer, it has been attempted to transfer this burdensome duty to the machine. To date, much time, effort and money has gone into the problem of automating feature extraction and pattern recognition, but so far the complexity of the problem has stymied attempts at completely automating the process.

A particular example of the feature extraction problem occurs in the construction of digital data bases. These data bases are usually maps of some sort with various features outlined that are in a digital matrix format. The most common format is to store data values in a two-dimensional matrix, where the row and column of the data entry in the matrix corresponds to the relative position of that entry on the ground. There are several ways to store this data, but the most common is a fully expanded format*. They contain such information as ground elevation, planimetric detail, or both. They are used in a variety of ways, including geologic, agricultural, and military applications. In the past, it has been costly in terms of time, effort, manpower, and money to create such a digital data base. The common method currently used to make such a data base consists of several phases. First, various information sources are gathered, such as maps, high-resolution aerial photography, microwave and IR imagery, optical stereo pairs, and ground data, such as type and extent of vegetation, urbanization, agriculture, etc., geological composition, weather and irrigation water table level, climate, and time of year. From all these diverse

*That is, a one-to-one ratio between the gathered data and matrix locations to store data, as opposed to a situation in which the data have been compressed in a retrievable manner.

information sources, two maps are constructed - an elevation map and a planimetry map. The elevation map contains the elevation of the ground, usually represented by equal-interval elevation contour lines. The planimetry map contains information about the various categories of microwave backscatter return, usually represented by outlined areas of homogeneous scatterer category. Then, each map is digitized, i.e., transformed from a physical piece of paper into a set of pairs of geometrical coordinates and attributes. These digitized points are normally stored on an intermediate storage medium, such as magnetic or paper tape. Ideally, this digitization process would map every point under consideration into a geometrical coordinate and attribute pair. Thus, the digitized map would be the final digital data base desired. However, practical limitations many times preclude this best-of-all-worlds situation. Therefore, what is usually done in the real world is to define on the maps regions of homogeneous content (lines of constant elevation, agricultural fields, etc.). Then only the boundaries of these regions are digitized. This cuts the sheer size of the data that must be handled by a factor of 50 or 100 for a moderately complex map. This data is then input to a series of computer programs which reconstruct the boundaries and "fill in" the regions described by these boundaries. When this process is done, the desired digital database is ready.

As mentioned above, this is a complicated and costly method of digital data base construction. Therefore, the problem is to facilitate this process by either finding a new method to create such data bases, or making the old process more efficient, or a combination.

2.0 GENERAL REQUIREMENTS

Even though the computer cannot solve the feature extraction problem by itself, it can be used as a device to assist the human operator and thus facilitate the feature extraction process. To achieve the optimum balance between human and computer efficiency, the computer should perform those acts which it can perform better or faster than the human, and similarly for the human operator. The computer is admittedly much faster than a human, and also has larger short-term and long-term memory capacities. The computer can thus perform many simple and repetitive manipulative chores better than a human. However, the computer lacks "intelligence." In the field of feature extraction, the computer cannot make judgment decisions and extract features nearly as well as a human.

Clearly, what is desired is a computer system, or integrated set of programs, that takes care of the bookkeeping and other "non-intelligent" jobs involved in feature extraction while allowing the human operator to make the decisions and guide the system intelligently. Such a system should be totally user-oriented in order to provide an atmosphere where the operator can concentrate on feature extraction rather than peripheral activities. Also, it must be interactive since the feature extraction process is a continuous one of defining and refining features.

It should be noted here that although this feature extraction system is intended to be useful in the general feature extraction field, its initial use will be oriented toward the extraction of planimetric and elevation information for the purpose of data base construction. Thus, the system will be designed and built with this purpose in mind. However, this limitation should not affect the system's usefulness in other areas of feature extraction because the process of extracting elevation contours or defining homogeneous regions on a map is basically the same as the process used in other areas of feature extraction, such as finding certain geometric shapes in a scene or finding cars in a parking lot. With this in mind, the feature extraction system will be a fully general system.

3.0 INPUT

The basic input to the system will be an image in digital format. If the actual input is not originally in this form, there should be some means of getting it into an acceptable form, such as a graphic digitizer. This image will probably almost always be represented as the two-dimensional matrix stored as digital grey-levels, probably between 32 and 256 levels (from 5 bits to 8 bits). Three-dimensional and higher-dimensional images can be easily stored within this two-dimensional framework. These two-dimensional coordinates will probably represent ground distance in some X and Y directions, where X and Y will normally be orthogonal, although this can be changed. However, for the sake of display purposes, the two-dimensional matrix will represent an orthogonal grid system with equally scaled coordinates. Thus, polar coordinates and other non-orthogonal systems will probably have to be converted into an orthogonal system in order to be stored as an image file and looked at on a monitor without distortion. This restriction on the form of input files is based on standard CRT capabilities. If there is a graphic display unit available with non-orthogonal rasters, this restriction may be waived. The types of input anticipated to be encountered are

(1) Maps
(2) Photographs
(3) Radar images
(4) Radar Simulations
(5) Prepared data bases
    (a) Elevation
    (b) Planimetry
(6) Other

In addition to image input, another type of input to the system will be operator commands which will be entered interactively from a keyboard. The list of legal commands will form the command language. The command language should be easy to use, understand, and remember; it should also be terse enough so that it doesn't hamper the operator, but flexible enough to handle all of the operations the operator might wish to use. The command language will be on a bi-level scheme; that is, there

will hopefully be a short and long form of discourse with the system. The long form should be fully explanatory and easy enough for a novice to use, while the short form should be as economical as possible to expedite the experienced user's interaction. Also the command "user", and for that matter, the whole system, will be designed in a modular fashion, so that changes and additions will be easy and straightforward to implement. This design aspect is particularly important since it is anticipated that many people will write and add modules to the basic system later. Special care will also be taken in the system design to try to make all modules as independent as possible. This will allow for a subset of the system to be implementable to perform specialized activities without dragging the whole system along. It also allows additional modules to be added easily.

4.0 OUTPUT

The output or final product of the system will be a matrix of values. This matrix could represent some sort of data base, i.e., planimetry, elevation, land use, geologic reserve locations, etc. It could also represent a map of an area, showing backscatter categories or other classifications. It could also represent a radar image or an enhanced photograph. These are all images of some sort. Notice that the output can also be used now as input. The output can be used in one of two ways: (1) in a digital format, as data to another program, or (2) in a pictorial format, either on a CRT, TV monitor, plotter, or printer. The final form of the output is dependent upon its further uses.

5.0 PHYSICAL REQUIREMENTS

The system as actually implemented will depend heavily on the hardware available. However, the high-level design of the system is independent of hardware with a few exceptions. The first is that the system, of course, needs a computer of some sort with reasonable memory, adequate off-line storage, and sufficient computing power. Another requirement is that there must be some sort of display device. Associated with this display device must be a data channel for transferring images to and from the computer in reasonable time, and also a means whereby the operator may access the display and convey information from it to the computer. These requirements can be met by a minicomputer with disk storage and a display CRT with either a cursor, framer, or light pen. Although this hardware may be considered standard, it is by no means the only configuration that is acceptable.

6.0 SUBSYSTEM

The system can be designed either as a complete operating system or as a subsystem operating under some higher-level operating system. If it were designed as an integral operating system, it would have the assets of efficiency and speed. With specially designed supervisory and support routines, the system could operate faster and with fewer interruptions than possible with a general purpose operating system. Also, I/O functions, in particular image transfers, could be made to operate faster and also speed up interactions with the operator. If the system were designed as a subsystem, it would have to rely on existing supervisory and support routines. However, experience has shown that most operating systems provide adequate supervisory and support routines, so that even though a specialized system could operate faster, the effort required to design, code, and test a complete operating system far exceeds the advantages derived from it. (But note that in some special circumstances, this rule might not hold - i.e., if I/O was really too slow to handle image transfers, etc.). Another advantage of a subsystem is its portability. If the system has to be moved to another hardware configuration, it would be much easier to redesign the feature extraction system interface with the operating system than to have to redesign the entire system to fit new hardware. Also, operating a subsystem mode would possibly allow other users to work in other subsystems in a time-sharing environment. In conclusion, it has been decided that even though a specialized operating system geared to feature extraction problems will be more efficient and faster, the extra effort needed to create the supervisory and support routines is unwarranted. Therefore, the feature extraction system will be kept as simple as possible and designed as a subsystem to operate under an existing operating system.

7.0 SYSTEM BREAKDOWN

The feature extraction (sub-) system will consist of an operating system interface, a file system, a command interpreter, and various function routines. The operating system interface is not really an individual module, but rather exists throughout the entire system as the means by which the system communicates with the operating system. The file system will handle image files. It will be simple, since, as indicated in INPUT, files will probably be restricted to matrices of data values. Within the file system will exist the interface to create, destroy, modify, and transfer files to and from storage (disk or tape), memory, and the display device(s).

The command interpreter is the executive of the system. It will be the program that accepts as input commands from the operator, parses them, and then invokes the proper routine to perform the actions indicated. This interpreter forms the interactive link between the operator and the computer. The command language, an intrinsic part of the command interpreter, has been previously discussed. Particular care will be taken in the command interpreter design to ensure that changes and additions to the command language can be easily incorporated into the command interpreter. Special documentation on how to modify parts of the system will be included in the final documentation.

These first three system parts form the skeletal framework of the system. The final system part is the package of various function routines. These programs will form the bulk of the system. They will be the last to be coded, and additional routines will probably be added as the system grows and more or different functions are needed in the system. Below is a list of possible functions grouped into categories which seem desirable to include in the system. This list is not necessarily complete, but represents a first-order view of functional capabilities desired.

A. Feature Extraction
   1. follow lines
   2. connect points
   3. clustering routines (find homogeneous regions)

4. gradient routines (find boundaries)
5. delineate features/designate regions/define boundaries
6. modify features (add, delete, fade, brighten, enhance)
7. zoom capability

B. Whole Image Processing
1. transfers
2. scale changes - expand/contract
3. sub-imaging
4. translation - rotation/flip/invert
5. polar-to-rectangular and vice versa
6. transforms - Laplace, FFT convolution
7. gradients
8. clustering
9. bias, scale, negate
10. sharpen, blur, average
11. threshold, binary image
12. statistical routines, histogram, averaging

C. Multi-Image Processing
1. add/subtract/multiply/divide
2. combine, merge
3. superimpose
4. multiple-band
5. split screen, flicker, fade in - fade out

D. Analog Processing
1. integration
2. transforms
3. linear combinations

E. Special Purpose Routines
1. evaluation functions
2. radar simulations
3. bio-medical applications
4. three-dimensional processing

F. Interface to Adapt Existing Routines and Files

8.0 CONCLUSIONS

It is felt that an interactive computer system would best meet the needs of the problem of feature extraction from imagery. A basic system design has been made and initial attempts at implementation are underway. Other uses for the system have not been explicitly explored, but many other applications could benefit from such a system.

APPENDIX N

IMAGE HANDLING AND PROCESSING

The following technical report (TR319-26) prepared by the Center for Research, Inc. University of Kansas, is included in this report.

THE UNIVERSITY OF KANSAS CENTER FOR RESEARCH, INC.

2291 Irving Hill Drive—Campus West
Lawrence, Kansas 66045

Telephone: 913-8644832

IMAGE HANDLING AND PROCESSING

Remote Sensing Laboratory

RSL Technical Report 319-26

M. McNeil
V. H. Kaupp
J. C. Holtzman

March, 1977

Supported by:

U. S. Army Engineer Topographic Laboratories

Fort Belvoir, Virginia 22060

CONTRACT DAAG 53-76-C-0154

ABSTRACT

This report examines some of the problems and difficulties encountered in image handling and processing. A major area of emphasis is image storage, examining physical storage requirements, I/O (input/output) transfer time, data compactness, reconstructive processing, sequential and random file structures, and compatability of file structures. Transfer and display of images is examined and transfer time is shown to be crucial to the success of any image handling system. Management of images is examined briefly, as is general manipulation of images, both external and internal.

1.0 INTRODUCTION

In working with images, certain problems invariably arise with respect to the handling and processing of data. This report examines some of the problems and presents some possible solutions to these problems. Throughout this report, the following definition of image will be used:

> An image is a structured file of data values which is normally displayed pictorially.

This concept of an image is analogous to that of a picture (map, photograph, scene) that is stored digitally in a computer. So an image is really a large data structure. In working with images, special actions need to be taken in order to successfully handle them. These special needs create special problems that need to be taken care of before anyone can work with images.

2.0 IMAGE STORAGE

2.1 sical Storage, I/O Transfer Time

Perhaps the most significant problem area concerns image storage, retrieval, and formatting. Nontrivial images invariably contain excessively large amounts of data, due to the very nature of them. The problem of how to store this vast quantity of data touches quite a few areas of interest. Of major interest is the problem of how to physically store the information. Normally, one would want to store the information in as small a space as possible, since storage space is usually limited. For long term storage, the standard physical devices used are disk and magnetic tape. Occasionally, circumstances dictate hard copy storage, which means punch cards or paper tape. Since it is easier to store small amounts of data on these physical storage media, rather than large amounts of data, it makes sense to try and store data as compactly as possible. Besides the physical storage space aspect, consideration must also be given to input and output (I/O) transfer time involved in accessing the data once it is stored. As a general rule of thumb, access time (I/O transfer time) is directly proportional to the amount of data being accessed. Thus, a large file will take longer to access than a small file, all other factors being the same.

2.2 Reconstructive Processing

However, if an image, or any file, is stored in its most compact form, then even though it will use the minimum physical storage space and be transferable from storage to memory in the minimum time, it will have the serious drawback that the data will not be immediately ready to use, since it is in a compacted form. Therefore, there must be some processing to reconstruct the original data values from its compacted form. Later, this same processing must be reversed to compact the data in memory to prepare it for storage. Thus, there is a tradeoff between minimizing storage and minimizing reconstructive processing.

2.3 Tradeoff Between Compactness and Reconstruction

2.3.1 No Compaction

At one end of the spectrum of tradeoff is simply doing no compaction on the data at all and storing it just like it appears in memory. Thus there would be no need to compact the data going from memory to storage or to reconstruct it as it comes into memory from storage. However, this would be the most space-consuming storage method, as there is no compaction at all. The tradeoff is obviously biased toward minimizing reconstructive processing. This storage scheme would be good to use in the situation where reconstructive processing would be burdensome (as would be the case in a very slow or heavily used CPU or where storage was not a problem (as would be the case if there was a lot of extra storage space with fast I/O transfer rates). An example of such a storage scheme is the method by which the current Pickwick Dam site data base is stored. This data base is a matrix of data values with 3169 rows of data, each row in turn containing 3169 data items. It is stored on magnetic tape (one main reel and one overflow reel) in sequential physical block mode, each physical block containing one complete row of data. This is approximately 3000 feet of magnetic tape. The data is not compacted at all. Each data item has 16 bits of significant information. Since the word size is 36 bits on the Honeywell 66/60, this means that fully 50%, or 1500 feet of tape, is not utilized at all. One reason why this method was chosen was that I/O transfer rates on the Honeywell 66/60 are fast, and performing reconstructive processing on such a large amount of data would use a lot of CPU time. Another reason was that it is the easiest and most straightforward. Thus, the user of the data base never has to worry about whether his data is in the right format or not.

2.3.2 Extreme Compaction

At the other end of the spectrum of tradeoff is performing as much compaction as possible on the data. Thus, storage is held to a minimum, but there is a need to perform quite a bit of reconstructive processing. In performing the compaction, the reduction process can get quite complicated as new, complex data structures are evolved to store data in a more compact format. Reconstuctive processing can become extremely complicated, but storage savings are usually on the order of several hundred, depending of course on how intricate the original image is and the exact compaction algorithm chosen. This storage scheme would be good to use in the situation

where storage space was at a premium (as would be the case with only limited disk space or very slow I/O transfer rate) or where reconstructive processing was not a problem (as would be the case with a very fast or lightly loaded processor).

An example of a highly compacted data storage technique is a data compression method used in handling elevation data. The technique used to compact the data is to only record those elevation values that have a certain characteristic. This characteristic is commonly set to be any elevation value which is a multiple of twenty feet. Thus, the data may be stored either as a collapsed matrix or as strings of 20-foot contour lines. This data compression technique is highly effective in compacting the data, bur reconstructive processing can be a problem. One- and two-dimensional interpolation routines can reconstruct the data, but it takes a long time. Also, another important aspect of this or any other data compression technique is the validity of the reconstructed data. Sometimes, not all of the data can be restored to its original appearance. Thus, some data is lost. If this process is used over and over again, it might result in a further degraded image every time the data is accessed. Thus, integrity of the data compaction scheme is an important aspect.

Some other extreme compaction schemes involve modeling the data mathematically. An example might be to combine the above mentioned scheme of picking the 20-foot contour levels, and then further compacting that data by modeling each contour line as a mathematical function like an "n^{th}" degree polynomial. In this way, the data could be compacted further. But again, this method suffers from the validity problem in that mathematical models can usually only approximate the data. Therefore, every time the data is reconstructed, it becomes progressively more distorted. Notice that this form of compaction actually changes the data itself besides just the method in which it is stored.

2.3.3 Optimum Balance

Between these two extremes lies a complete spectrum of data storage schemes that attempt to optimize storage and processing for any particular situation. Most of these methods choose some data compaction scheme that falls short of extreme compaction. In fact, some of the first-order data

compaction algorithms generate storage savings on the order of five or ten times the initial size while incurring only a very modest processing overhead.

Any one of the middle-of-the-road schemes would be good to use in a normal situation where there needs to be some balance between data storage and reconstructive processing. A particular example that is quite commonly used is packing. In a packing scheme, several data items which are smaller than the computer word size are packed into a single word. For example, if the data items contain only 12 bits of information and each computer word has 36 bits, then three data items can be packed easily into one computer word. Notice that packing does not change the data itself, but only stores it more compactly. This scheme is successful because a computer word is usually the smallest piece of information that can be accessed and stored and also because bit shifting manipulation is fast.

Sometimes, packing on word boundaries doesn't work. For instance, if the data item are 10 bits long and the word size on the machine is 36 bits, then only 3.6 data items can fit into one word. In this case, one can either pack only three data items into one word, leaving six bits of every word unused, or one can choose to pack across word boundaries, i.e., fit three data items and the first six bits of the fourth data item into one word and put the last four bits of the fourth data item into the beginning of the next word. This latter scheme uses all of the available storage space, but reconstruction is much more difficult, as reconstructing data items spread across word boundaries is tricky to do. However, one might be forced to resort to crossing word boundaries if the data items are longer than the word size.

It has been assumed that all data items are of the same size, which will be the normal case. However, sometimes variable data sizes must be accommodated. In this case, one alternative is to force all the data items to be the same size by filling in with zeroes. Another alternative is to place markers at the end of every data item. This scheme will work satisfactorily if a suitable marker can be found which does not conflict with any of the data items. Data storage is much more complicated for variable-sized data items. Since almost all data base applications will deal with uniform data, it will be assumed that this is the case.

Two other examples of data compaction and storage will be mentioned. The first called run length coding, is a good scheme to use whenever the data is constant over areas of the image. In this method, which is similar to the elevation storage technique discussed earlier, basically the outlines of features within the image are stored. Within a row of data, the first entry is recorded along with the number of times it occurs consecutively. Then, when the data changes to a new value, this new value is recorded along with the number of times it occurs consecutively, and so on until the end of the line. In this way, the boundaries of regions of constant data are recorded, or outlined. This scheme is well-suited to storing a category data base, or any other image where only the outlines need to be remembered. In this way, the exact same image can be reconstructed by a simple fill-in process. This fact distinguishes it from the elevation contours method discussed earlier, which only approximates the image.

The second example is a similar scheme but better suited towards handling images like the elevation contours where the data is not constant, but changing by small increments. In this method, a base value for a row (or column) of data is given, usually the first value encountered or the average value. From then on, offsets from the base value are used to describe the data. This process can either be static or performed as a running or relative offset. The following example illustrates both methods:

STATIC OFFSET

DATA	INITIAL VALUE	AUG. VALUE	RUNNING OFFSET
71 716	71 716	71 719 - 3	71 716
71 718	+ 2	- 1	+ 2
71 721	+5	+ 2	+ 3
71 727	+11	+ 8	+ 6
71 730	+14	+11	+ 3
71 730	+14	+11	+ Ø
71 725	+ 9	+ 6	- 5
71 717	+ 1	- 2	- 8
71 710	- 6	- 9	- 7
71 703	-13	-16	- 7

Using either of these methods, the number of data entries is not reduced, but rather simplified. In the above example, for instance, each initial data entry needs at least 17 bits to accurately represent it in a computer. The static offset method needs 17 bits for the initial value, but only 5 bits for each entry thereafter. The running offset again needs 17 bits for the initial value, but then only four bits thereafter. Each method has its own particular advantages. Notice that although the number of entries has not been reduced, the data has been changed so that it requires much smaller storage space. This scheme would be a good one to use on elevation data, which exhibits the kind of data which works best in such a scheme.

2.4 Sequential and Random File Structures

In addition to these different methods of storing images (large data files), there are several techniques for physically storing the data on the peripheral devices. On such devices as magnetic tapes, cassettes, paper tape, and cards, data must be stored sequentially.

When data is stored sequentially, in order to access data which is not currently being looked at, the file must be rewound and searched through until the desired record is found. However, data can be arranged into blocks of data in such a way as to maximize the use of the device. Then, instead of looking word by word for a record one can go to that block, but still starting from the current location or the beginning of the file. Within physical blocks, data may additionally be arranged into logical records to further facilitate access.

Sequential files are fine for data that will be used sequentially, i.e., in order. However, if data is to be used at random from within a file, then a random file structure would be desirable. In a random file, any record can be accessed in approximately the same time as any other record. This is usually accomplished by having an address vector with the physical disk address of each record. Thus, there is an overhead in storage space and to some extent speed. To determine if a random file structure is better than a sequential file structure, this overhead must be weighed against the utility of fast random access to data. If the image file is to

be accessed only to transfer it as an entire image, then a sequential file would be best. If, however, the capability of randomly selecting data from the image file is wanted, such as would be the case with in memory editing functions, then a random file structure would be best, as direct access would negate the need for constantly rewinding and searching sequentially.

2.5 Compatibility

Another major consideration in determining image file structure and storage is compatibility. It would be desirable to store an image in a format where it would be easily compatible with other images and files. This factor would be important if images were to be used off site, since the image must be in such a format that the new site could read it and use it. There are often severe hardware, software, and interface compatibility problems within even the same site. Different programs must be able to access different files successfully, since often image analysis involves working with multiple images. This consideration in file design suggests the standardization of image storage techniques. But this must be weighted against some of the other factors already mentioned, as different circumstances and situations dictate different storage techniques.

3.0 TRANSFER AND DISPLAY

3.1 Display Devices

Another significant problem area concerns the transfer and display of images. As noted before, images are data files that are normally viewed as pictures. Obviously, there must be some medium on which to display and view the images. There are many such display devices, including:

1. Graphic terminals and CRT's
2. Television monitors
3. Oscilloscopes
4. Plotters
5. Film

Many of these devides are set up to receive data in a very particular format. For example, television monitors normally require block sequential transfers. Such restrictions will definitely affect the technique used to store the data, as data should be stored in such a way as to require the minimum conversion. Also, many times the I/O data channels are of a particular form. Thus, if there is a 16 bit data bus and 12 bit data items provisions must be made to account for the discrepancy. In some instances, this is taken care of by the display interface. In others, special preprocessing must be applied to images before they can be transferred and displayed. Each different hardware device will affect a system differently, unless a uniform interface can be designed for all devices.

3.2 Transfer Time

The biggest problem involved with the transfering of images is time. The normal path that data has to follow in order to get from storage to a display device is to first travel from the storage device to memory. While the data is in memory, it is optionally processed to reconstruct the image as if it was stored in a compacted form. Then the data is further processed to put it into the correct format to be passed to the display device. The data is finally put on the data channel and sent to the display device, where it is finally viewed. This path is quite long and complex,

and remembering the huge amount of data present in an image that must be transferred, it is not surprising that images transfers take a long time to complete. This time factor causes much delay and degrades the whole system since the user is waiting for his image to be transferred and can't do anything else until it is completed. Meanwhile, valuable resources are being used, and if the system is a single-user system, then the entire system is log-jammed until the transfer is done. Thus, transfer time can heavily influence the effectiveness and usefulness of an image handling system, and so care should be taken in the early design to try to minimize transfer time.

4.0 IMAGE MANAGEMENT

4.1 Image Control Blocks

Management of images is another major area of concern. There must be some way of maintaining control over images. The manner in which this problem is commonly solved is to create special areas where control information is kept. These areas are called Image Control Blocks (ICB). These ICB's should contain all the control information about a particular image that will be needed anywhere by the system. This information is then used by the system, and can be changed by the user only in special circumstances. ICB's normally include the following information:

1. File name
2. File format (how it is stored)
3. File location (where it is stored)
4. Size of physical records
5. Size of logical records
6. Random/sequential
7. Size of data item
8. Size of image
9. Number and size of subimages
10. Open/closed (accessible)
11. Other

4.2 Image Descriptor Records

Besides control information, many times there is other descriptive information associated with an image. This information is not used by the system to manage the image, but rather it is used to present descriptive information about the image to the user. This type of information is contained in areas called Image Descriptor Records (IDR). The IDR's are in turn controlled by the ICB. IDR's normally contain such information as:

1. File name
2. Data created
3. User name
4. Last date changed

5. Last data accessed
6. Changes to original image
7. Where did original image come from
8. What information does this image contain
9. Other

5.0 MANIPULATION OF IMAGES

5.1 Whole Image Manipulation

The last major area of interest concerning images is the manipulation of images. In working with images, the occasion quite often arises that some changes need to be made in the image. These changes can be made in several ways. One class of modification schemes acts on entire images. Functions in this class include thresholding, coordinate transformation, scaling, bias, expansion, contraction, subimaging, functional transformations, etc. A subclass of modification schemes includes multiple image functions, such as appending, merging, arithmetic combinations, etc. Since all these functions operate on the entire image, they usually take a long time because of the amount of data involved. Such functions are usually well-specified and parameterized for the user's convenience. A major problem with these functions is that it takes a long time to verify the results. Besides taking a long time to run the functions initially, then the results have to be transferred in order to evaluate them. For instance, if a thresholded image is required, the resultant thresholded image usually requires verification that the level of quantization was correct. These functions act as a guide for the user, usually performing feature enhancement functions over the entire image. For further information on these functions, see [1].

5.2 Image Editing and Modification

Another class of modification schemes act on individual features within an image. These schemes differ from whole image processing in that only selected sub-portions of an image are affected, rather than the entire image. This class is further divided into two sub-classes. One sub-class, called image editing, just manipulates data within an image by processing the data in memory without ever transferring it to a display device. Examples of this type of processing include string manipulations, inserting and deleting lines, changing and replacing data items, etc. The other sub-class, called image modification involves manipulation of the data in memory while the image is being viewed on a display device. The difference between image editing and image modification is that in image modification,

clues are taken from examination of the scene presented by the image on the display device. Examples of image modification techniques include various interactive feature extraction techniques, as discussed elsewhere[1].

[1] McNeil, M., J. L. Abbott, V. S. Frost, V. H. Kaupp, and J. C. Holtzman, "Automated Techniques in Feature Extraction," TR 319-24, Remote Sensing Laboratory, The University of Kansas, June, 1977.

6.0 CONCLUSIONS AND RECOMMENDATIONS

Many of the problems associated with image handling and processing have been discussed, with possible solutions proposed. It is evident that the specific choice of a solution to a problem is dependent upon several factors, including the hardware and software available, the type of problem and the type and usage of the image involved. Because images contain large amounts of data, a primary problem is how to store and access the data quickly and efficiently. The tradeoff between compactness of data storage and reconstructive processing has been explored to some lengths, with the conclusion that the specific form of storage be determined by considering hardware storage and processing capabilities, I/O transfer rate, file structure, compatibility, and the type of data nad its intended use. In addition, specific algorithms have been discussed and their usefulness with respect to certain types of data explored. Display device capability and transfer times to and from memory have been shown to be a very influencing factor in system utility and ultimate usefulness. Management of images will probably be handled in the standard way with an Image Control Block and Image Descriptor Records. Manipulation of images was briefly discussed, identifying several areas of manipulation to be used in an image handling and processing system.