

ETL-0530

DTIC FILE COPY

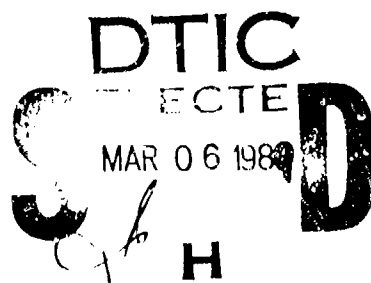
AD-A205 195

Linear Feature Extraction from Radar Imagery: SBIR-Phase II, Option II

David L. Milgram
Philip Kahn
Gary D. Conner
Daryl T. Lawton

Advanced Decision Systems
1500 Plymouth Street
Mountain View, California 94043-1230

December 1988



Approved for public release; distribution is unlimited.

Prepared for:

U.S. Army Corps of Engineers
Engineer Topographic Laboratories
Fort Belvoir, Virginia 22060-5546



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.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188
Exp. Date: Jun 30, 1986

1a. REPORT SECURITY CLASSIFICATION Unclassified			1b. RESTRICTIVE MARKINGS Unclassified		
2a. SECURITY CLASSIFICATION AUTHORITY None			3. DISTRIBUTION / AVAILABILITY OF REPORT Approved for public release; distribution is unlimited.		
2b. DECLASSIFICATION / DOWNGRADING SCHEDULE None			5. MONITORING ORGANIZATION REPORT NUMBER(S) ETL-0530		
4. PERFORMING ORGANIZATION REPORT NUMBER(S) TR-3139-03			7a. NAME OF MONITORING ORGANIZATION U.S. Army Engineer Topographic Laboratories		
6a. NAME OF PERFORMING ORGANIZATION Advanced Decision Systems		6b. OFFICE SYMBOL (if applicable)		7b. ADDRESS (City, State, and ZIP Code) Code: CEETL-RI-D Fort Belvoir, Virginia 22060-5546	
6c. ADDRESS (City, State, and ZIP Code) 1500 Plymouth Street Mountain View, California 94043-1230		8a. NAME OF FUNDING / SPONSORING ORGANIZATION		9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER DACA72-86-C-0004	
8b. OFFICE SYMBOL (if applicable)		10. SOURCE OF FUNDING NUMBERS			
8c. ADDRESS (City, State, and ZIP Code)		PROGRAM ELEMENT NO.		PROJECT NO.	
		TASK NO.		WORK UNIT ACCESSION NO.	
11. TITLE (Include Security Classification) Linear Feature Extraction from Radar Imagery: SBIR Phase II, Option II					
12. PERSONAL AUTHOR(S) David L. Milgram, Philip Kahn, Gary D. Conner, Daryl T. Lawton					
13a. TYPE OF REPORT Final Technical Report		13b. TIME COVERED FROM 6/87 TO 6/88		14. DATE OF REPORT (Year, Month, Day) 1988 December	
				15. PAGE COUNT 99	
16. SUPPLEMENTARY NOTATION					
17. COSATI CODES			18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)		
FIELD	GROUP	SUB-GROUP			
08	02		SAR, Feature Extraction, Edge Detection, Terrain Analysis, Image Understanding		
19. ABSTRACT (Continue on reverse if necessary and identify by block number)					
The goal of this effort is to develop and demonstrate prototype processing capabilities for a knowledge-based system to automatically extract and analyze linear features from Synthetic Aperture Radar (SAR) imagery. This effort constitutes Phase II funding through the Defense Small Business Innovative Research (SBIR) Program. Previous work examined the feasibility of and technology issues involved in the development of an automated linear feature extraction system. This final report documents this examination and the technologies involved in automating this image understanding task. In particular, it reports on a major software delivery containing an image processing algorithmic base, a "perceptuai structures" manipulation package, a preliminary hypothesis management framework and an enhanced user interface.					
20. DISTRIBUTION / AVAILABILITY OF ABSTRACT ☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT. ☐ DTIC USERS			21. ABSTRACT SECURITY CLASSIFICATION Unclassified		
22a. NAME OF RESPONSIBLE INDIVIDUAL E. James Books			22b. TELEPHONE (Include Area Code) (202) 355-2728		22c. OFFICE SYMBOL CEETL-IM-T

Continuation of Block #19:

In addition, the report discusses three terrain feature finders (for forested areas, bridges, and roads) and presents results showing their application to SAR imagery.

Finally, the report makes specific recommendations for the completion of research and development leading to an autonomous linear feature extraction prototype system.

PREFACE

This report describes work performed under contract DACA72-86-C-0004 for the U.S. Army Engineer Topographic Laboratories, Fort Belvoir, Virginia 22060-5546 by Advanced Decision Systems, Mountain View, California 94043-1230. The Contracting Officer's Technical Representative has been Dr. P.F. Chen.

The authors would like to thank Dr. Pi-Fuay Chen and Mr. Richard Hevenor for their many helpful suggestions throughout the course of the effort.

Accession For	
NTIS GRA&I	☒
DTIC TAB	☐
Unannounced	☐
Justification	
By	
Distribution/	
Availability Codes	
Dist	Avail and/or Special
A-1	



TABLE OF CONTENTS

	Page
1. EXECUTIVE SUMMARY	1-1
1.1 BACKGROUND OF PROBLEM	1-1
1.2 APPROACH	1-2
1.3 SUMMARY OF ACCOMPLISHMENTS	1-3
1.3.1 Phase I	1-3
1.3.2 Phase II - Base Contract	1-4
1.3.3 Phase II - Option I	1-4
1.3.4 Phase II - Option II	1-5
1.4 CONCLUSIONS	1-6
1.4.1 Lessons Learned	1-6
1.4.2 Expected Benefits	1-7
1.5 RECOMMENDATION FOR FUTURE RESEARCH	1-8
1.5.1 What is Needed	1-8
1.5.2 A Plan for Automatic Terrain Feature Extraction	1-8
1.6 ORGANIZATION OF THIS DOCUMENT	1-11
2. TECHNICAL APPROACH	2-1
2.1 OVERVIEW OF CONCEPTUAL SYSTEM ARCHITECTURE	2-1
2.1.1 Motivation	2-1
2.1.2 System Overview	2-2
2.1.3 Image Structure Data Base	2-2
2.1.4 Perceptual Grouper	2-4
2.1.5 Terrain Feature Modeler	2-6
2.1.6 Hypothesis Manager and the Terrain Hypothesis Database	2-7
2.1.7 System Controller	2-8
2.2 FUNCTIONAL COMPONENTS	2-8
2.2.1 Perceptual Grouper	2-8
2.2.2 Terrain Feature Modeler	2-9
2.2.3 Hypothesis Manager	2-27
2.2.4 System Controller	2-30
2.3 RECOGNITION ALGORITHMS	2-31
2.3.1 Perceptual Grouper	2-31
2.3.2 Terrain Object Modeler	2-46
3. DEMONSTRATION OF SAR TERRAIN FEATURE EXTRACTION	3-1
3.1 INTRODUCTION	3-1
3.2 FOREST FINDER	3-1
3.3 BRIDGE FINDER	3-11
3.4 ROAD FINDER	3-19
3.5 EVALUATIONS AND CONCLUSIONS	3-28

4. GLOSSARY .

4-1

5. REFERENCES

5-1

LIST OF FIGURES

	Page
1-1: Research and Development Plan	1-9
2-1: Conceptual System Architecture	2-3
2-2: Simple Example of Image Structure Creation by the Perceptual Grouper	2-5
2-3: Structural Hierarchy for Terrain Feature Models	2-10
2-4: Example of <i>Image Feature Properties</i> and their Relationship to Terrain Features	2-12
2-5: Local Region Properties for Bridge Terrain Feature Example	2-15
2-6: Specifying Point Relationships	2-17
2-7: Prototypical Support Area Shapes	2-18
2-8: Complex Support Area	2-19
2-9: Support Areas and a Central (Bridge) Region	2-20
2-10: Perimeter Relationships	2-21
2-11: Using Hierarchical Terrain Representation to Recognize Terrain Features	2-24
2-12: Mapping from Image Structures to Terrain Feature Models	2-25
2-13: Hypothesis Evaluation and Reasoning	2-29
2-14: Defining Brightness by Dividing Intensity into Fixed Ranges	2-32
2-15: Brightness Illusion: the Two Center Squares have the same Brightness	2-33
2-16: Perceptual Brightness Algorithm BRIGHT Range Definition	2-35
2-17: Perceptual Brightness Calibration	2-37
2-18: Algorithm for Extracting ROUGHLY TEXTURED Image Areas	2-39
2-19: Computing Region Properties: CENTROID, ENDPOINTS, LENGTH, WIDTH, and STRAIGHTNESS	2-41
2-20: Determining Perimeter which Faces Illumination	2-45
3-1: Forest Terrain Model. (a) Formative Model, (b) Perimeter-based Local Region Properties for Forest Terrain Model	3-2
3-2: Original SAR Image of Forested Terrain	3-3
3-3: Original Image Convolved with a Gaussian with (a) $\sigma \approx 3$, and (b) $\sigma \approx 4$	3-4
3-4: Bandpass Image <i>B</i> Obtained by Subtracting Figure 3-3a from 3-3b	3-6
3-5: Texture Elements Produced by Thresholding Areas in the Bandpass Image in Figure 3-4	3-7
3-6: Rough Texture Elements (Textels) Produced by Merging the Binary Images Shown in Figure 3-5	3-8
3-7: ROUGHLY TEXTURED Regions (Small Texture Regions have been Filtered)	3-8
3-8: Holes in ROUGHLY TEXTURED Regions (in red)	3-9
3-9: Initial Forest Terrain Hypotheses	3-9
3-10: Local Region Supports for Central Forest Region Hypotheses (in green)	3-10

3-11: Magnified View of Central Forest Region Support at the Center Bottom of Figure 3-10	3-12
3-12: Final Forest Hypotheses which Possess Local Region Support	3-12
3-13: Bridge Terrain Model. (a) Formative Model, (b) Point-based Local Region Properties for Bridge Terrain Model	3-13
3-14: Original SAR Image Which Contains a Bridge	3-14
3-15: Intensity Histogram for Image Shown in Figure 3-14	3-14
3-16: Gradient Magnitudes for Image Shown in Figure 3-14	3-15
3-17: Gradient Histogram for Gradient Image Shown in Figure 3-16	3-15
3-18: Calibration Points Selected by the Perceptual Brightness Algorithm	3-17
3-19: Perceptually BRIGHT Regions (in green, blue, red)	3-17
3-20: Local Region Support Areas for Central Bridge Hypotheses (in green)	3-18
3-21: Zoom-in of Local Region Support Areas Which Do Not Confirm the (Incorrect) Central Bridge Hypothesis (in green)	3-18
3-22: Zoom-in of Local Region Support Areas Which Confirm the (Correct) Central Bridge Hypothesis (in green)	3-20
3-23: Final Bridge Hypothesis Which Possesses Local Region Support	3-20
3-24: Model of Road Segment Feature	3-21
3-25: Original SAR Image Which Contains a Road Network	3-22
3-26: Band Pass Image Results	3-23
3-27: Medial Axes of Candidate Road Segments	3-25
3-28: Longer Segment Results After Shrinkage	3-26
3-29: Fitting Straight Lines to the Medial Axes	3-27
3-30: Merged Line Segments	3-29
3-31: Segments Joined Over Longer Gaps	3-30
3-32: Result over Extrema Background	3-31

1. EXECUTIVE SUMMARY

Advanced Decision Systems (ADS) is pleased to submit this final technical report on research undertaken during the Linear Feature Extraction from Radar Imagery effort (contract #DACA72-86-C-0004). The goal of this effort has been to develop and demonstrate prototype processing capabilities for a knowledge-based system to automatically extract and analyze linear features from synthetic aperture radar (SAR) imagery. This effort constitutes Phase II funding through the Defense Small Business Innovative Research (SBIR) Program. The previous Phase I (contract DACA72-84-C-0014) work examined the feasibility of and technology issues involved in the development of an automated linear feature extraction system [14]. This final report reviews the approach taken and discusses algorithms which implement the automatic recognition of three significant terrain feature classes: natural terrain features (e.g., forests, fields), cultural features (e.g., bridges), and extended cultural features (e.g., roads).

1.1 BACKGROUND OF PROBLEM

A vitally important problem facing the Department of Defense is the ability to quickly and efficiently analyze remotely sensed image data. This analysis is used for a variety of applications ranging from automated map making/updating to a variety of surveillance tasks, to other military and commercial remote sensing applications. An increasingly important and useful sensing capability is provided by synthetic aperture radar (SAR) imagery.

Imaging radar sensors provide all-weather, day/night, and cloud penetration capabilities for a variety of applications. Technical capabilities now allow enormous volumes of such imagery to be automatically produced in relatively short periods of time. However, the current methods for analysis and interpretation of radar imagery largely consist of manual examination by human experts. As the quantity of imagery expands, the requirements for timely and efficient feature classification and the scarcity of radar image interpreters point to the need for an automated system for feature extraction and classification.

Linear features such as roads, rivers, bridges, and railroads are major landmarks in such imagery. Extracting and analyzing such features are prerequisites for most analysis applications. Traditional linear feature extraction techniques (edge detection and region segmentation) tend to perform adequately for low noise, high resolution visible imagery. However, the relatively poor quality and the complexity of the observed scenes in radar imagery make these feature extraction techniques less effective.

The ability to automatically detect and analyze linear features will have a major payoff for numerous applications. Technology to provide such an automated capability is emerging from the fields of image understanding (IU) and artificial intelligence (AI). Such a system could incorporate knowledge about the scene and use context (from the image or external sources such as digital terrain maps or terrain object models) to intelligently guide and interpret the extraction process. The results of the Phase I SBIR effort were encouraging in showing the feasibility of this approach. An automated system would greatly enhance the

Army's capability for aerial cartography, change detection, aerial surveillance, and autonomous navigation. The goal of this effort is to pave the way for such a system by developing a largely automated terrain/image analysis workstation prototype.

There has been much work in artificial intelligence, computer vision, and graphics that satisfies the individual requirements for object modeling capabilities. Little has been done to integrate these diverse fields, especially for the domain of SAR imaging. To date, the only vision systems that can interpret natural scenes limit themselves to very restricted environments while other systems are restricted to artificial objects and environments. A system which uses well defined shape attribute inheritance among a set of progressively more complex object models, and which generalizes attachment relations to handle uncertainty begins to fulfill the basic requirements. This system must also generate constraints on image features from object models. Care must be taken so that constraints on image structures generated from the abstract instances of object models are specific enough to generate initial correspondences between models and image structures. A rich set of image feature descriptions and robust object models that can adjust the segmentation process directly during their instantiation are also crucial to an automated system. Such object models are to be produced by ADS during the Option II phase of this effort for a limited set of features. These minimal object models must be able to direct constrained searches against image data. Models must eventually be capable of supporting learning and handling uncertainty in the matching of image feature descriptions to multiple terrain features.

The basic motivations for such a system stem from the poor results associated with the undirected application of low level image processing techniques. Environmental objects such as roads and rivers are semantic entities whose extraction requires contextual and object-specific knowledge which cannot be easily incorporated into, for example, low level filtering operations. In fact, it has become clear that a general and expandable system will have to incorporate processing which reflects the actual reasoning involved in expert SAR image interpretation.

The purpose of the Phase II Effort and particularly of this final option is to complete the design of an automated linear feature extraction system for SAR imagery and to demonstrate this design in a prototype software embodiment.

1.2 APPROACH

The major steps of the Phase II effort have been to:

1. Develop the appropriate working environment to manipulate and process imagery.
2. Develop and experiment with various segmentation and feature extraction algorithms.
3. Determine significant terrain object feature properties and construct representative object models.

4. Experiment and evaluate model to image feature matching schemes.
5. Develop an approach for managing competing and conflicting hypothesis matches.
6. Develop feature finders/predictors to support or contradict an expected terrain feature's existence.
7. Implement a display interface to support the above processing steps.

Once the proper environment has been established, the system for determining and extracting terrain features can be extensively tested. The purpose of these experiments is to further establish the role of autonomous feature extraction from SAR imagery and, indeed, the importance of SAR imagery to map generation.

1.3 SUMMARY OF ACCOMPLISHMENTS

The major results and research accomplishments of this effort are:

- demonstration and delivery of a software capability for extracting forested areas, bridges, and roads from SAR imagery
- development and delivery of a LISP machine-based testbed environment for undertaking experimentation and analysis on terrain feature extraction
- development of a set of SAR-based extraction algorithms
- development of a research plan and recommendations for completing implementation of a real-time terrain feature extraction capability

The following describes in more detail the efforts accomplished on both the Phase I effort and on the three portions of the Phase II effort (Base, Option I, and Option II).

1.3.1 Phase I

The major accomplishments of the Phase I effort were to:

- Review and implement several edge and region extraction routines from optical image processing on SAR aerial imagery. Routines were evaluated for their performance in order to determine which would be valuable for integration into the general system.
- Obtain a better understanding of the nature of SAR aerial imagery and its requirements for interpretation.

- Investigate a variety of techniques for representing the properties of environmental objects such as roads and rivers in SAR imagery.
- Design and begin component implementation of a model-based vision system for the extraction of linear features from SAR aerial imagery. In particular, ADS implemented an initial image structure data base and experimented with associated perceptual grouping rules and simple SAR object models.

A comprehensive report of Phase I results is available [14].

1.3.2 Phase II - Base Contract

The work performed by ADS under the Base Contract portion of Phase II addressed three different problem areas.

The primary work area focused on the continuation of the design produced in the Phase I SBIR effort. The results of that design are described in [26].

The second major area in which ADS pursued the project goals was the development and the design of a software environment in which to perform experiments and begin to build the eventual prototype system. The basic framework of this software was delivered to ETL in May 1987. The delivery emphasized neighborhood and display operations. The software also contained the necessary software "hooks" for future expansion into the other system components.

Finally, the last area of work undertaken as part of the Base Contract was the continued experimentation with the government provided radar imagery. Experimentation included algorithm surveys, hand processing sample imagery, and actual algorithm implementation. This work and ADS's general understanding of machine vision, has been continually supporting the design and development of the components of a model based vision system for linear feature extraction.

1.3.3 Phase II - Option I

The bulk of the work accomplished under this effort (described in [27]) pertained to the continuing effort to embody the system design in software. A major software delivery to ETL of the processing framework was made in September 1987. The software included the following:

- Many of the relevant image processing routines used at ADS (converted to run under Version 7 of the Symbolics LISP Machine operating system).
- The software for creating, manipulating, accessing, and editing image structures (also called "perceptual structures").
- The preliminary framework of the hypotheses database. (This database

contains hypotheses about extended image structures. Functions that provide for the creation of these structures are embodied in the "filter" functions.)

- Enhanced user interface to display the image structures.

A User's Guide was prepared to accompany the delivered software. The guide is written with the expert Symbolics Lisp Machine user in mind. At the suggestion of ETL, a supplemental guide was issued to address the needs of those users not intimately familiar with the Symbolics environment. In addition to the documentation, ADS held two tutorial sessions at ETL. The first session was a general "demonstration" of the software delivered. The second session was oriented towards familiarizing the user with the software. Given the size and complexity of the developmental environment, a subsequent visit was scheduled in December 1987 to further assist ETL personnel in the use of the system. During this visit some software "bug" fixes were also accomplished.

Work was also initiated on the recognition procedures. The details of the various terrain features were studied. In addition to the standard properties of the individual features, of particular interest were the internal and external structures of the features. For example, the apparent image-based structure of a patch of forest may be comprised of the textured area representing the bulk of the forest, the bright leading edge of the patch, and the trailing shadowed region. All three portions have entirely different "visual" characteristics, but each is an important component of the recognition of the forest patch. An example of external structure is best illustrated by a bridge. Typically, a bridge is detected as a long, thin bright region. However, this is not a unique signature by itself. If this bright region has roads extending from both ends and is surrounded on each side by water, then a unique signature for a bridge begins to form. The bulk of the effort in object structure recognition took place in Option II.

A source of difficulty for the Linear Feature II project was the incompatibility of software environments at ADS and ETL, which occurred when ETL installed the Version 7 Operating System (OS) on the Symbolics LISP Machine. ADS's commitment to deliver software onto the Version 7 OS required extra effort and overhead to port software between versions, which caused delays in the delivery of important additions and bug fixes to ETL.

1.3.4 Phase II - Option II

During the final segment of this multi-phase program, ADS completed its analysis of the knowledge and model structure required to represent terrain and cultural features. The model consists of a number of layers corresponding to a hierarchical view of the appearance of the object and its surroundings. The recognition of the terrain feature is accomplished by an architecture which attempts to match the predictions contained within the feature model with structure extracted from the image. However, there is no a priori constraint on which layer of the model is active at a given time. In practice, we implemented the architecture as a bottom-up extraction step followed by a top-down model verification step for simplicity and clarity.

The recognition according to the feature models was accomplished by the construction of "finders" - algorithms which find instances of natural and man-made features in the SAR imagery. Three finders were developed for forests (as instances of natural area features), for roads (instances of man-made features which are of extended length), and bridges (man-made features of local extent). The demonstration of the finders completes this ADS study into automated linear feature extraction.

ADS delivered and installed all software to the customer site and described its philosophy, approach, and methodology at a final review open to ETL personnel. ADS also described the operation of the software to interested personnel at a separate tutorial session.

This work has made two important advances. First, a processing paradigm has been developed which compresses and encodes the input into symbols representing meaningful chunks. The net effect of this transform is to permit more advanced segmentation and grouping algorithms to make the process more reliable and efficient. Examples of such structures are textured regions which correspond to interiors or boundaries of fields, forests, dirt road segments or similar features.

The second important improvement has been the use of high level feature models to guide the extraction, interpretation and attribution of the features. In object recognition applications, these feature models are the shapes that are stored in the shape library. For those terrain objects whose shapes are typically constrained (e.g., buildings, bridges and radio towers), the feature models specify, in fact, the shapes which will be positioned in the scene model. However, for terrain objects whose shapes are less constrained (e.g., rivers, roads and forests), the feature models contain rules to guide the image structure extraction routines and also rules to specify how image structures corresponding to the same feature are to be merged. The major advantage over strict bottom up processing is that knowledge from multiple sources can be used to guide the feature extraction. Another advantage is that feature models can share domain knowledge with the user and thus anticipate user interactions, thereby simplifying them.

This effort has identified key elements of a plan leading to the autonomous intelligent behavior required of an Automated Linear Feature Extraction system.

1.4 CONCLUSIONS

1.4.1 Lessons Learned

We draw a number of conclusions from this work:

- SAR as an imagery source yields significant information about numerous classes of terrain features.
- Terrain feature recognition is a complex process requiring knowledge not only of the feature's intrinsic "signature" but also of its component

structure and its relations to other local terrain features. In fact, the recognition of terrain features appears to depend as much on external relationships to other neighboring objects, such as adjacency or collinearity, as on intrinsic factors such as shape and texture. This contextual information can make up for information loss due to low resolution.

- The recognition process itself cannot be strictly "bottom-up" but must allow reliable knowledge from any source to guide the recognition.
- A model-based reasoning paradigm is most appropriate for terrain feature recognition and should incorporate:
 - o levels of abstraction
 - o 3D geometric/sensor reasoning
 - o heuristic models as well as analytic models
 - o models of feature interrelations and scene-level context

1.4.2 Expected Benefits

The significance of a competent system for automatic exploitation of terrain imagery will be felt in a wide variety of military applications:

- Situation Assessment and Target Recognition
 - o Area Limitation (Identify and prioritize those areas likely to contain the targets; delete from consideration those areas which cannot contain targets.)
 - o Hypothesis Verification (Process the area surrounding a candidate target to determine if the identification is consistent with the surrounding terrain.)
- Database Generation (Describe the contents of the field of view as a juxtaposition of terrain features.)
 - o Map Making and Updating (Use the description to form or edit a geographic database.)
 - o Mission Planning and Visualization (Use the description in conjunction with an elevation or other 3-D map to simulate views other than the original sensor perspective.)

1.5 RECOMMENDATION FOR FUTURE RESEARCH

1.5.1 What is Needed

A continuation of the effort begun here should strive to extend the existing terrain feature models, especially by incorporating higher level knowledge (e.g., roads lead into bridges, bridges cross rivers; rivers may be dry in the summer-time). This knowledge may be available from existing terrain data bases, elevation maps, or reports. Also, while little effort was directed to issues of control and focus of attention, the ability of a system to maintain robust recognition behavior depends on a clear strategy to handle competing hypotheses and select a sequence of actions which converge to a "best" scene interpretation.

We recommend that an evolutionary approach be taken to building the Automated Terrain Feature Extraction system. In particular, the first steps should address the extension of the current algorithmic base to include additional segmentation approaches upon which new finders will depend. The crucial paradigm is to develop intelligent algorithms with meaningful parameters which can build objects with descriptions used for further reasoning steps. In this way, a rich vocabulary of primitives will be available for further knowledge-based terrain feature models.

1.5.2 A Plan for Automatic Terrain Feature Extraction

The following plan (see Figure 1-1) discusses the steps leading to an operational system. The key to building an automated feature extraction system is to have competent low-level segmentation and processing algorithms which accept high-level guidance from knowledge-based models. The competency of the algorithms is not inherent but comes about because the algorithm can extract image structures related to model predictions when given the right set of parameters. This effort has developed a set of such algorithms based on the needs of a few finders. This tech-base is barely sufficient to the finders studied. More complete and knowledgeable finders will require some additional lower-level algorithms to supplement the algorithm base.

Once this is done, the testbed will consist of a rich mix of algorithms and models. The performance of these components must be studied and characterized based on a larger set of test imagery. The major performance factors which are relevant at this point are:

- the ability of the segmentation algorithms (even under human guidance!) to extract competent descriptions of known existing features,
- the intelligence of the models to specify parameter values for the lower-level segmentation algorithms which are similar to "optimal" choices made by human operators,
- the end-to-end recognition rate for known targets and false alarms,

- **EXPLOIT THE EXISTING TESTBED**
 - **EXTEND EXISTING TERRAIN FEATURE FINDERS TO WIDER KNOWLEDGE BASE**
 - **BUILD ADDITIONAL LOW-LEVEL EXTRACTION ROUTINES FOR THE FINDERS**
 - **CHARACTERIZE THE PERFORMANCE**
- **EXPLORE ALTERNATIVE HARDWARE**
 - **INEXPENSIVE GENERAL PURPOSE WORKSTATIONS - SUN, MAC II**
 - **ADVANCED ARCHITECTURES - WARP, CONNECTION MACHINE**
- **BUILD THE COMPLETE PROTOTYPE**
 - **ADDRESS OUTSTANDING RESEARCH ISSUES**
 - **PROTOTYPE ADDITIONAL TERRAIN FEATURE FINDERS**
 - **EVALUATE PERFORMANCE SHORTFALLS**
- **BUILD A REALISTIC SYSTEM FOR TERRAIN DATABASE GENERATION**
 - **REQUIREMENTS ANALYSIS: IMAGERY, TERRAIN FEATURES, PERFORMANCE, USER INTERFACE**
 - **DEVELOP COMPLETE TERRAIN FEATURE MODELS**
 - **DESIGN AN OPERATIONAL SYSTEM**
 - **BUILD THE SYSTEM**
 - **TEST AND EVALUATE THE SYSTEM**

Figure 1-1: Research and Development Plan

- some indication of the major throughput bottlenecks.

One issue associated with the performance analysis and the further expansion of the testbed is the hardware environment. The current Symbolic hardware has become difficult to maintain and expensive to expand. It is recommended that an alternative hardware base be considered, specifically a Sun or Mac II environment. Lucid Common LISP would be the appropriate target language to which to convert the existing LISP code. This would also be an opportune time to consider the value of advanced architecture computers in improving the testbed environment.

Once the lessons of the performance analysis and the issues of the hardware environment are assimilated, the major remaining research issues should be addressed:

- Extend the terrain model base
 - o model additional terrain features
 - o explore additional segmentation strategies
 - o incorporate a formal method of evidence accrual
- Develop extended control systems
 - o build a general system controller
 - o add a model-driven re-segmentation capability
- Utilize existing terrain databases
 - o guide initial terrain feature extraction
 - o automate terrain database revisions and updates
 - o assist both terrain and target change detection
- Extend heuristic sensor models
 - o model IR, visible, multi-spectral, ...
 - o fuse multi-sensor information

Once the research issues are addressed and the performance of the prototype has been evaluated, it will then be possible to begin the design of an operational Linear Feature Extraction system.

1.6 ORGANIZATION OF THIS DOCUMENT

Section 2 describes the technical approach taken for the development of the Linear Feature Extraction (LFE) System. It begins with an overview of the architecture and continues with an in-depth description of each component. It concludes with a discussion of the recognition approach.

The descriptions and results of the prototype terrain feature recognition algorithms ("finders") developed for this project are presented in Section 3.

Section 4 contains a glossary of relevant terms. References are provided in Section 5.

2. TECHNICAL APPROACH

This section describes the technical approach designed by ADS to reliably identify terrain features in SAR imagery. The technical approach is developed in a top-down fashion. That is, the description begins at a conceptual level and then becomes progressively more detailed.

The first section overviews the conceptual system architecture and the main components in the proposed LFE system. The next section describes these system components in greater detail. The last section then describes algorithms developed to demonstrate the technical approach as shown in Section 3.

2.1 OVERVIEW OF CONCEPTUAL SYSTEM ARCHITECTURE

2.1.1 Motivation

The following general observations have influenced the development of the LFE technical approach:

- The representation chosen for terrain features is critical to their reliable extraction.
- Because individual low-level image features (e.g., edges, texture, homogeneity) are fairly sensitive to noise, reliable image interpretation should be based upon a preponderance of terrain feature evidence.
- Terrain features should have spatial descriptions which relate to their appearance in SAR imagery.
- The description of terrain features in imagery should occur at several levels of abstraction (e.g., pixels, regions, terrain objects).
- The models of terrain features should explicitly incorporate contextual constraints at all levels of abstraction (e.g., pixel-level constraints, region-level constraints, object-level constraints).
- The design should support parallel processing.

Before describing each of the components of the system architecture, it may be useful to define some terms used throughout the following sections.

Top-down (model-driven) information is derived from models of terrain features. For example, top-down information from a road model can be used to predict road width and to direct search in the image. Conversely, *bottom-up* (data-driven) information is derived from the imagery without specific reference to

terrain features. For example, bottom-up information can be used to recognize all parallel lines in the image. In summary, bottom-up information flows from imagery to terrain features and top-down information flows from terrain feature models to image structures.

High-level (more abstract) information is provided at the terrain object level. This level of abstraction allows the terrain features to be discussed as entities with specific functional attributes. Conversely, *low-level* (less abstract) information describes visible properties of the imagery without specific reference to terrain objects (e.g., bright areas). This level allows the description of the image to be in terms other than individual pixel values. There is a continuum between high-level and low-level information that forms the intermediate levels of abstraction. In general, any information which relates specifically to an object model is high-level; information which describes image properties can be mid- to low-level.

2.1.2 System Overview

The conceptual system architecture is shown in Figure 2-1. A *Query Manager* specifies the terrain features sought and the sensor context; execution of the query produces extracted terrain features. There are two core databases in the system: the *Image Structure Data Base* (ISDB) and the *Terrain Hypothesis Data Base* (THDB). These databases store intermediate processing results and interpretations derived by the system. There are four main processing components of the system which control the contents of these databases: the Perceptual Grouper, the Terrain Feature Modeler, the Hypothesis Manager, and the System Controller. In general, the interpretation process consists of the application of these main processing components to create and verify terrain hypotheses which are supported by the databases.

The next subsections briefly describe each of these components.

2.1.3 Image Structure Data Base

The Image Structure Data Base (ISDB) contains segmented image structures which describe properties of the input imagery. Image structure descriptions in the ISDB include the type of process which extracted it, when it was created, the relevant parameters for that image structure, and appropriate access descriptors. Some examples of segmented image structures are lines, parallel line pairs, homogeneous or textured regions, etc. These structures provide additional levels of abstraction which yield a precise and useful description of image structure.

The interpretation process affects the ISDB in two basic ways. In one, the Perceptual Grouper invokes bottom-up processes which use current ISDB image structures to create more abstract image structure descriptions; these more abstract, newly created structures are then placed into the ISDB. This bottom-up image description process is shown in Figure 2-1 by the bidirectional dataflow arrow between the Perceptual Grouper and the ISDB (the Perceptual Grouper will be discussed in greater detail). This bottom-up process starts with the SAR imagery as the initial contents in the ISDB. The other interpretation process uses a top-down Terrain Feature Modeler to interpret image structures in the ISDB. This top-down interpretation process is shown in Figure 2-1 by the dataflow

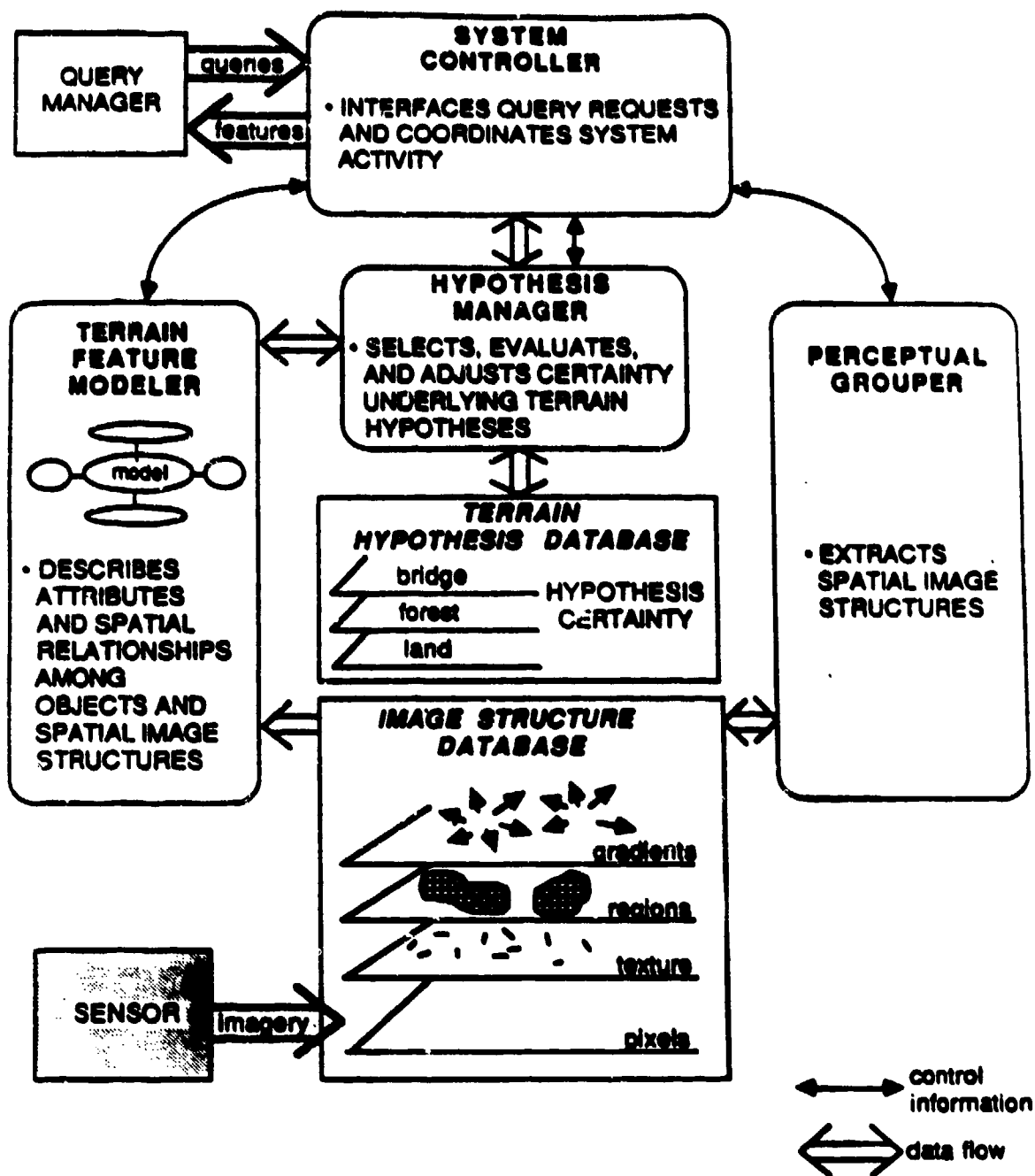


Figure 2-1: Conceptual System Architecture

arrow from the ISDB to the Terrain Feature Modeler.

Interactions with the ISDB take the form of queries and entries which specify particular image structures and relations. These queries are interpreted into the primitive attributes and relations used in the database and they are implemented in a library of functions and methods associated with the ISDB. For example, finding bridges can involve finding all BRIGHT image areas which are STRAIGHT and THIN¹. Note that when interpreting queries, it is important to consider how attributes such as BRIGHT, STRAIGHT, and THIN map onto particular parameter attribute ranges for image structures in the ISDB.

Results from queries to the ISDB are used in the recognition task to identify salient image features. Because queried structures in the ISDB can be displayed graphically, they form the basis of a user interface.

2.1.4 Perceptual Grouper

The Perceptual Grouper is the bottom-up component of the system which extracts spatial image structures.

The input imagery describes the image in terms of pixels of varying intensity. Though these intensities bear some physical relationship to the imaged environment, they do not directly provide a sufficient description of the image which can be used to define and reliably identify terrain models. The Perceptual Grouper provides a richer image description by aggregating portions of the image (e.g., pixels) into more abstract image structures (e.g., lines). As shown by the double arrow in Figure 2-1, the Grouper retrieves and places these image structures into the ISDB.

The Perceptual Grouper consists of a library of routines for extracting edges and lines, regions, texture, shape and other image features. A grouping routine inputs image structures and processing parameters in order to produce another image structure which describes another useful property of the image. Initially, the SAR image is the only image structure in the ISDB; the first grouping routines to be invoked are therefore low-level pixel-based processes (e.g., edge detectors). The resulting image structures are then placed in the ISDB for use by subsequently invoked grouping routines or the Terrain Feature Modeler. In general, grouping routines produce image structures which are more abstract than the input image structures.

Figure 2-2 provides a simple example of image structures which can be produced by Perceptual Grouper routines. *Edge* image structures can be produced by detectors which extract fairly straight local intensity boundaries; these edges are then placed into the ISDB. Because these edges form two colinear groups, they may be aggregated by a line finder into two *line* image structures; the edges can thus be retrieved from the ISDB, grouped into lines, and the resulting lines can be placed in the ISDB. Since these two lines are parallel to one another, a parallel

¹ In addition to the naturalness of class descriptors (e.g., BRIGHT), research in *fuzzy logic* has shown that the use of class descriptors to describe valued ranges results in better system performance [22].

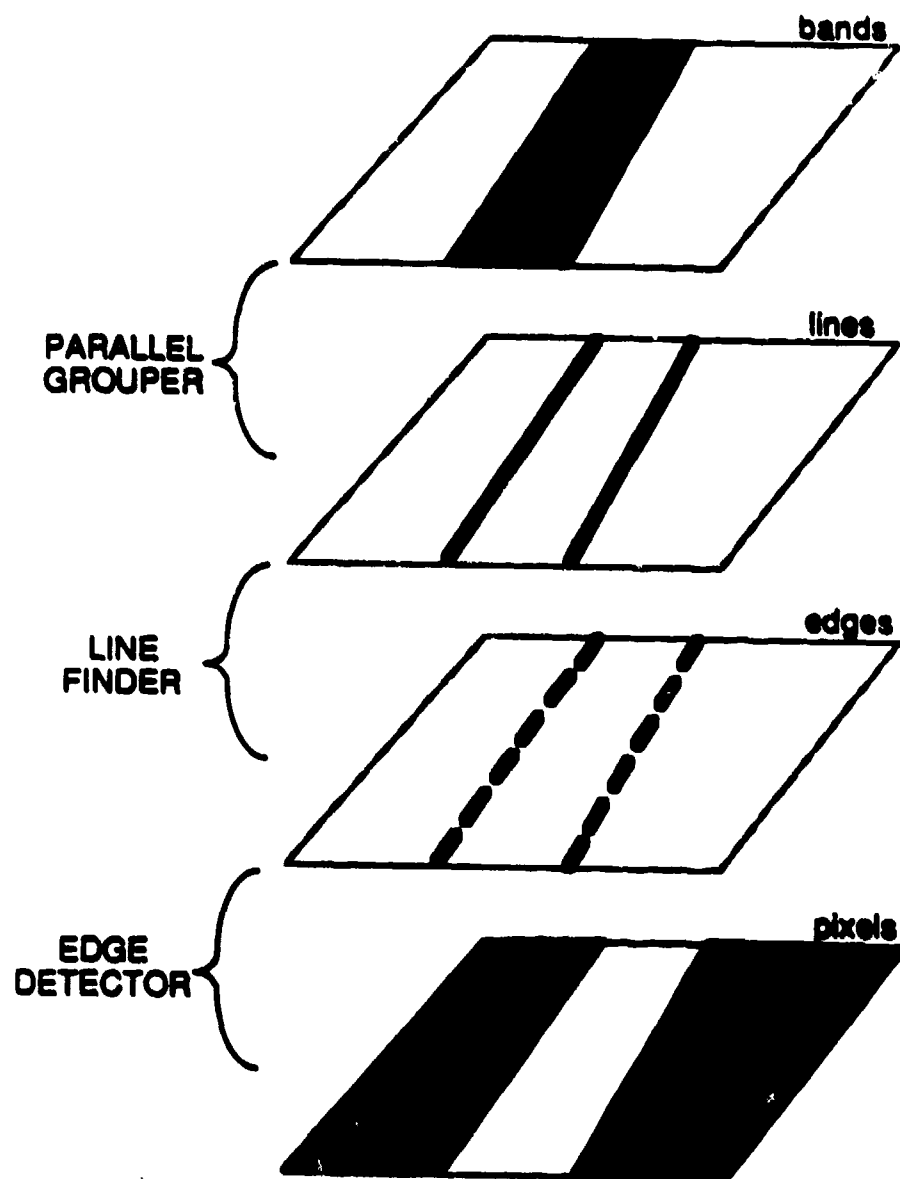


Figure 2-2: Simple Example of Image Structure Creation by the Perceptual Grouper

grouper can be invoked to form a *band* image structure; the lines are retrieved, grouped, and the band image structure is placed in the ISDB. Note that each routine in this example produces an image structure that is more abstract than its input structures; that is, edges are more abstract than pixels, lines are more abstract than edges, etc.

Section 2.2.1 discusses the Perceptual Grouper in greater detail.

2.1.5 Terrain Feature Modeler

The Terrain Feature Modeler predicts attributes and spatial relationships among terrain features and spatial image structures. The Modeler contains all information regarding terrain feature structure and characteristics. For each terrain feature, this information is embodied in models which:

- define a bottom-up mapping from image structures in the ISDB to terrain feature models (terrain hypothesis generation), and
- define top-down descriptions of terrain features which can be used to evaluate terrain hypotheses selected by the Hypothesis Manager (terrain hypothesis evaluation).

More details about this terrain feature representation are provided in Section 2.2.2 which discusses its underlying structural abstraction hierarchy.

The modeler has two primary roles in the interpretation process: bottom-up hypothesis generation and top-down hypothesis evaluation.

The bottom-up component of the Modeler examines image structures in the ISDB (as shown in Figure 2-1 by the arrow from the ISDB to the Terrain Feature Modeler) in order to infer the presence of terrain features. These terrain hypotheses are provided to the Hypothesis Manager (as shown in Figure 2-1 by the arrow from the Modeler to the Hypothesis Manager). Thus, the Modeler is responsible for bottom-up terrain hypothesis generation.

The Modeler is also responsible for evaluating terrain hypotheses that are selected by the Hypothesis Manager (as shown in Figure 2-1 by the arrow from the Hypothesis Manager to the Terrain Feature Modeler). Because initial terrain hypotheses can be incorrect (especially since they are derived from inherently noisy bottom-up information), accurate and reliable terrain feature recognition requires that hypotheses be judged for correctness. Evaluating terrain feature hypotheses requires specific terrain feature information; thus, the Modeler is responsible for evaluating terrain hypotheses. As an example, bridges are usually surrounded on either side by water and they are generally surrounded at either end by land. A bridge model which incorporates these structural relationships can evaluate a bridge hypothesis by seeking confirmation from surrounding spatial structures.

Section 2.2.2 discusses the Terrain Feature Modeler and its underlying representation in greater detail.

2.1.6 Hypothesis Manager and the Terrain Hypothesis Database

The Hypothesis Manager selects terrain hypotheses and evaluates them using top-down contextual terrain information provided by the Modeler in order to better determine the certainty underlying terrain hypotheses. These hypotheses are contained within the Terrain Hypothesis Data Base (THDB). The Hypothesis Manager is responsible for:

- selecting terrain hypotheses from the THDB whose certainty factors are to be evaluated and updated,
- evaluating terrain hypotheses by invoking the top-down part of the Terrain Feature Modeler which is responsible for testing hypotheses,
- using the Modeler evaluation of the terrain hypothesis to update the certainty assigned to the hypothesis in the THDB, and
- determining when the current terrain hypotheses are reliable and consistent with each other, the imagery, and other contextual information.

The Hypothesis Manager selects terrain hypotheses from the Terrain Hypothesis Data Base (THDB) (as shown in Figure 2-1 by the arrow from the THDB to the Manager). Because the evaluation of a hypothesis requires specific terrain feature information, the Manager obtains hypothesis evaluation from the Terrain Feature Modeler (as shown in Figure 2-1 by the arrow from the Modeler to the Manager). The Manager uses these evaluations to update the certainty assigned to hypotheses in the THDB (as shown in Figure 2-1 by the arrow from the Manager to the THDB).

Under some circumstances, the Query Manager may provide some initial object hypotheses. For example, when a terrain map is available, map features may be used for initial terrain hypotheses. The System Controller passes this contextual information to the Hypothesis Manager (as shown by the downward dataflow arrow in Figure 2-1) which then places them in the THDB.

The Hypothesis Manager is responsible for determining when terrain hypotheses are reliable and consistent with one other, with the imagery, and with other contextual information. The System Controller uses this determination to decide when terrain recognition is complete for a given feature. When recognition is complete, the Manager passes the extracted terrain feature descriptions to the System Controller (as shown by the upward dataflow arrow in Figure 2-1); the Controller then outputs the extracted features to the Query Manager.

Section 2.2.3 will discuss the Hypothesis Manager and the Terrain Hypothesis Database in greater detail.

2.1.7 System Controller

The System Controller has two primary tasks; it handles query requests and it coordinates system activity. As shown in Figure 2-1, the Controller receives queries from the Query Manager. After translating these queries into system operations, the Controller coordinates system activity and the interpretation process. Figure 2-1 uses thin arrows to show control information. The Controller uses information provided by the Hypothesis Manager, Modeler, and Grouper in order to determine when terrain recognition is complete. The Controller then obtains the final terrain hypotheses from the Hypothesis Manager (the upward dataflow arrow in Figure 2-1), formats them into output form, and returns the extracted features to the Query Manager (as shown in Figure 2-1).

Section 2.2.4 discusses System Controller issues in greater detail.

2.2 FUNCTIONAL COMPONENTS

The previous section presented a conceptual system architecture for the automated extraction of terrain features. This section provides a more in-depth description of the components in this architecture.

2.2.1 Perceptual Grouper

The original form of a SAR image is merely a collection of pixels of varying intensity. Because these raw pixel values do not form a sufficiently rich language to describe the appearance of terrain features, it has been well-established that more abstract image structures (e.g., edges, lines, textured regions) are required in order to support the description and recognition of objects in imagery [1, 2, 5, 14, 16, 17, 19].

Perceptual grouping is the systematic spatial aggregation of more primitive image structures into more abstract image structures [17]. Because the aggregation of pixels necessarily partitions areas of the image, this process can be considered a form of *image segmentation*. Image segmentation is concerned with breaking an image into structural components (e.g., regions, lines, edges) that can be used to support the interpretation process.

The Perceptual Grouper places image primitives into the ISDB which the Terrain Modeler uses to form bottom-up descriptions of terrain models. These bottom-up descriptions are used in the interpretation process to form hypotheses about where terrain features occur in the imagery. The Grouper has two important and related functions. First, it must partition the image into segments which bear a structural relationship to the appearance of terrain features in SAR imagery. For example, lines relate to roads, heavily textured regions relate to forest, dim image areas relate to water, roads, and shadows, etc.; [20] enumerates some relationships between image structures and terrain features. Secondly, the Grouper must provide a description of extracted features which can be used to construct bottom-up terrain feature models. For example, BRIGHT image regions can be identified, but a spatial description of these regions is required in order for the Modeler to ascribe semantic descriptions to these regions (e.g., for a bridge-like region, to determine: whether the region is straight, which are the ends and the

sides of these regions, what is the minimum width of a bridge, what is the orientation of the bridge relative to expected surrounding structures such as water or land). The next section, which discusses the Modeler, examines terrain modeling issues in greater detail.

A detailed study and definition of perceptual grouping and organization may be found in [17]. Section 2.3.1 describes some image structure extraction algorithms that were investigated as part of this LFE effort.

2.2.2 Terrain Feature Modeler

The Terrain Feature Modeler is responsible for representing and maintaining all terrain feature specific knowledge. Its bottom-up role is to relate image structures in the ISDB to object models in order to create initial terrain hypotheses. Its top-down role in the interpretation process requires that it evaluate terrain hypotheses provided to the Modeler by the Hypothesis Manager.

A large part of the LFE effort has concentrated on terrain feature model definition and their measurement and verification from imagery. This effort led to the development of a highly spatial description of terrain features which integrates bottom-up and top-down model description. These terrain models are based upon a well-developed structural hierarchy which relates the description of terrain features in a model to their measurable occurrence in SAR imagery.

The next subsection describes the structural hierarchy which underlies the terrain feature models used by the Terrain Feature Modeler. It describes a geometric and hierarchical representation of terrain knowledge; this hierarchical representation lays a strong algorithmic basis for understanding and implementing an automated terrain feature extraction system.

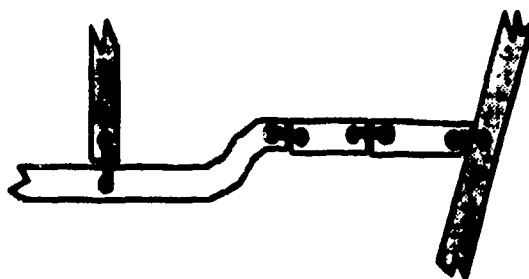
2.2.2.1 Hierarchical Representation of Terrain Features

The terrain feature representation determines what aspects of terrain features may be expressed. It also largely determines the ability to instantiate and measure components of terrain feature models.

The description of a terrain feature does not occur at a single level of abstraction. Terrain features, and objects in general, have an inherent hierarchy of structure. For example, cars are a type of vehicle, cars are comprised of bodies, engines, wheels, etc.; wheels are comprised of tires, rims, lug nuts, valve stems, etc; tires have tread, sidewalls, internal belt structure, etc. Similarly for terrain features, there are inherent hierarchies with which terrain features may be well modelled.

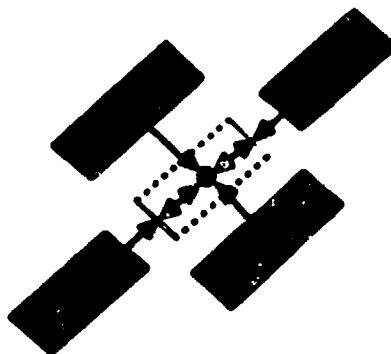
Figure 2-3 illustrates a structural hierarchy for terrain features. The basis of this structural hierarchy is the *locality of model information*. At the bottom level of the hierarchy are *image feature properties*. These properties encompass very local, pixel-based properties of terrain features in imagery (e.g., brightness, dimness, edges, texture). These image feature properties are then used to aggregate pixels into larger regions. *Region properties* describe the geometric and statistical properties of an aggregate of image pixels (e.g., its area, perimeter, shape,

**INTER-OBJECT
PROPERTIES**



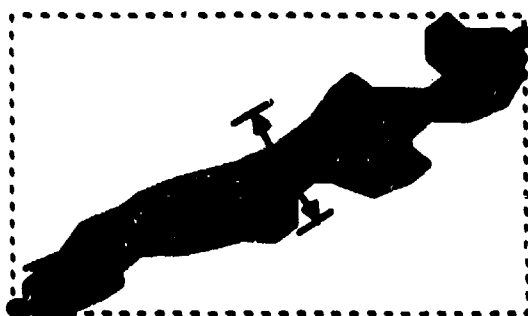
Bridges and
roads form
road networks

**LOCAL REGION
PROPERTIES**



Bridges link roads
across water

**REGION
PROPERTIES:**



area
perimeter
length & width
centroid
orientation
MBR
number & area
of holes
cohesiveness
straightness
mean/median/SD

**IMAGE
FEATURE
PROPERTIES:**



brightness/dimness
edges/lines
texture

Figure 2-3: Structural Hierarchy for Terrain Feature Models

length, width, centroid). These region properties provide a more abstract way in which terrain features may be described and modelled. The next higher level in the structural hierarchy describes *local region properties* which define the relative spatial relationships between neighboring image areas (e.g., the fact that bridges link road segments across water can be used to specify relationships between bridge regions and their surrounding objects). At the top level of the hierarchy are *inter-object properties* which provide global consistency checking and more abstract reasoning (e.g., bridges and roads form road networks which may link buildings).

A basic characteristic of the hierarchy shown in Figure 2-3 is that the *locality of constraints* increases when moving down the hierarchy to the level of image feature properties. If n is the number of elements at each level of processing, then the size of n increases when moving down the hierarchy. That is, since higher levels form larger spatial aggregates, there must be fewer elements at each higher level. As the locality of reference increases when moving up the hierarchy, the noise immunity can improve since there are more constraints which can be used to eliminate unmodelled data. For terrain features, these additional constraints at higher levels in the hierarchy provide greater data/reasoning abstraction and terrain hypothesis confidence.

The next sections describe each level in the structural hierarchy (shown in Figure 2-3 in greater detail).

2.2.2.2 Image Feature Properties

Image feature properties describe pixel-level properties of terrain features in SAR imagery. These properties are used to associate regions of an image which are related by some pixel-level metric to the physical appearance of terrain features in imagery.

The extraction of image feature properties uses pixel intensities within a small local neighborhood. For example, a simple edge detector looks at a 3 x 3 local pixel neighborhood. Image feature properties extracted by these detections are *suggestive* of terrain features, but they by no means provide a reliable indication of terrain feature presence. It is important to recognize that no single image feature works *all* the time. Noise, computational limitations, and other factors undermine the relationship of extracted image properties to terrain features. That is, the mapping from image features to terrain features is not very reliable. A key observation is that *robust terrain feature recognition is derived from a preponderance of image feature evidence*. For example, Figure 2-4 shows some image structures and their relationship to some terrain features. Bridges appear as BRIGHT image areas, which are HOMOGENEOUS, and bounded on either side by LINES. Though any one of these image features may be unreliable, together they form a more trustworthy indication of the presence of a bridge terrain feature. A more detailed enumeration of the mapping from image structures to terrain features may be found in [20].

As mentioned previously, the Perceptual Grouper is responsible for extracting image feature properties and placing them in the ISDB. These image feature properties are then used by the Modeler as the lowest level in the structural hierarchy for terrain features. Section 2.3.1 discusses some of these image feature

TERRAIN FEATURES	brightness	mod.intensity	dimness	high texture	med. texture	homogeneity	scale	lines
bridge	●						●	●
water				●			●	
road				●			●	●
clearing			●	●		●		
forest	●	●	●	●	●			
shadow				●			●	
power lines	●							●

Figure 2-4: Example of *Image Feature Properties* and their Relationship to Terrain Features

properties and their extraction from SAR imagery.

2.2.2.3 Region Properties

Contiguous pixels which share similar image feature properties can be aggregated into *regions*. These regions provide a more abstract tool for describing terrain features. *Region properties* describe terrain feature properties for these aggregated regions of pixels.

Region properties descriptively embody important aspects of terrain feature structure. For example, if we wish to define a terrain model for bridges, there are region properties that capture important aspects of bridges. Though bridges usually appear as BRIGHT image regions, all BRIGHT image regions need not be bridges. It is possible to further distinguish and describe bridges by noting that bridge regions (and their associated BRIGHT appearance in imagery) should have particular properties; bridges are usually fairly straight, long, thin, bounded in width, etc.

The second important role of region properties is to provide a local coordinate system which allows the next level in the hierarchy to relate them to other areas in the image. For example, a line and its endpoints fit to a "bridge" region can denote the "ends" of the bridge and the "sides" of the bridge. This local region coordinate system captures an important aspect of terrain feature descriptions.

There are two basic ways to describe image region properties of terrain features: geometric and value-based. Geometric region properties describe the two-dimensional shape of the image region (e.g., area, length, width). Value-based region properties specify the characteristics of the underlying image feature properties contained within the extent of the region (e.g., average intensity, standard deviation of intensity). The following region properties have been found useful for describing terrain features:

- Geometric-based region properties:

- o area
- o perimeter (e.g., length, two-dimensional shape)
- o length and width
- o centroid
- o best-fit line and endpoints
- o upright minimum bounding rectangle (MBR)
- o number and area of holes (i.e., area within the outer perimeter not considered part of the region)
- o straightness

- o cohesiveness (e.g., the ratio of holes to region area)
- o perimeter facing illumination vs. perimeter occluded from illumination from region
- o degree to which perimeter borders other regions
- Value-based region properties:
 - o Properties which can be computed for an entire area or along the perimeter
 - mean/median
 - measure of variability (e.g., standard deviation, entropy)

[1, 12, 16, 19] and Section 2.3.1 discuss how some of these region properties can be computed.

2.2.2.4 Local Region Properties

Region properties describe contiguous parts of the image which are similar in some respect (e.g., bright regions). *Local region properties* allow relationships among regions and surrounding image areas to specify more complex terrain feature structure. These local relationships arise because terrain features do not occur in isolation. Rather, terrain features have functional relationships with other regions and areas of the image. These functional relationships specify the local contextual structure of terrain features relative to other regions, features, and image areas.

Support areas specify the locations in which surrounding structures are expected to occur relative to a *central region*. In the example shown in Figure 2-5, one functional role of bridges is to link a road across a body of water. Put in terms of local region properties, bridges are usually surrounded on either side by water, and land/road surrounds them on either end. In this example, the bridge is the central region; water and road/land are the surrounding structures which can be found in support areas.

There are two primary issues which impact the specification of local region properties: geometric specification of the relationship between a central region and support area(s) and measurement of support area properties. Using Figure 2-5 to illustrate, there is the need to define the position and shape of the support areas relative to the central area (bridge), and the support areas need to be evaluated (e.g., test if they contain water and land/road).

2.2.2.5 Geometric Relationship Between Central Region and Support Area(s)

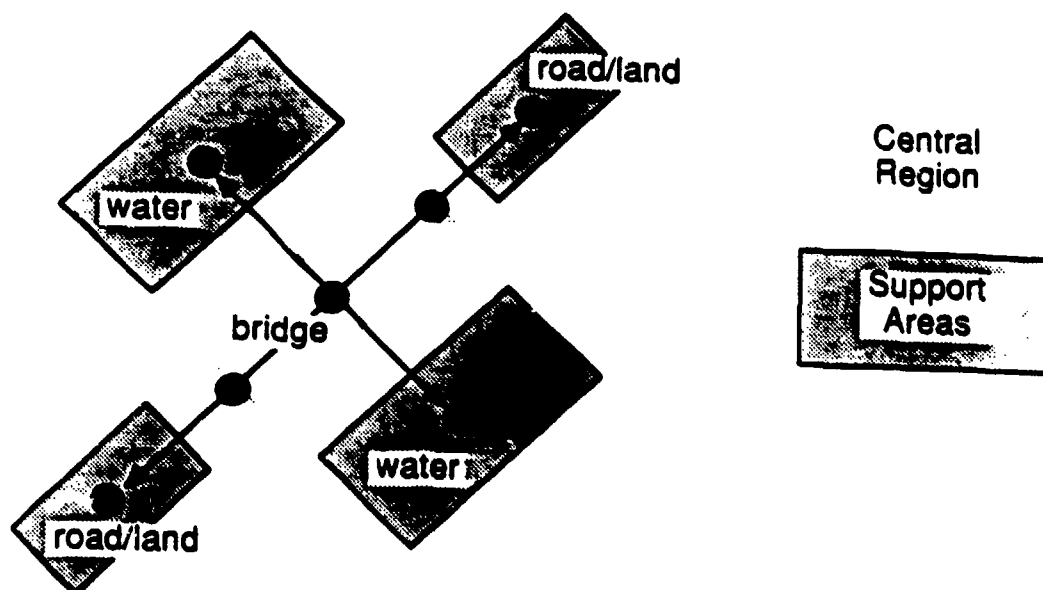


Figure 2-5: Local Region Properties for Bridge Terrain Feature Example

Two techniques have been identified for geometrically specifying the position of surrounding areas relative to a central region. These are point and perimeter relationships. *Point relationships* relate the central region to its surrounding areas by locating a point in each surrounded area with respect to a point on the central region (as in Figure 2-5); point relationships are useful for specifying the geometry of relative position of regions. *Perimeter relationships* specify surround areas which closely follow the shape of the central region perimeter; perimeter relationships capture attributes of local shape (e.g., shadow).

Point relationships are described in polar coordinates. Let (c_x, c_y) be a point in the central region (e.g., its centroid) and (s_x, s_y) a point in the support area (e.g., the center of a rectangle). As shown in Figure 2-6, the relative geometric position of these two points can be specified by d_p and θ_p : d_p specifies the euclidean distance between (c_x, c_y) and (s_x, s_y) ; θ_p , the position of (s_x, s_y) about (c_x, c_y) .

After defining the point relationships, the shape and orientation of the support area(s) can be further restricted. Since the description of arbitrary support area shapes is extremely complex, a small set of prototypical shapes can be used to describe the support areas. For example, Figure 2-7 show three prototypical support area shapes that were found useful: the rectangle (which is specified by three parameters), the arc (also specified by three parameters), and the circle (specified by one parameter). These three support area shapes may be composed to create more complex support areas. For example, Figure 2-8 shows a complex support area (shaded) resulting from the intersection of an arc and a rectangle.

Figure 2-9 demonstrates how local region properties for a bridge terrain feature determine the point relationships and support area shape and orientations. The central region is the "bridge" region. This region has length l , width w , a centroid and two endpoints. The water and road/land support areas have been specified using rectangles. Water is expected to be on "both sides" of a bridge and road/land occur on "both ends" of a bridge. This can be seen by the point relationships between the central bridge region and the supporting areas. For example, the point relationship between the bridge and the top water support area has length $1.1(w)$ and it is perpendicular to the main axis of the bridge region. The shape of this water support region is specified by constants multiplied by central bridge region properties: the rectangle bounding the top water support region has height $.9(w)$ and length $.8(l)$. The definition of local region properties for other terrain features is analogous.

The support area shapes used for point relationships cannot express all local region properties. *Perimeter relationships* specify surround areas which closely follow the shape of the central region perimeter. These relationships can specify support areas which are highly related to the boundary structure of the central region (e.g., shadow).

Figure 2-10 shows how perimeter relationships are defined by three parameters: band width, region band separation, and a subset of the central region perimeter. *Band width* specifies the width of the support region. The *region band separation* specifies the distance of the support region from the perimeter of the central region. If the entire central region perimeter shown in Figure 2-10 is used, the resulting support region would surround the central region (as shown by the shaded and hatched areas). The third parameter when defining perimeter

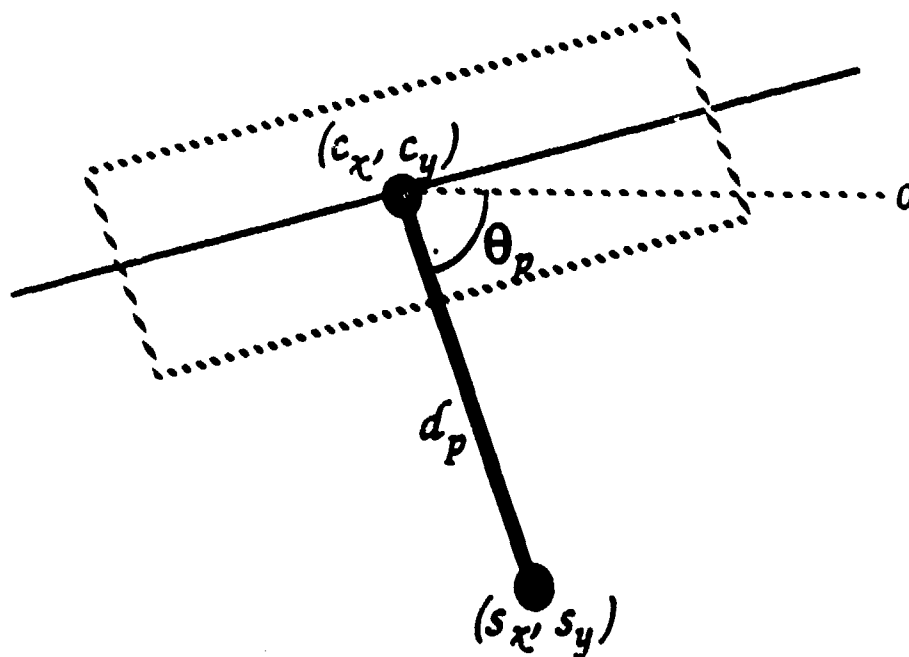
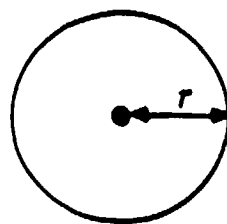
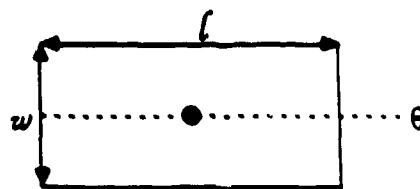


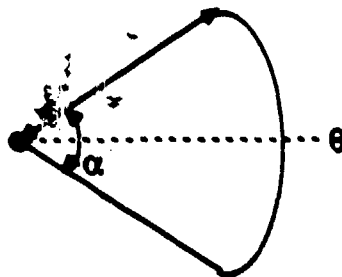
Figure 2-6: Specifying Point Relationships



circle
(radius)



rectangle
(width, length, orientation)



arc
(radius, orientation, arc angle)

Figure 2-7: Prototypical Support Area Shapes

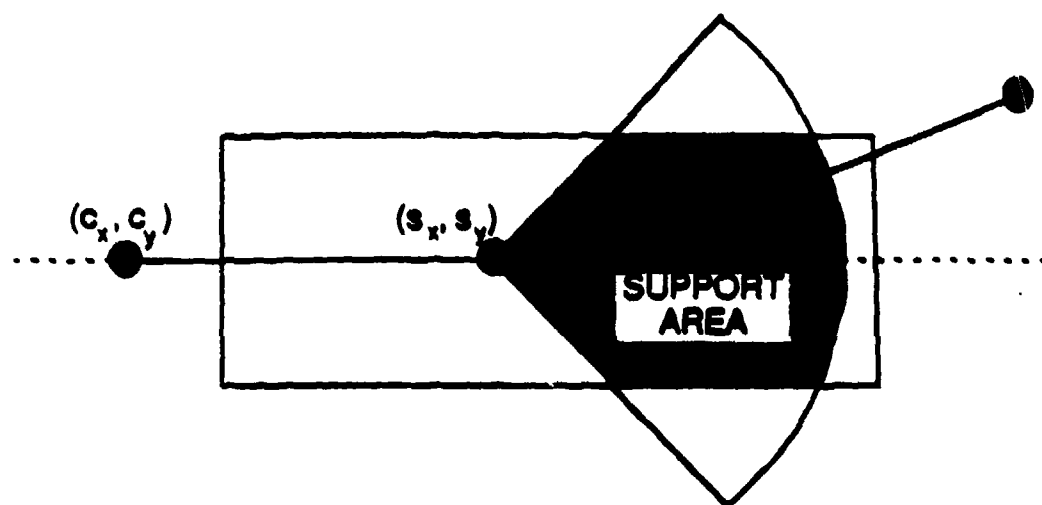


Figure 2-8: Complex Support Area

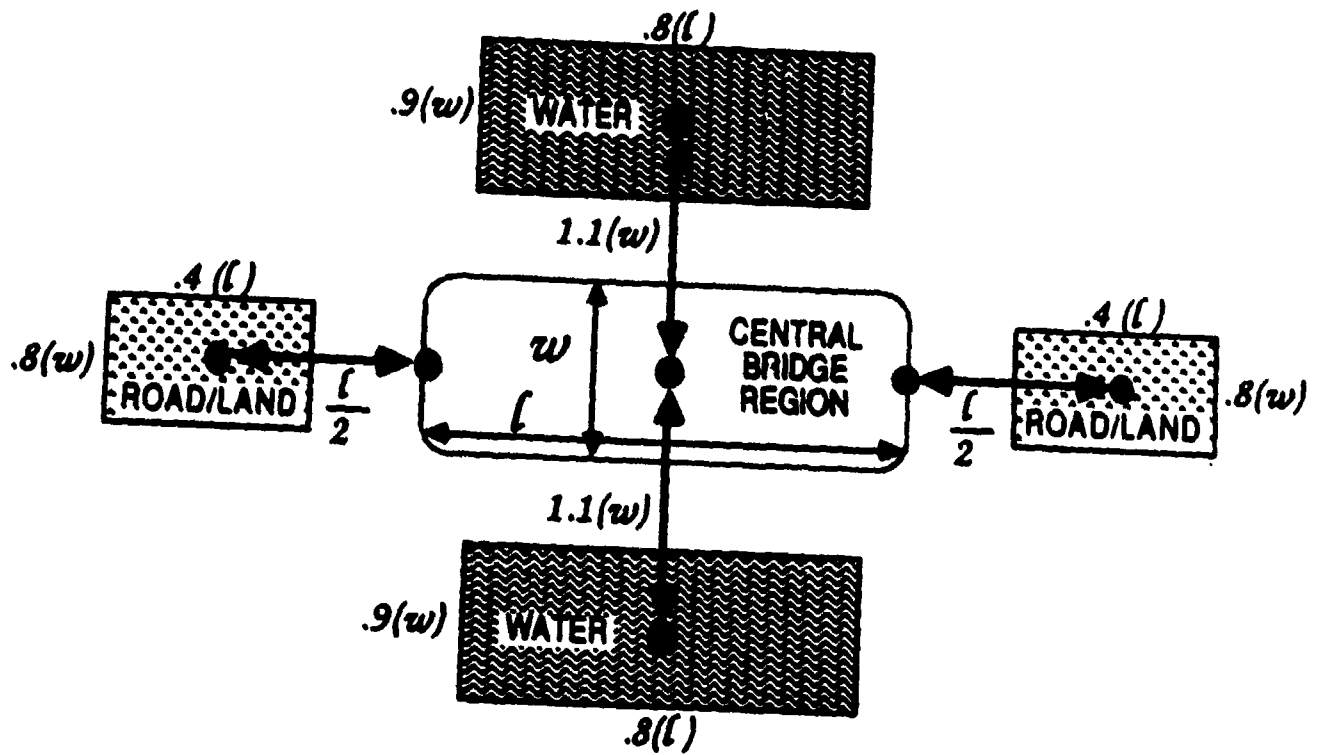


Figure 2-9: Support Areas and a Central (Bridge) Region

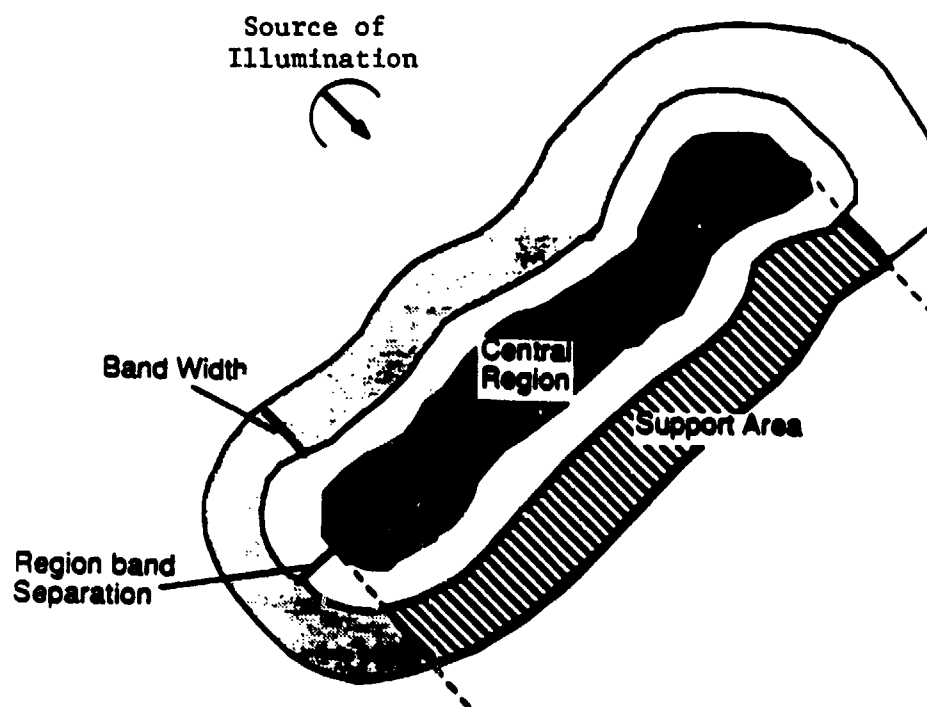


Figure 2-10: Perimeter Relationships

relationships allows a smaller subset of the central region perimeter to restrict the final support area configuration. For example, assume that the central region shown in Figure 2-10 is a forest region with the source of illumination as indicated. It could be expected that this forest would cast a shadow which closely follows the central region perimeter farthest from the illumination. This support area (the hatched area in Figure 2-10) can then be created by restricting the perimeter relationship to that part of the central region perimeter which is hidden from the source of illumination by the area of the central region. When the central region perimeter has concavities and the band width and region band separation are sufficiently large, it is possible that the resulting support area will overlap the central region; if this is not desired, mutual exclusion may be enforced by eliminating portions of the support area which overlap the central region.

2.2.2.6 Measurement of Support Area Properties

A support area specifies the location in which a surrounding structure is expected to occur relative to a central region. Expected structures in these support areas may be terrain features (e.g., water, road, land), structural clues afforded via the imaging process (e.g., shadows), or other structures which locally relate to central terrain feature regions.

In order to determine whether the support area contains the expected structures, properties of the support area must be measured and evaluated. The extent to which support areas contain expected structures can be used to evaluate the central region hypothesis. If a support area provides suitable evidence that the expected structure is contained within it, the support area is considered to uphold the conjecture of the central region hypothesis.

A good way to evaluate search regions for expected structures is to create an independent routine for each structure which can be expected to occur within a support area. Each routine is responsible for evaluating whether the expected structure is contained within a support area; it returns its evaluation as a measure of belief (e.g., "strongly indicative evidence of structure presence", "no indication of presence"). The form of each routine depends upon what structure is being evaluated. If there is a routine EVAL-SEARCH-AREA-FOR-WATER, it may choose to examine the dimness and homogeneity of the underlying image (since these are visible properties of water in SAR imagery). Rather than examine image-level properties, the routine can examine active hypotheses in the search area which support or deny the existence of the expected structure. Alternatively, when map information is available, this routine can use the support areas to form map database queries in order to evaluate the search area.

2.2.2.7 Inter-Object Properties

Though local region properties capture an important aspect of terrain features, more global terrain context captures an essential aspect of terrain feature representation. Terrain features do not occur in isolation. Rather, each terrain feature affects and is affected by other terrain features with which it is functionally related. For example, a bridge exists only to continue portions of a road network over otherwise intractable terrain, road networks connect destinations (e.g., buildings), vehicles travel on roads, roads cannot travel directly up steep

mountains, etc. These more global inter-object properties embody an important aspect of terrain features which is not described by local region properties. These properties are described at the terrain feature level, so they are particularly suited for answering user queries and the construction of maps.

2.2.2.8 Terrain Recognition using the Hierarchical Terrain Feature Representation

The previous subsection presented a structural hierarchy for the representation of terrain features. We now discuss how this hierarchy may be used to recognize terrain features.

Bottom-up terrain information provides object hypotheses and it is generally required to start the interpretation process. Conversely, top-down terrain information allows terrain feature knowledge and context to guide the interpretation process. Rather than advocate the use of one over the other, it is important to recognize that *both* bottom-up and top-down terrain information is generally required to provide reliable terrain feature extraction. Therefore, the terrain feature representation should integrate both sources of information.

Figure 2-11 shows how the hierarchical terrain representation may be used to recognize terrain features; bottom-up and top-down portions of the processing have been identified. Bottom-up mapping from image structures to terrain feature models produces initial terrain feature hypotheses. Top-down information is then used to evaluate and rank these hypotheses and integrate information across terrain feature types. The levels in the structural hierarchy which were discussed above have been labelled in Figure 2-11. The next sections discuss how these bottom-up and top-down processes are performed within the conceptual system architecture presented in Section 2.1.

2.2.2.9 Bottom-Up Terrain Modeling: Mapping from Image Structures to Terrain Feature Models

Initially, let us assume that there are no terrain hypotheses and the system is given an input SAR image. Though initial hypotheses may be provided using a map database or user-specified context, this only makes the problem more tractable. We assume that the first step in the interpretation process must extract more abstract image structures and map them to terrain models to provide initial terrain feature hypotheses.

The Perceptual Grouper extracts image structures and region properties (as shown in Figure 2-11 and discussed in the previous section) and places these image structures into the Image Structure Data Base (ISDB). As shown in Figure 2-11, the Terrain Feature Modeler uses the bottom-up portion of terrain models to map these image structures to models in order to form terrain hypotheses.

Figure 2-12 shows an example mapping from image structures to terrain feature models. The trees in a forest cast alternating shadows and high returns for unoccluded tree leaves; this results in a dappled texture in forest regions. The frequency of this texture can be modelled using sensor platform position, resolution, and bounded tree size; thus thresholded bandwidth limited SAR imagery

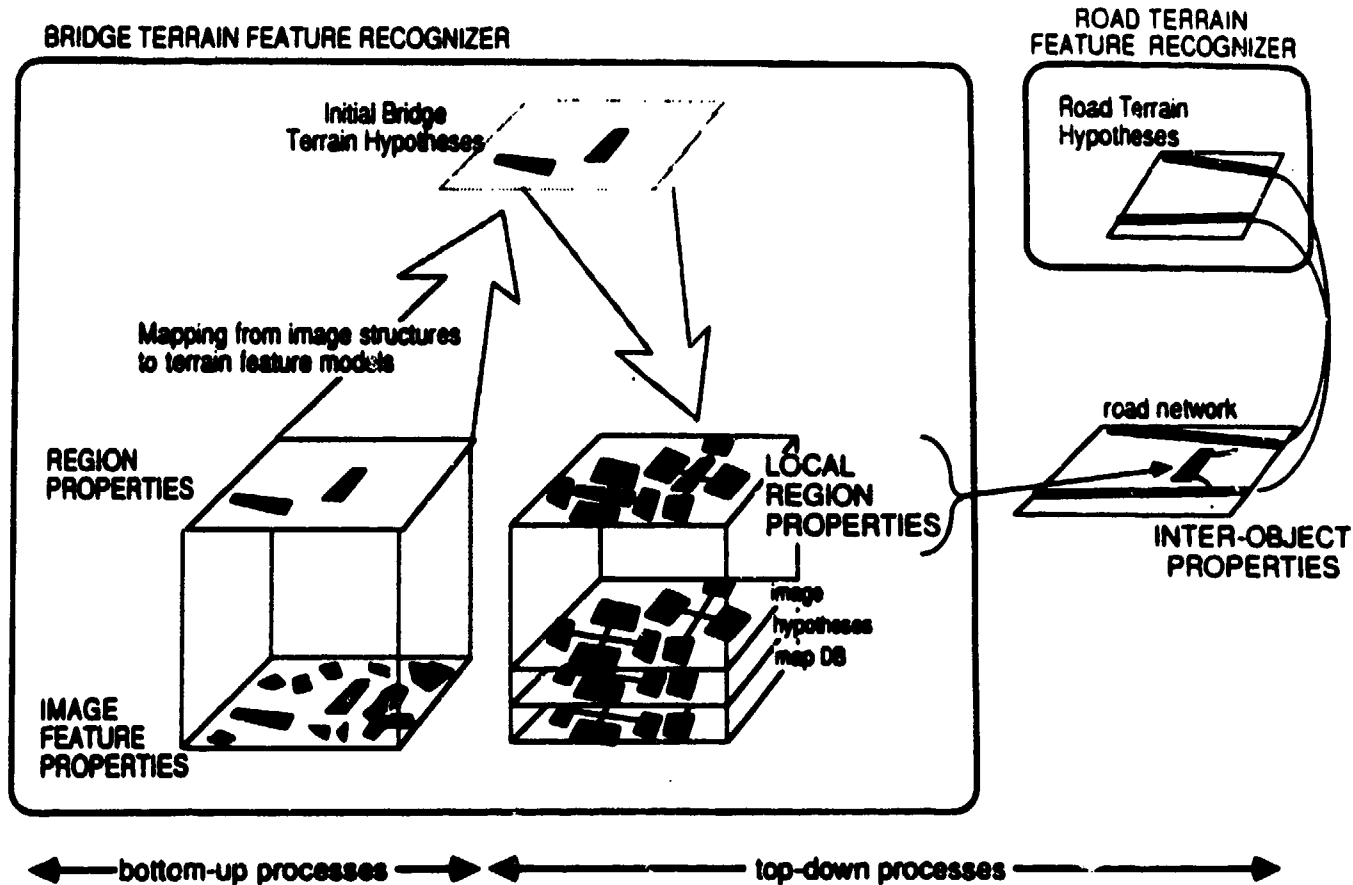


Figure 2-11: Using Hierarchical Terrain Representation to Recognize Terrain Features

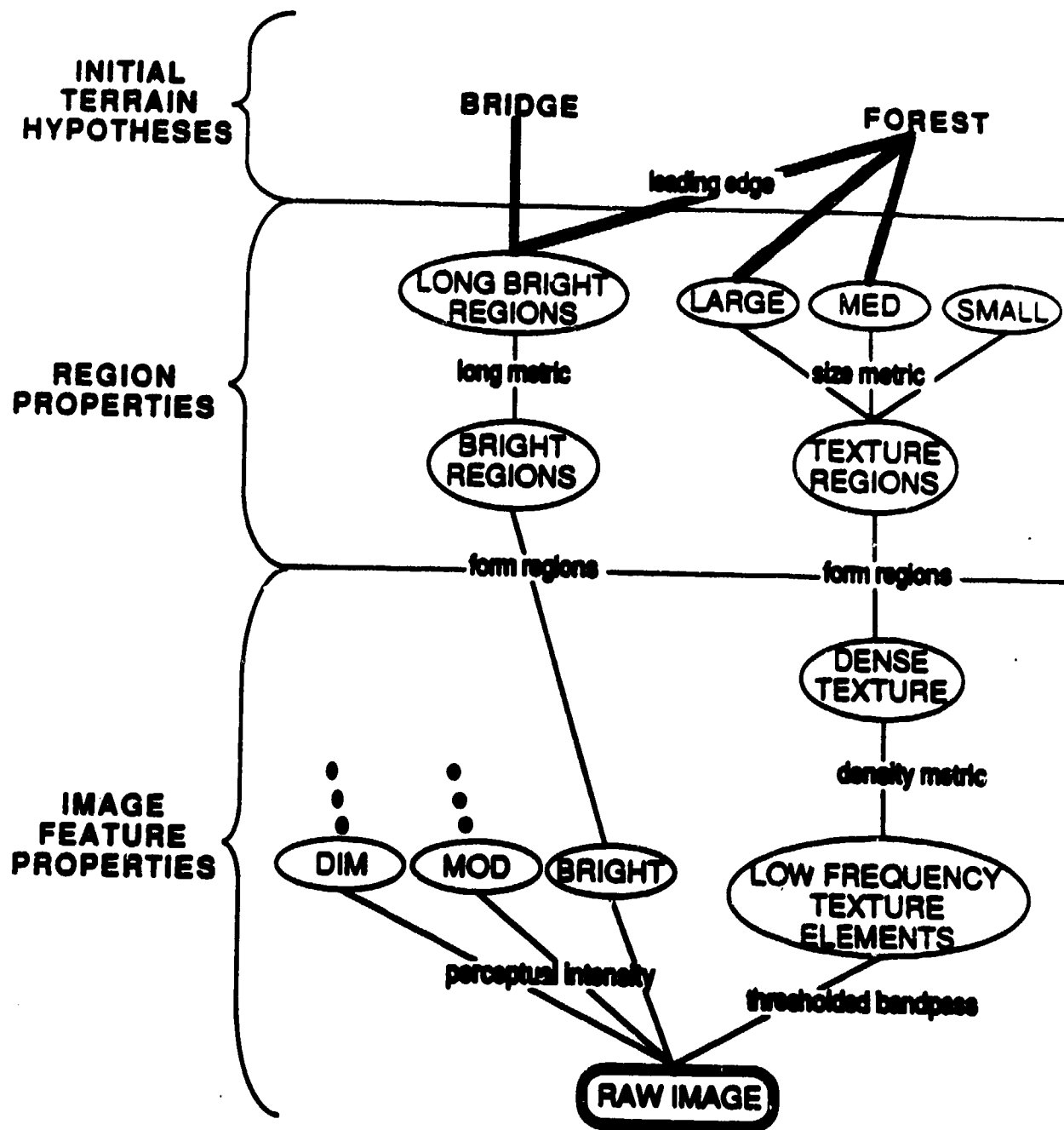


Figure 2-12: Mapping from Image Structures to Terrain Feature Models

provides texture elements which are highly associated with forest texture regions. Dense patches of these texture elements occur in forest regions. Since there is a minimum size on a forest region, medium and large dense bandwidth texture regions are thus suggestive of forest regions.

Continuing the examples shown in Figure 2-12, bridges tend to have many reflective dihedrals, so their return (i.e., brightness) from SAR sensors is generally very high. Bridges are also generally long and thin since the width of roads is generally far less than the spanned distance; bridges thus appear as long, bright regions. Additional constraints are available (e.g., bridges have bounded width, they are generally straight). These long bright regions also occur at the leading edge of forests (i.e., the side of a forest region which faces the illumination); this occurs because of the increased leaf surface and increased presence of dihedrals at the leading edges of a forest. Thus, long bright regions are suggestive of both forest leading edges and bridges. As shown in Figure 2-11, local region properties and inter-object properties can distinguish the most plausible hypotheses.

Bottom-up models of terrain features are built up in an analogous fashion for each terrain feature of interest.

2.2.2.10 Top-Down Terrain Modelling: Using the Terrain Model to Evaluate Terrain Hypotheses

As has been discussed, one role of the Terrain Feature Modeler is to evaluate terrain hypotheses selected by the Hypothesis Manager. This requires that top-down terrain structure information be used to determine the extent to which the contextual presentation of the terrain feature is consistent with the terrain hypothesis. As shown in Figure 2-11, this can be done using the local region and inter-object properties of terrain feature models.

The previous section described how local support regions may be constructed in order to specify local support areas and local region properties. Figure 2-9 provides an example of a local support area structure for a bridge terrain feature. Figure 2-10 illustrates a shadow support area for an example forest terrain feature. The extent to which expected structures appear in these support areas (e.g., water, road/land, trailing shadow) determines the confidence and supporting evidence underlying the terrain hypothesis.

The Modeler must perform three tasks in order to use local region properties to evaluate a terrain hypothesis. First, it must define the support areas for that terrain hypothesis (e.g., for a bridge, the water and road/land support areas). Second, it must evaluate the extent to which the expected structures appear in the support areas (e.g., to what extent does image, current hypothesis, and/or map database information support the conclusion that water is contained within the water support area). The third and final step requires that the evaluation of individual support areas be combined into an overall evaluation which specifies the plausibility of the central terrain feature hypothesis. As previously discussed, the Modeler returns this overall evaluation to the Hypothesis Manager which is responsible for hypothesis reasoning and maintenance.

Combining individual support area evaluations into an overall central terrain feature evaluation leads to some interesting observations. All support areas

do not contribute equally to the overall evaluation of the central terrain hypothesis. Using a bridge as an example, if one water area and both road/land support areas evaluated well (i.e., there is strong confidence that they contain water and road/land) and the remaining water support area shows no evidence of water, this does not translate into a "75 confidence" in the bridge hypothesis; instead, the failure to find supporting evidence of water in that single support area is sufficient evidence to seriously question the validity of the central terrain hypothesis. In other words, when combining support area evaluations, the Modeler must take into account the structural importance of each support area to the central terrain hypothesis. Each terrain feature must thus have a unique combination function which integrates individual support area evaluations into an overall central region hypothesis evaluation.

The top-down evaluation of local region properties attempts to eliminate terrain hypotheses which are not locally plausible. Because locally plausible terrain hypotheses may lack more global consistency, inter-object properties can be used to further evaluate current terrain hypotheses. As shown in Figure 2-11, this can involve the creation of more abstract terrain features (e.g., road networks) which are composed of terrain feature subparts (e.g., bridges and roads); consistency is then verified within this more abstract terrain feature structure.

2.2.3 Hypothesis Manager

As terrain recognition proceeds, terrain hypotheses are created. Initial hypotheses are subject to noise and error, so subsequent reasoning is required to achieve reliable terrain feature recognition. The Hypothesis Manager is responsible for selecting, evaluating, and adjusting terrain hypotheses which are maintained in the Terrain Hypothesis Data Base (THDB). Two issues are particularly relevant to the Hypothesis Manager: the representation of terrain hypothesis confidences, and, the methodology of hypothesis reasoning.

A well-known problem in early AI systems arose from the use of numeric confidences which had no well-known meaning or interpretation [2, 22]. For example, MYCIN expressed confidences in medical diagnoses as a numeric quantity; +0.5 expressed positive belief in a diagnosis, while -0.95 expressed a strong disbelief in a diagnosis. These numeric confidences have several problems. First, the meaning of each numeric confidence has no specific meaning nor interpretation; for example, does 0.75 mean that the system is very certain? Second, it is difficult to express the usefulness of relative confidences when the range of numeric confidences exceeds the number of discretely meaningful confidence classes; for example, does a hypothesis with belief 0.75 express more confidence than a hypothesis with belief 0.74? Third, because of the previous two problems, it is difficult to derive meaningful combination functions which are based on these numeric confidences. This last difficulty can be overcome in part using Bayesian approaches (e.g., as done in [3], though some argue that statistical properties do not strictly hold for AI problems).

An alternative specification of hypothesis confidences uses discrete ("fuzzy") classes [22]. Each "fuzzy" class expresses confidence as a discrete semantic unit. An example of a fuzzy confidence set with nine classes might be:

- DEFINITELY SO (POSITIVE AXIOM)
- STRONGLY INDICATIVE
- INDICATIVE
- WEAKLY INDICATIVE
- UNDECIDED
- WEAKLY NOT INDICATIVE
- NOT INDICATIVE
- STRONGLY NOT INDICATIVE
- DEFINITELY NOT (NEGATIVE AXIOM)

These fuzzy sets provide better reasoning abilities and they more accurately express the meaning underlying hypothesis confidences [22]. Combination rules are also more straightforward when using these fuzzy sets. For example, several WEAK INDICATIONs of a particular hypothesis may allow that hypothesis' confidence to be upgraded to INDICATIVE. These fuzzy approaches also make hypothesis reasoning less opaque and more available to human analysis and understanding. Because of these and other reasons, it seems that discrete confidence classes and fuzzy set theory provide a good representation for terrain hypothesis confidences.

The Hypothesis Manager reasons about terrain hypotheses in order to drive the system to reliable final hypotheses. The Manager is responsible for selecting, evaluating, and adjusting terrain hypotheses. The method by which the Manager performs this reasoning critically affects the successful and reliable extraction of terrain features.

Figure 2-13 shows how hypothesis evaluation and reasoning can be performed by the Hypothesis Manager. The Manager determines whether the terrain recognition task is completed by examining whether conflicts among competing hypotheses have been resolved and the current hypotheses have sufficiently high confidence. If terrain features do not overlap (i.e., if they are spatially mutually exclusive), conflicts can be simply detected when more than one hypothesis describes a given image position. If there is not mutual exclusion among terrain features of interest, a more explicit definition of competing hypotheses is required (e.g., explicitly defining which hypotheses are mutually exclusive and which are mutually compatible).

As shown in Figure 2-13, when conflicts have not been eliminated or confidences are not high, the manager then selects a central terrain feature hypothesis (H_i) to spatially reason about. A good selection strategy is to select the best (i.e., most confident) hypothesis. This best-first strategy supports *island-driving*, which allows hypotheses with high certainty (i.e., "islands") to "drive" subsequent hypothesis reasoning [2]. Care must be taken with such

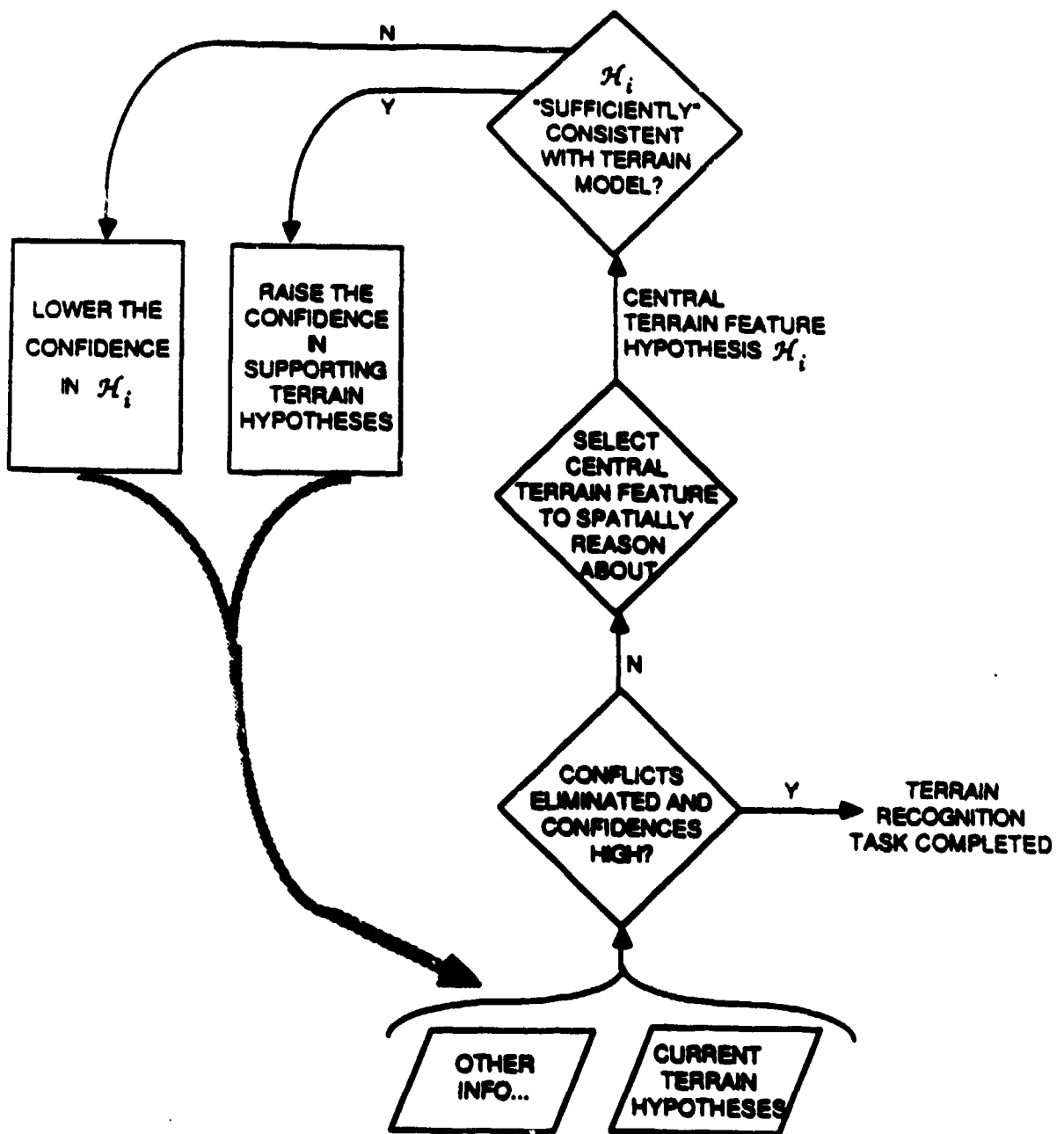


Figure 2-13: Hypothesis Evaluation and Reasoning

strategies that the same hypothesis is not repeatedly selected (e.g., after selection, a hypothesis cannot be reselected for a given period). Alternative hypothesis selection strategies can be employed [2].

After the central terrain feature hypothesis H_i has been selected, the Manager needs to determine whether the hypothesis is well-founded and consistent with the terrain model. Since this requires terrain-specific knowledge, the Terrain Feature Modeler must be queried by the Manager to answer these questions. As discussed earlier, the Modeler can evaluate a terrain hypothesis by examining the consistency of Local Region Properties and Inter-Object Properties with the hypothesis' terrain model. The Modeler then returns this hypothesis evaluation to the Manager.

As shown in Figure 2-13, if the Modeler finds that the central terrain hypothesis H_i is not sufficiently consistent with the terrain feature model, then the Manager should lower the confidence in H_i . Conversely, if the Modeler finds H_i is consistent with its terrain model, the Manager can raise the confidence in supporting terrain hypotheses. For example, if a bridge hypothesis is supported by the existence of surrounding structures (e.g., water and road/land), then the confidence in those surrounding structures may be raised (i.e., the confidence in water and road/land can be increased). Care must be taken that the hypothesis selection algorithms and the raising/lowering of confidences does not lead to hysteresis or unstable confidence reasonings.

The Manager cycles through the hypothesis evaluation and reasoning cycle until it determines that the terrain recognition task has been completed; the Manager then notifies the System Controller that it considers the recognition task completed. At this time, the current terrain hypotheses represent the final terrain features recognized by the ATFE system, and the System Controller can output these identified terrain features to the Query Manager.

2.2.4 System Controller

As mentioned earlier, the System Controller has two primary tasks: it interfaces query requests and it coordinates system activities (see Figure 2-1). Detailed discussion of system control is more appropriate for the next phase of research, though we have identified some techniques and observations which are relevant.

An important aspect of the architecture is that control can be largely decentralized to support distributed processing. For example, the system could be implemented as a segmented *blackboard system* with independent *knowledge sources* that autonomously read from and post to blackboards [2]. Implemented in such a way, the Terrain Feature Modeler, Perceptual Grouper, and Hypothesis Manager routines operate as Knowledge Sources which independently gather information; these knowledge sources then asynchronously post events and information to other knowledge sources via a blackboard. Using such a control architecture, the System Controller monitors blackboard posting and knowledge source processing in order to determine how to best allocate resources by coordinating the multiple parallel system activities. Work on these blackboard architectures has demonstrated their power and it has developed effective controller designs and strategies [25].

Independent of specific control architectures, the efficient allocation of computational resources is an interesting aspect of the controller. Meta-knowledge (i.e., knowledge about how the system operates and what costs are assigned to tasks) can be used to establish global strategies, intermediate processing results can be used to provide a *focus of attention* (e.g., *island-driving* [2]), and user queries can specify constraints which determine a more efficient system strategy (e.g., "I'm only interested in bridges").

2.3 RECOGNITION ALGORITHMS

This section describes algorithms which were developed and used to demonstrate the Phase II technical approach on bridge and terrain features in SAR imagery. Section 3 describes the application of these algorithms in order to automatically extract terrain features.

2.3.1 Perceptual Grouper

2.3.1.1 Image Feature Properties

Many low-level computer vision and image processing techniques, features, and extraction algorithms have been developed. For example, Canny edges [5], lines [12], homogeneous intensity regions [19], and other techniques for low-level image feature extraction have been implemented in the ADS Powervision environment [1, 14]. Rather than describe these published algorithms, we describe two new algorithms that were developed as part of the Phase II effort to determine BRIGHT and ROUGHLY TEXTURED areas in imagery.

2.3.1.2 BRIGHT Image Feature Property

When analyzing imagery, terrain features will often be described as "bright". In SAR imagery, these bright image areas correspond to high radar signal returns.

Figure 2-14 shows one approach for defining image brightness which divides intensity into fixed ranges. For example, BRIGHT could be defined as the upper 30% of the intensity range. This is a good approach when SAR intensity calibration is always the same and the brightness range bears a fixed, absolute relationship to the imagery. In general, however, this is not the case. Sensor fluctuation, changes in calibration, gross environmental changes (e.g., radar return changing after a rainstorm or morning dew), and other effects make this approach brittle and inaccurate.

An alternative way of determining BRIGHT image areas is suggested by the human visual system. It has been found that the perception of brightness depends upon the local rates of intensity variation [10]. Figure 2-15 demonstrates how this principle results in a well-known optical illusion. Though the two smallest squares are identical in gray value, the square within the darker background is perceived to be brighter than the square within the lighter background. This

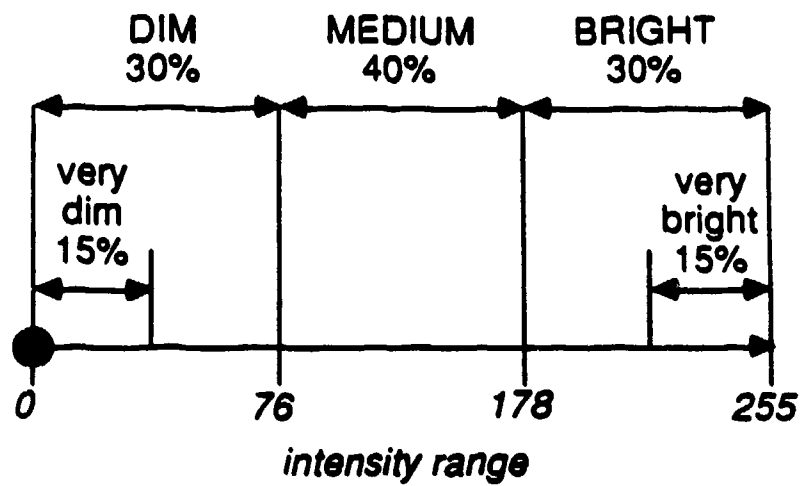


Figure 2-14: Defining Brightness by Dividing Intensity into Fixed Ranges

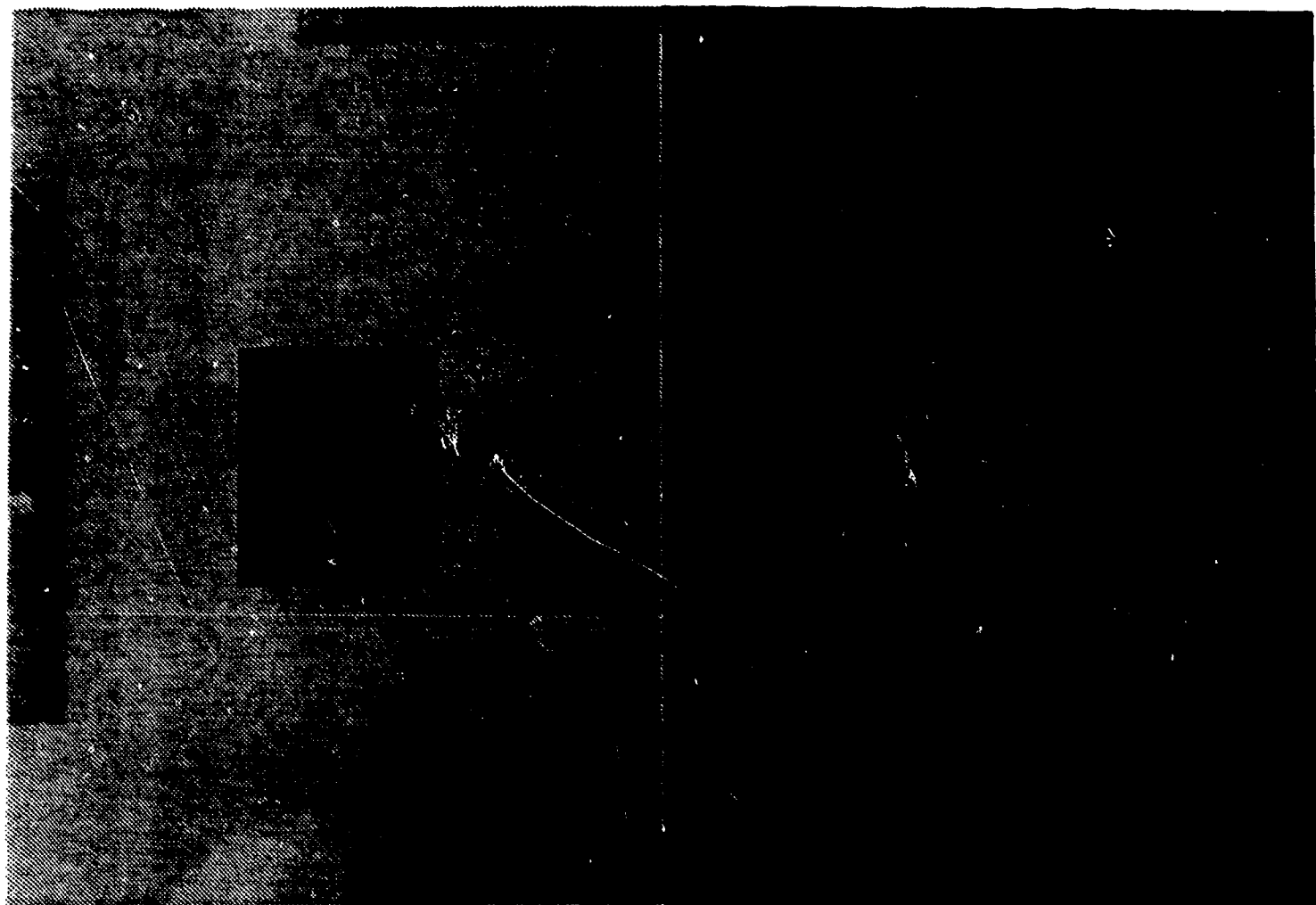


Figure 2-15: Brightness Illusion: the Two Center Squares have the same
Brightness

effect arises from the larger intensity discontinuity resulting from the darker background. This perceptual phenomena suggests that the determination of BRIGHT image regions should calibrate at image areas where large discontinuities in intensity occur.

Figure 2-16 illustrates the *perceptual brightness* algorithm which was developed to discriminate BRIGHT image areas. The algorithm calibrates by examining intensity changes at image areas where large intensity discontinuities occur. This calibration provides the range of intensities which are considered bright, and this range can be used to threshold the original image to provide a BRIGHT binary image (i.e., bright pixels are 1; others are 0). A connected components algorithm [1, 12] may then be applied to this binary image to produce aggregated pixel regions.

Let us specify the BRIGHT range as $([I], [I])$. The top of the BRIGHT range $[I]$ is assumed to be the greatest image intensity value. The task is then to find the lower bound $[I]$ of the BRIGHT range.

The gradient magnitude of image intensity specifies local intensity discontinuities which are used to specify the BRIGHT intensity range. As shown in Figure 2-16, the first step in the algorithm computes the gradient magnitude. Gradient magnitude is given by $M(x,y) = \sqrt{\Delta x^2 + \Delta y^2}$. Many techniques for computing the gradient may be used (e.g., Roberts, Prewitt, Sobel) [1, 12, 13], though some techniques have better immunity to noise and spatial aliasing [13]. In the experiments presented in Section 3, the derivatives were computed as $\Delta x = I(x,y) - I(x-1,y)$ and $\Delta y = I(x,y) - I(x,y-1)$. Good results were obtained even though this highly local method of computing image derivatives is particularly susceptible to noise and aliasing effects; alternative techniques may be used (e.g., Prewitt).

The bottom of Figure 2-16 shows a characteristic histogram of gradient magnitude. The rapidly decreasing occurrence of large gradients arises from the geometry of an image [12]. For example, the greatest number of large gradients occurs when the image is a checkerboard pattern of minimum and maximum intensity values. Quite the opposite effect is found in natural images in which the majority of image pixels are contained within fairly homogeneous image regions; this results in the characteristic histogram shown in Figure 2-16.

The perceptual brightness algorithm seeks to calibrate brightness at areas in the image which have large intensity discontinuities. This requires that all image areas whose gradient magnitude exceeds some threshold $[M]$ be identified as calibration points. The threshold $[M]$ cannot be directly specified without suffering from the same problems which occur for the fixed range technique shown in Figure 2-14. Instead, the gradient threshold $[M]$ can be simply determined by specifying the number of desired calibration points and using the gradient histogram to determine $[M]$. The perceptual brightness algorithm assumes that some fixed small percentage of the image is used to calibrate brightness; a percentage of 0.1% was used, though as will be described, the exact percentage is not particularly important. The total mass under the histogram shown in Figure 2-16 is the number of pixels contained within the image. All image pixels whose gradient magnitude exceed 0.1% of mass (this cutoff $[M]$ is shown in Figure 2-16) are used to calibrate brightness. That is, the gradient threshold $[M]$ is defined as the smallest gradient magnitude among the largest 0.1% of pixel gradients.

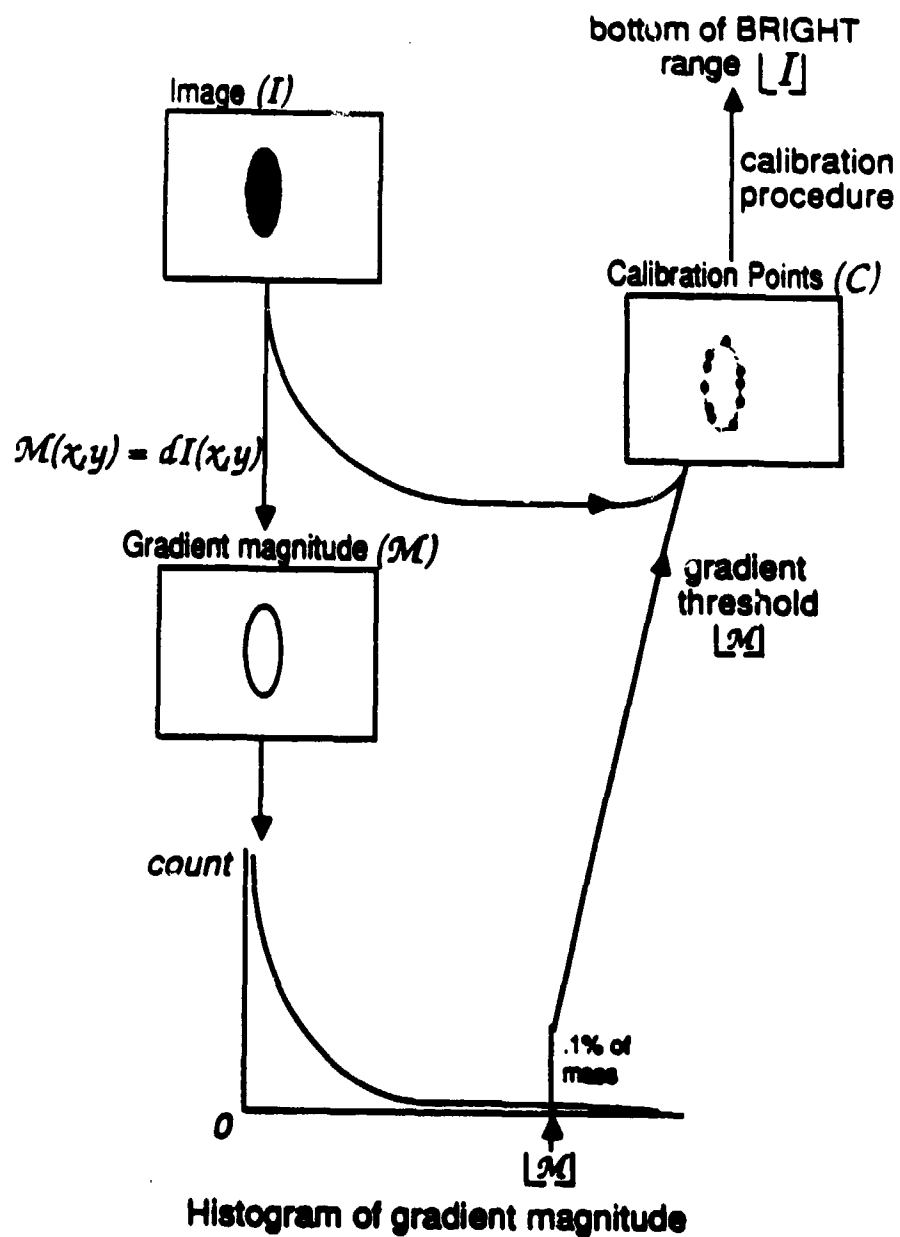


Figure 2-16: Perceptual Brightness Algorithm BRIGHT Range Definition

At this point, the calibration points have been identified (shown in Figure 2-16 as image C). Figure 2-17 shows the calibration points (in red) selected in a portion of a SAR image; 0.1% of mass was used. At this point, it becomes fairly evident why the technique is fairly insensitive to change in the mass percentage cutoff. Suppose, for example, that a five-fold increase in the mass cutoff was used (i.e., from 0.1% to 0.5%). This would result in a five-fold increase in the number of calibration points. As is evident from Figure 2-17, this would merely add additional calibration points along the same intensity boundaries of the objects. Though the algorithm is fairly insensitive to changes of this magnitude, it is clear that changes of many orders of magnitude would undermine the determination of calibration points (e.g., if 10% of mass was used). The main point to be made is that this mass cutoff may be empirically determined and applied to a large set of similar imagery while still producing comparable results. This stability is not afforded by the fixed range approach (e.g., as shown in Figure 2-14).

The final step of the perceptual brightness algorithm must examine the underlying image intensities at the calibration points in order to determine the bottom of the BRIGHT range $[I]$. As can be seen from Figure 2-17, the calibration points occur at the boundary of bright regions. Many techniques may be used to determine $[I]$.

In one approach, the bottom of the bright range may be defined as the average intensity among all of the calibration points; that is, $[I] = \frac{\sum C(x,y)}{n_c}$ where $C(x,y)$ is a calibration point and n_c is the total number of calibration points. Because the determination of spatial derivatives can affect the pixel registration between the gradient image and the intensity image (due to aliasing and the use of larger convolution masks), this approach can sometimes sample at slightly displaced image locations. The use of the average somewhat overcomes this effect; the median performs better than the mean in this respect.

An alternative technique was used to produce the results described in Section 3.3. Potential registration problems were overcome by processing the intensity image with a 3x3 maximum filter; that is, each pixel value is replaced by the maximum value within a 3x3 window centered at that pixel. This maximum filtered image is then measured at the calibration points. The bottom range of BRIGHT was then defined as the smallest maximum filtered image value sampled among all the calibration points. That is, $[I] = [F(x,y) | C(x,y)]$ where $F(x,y)$ is the maximum filtered image which is sampled at every calibration point $C(x,y)$. This approach produced good results in the SAR imagery we examined.

Once the BRIGHT intensity range ($[I], [I]$) is determined, it may be used to threshold the original intensity image. A connected components algorithm [1,12] may then be applied to this BRIGHT binary image to produce BRIGHT image regions.

The form of the perceptual brightness algorithm described here assumes that a global intensity range can adequately describe bright image areas. This assumes constant illumination, which may not always occur. Additionally, this approach allows large, high contrast objects to dominate smaller, lower contrast objects in the image. Because of these and other reasons, the gradient histogram and subsequent image sampling can be produced using a smaller image window (as in the human visual system [10]).



Figure 2-17: Perceptual Brightness Calibration

2.3.1.3 ROUGHLY TEXTURED Image Feature Property

Area terrain features often possess a roughly textured pattern. Texture is an important and complicated area of research within computer vision [1, 16], so we restrict ourselves to determining image areas which contain a simple mottled texture. The application of this algorithm to a SAR image in order to discriminate forest terrain is shown and described in Section 3.2.

Figure 2-18 shows a bandpass frequency-based approach for discriminating simple mottled (rough) texture in imagery. Many techniques may be used to create bandpass images (e.g., fourier analysis, optimal filters, etc.) [1, 16, 21]. The approach shown in Section 3.2 used the difference of two low-pass images which were created by convolving the original image with two-dimensional gaussian masks of differing width σ . An equivalent and more efficient approach results from composing the two gaussian masks into a single convolution mask and convolving this with the original image [1, 21]. Additional efficiencies can be obtained by noting that these masks are separable, and hence, the bandpass can be produced by sequentially convolving two one-dimensional convolution masks over the original image [4].

Mottled texture in the image area produces non-zero values in the bandpass image. Positive values in the bandpass image occur where bright texture within the bandpass occur. Similarly, negative values in the bandpass occur where dim texture within the bandpass occur. As shown in Figure 2-18, the bandpass is thresholded into two binary images which specify darker and brighter portions of the bandpass image.

A distinguishing characteristic of mottled (rough) texture is that component texture elements are fairly bloblike. It is thus desirable to eliminate adjacent pixel clusters (i.e., blobs) which are straight. Additionally, very small blobs may be eliminated. (Note that this stage of the algorithm shifts from image feature properties to description of region properties in order to perform intermediate filtering of pixel candidates; the final output, however, is an image which describes image feature properties.) This step of the algorithm aggregates pixels into regions, filtering regions which are straight or very small, and then projects the remaining regions back into a binary image. The projection back into a binary image facilitates the next stage of the algorithm.

Mottled texture contains alternating patches of bright and dark spots. Several approaches may be used to aggregate these blobs into a single entity (e.g., density metrics, search-based approaches, region growing). The technique which is shown in Section 3.2 expands individual texture blobs by two pixels which causes them to merge into a larger mottled texture blob. This region growing can be performed by computing a chamfer [1] and thresholding at the expansion distance. As shown in Figure 2-18, the binary thresholded/filtered bandpass images are merged to produce a binary image containing individual bright and dim texture elements. Adjacent texture elements are merged by growing this binary image by two pixels. The resulting image is a binary image which identifies roughly textured areas in the original image.

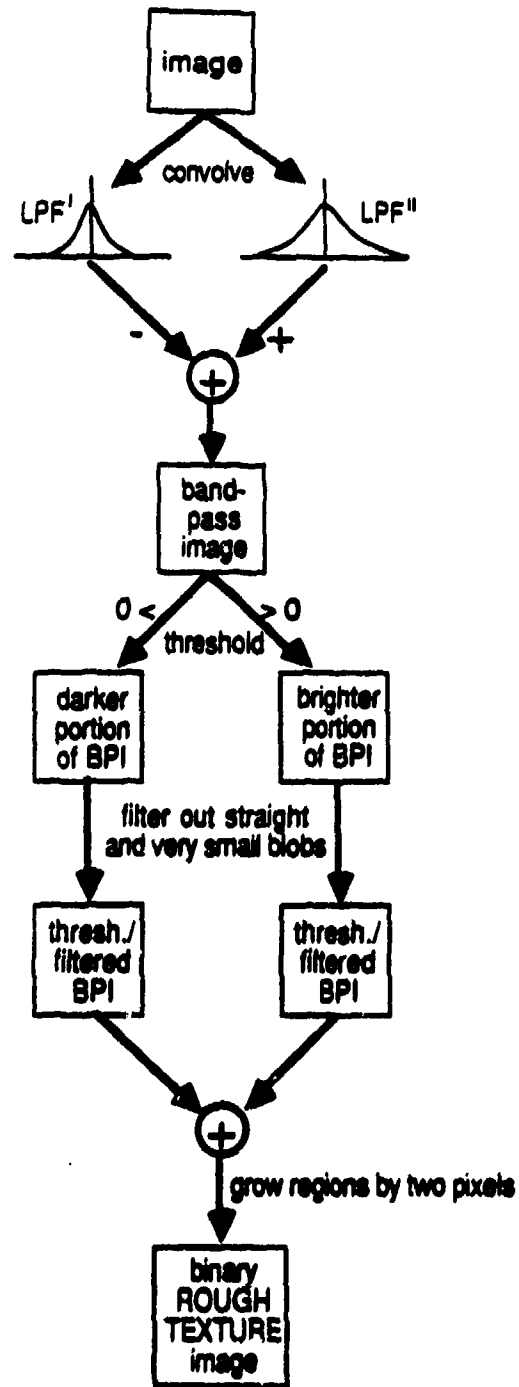


Figure 2-18: Algorithm for Extracting ROUGHLY TEXTURED Image Areas

2.3.1.4 Region Properties

Many techniques have been developed for the description of region properties. Many of these descriptive attributes are directly provided by the Powervision environment (e.g., perimeter points and length, area points and statistics, bordering regions). Other region descriptions can be trivially computed using these existing attributes (e.g., cohesiveness [1, 19], number and area of holes (i.e., other regions within the outer perimeter) [19], perimeter statistics). Here we describe some algorithms which were developed for this Phase II effort in order to compute region properties; a description of other region attributes and their computation may be found in [1, 12, 14, 16, 19].

2.3.1.5 LINE, ENDPOINT, WIDTH, LENGTH, and STRAIGHTNESS Region Properties

Orientation, straightness, width are among the most useful region properties. As in Section 2.2.2, these descriptions provide a local coordinate system which is useful for relating regions to terrain features. For example, bridge terrain features usually appear LONG and STRAIGHT. This section describes an efficient way to compute region properties described in an earlier paper [12].

The computation described below to fit a line to a region also provides other useful region descriptions which are based upon statistical properties of region pixel distributions. A technique based upon the principal axis is used. Specifically, we describe a technique for determining the best-fit LINE and ENDPOINTS, LENGTH, WIDTH, and STRAIGHTNESS of a region. The partial statistics derived from region pixels can be computed in a single pass over the image; region descriptions can then be derived with final computations on these partial statistics. A fuller discussion of the principal axis computation may be found in [1, 21].

Referring to Figure 2-19, the following steps are used to determine the stated region descriptions:

1. Accumulate region statistics
2. Compute region CENTROID (c_x, c_y) and the scatter matrix
3. Compute eigenvalues V_s and V_l
4. Compute principal axis ORIENTATION θ
5. Fit a line to the region with ENDPOINTS (p_1, p_2)
6. Compute the LENGTH l of the region
7. Compute the WIDTH w of the region

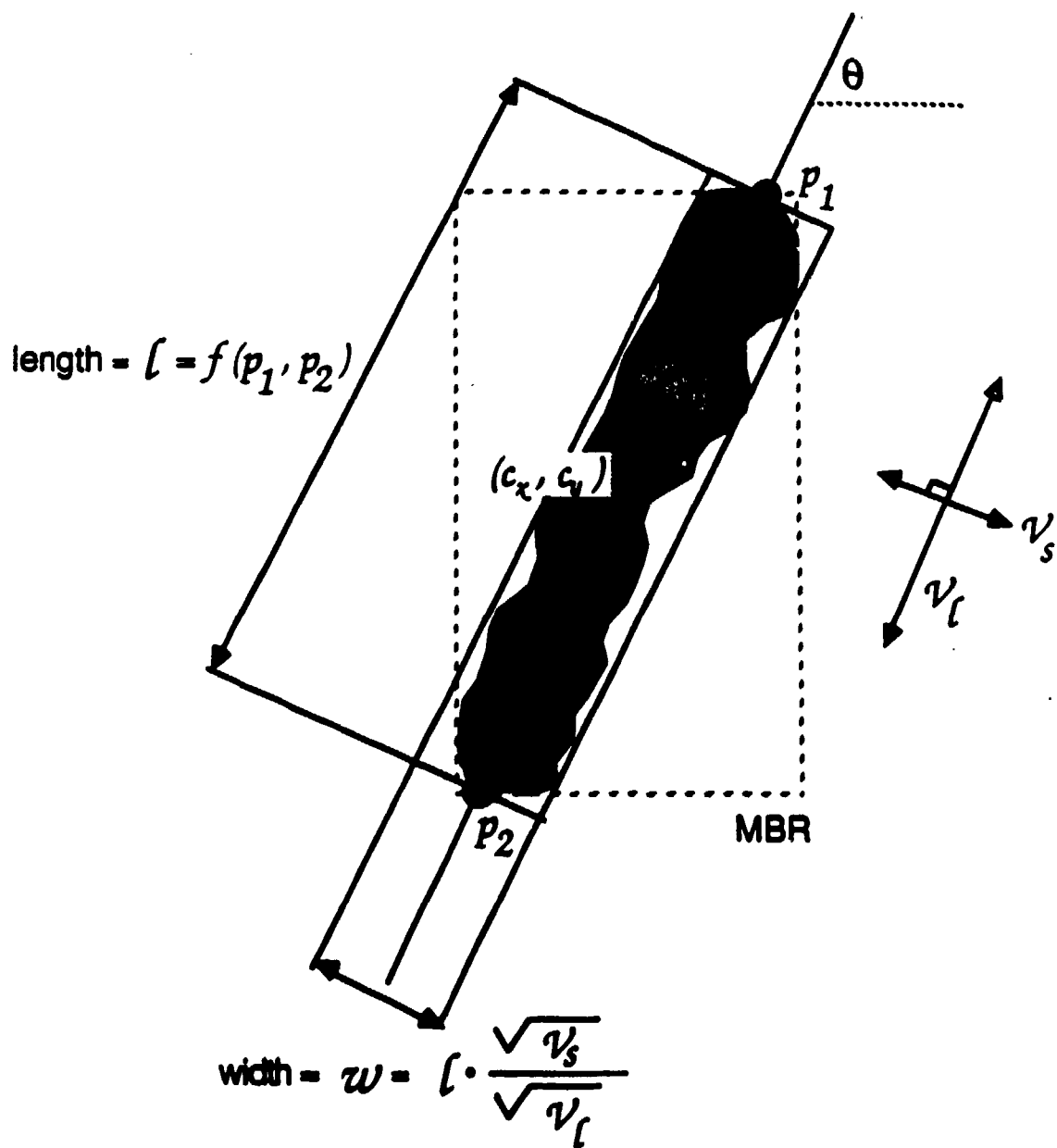


Figure 2-19: Computing Region Properties: CENTROID, ENDPOINTS, LENGTH, WIDTH, and STRAIGHTNESS

Region statistics are accumulated in order to compute the scatter matrix and determine the centroid for a region. The following computes the centroid:

$$c_x = \frac{\sum_{X \in R} x}{n} \quad (2.1)$$

$$c_y = \frac{\sum_{Y \in R} y}{n} \quad (2.2)$$

where R is the region being described, (c_x, c_y) is the centroid and n is the number of pixels(area) in the region; for the scatter matrix $\begin{bmatrix} a & b \\ b & c \end{bmatrix}$ the elements may be defined as

$$a = \sum x^2 - \frac{(\sum x)^2}{n} \quad (2.3)$$

$$b = \sum xy - \frac{\sum x \sum y}{n} \quad (2.4)$$

$$c = \sum y^2 - \frac{(\sum y)^2}{n} \quad (2.5)$$

As described in [12], this formulation allows the statistics to be efficiently computed in a single pass over the image. Thus, the following sums must be accumulated: $\sum x$, $\sum y$, $\sum x^2$, $\sum y^2$, and $\sum xy$. As will be discussed, the upright minimum bounding rectangle (MBR) is used to fit a line to the region, where

$$MBR = [([x], [y]), ([x], [y])]$$

so these minimum and maximum values also need to be maintained. Once these sums are accumulated, the centroid and scatter matrix may be computed.

Numeric capacity is important when accumulating the statistics, especially sums of the square of image coordinates, and for large regions, these numbers can sometimes cause numeric overflow. Though not a serious concern, care should be taken to ensure that partial sums never exceed the representation. This potential problem can usually be eliminated by moving the origin to the center of the image or region, scaling down all pixel coordinates, restricting the number of pixels sampled from the region, or simply using large numeric representations.

The principal axis which describes the aggregate of pixels within a region can be determined from the eigenvalues which can be derived from the scatter matrix. The characteristic equation of the scatter matrix is

$$\begin{bmatrix} a & b \\ b & c \end{bmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \lambda \begin{pmatrix} x \\ y \end{pmatrix} \equiv \begin{vmatrix} a-\lambda & b \\ b & c-\lambda \end{vmatrix} = 0$$

which can be solved using the quadratic formula

$$\frac{a + c \pm \sqrt{(-a-c)^2 - 4(ac-b^2)}}{2}$$

A less expensive form to compute is

$$\frac{a+c}{2} \pm \sqrt{\frac{(a-c)^2}{4} + b^2}$$

The solutions to the characteristic equation yield two eigenvalues. The smaller eigenvalue V_s is obtained by subtracting the second term, and the large eigenvalue V_l is obtained by adding the second term. The ORIENTATION of the principal axis is given by

$$\theta = \arctan \left(\frac{V_s - a}{b} \right)$$

where a and b are scatter matrix values, and V_s is the smaller eigenvalue.

The principal axis θ determines the orientation of the line which best fits a region (in the least-squares sense); the region centroid (c_x, c_y) anchors the position of that line. The endpoints of the line segment must still be determined.

Conceptually, it is difficult to define where the endpoints of a line fitted to a region *should* be placed. A reasonable definition would place the endpoints where the two region pixels farthest from the centroid project perpendicularly onto the principal axis. Though intuitive, this solution has problems with irregularly shaped regions, it is relatively expensive to compute, and it requires an extra pass through the image (since the projection cannot be done until after the principal axis been determined).

As illustrated by the ENDPOINTS (p_1, p_2) in Figure 2-19, a simpler way to compute line endpoints intersects the principal axis line with the upright minimum bounding rectangle (MBR). As discussed, the MBR can be easily determined when the statistics are accumulated. This method is efficient because defining the bounding box and intersecting it with the principal axis line is computationally simple. This technique causes some anomalies (e.g., rotating a non-circular region in the plane changes the endpoints somewhat), but it works extremely well for elongated support regions. Because it is not meaningful to fit a line to an irregular region (as when the ratio of the eigenvalues is large), unusual fitting of lines to highly irregular regions is not a serious concern.

The best-fit line fit to the region simply provides the length of the region. That is, region LENGTH may be defined as $l = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$ where x_1, y_1 and x_2, y_2 are the coordinates of the endpoints p_1 and p_2 .

Determining region width can be a bit more problematic. One possible approach determines the farthest region pixels from the principle axis. This approach is computationally expensive since it must determine the distance of region pixels from the principal axis. Further, even a small aberration in the perimeter of a region can cause a large change in assigned region width.

By noting that the eigenvalues measure the distribution of region pixels about the principal and minor axes, they can be used to provide an inexpensive way to determine region width. That is, $w = l \left[\frac{\sqrt{V_s}}{\sqrt{V_l}} \right]$ where w is region

WIDTH, V_2 and V_3 are the eigenvalues, and l is the region LENGTH as determined by the ENDPOINTS. In addition to the computational simplicity of this approach, it is insensitive to minor variations or irregularities in the region perimeter.

When region LENGTH and WIDTH are available, region STRAIGHTNESS may be defined by their ratio. That is, $s = \frac{w}{l}$ where s is the measure of STRAIGHTNESS, w is region WIDTH, and l is region LENGTH. Note that $0 \leq s \leq 1$. As $s \rightarrow 0$, a region becomes more elongated. Conversely, as $s \rightarrow 1$ a region becomes more spherical.

2.3.1.6 ILLUMINATED PERIMETER Region Property

When referring to the perimeter of a region, it is often quite useful to refer to the direction of illumination. For example, for a forest terrain feature in a SAR image, the perimeter which faces the illumination usually has a white band; similarly, the shadow cast by the forest region usually results in a wide black band on the nonilluminated trailing perimeter. Section 3.2 demonstrates how these white and dark bands, in conjunction with a concept of (non)illuminated perimeter, can be used to more reliably determine the location of forest terrain features. This section describes how the illuminated and nonilluminated portions of a region perimeter may be determined.

It is assumed that the perimeter for a region is maintained as a circular list of perimeter points; this is the representation afforded by Powervision (see Appendix B). The task is then to determine, for each point in the perimeter, whether that perimeter point faces towards or away from the illumination. The algorithms discussed here first find a region perimeter point which is known to face the illumination source. The algorithms then walk along the perimeter from the starting point marking the visibility of each perimeter point. As the algorithm walks along the perimeter, it examines the relationship between the orientation of the local perimeter relative to the source of illumination in order to determine whether the current perimeter is illuminated. The algorithms terminate when the algorithm returns to the initial perimeter position.

Many SAR images are synthesized according to the convention that the (radar) illumination originates vertically from the top to the bottom of the image (as shown in Figure 2-20). In this case, there is a very simple algorithm for determining which portions of the region perimeter face the illumination source.

As shown in Figure 2-20, the starting perimeter point is selected as the top-most point of the region (i.e., $P([x], y)$). Since the illumination is known to originate from above, this ensures that $P([x], y)$ faces the illumination source; a toggle variable V is thus initially set to *illuminated*. The algorithm walks along the perimeter from this starting point $P([x], y)$. As shown in Figure 2-20, let us assume that the perimeter walk initially moves from left to right (i.e., increasing in column values). Perimeter positions are marked as "illuminated" until the direction of the walk changes (e.g., the direction of the walk changes from left-right to right-left). Perimeter points along the walk are subsequently marked as "not illuminated" until the direction of the walk changes again. This toggling continues as the walk continues and the direction of the walk changes. An example result from this algorithm is shown in Figure 2-20.

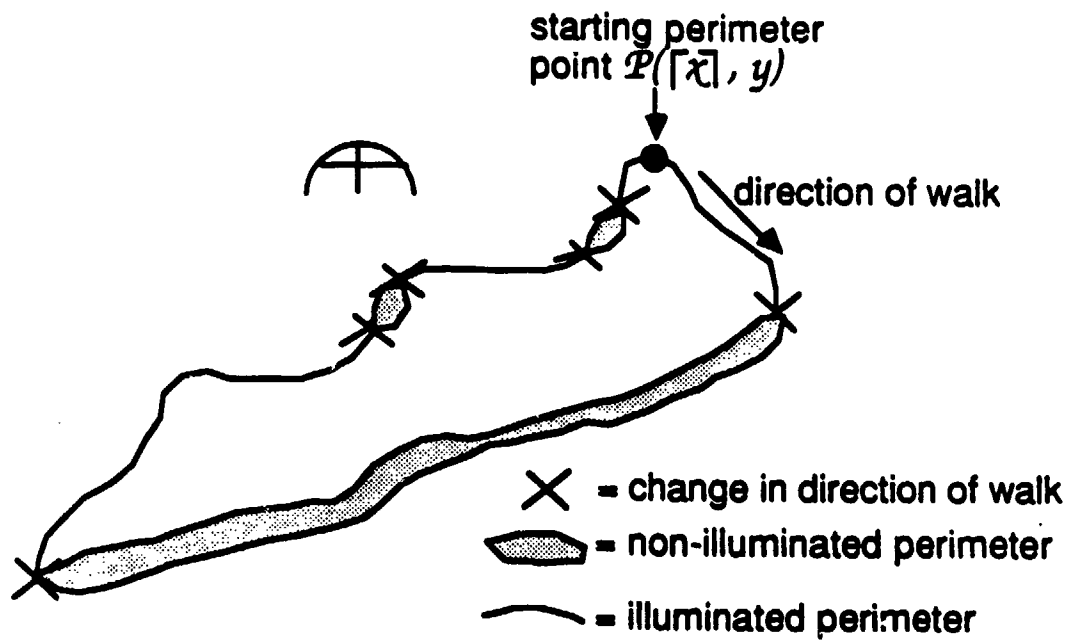


Figure 2-20: Determining Perimeter which Faces Illumination

When the source of illumination can originate from an arbitrary direction (i.e., not always from directly above), the above algorithm must be modified. In this less restricted case, changes in the direction of the walk (as shown in Figure 2-20) can be determined when the tangent of the perimeter crosses the direction of illumination.

2.3.2 Terrain Object Modeler

Only a subset of image feature and region properties are applicable to any given terrain feature. For example, LONG, STRAIGHT, BRIGHT regions with minimum WIDTH are applicable to bridges appearing in SAR imagery (see Section 3.3). The Modeler can specify image feature and region properties for each modelled terrain feature by specifying the relevant parameter ranges. Operationally, the System Controller can use this information to activate routines within the Perceptual Grouper, or alternatively, the parameters may be used to filter image structures contained within the ISDB; the two approaches differ primarily in overall computational effort and control complexity. A filtering approach was used in the experiments described in Section 3. The filtering of image structures based on image and region criteria is especially well-supported in the Powervision environment. This approach is especially useful since extraction routines can produce false positives which are subsequently filtered; conversely, false negatives need to be avoided in a filter-based paradigm.

Local Region Properties require search areas to be defined. Search areas can be specified using the techniques presented in Section 2.2.2 and standard computer graphics techniques (e.g., [8]). The resulting search areas can then be represented as a graphical object or by using bit planes.

Inter-Object Properties can be represented using a semantic network representation where terrain feature objects are vertices and inter-object properties are defined by relational edges [1, 2, 14]. These semantic networks typically represent spatial relations among objects quite coarsely (e.g., bridges CONNECT roads). Finer resolution in spatial relationships can be obtained by examining terrain feature descriptions at lower levels in the terrain feature structural hierarchy (i.e., local-region, region, or image feature properties). These semantic networks can support *constraint-based reasoning* [2].

3. DEMONSTRATION OF SAR TERRAIN FEATURE EXTRACTION

3.1 INTRODUCTION

A recognition procedure can be thought of as mapping parts of the imagery to terrain features. It is the task of a recognition procedure to organize the image into useful image features (e.g., the structural hierarchy shown in Figure 2-3). The recognition procedure has both top-down and bottom-up processing. These processes are broken down into several levels of abstraction; previous sections have described the architecture and its operation.

This section demonstrates the extraction of forest and bridge terrain features from SAR imagery using the technical approach and techniques described in Section 2. The forest example uses a texture-based technique to extract initial forest hypotheses; local region properties using perimeter-based search areas are then used to identify and eliminate false positives. The bridge example uses brightness to extract initial bridge hypotheses; local region properties using point-based search areas are then used to eliminate false positives. The road example uses local extrema properties to extract thin segments whose colinear properties are used to build longer and larger road elements. These features were chosen because they demonstrate the use of a variety of local region properties.

A key aspect of both examples is that bottom-up processing is used to generate terrain hypotheses and top-down model information is used to eliminate contextually inconsistent hypotheses. This approach achieves the sensitivity of bottom-up processing and the robustness of top-down modelling.

3.2 FOREST FINDER

Forested areas in SAR imagery possess distinctive "texture" with a trailing radar shadow and a leading band of high return. As shown in Figure 3-1, the appearance of forest terrain can be defined using a formative model of the canopy. The canopy is composed of leaves on individual trees that when viewed *en masse* produces a dappled pattern of high returns (individual trees which give strong returns) and low returns (where trees cast a radar shadow). This characteristic rough texture distinguishes it from other terrain features (see Figure 3-2). For example, the field areas contain small elements (i.e., wheat) which result in higher spatial frequencies.

The ROUGHLY TEXTURED algorithm described in Section 2.3.1.3 and region attributes were used to produce initial forest terrain hypotheses. The algorithm for this band-pass (frequency-based) texture metric is shown in Figure 2-18.

Figure 3-2 shows the original SAR image of forested terrain. As shown in Figure 2-18, a bandpass image is produced using the difference of two low-pass filtered (LPF) images which differ in their cutoff. Gaussian filters attenuate higher frequencies, hence, they can be used as low-pass filters [21]. Figure 3-3a shows the original image convolved with a symmetric gaussian mask with $\sigma \approx 3$; Figure 3-3a

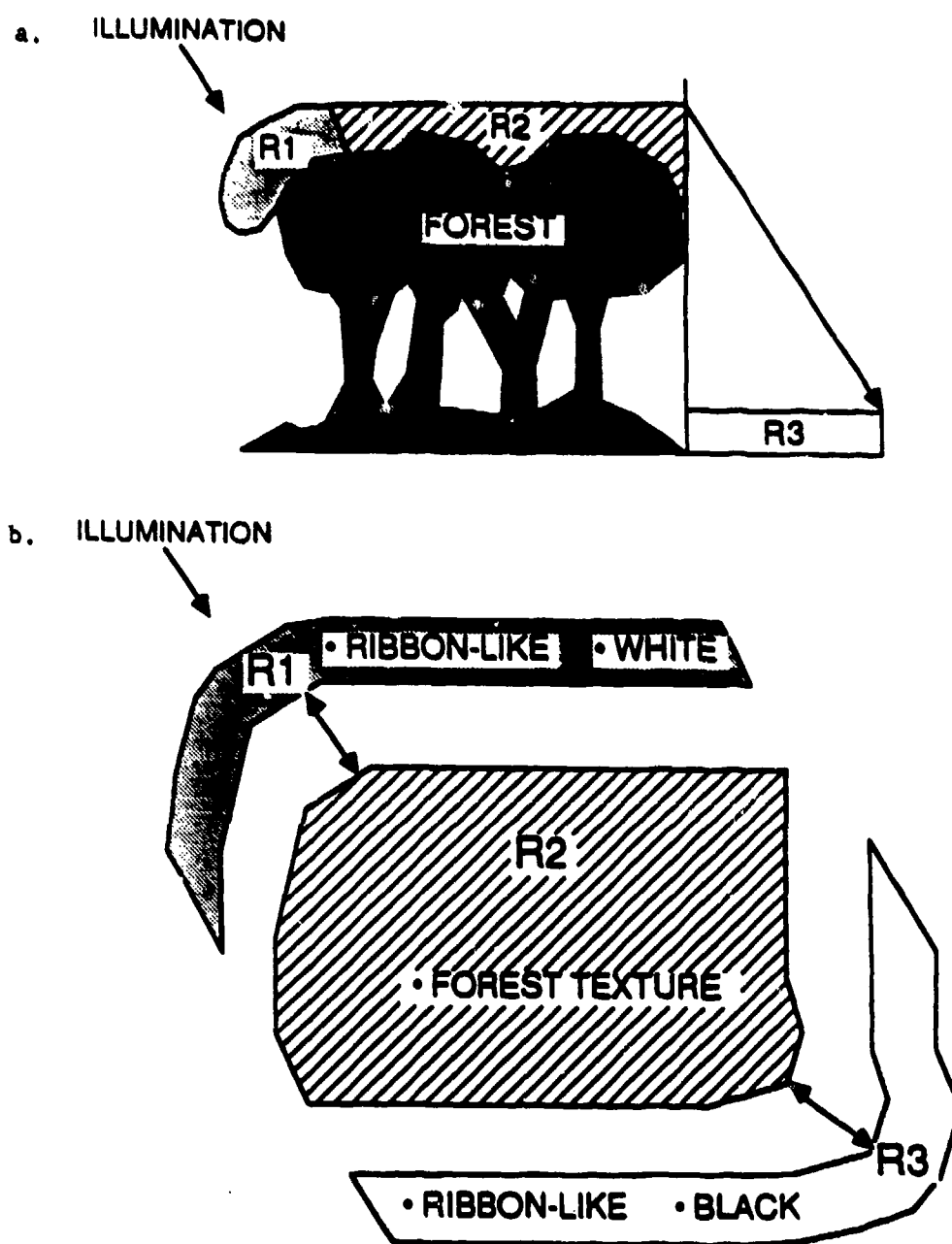


Figure 3-1: Forest Terrain Model. (a) Formative Model, (b) Perimeter-based Local Region Properties for Forest Terrain Model



Figure 3-2: Original SAR Image of Forested Terrain



(a)



(b)

Figure 3-3: Original Image Convolved with a Gaussian with
(a) $\sigma \approx 3$, and (b) $\sigma \approx 4$

has a lower frequency cutoff since it uses a gaussian mask with $\sigma \approx 4$. The relationship between the spread of the gaussian σ and frequency is discussed by Burt [4]. The bandpass image B (shown in Figure 3-4) was obtained by subtracting Figure 3-3a from 3-3b.

Notice in the bandpass image shown in Figure 3-4 that forested image regions contain a dappled texture composed of large bright and dark blobs. As shown in Figure 2-18 (described in Section 2.3.1.3), these blobs are the texture elements ("textels") used to determine ROUGHLY TEXTURED image areas. Thresholding the bandpass such that $B(x,y) > 0$ produces bright blobs (shown in red in Figure 3-5a); thresholding such that $B(x,y) < 0$ produces dark blobs (shown in green in Figure 3-5b). Forest has a rough dappled texture, so component textels should be blob-like; the textels should not be very small or highly linear. The textels shown in Figure 3-5 (red and green) thus contain three or more pixels and they are not very straight (i.e., they have more than a 1:5 width/length ratio). The rough texture elements (textels) shown in Figure 3-5 have been merged to form the binary textel image shown in Figure 3-6.

Forested areas are distinguished in Figure 3-6 by a high density of textels. The final step in the algorithm must aggregate dense textel areas into ROUGHLY TEXTURED regions. Because rough texture contains blobs alternating from bright to dark, the textels in a dense region are very close to one another. Dense textel regions may thus be obtained by growing (i.e., enlarging) textel regions by one or two pixels; this effectively merges textels together to form ROUGHLY TEXTURED regions.

Clearly, forest regions have a minimum size. Figure 3-7 shows the ROUGHLY TEXTURED regions produced by the above process. Small regions have been eliminated (filtered). Note that in Figure 3-7 the minimum region size is still quite small in order to avoid excluding actual forest regions.

As shown in Figure 3-7, the ROUGHLY TEXTURED regions contain holes (i.e., contained regions which are not considered part of the textured regions themselves). Figure 3-8 highlights these holes in red. The holes are caused by breaks in the texture. For forested regions, they identify small breaks and clearings within the forested areas. These interior regions can be considered terrain features in their own right (with potential for area limitation use for detecting targets or other features). Since we are seeking forest terrain, we chose to consider small clearings part of the forest regions (see Figure 3-9).

Figure 3-9 shows the ROUGHLY TEXTURED regions which are considered initial forest terrain hypotheses. As discussed in Section 2, top-down models of forest terrain may now be used to evaluate these hypotheses.

Figure 3-10 shows the perimeter-based local region properties (see Section 2.2.2.5) which can be used to evaluate the initial forest terrain hypotheses shown in Figure 3-9. The search area labelled R1 is the bright thin leading edge of the forest. The search area labelled R2 is the dark trailing radar shadow that is cast by the forest. The direction of illumination and maximum tree height places a boundary on the width of these support areas.

The local region support areas are shown in Figure 3-10. The leading edge support areas are outlined in red; the trailing shadow support areas are outlined

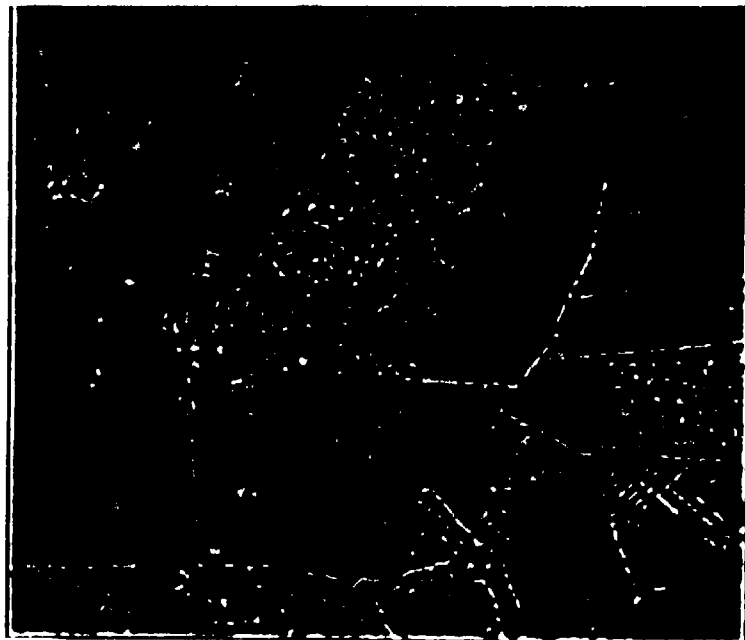


Figure 3-4: Bandpass Image *B* Obtained by Subtracting Figure 3-3a from 3-3b



(a)



(b)

Figure 3-5: Texture Elements Produced by Thresholding Areas in the Bandpass Image in Figure 3-4

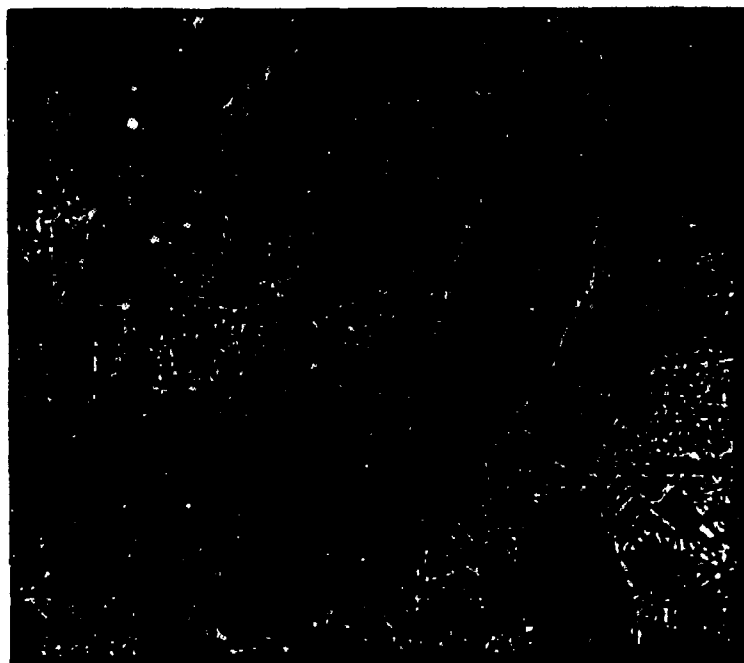


Figure 3-6: Rough Texture Elements (Textels) Produced by Merging the Binary Images Shown in Figure 3-5

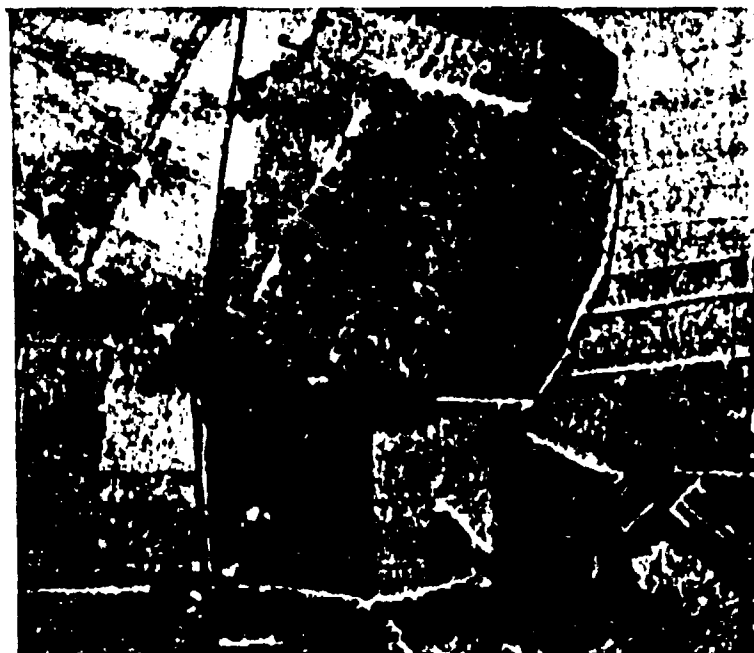


Figure 3-7: ROUGHLY TEXTURED Regions (Small Texture Regions have been Filtered)

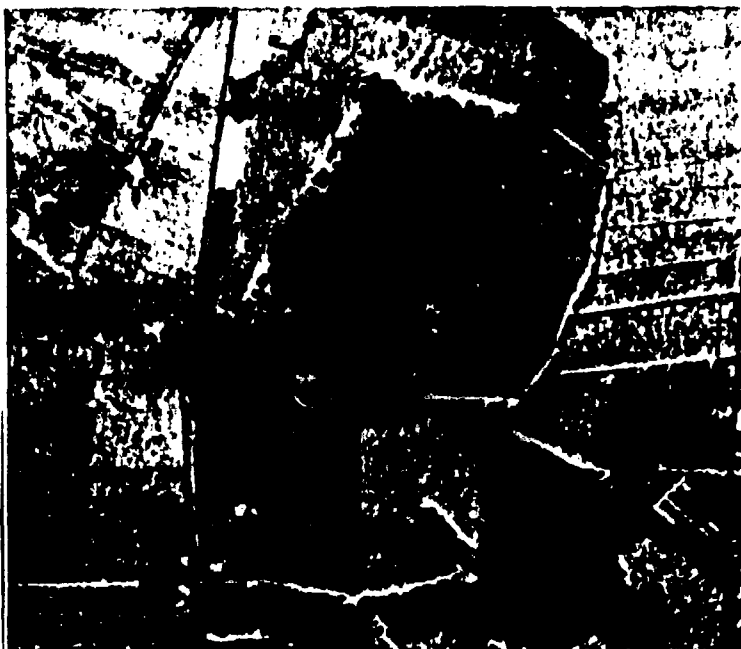


Figure 3-8: Holes in ROUGHLY TEXTURED Regions (in red)



Figure 3-9: Initial Forest Terrain Hypotheses



Figure 3-10: Local Region Supports for Central Forest Region
Hypotheses (in green)

in white. The ILLUMINATED PERIMETER algorithm described in Section 2.3.1.6 was used to determine leading and trailing region perimeters.

Incorrect initial forest hypotheses will lack the bright leading bands and trailing shadows in the local region support areas. This lack of context is used to eliminate false positives. Conversely, correct forest hypotheses will receive local region support. Figure 3-11 zooms in on the search areas which support a central forest region.

Figure 3-12 shows the final forest hypotheses which were found to possess local region support. A comparison of Figures 3-9 and 3-12 shows which hypotheses failed to demonstrate contextual consistency (as determined by search areas).

It is worthwhile to note that the search areas can be used to improve forest region discrimination. For example, a comparison of the large central forest region (Figure 3-12) and the original SAR image (Figure 3-2) shows that the roughly textured region overestimated the forested area; this occurred at the bottom left of the region. By noting where the shadow ends at the left bottom of the region, the small oversegmented part of the region can be discarded. Similar techniques can be used to correct minor region boundary misplacements (e.g., at the top of the large region).

3.3 BRIDGE FINDER

As shown in Figure 3-13, the appearance of bridge terrain features are predicted using a formative model of bridges. Bridges usually have large radar returns (from metal construction and dihedrals), thus they typically appear as BRIGHT, STRAIGHT regions.

The *perceptual brightness* algorithm (see Section 2.3.1.2) and region attributes were used to generate initial bridge terrain hypotheses. A familiarity with the algorithm described in Section 2.3.1.2 is assumed.

Figure 3-14 shows the original SAR image which contains a bridge. Although bridges are in the higher intensity range of the image, a simple unguided thresholding will not isolate the bridge areas. Consider, for example, the many different images which would have the same intensity distribution as Figure 3-14. It is clear that a more sophisticated method of determining "bright" is required to create bridge hypotheses. In the discussion of the *perceptual brightness* algorithm in Section 2.3.1.2, intensity discontinuities were found to be a good way to discriminate bright image areas. The *perceptual brightness* algorithm is illustrated in Figure 3-15.

As shown in Figure 3-15, the first step in the *perceptual brightness* algorithm computes gradient magnitude. Figure 3-16 shows gradient magnitudes in which rapid intensity discontinuities appear bright and more homogeneous image areas appear dim. Figure 3-17 shows the characteristic histogram of gradient magnitude (scaling the histogram for display has compressed the detail of the higher gradient magnitude values). As described in Section 2.3.1.2, the upper and lower boundaries of the BRIGHT intensity range are determined from a small, fixed number of calibration points. For the results described here, .1% of the image



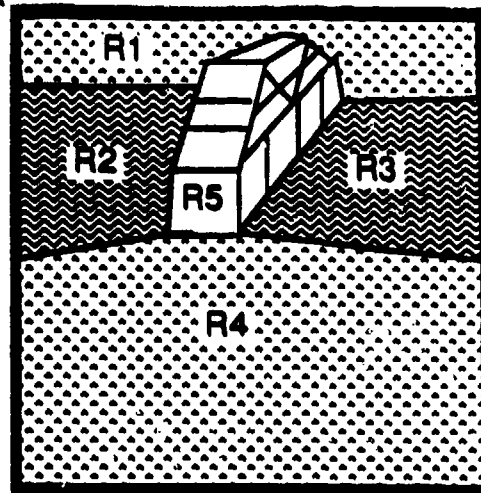
Figure 3-11: Magnified View of Central Forest Region Support at the Center Bottom of Figure 3-10



Figure 3-12: Final Forest Hypotheses which Possess Local Region Support

a.

ILLUMINATION



b.

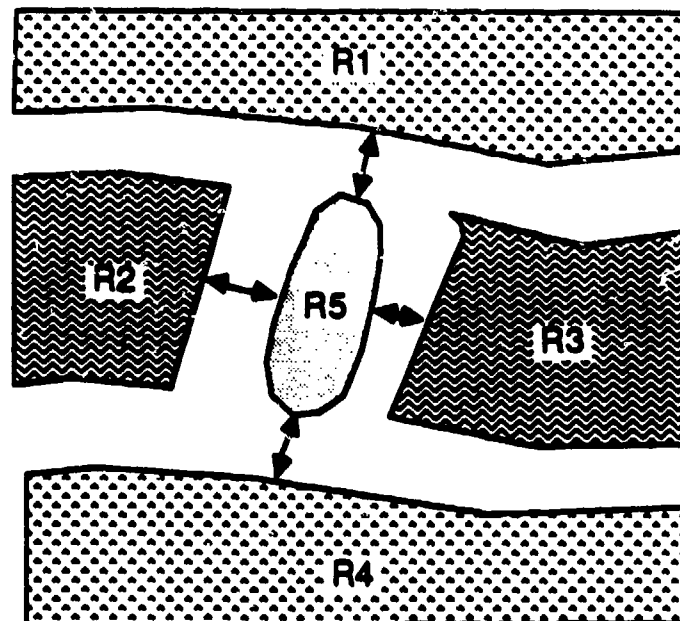


Figure 3-13: Bridge Terrain Model. (a) Formative Model, (b) Point-based Local Region Properties for Bridge Terrain Model



Figure 3-14: Original SAR Image Which Contains a Bridge

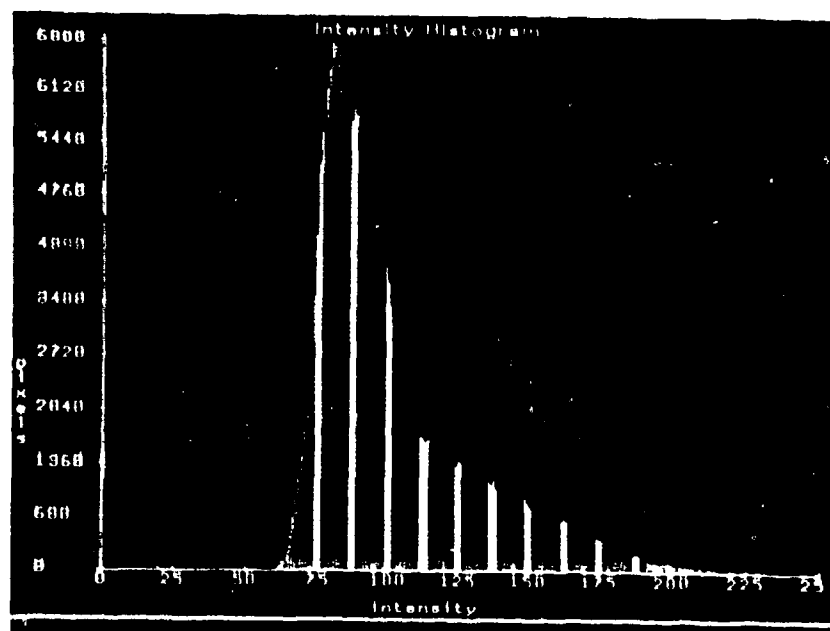


Figure 3-15: Intensity Histogram for Image Shown in Figure 3-14



Figure 3-16: Gradient Magnitudes for Image Shown in Figure 3-14

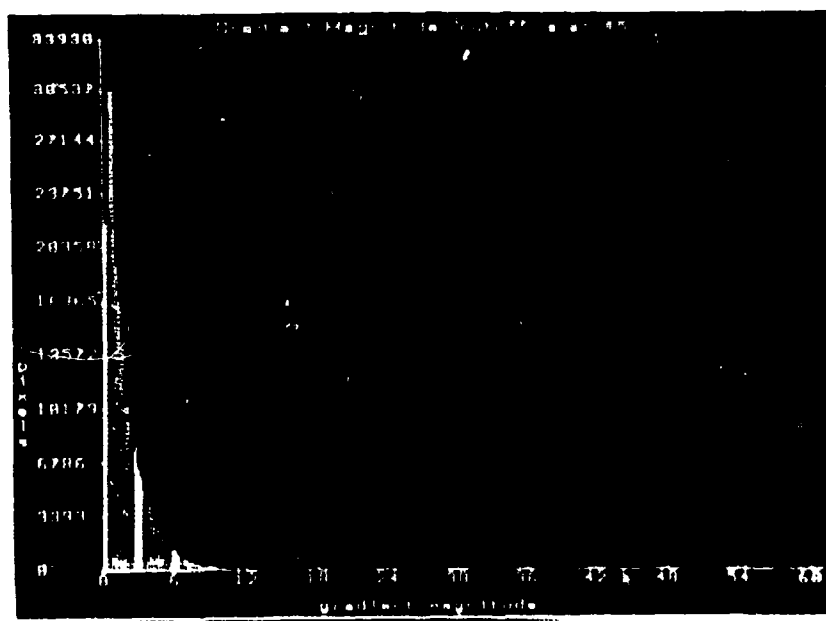


Figure 3-17: Gradient Histogram for Gradient Image Shown in Figure 3-16

points were used for calibration. A given pixel was chosen as a calibration point if it was among the largest .1% of gradient magnitudes. In the gradient histogram shown in Figure 3-17, the lowest gradient magnitude among the largest .1% magnitudes was found to be 45; that is, 99.9% of the gradient values are below 45. As illustrated in Figure 2-16, this threshold value of 45 is used to produce the calibration points (in red) shown in Figure 3-18; a pixel is red when its gradient exceeds 45. As described in Section 2.3.1.2, the bottom of the BRIGHT range was defined as the minimum value of all calibration points which were sampled from a maximum filtered image¹ (3X3 neighborhood). This calibration procedure determined that the BRIGHT intensity range is (169, 241).

Figure 3-19 shows BRIGHT image areas in green, blue, and red; a pixel is considered BRIGHT if its value is within the bright intensity range (169, 241). Because bridges have a minimum possible size, larger regions (i.e., ≥ 40 pixels) are shown in blue and red. Eliminating very small regions can greatly decrease computational effort, and this minimum threshold can be simply determined from platform parameters and an empirical estimate of the smallest realistic bridge size.

Because the width of a bridge is usually far less than its length, bridges characteristically appear as STRAIGHT regions of high return. Figure 3-19 shows those BRIGHT regions (in red) which have a width/length ratio less than 1:10; these regions become the initial bridge hypotheses. A larger width/length ratio (e.g., 1:4) generates more initial hypotheses and the ratio can be selected based upon the maximum realistic bridge width/length ratio. In general, it is better to use a conservative (i.e., larger) ratio to avoid false negatives (i.e., incorrectly excluding a terrain feature hypothesis).

The red regions in Figure 3-19 are the initial bridge hypotheses. As shown in Figure 2-11, top-down information in the form of local region properties can be used to evaluate these hypotheses.

Figure 3-20 shows the point-based local region support areas for two central bridge hypotheses (shown in green). Water support areas are outlined in white; road/land support areas are outlined in red. If a central bridge region hypothesis is well-founded, these search areas should contain expected properties (i.e., water and road/land).

Though other, more accurate, metrics are possible, the search area properties were based upon statistical properties. A search area which had intensities with a small standard deviation and average was considered to have water "properties." Conversely, road/land search area properties were considered present if intensities in the area had a significant standard deviation and an average in the upper half of the intensity range. Though these metrics are crude (especially for road/land), the next paragraphs describe how they were able to eliminate incorrect initial hypotheses.

Figure 3-21 shows a zoom-in of local region support areas (see Figure 3-20) which do not confirm the (incorrect) central bridge hypothesis. Because the central region in Figure 3-21 (in green) is actually part of an island, and not a bridge, the leftmost support area (in red) does not contain road/land properties. Instead, this

¹ A maximum filter sets each pixel value to the maximum value within a fixed window.



Figure 3-18: Calibration Points Selected by the Perceptual Brightness Algorithm



Figure 3-19: Perceptually BRIGHT Regions (in green, blue, red)

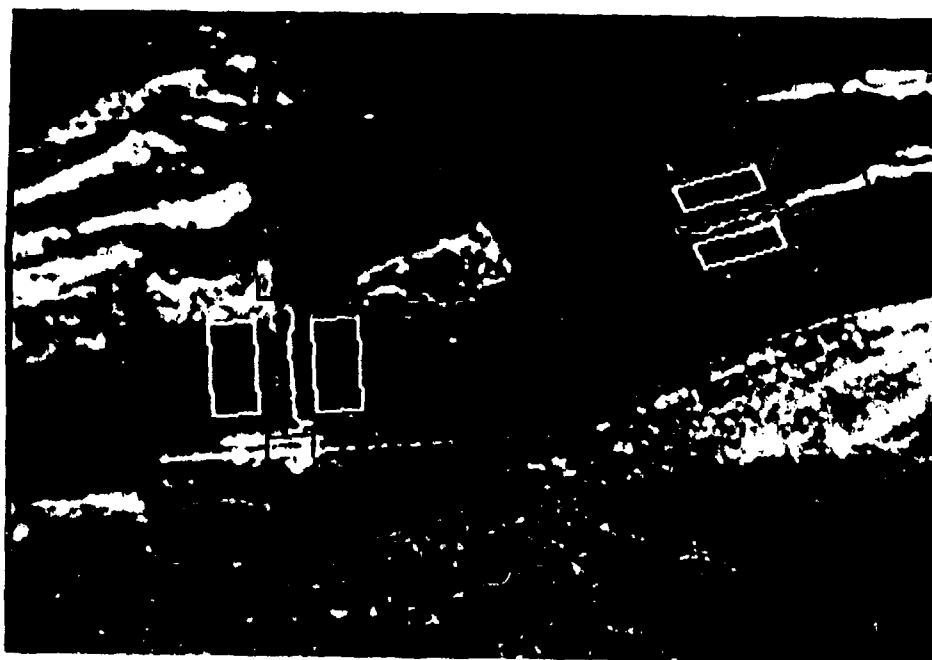


Figure 3-20: Local Region Support Areas for Central Bridge Hypotheses (in green)

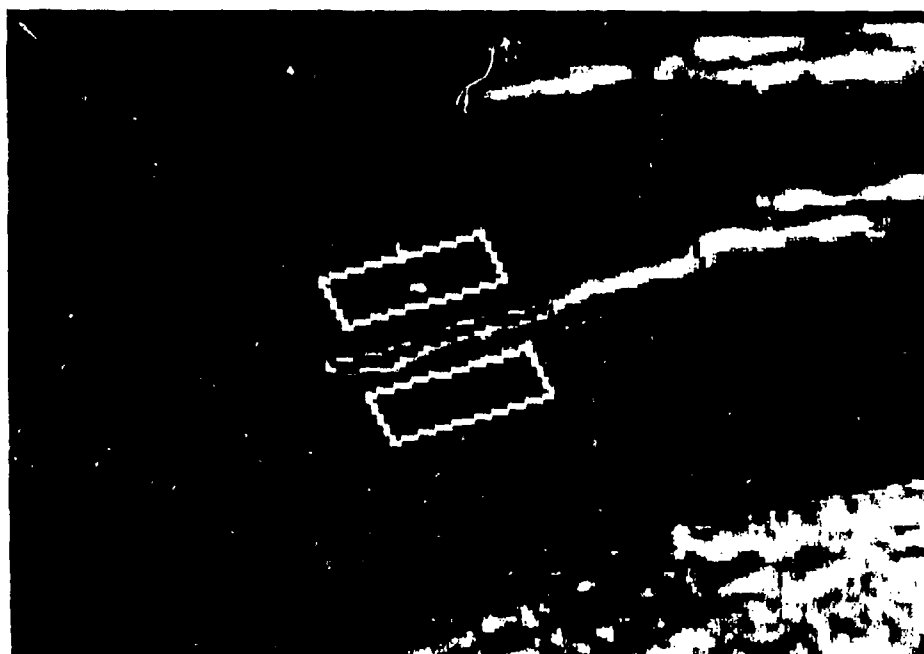


Figure 3-21: Zoom-in of Local Region Support Areas Which Do Not Confirm the (Incorrect) Central Bridge Hypothesis (in green)

left-most support area contains water. Because this is clearly inconsistent with the functional use of bridges, this initial bridge hypothesis may be discarded (or downgraded in confidence).

Figure 3-22 shows a zoom-in of local region support areas which *confirm* the (correct) central bridge hypothesis. Because the central region is in fact a bridge, the support areas display expected properties. That is, the water support areas (outlined in white) possess properties of imaged water (e.g., the areas are dim and fairly homogeneous). Similarly, the road/land support areas (outlined in red) possess properties of imaged road/land (e.g., non-homogeneous, presence of moderate returns). Because the central bridge hypothesis is confirmed by its local support regions, it is retained as a valid bridge hypothesis.

Figure 3-23 shows the final forest hypothesis which was found to possess local region support. A comparison between Figures 3-19 (in red) and 3-23 shows which hypotheses failed to demonstrate contextual consistency (as determined by search areas).

3.4 ROAD FINDER

The representation of roads differs from those of the other features considered. Forests and bridges are features described as areas and they are viewed as individual entities of finite extent. Roads are not "finite" in the sense that roads are important because of their ability to link places which are distant from one another. Roads are generally represented as a network; that is, a structure consisting of junctions (and ends) linked by uninterrupted (not necessarily straight) segments. Our concern in this section is how to recognize and extract uninterrupted road segments and junctions.

Road segments have characteristic shapes but no single internal cue. In other words, roads are recognized not because of a specific intensity property but rather because of the particular shape characteristics associated with a number of intensity properties.

Our model for road segments (Figure 3-24) recognizes them as thin entities distinguishable by being either darker or lighter than their surroundings. These entities are then joined into segments based on grouping considerations. In particular, we structure the grouping by considering long segments as "seeds" for the grouping process. That is, the grouping process considers longer entities first. There are good reasons for this. Longer entities have more stable descriptions based on a greater sample of constituent pixels. These descriptions, when matched to the road model predictions, are more likely to give a correct classification as road. Also, for these entities it is easier to measure directionality of the endpoints, thereby simplifying the search for continuations. Finally, their descriptions provide clues as to the descriptions of their candidate continuations, serving themselves as a "model" for the continuation entity.

The following description illustrates processing steps as applied to an image containing a road network (Figure 3-25). In the first step, the imagery is subjected to a band pass filter. The purpose of this filter is to permit the analysis of the intensity characteristics without assumptions concerning the exact gray levels at which road segments may be found. The band pass image (Figure 3-26) results

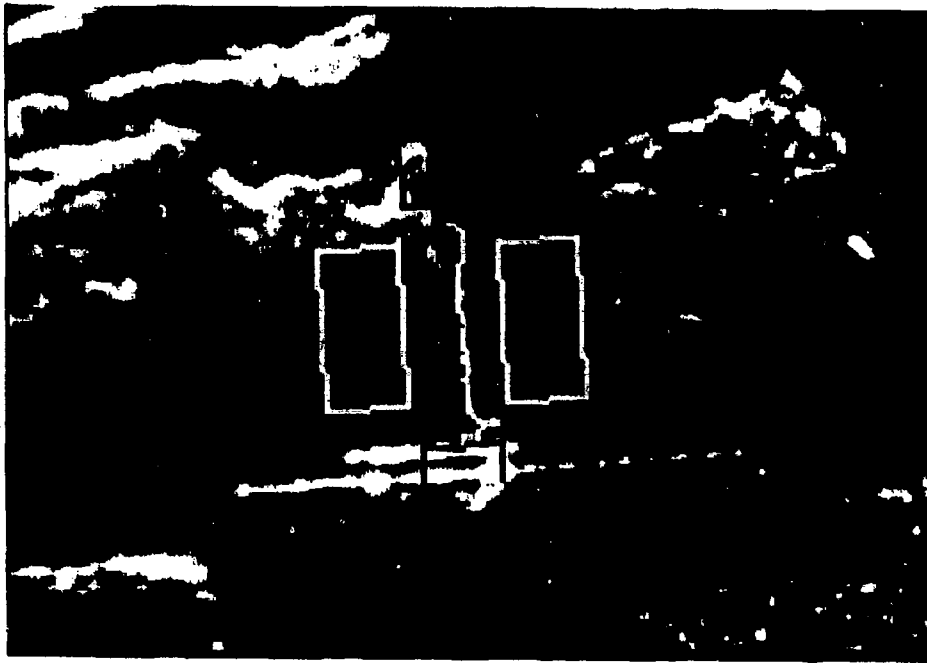


Figure 3-22: Zoom-in of Local Region Support Areas Which Confirm the (Correct) Central Bridge Hypothesis (in green)

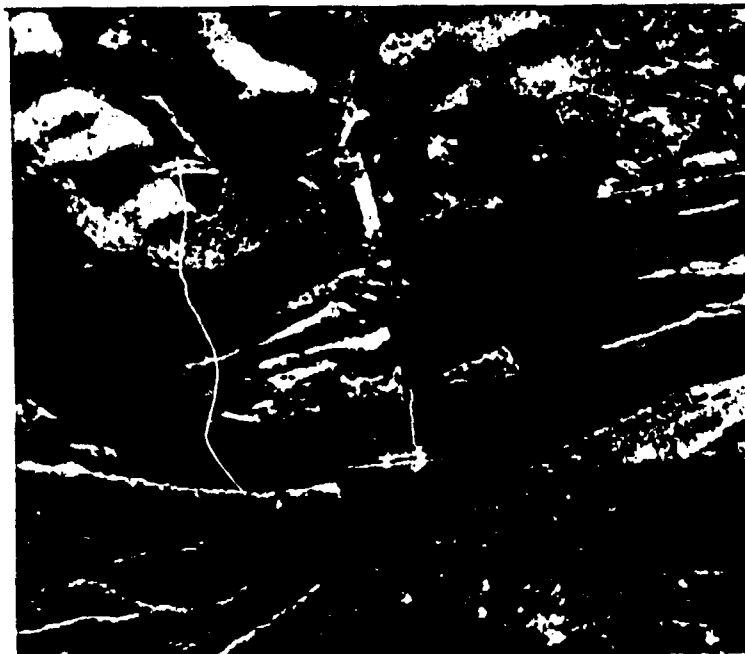


Figure 3-23: Final Bridge Hypothesis Which Possesses Local Region Support

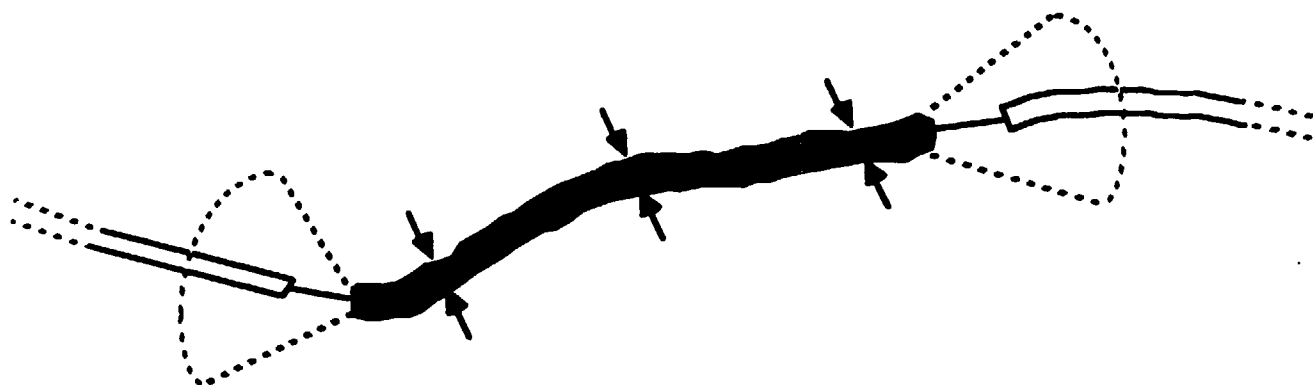


Figure 3-24: Model of Road Segment Feature

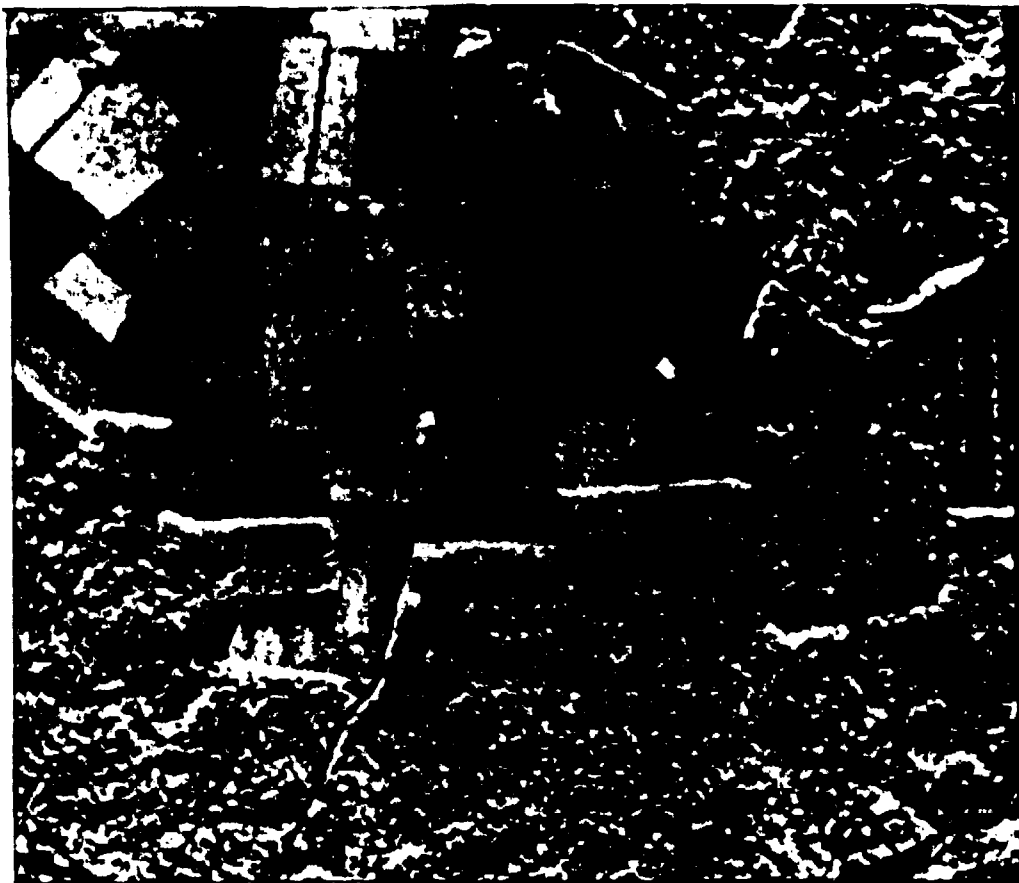


Figure 3-25: Original SAR Image Which Contains a Road Network

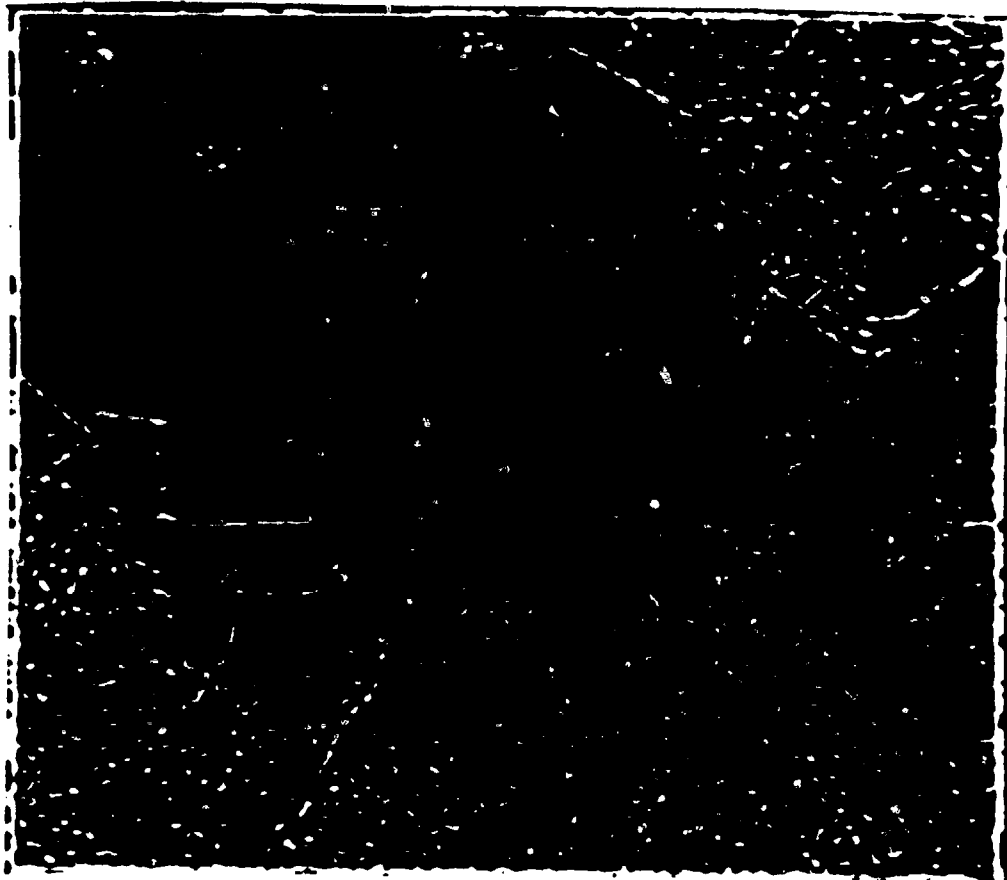


Figure 3-28: Band Pass Image Results

from the difference of two low-pass filtered versions of the original image. In effect, this computes a difference of Gaussians (also known as a Generalized Laplacian). In this image, extrema correspond to bright and dark points. The roads themselves tend to lie along these extrema. However, extrema by themselves are simply individual points which may form regions whose shape properties are inappropriate.

An intelligent thinning mechanism is required to reduce an area type property to a linear property. The medial axis transform (MAT) is an algorithm which converts regions into a network of lines by computing a "skeleton" for the region. The medial axis skeleton corresponds to the set of points which are equal in distance from the boundary of the region. Each medial axis point is the center of a maximal circle which fits wholly within the region. Thus each medial axis point is characterized by its location and radius.

The algorithm for computing the MAT follows this definition. First, the boundaries of the regions for which the MAT is to be computed are identified (as a by-product of the connected component algorithm or explicitly by tracking each contour or even by the simple method of deleting interior points - that is, points with no neighbors outside the region). Each such boundary point is labeled with a zero (in addition to its region label) indicating zero distance from the boundary. A series of operations is then applied simultaneously to all image pixels. At each iteration, each pixel adjacent to a labeled pixel is itself marked with a label one higher than that of the adjacent labeled pixel. This parallel propagation of labels is called the "distance transform" or "chamfering" since the result of the process is a labeling of all pixels by their distance from the closest boundary point. The medial axis of any region is the set of pixels with locally maximal labels in the interior of the region. The radius corresponding to the pixel is the label that has been computed for it.

Now it is clear how road segment candidates can be extracted. These are the elongated regions of extrema whose medial axes have roughly the same radius and whose radius is within the allowable range for roads. Figure 3-27 displays the medial axes of candidate road segments which will be subjected to further relational analysis. Tiny stubs ("nubbins") have been deleted by a process of shrinkage at endpoints. The resulting longer segments are shown overlaid on the original image and as individual entities in Figure 3-28.

Joining road segment candidates into longer road segments is based on an analysis of segment-to-segment relationships. This analysis is simplified if each segment is represented by a straight line. Figure 3-29 is the result of fitting straight lines to the medial axes. The line fitting (the Ramer algorithm) is adaptive; that is, multiple lines are used to represent a single segment if the distance to the furthest point of the points being fit is above a preset threshold. This may cause a curved segment to break into a number of fitted line segments. This is not an error since the processing which caused the curved segment to be extracted in the first place is not perfect and the breaking of curved segments at curvature maxima permits high-level knowledge of collinearity and adjacency to coalesce the line segments. In Figure 3-29, color is used to differentiate separate line segments.

Two heuristics operate to join line segments: grouping by collinearity and grouping by endpoint adjacency. In the first, segments are linked if they appear to be part of the same line - that is, their line parameters are approximately

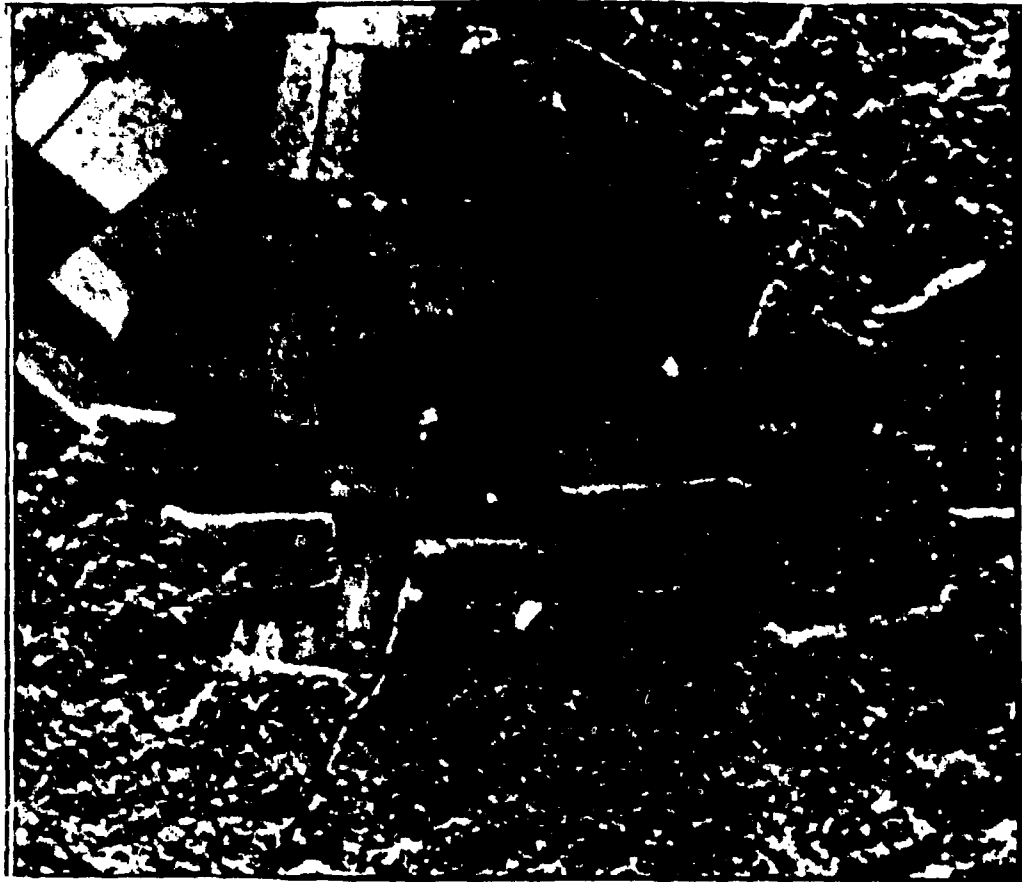


Figure 3-27: Medial Axes of Candidate Road Segments



Figure 3-28: Longer Segment Results After Shrinkage

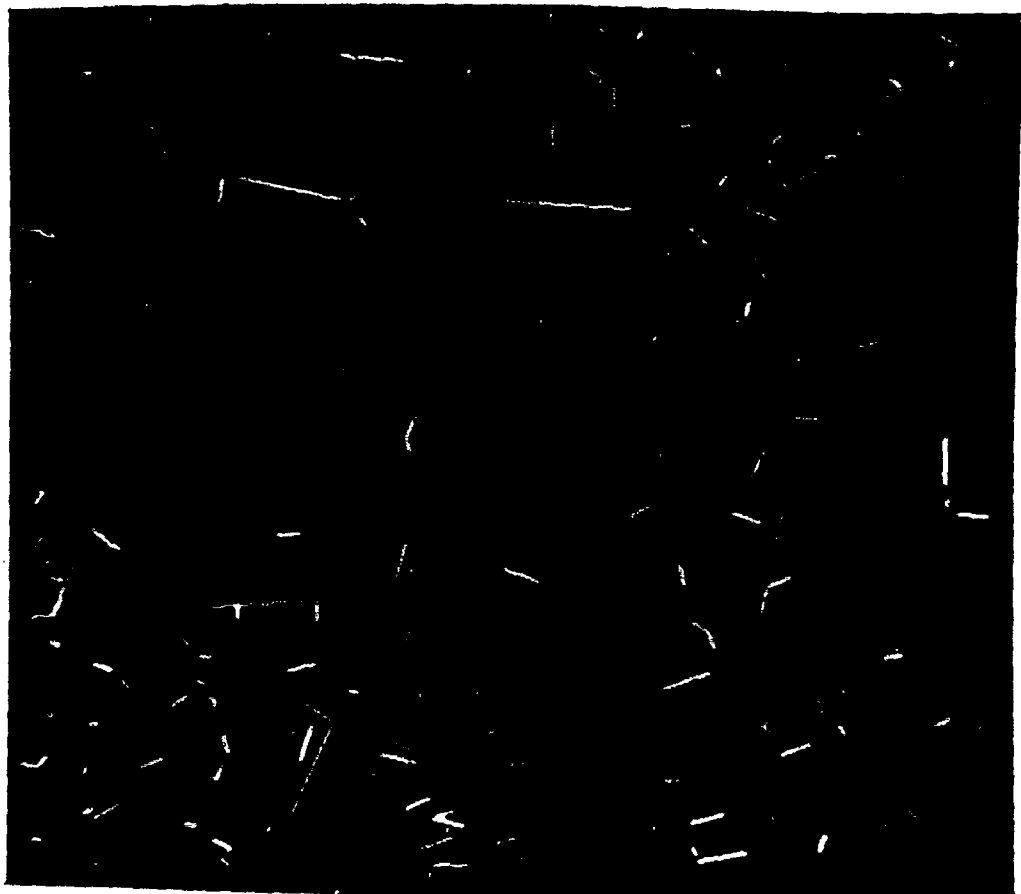


Figure 3-29: Fitting Straight Lines to the Medial Axes

equal.

The other segment joining heuristic is to merge segments whose endpoints are adjacent (i.e., proximate to one another).

These two heuristics are combined into a single co-extensivity property which operates over the prototypical support areas shown in Figure 2-7. That is, a line segment is merged with the current segment if one of its endpoints lies within the support area and its orientation matches that of the current segment. There are several parameters to be selected: the shape and size of the support and the degree of match of the segment orientations. The rule can then be applied recursively either with constant parameters or with parameters that vary as the process becomes more stable.

Figure 3-30 shows in color which line segments have merged using a rule which closes only small gaps and requires a high degree of orientation match. This application of the rule is followed by a further pass (Figure 3-31) permitting segments to be joined over longer gaps but still requiring close orientation match. This result is displayed over the extrema background (in Figure 3-32) to show the relation of the linked segments to the image as a whole.

3.5 EVALUATIONS AND CONCLUSIONS

The finders developed in the course of this project are examples of a whole class of feature finders that need to be constructed in order to recognize and extract typical linear features. In particular, the sponsor has identified the following feature classes as being of significant interest: roads, rivers, bridges, railroads, powerlines, forest regions, bare fields, and airports. Of these, finders for bridges, forest, and roads have been built as described above. Testing these finders has revealed both the power of the approach and the direction that future work must take.

Our approach exploits the hierarchic nature of the feature models by building finders that operate at multiple levels of evidence. The Image Understanding literature has focused on the lower levels of evidence which account for the local properties of the pixels and regions which constitute the feature. Numerous approaches for recognition are based on the ability of algorithms to function well at these lower levels. However, these algorithms tended to fail when confronted by the variability of real world imagery. The examples we have chosen to implement show how the introduction of higher level knowledge is valuable in feature recognition by guiding the lower level algorithms and by introducing the available context as a source of additional evidence. The power of the approach is particularly apparent in the bridge example, where the absence of a "land" signature at one end of a bridge hypothesis is used to critique and dismiss the incorrect hypothesis.

Nonetheless, the scope of the experiments conducted in the course of the project was constrained both by the limited available resources and by the lack of a comprehensive set of terrain images to provide multiple examples of verified feature types. The effect of this is that the algorithms and the models have not undergone thorough testing to identify the extent and quality of their recognition performance. This affects not only how well the finders recognize the features for



Figure 3-30: Merged Line Segments

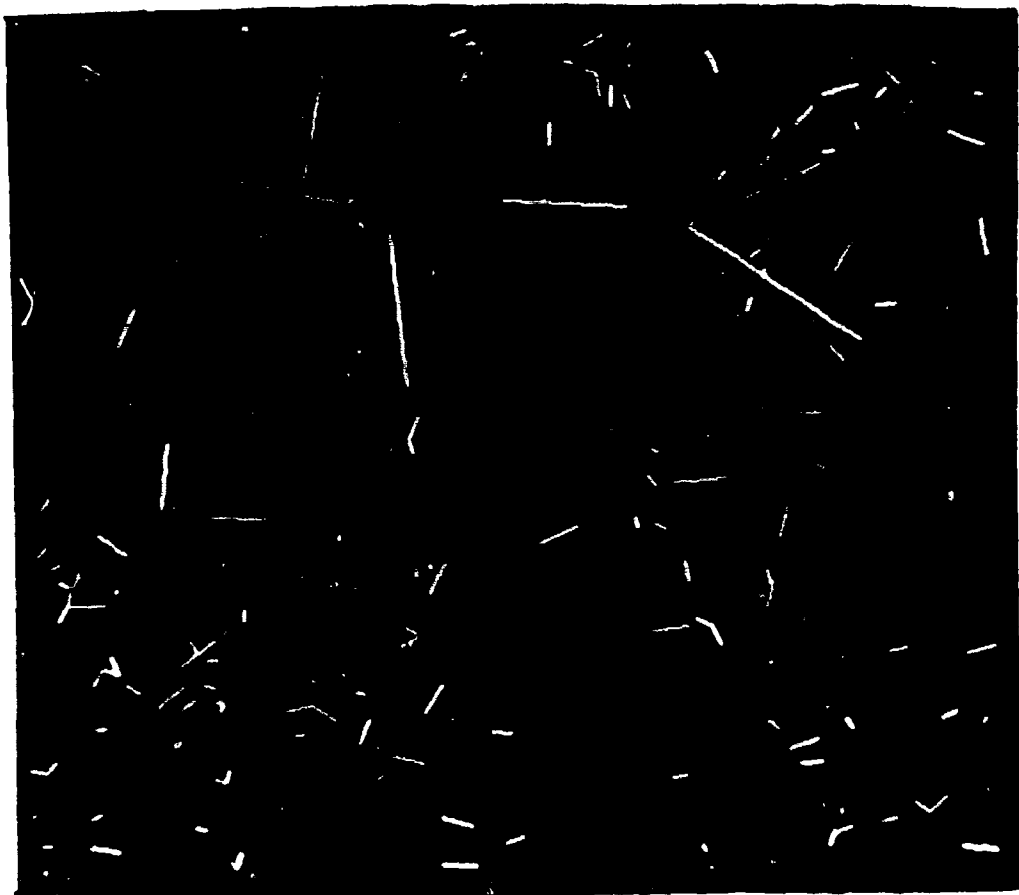


Figure 3-31: Segments Joined Over Longer Gaps



Figure 3-32: Result over Extrema Background

which they are intended but also how often they are confused by other terrain features sharing properties with the intended feature class.

Furthermore, the models which were developed are incomplete and do not avail themselves of significant contextual cues. For example, the bridge finder currently checks for water and land signatures; a more complete model would permit a bridge to be recognized even if there were no water "under" it, if, for example, terrain elevation data could show that the area under the bridge was a local minimum lying along the drainage pattern for the region and that a lack of water could be explained by summer or dry spell.

This additional model knowledge will come out of the recommended additional testing and evaluation. The models are developed and improved based on a cycle which tests, identifies mistakes and inadequacies in the recognition and enhances the model base to provide sufficient knowledge to avoid the diagnosed problem. Another fertile source of expertise to complete the recognition models for terrain features can be found in the training literature for photointerpreters and in discussions with them regarding specific imagery.

Finally, our experiments have convinced us that the continuation of the effort discussed in this final report will lead to the development of feature finders for the additional feature classes mentioned above. The reader is also directed to Sections 1.4 and 1.5 of the Executive Summary for additional conclusions and recommendations.

4. GLOSSARY

angle of illumination (regard) The direction in which the radar beam is emitted from a SAR sensor.

ATFE An acronym for *Automated Terrain Feature Extraction*.

blackboard system A decentralized AI architecture in which autonomous knowledge sources read from and post to a shared data area through blackboard manager(s).

bottom-up A term used in the field of Artificial Intelligence to describe a data-driven technique for search in which properties of the data drive system processing (e.g., the search for terrain features). Opposite in meaning to *top-down*.

direction of illumination The direction in a SAR image from which the radar beam is emitted. Generally, the direction of illumination is considered to come from the top of the SAR image.

finder A *recognition procedure* defined for a specific terrain feature. For example, a bridge finder is responsible for hypothesizing instances (occurrences) of a bridge in a SAR image.

focus of attention A goal directed narrowing of a problem space.

functional terrain feature properties A definition of the functional relationships between a given terrain feature and other terrain features.

Image Structure DataBase (ISDB) A component of the CSI conceptual system architecture; it contains descriptions of image structure which are produced by the *perceptual grouper*.

island-driving A reasoning control technique where hypotheses with high certainty (i.e., "islands") are used to "drive" subsequent hypothesis reasoning.

knowledge source A routine which interprets a well-defined (though perhaps restricted) domain (i.e., a domain expert).

Medial Axis Transform (MAT) A technique for computing a skeleton (linear) representation of an area region; a method of region thinning.

perceptual grouping A term defined by [17] which describes the systematic spatial aggregation of more primitive image structures into more abstract image structures.

platform parameters The parameters which specify the position and orientation of the SAR sensor in three-dimensional space (e.g., elevation, angle of regard, latitude and longitude).

recognition procedure An algorithmic specification of the operations required to hypothesize instances of a terrain feature in a SAR image. A recognition

procedure for a specific terrain feature is called a *finder*.

measurable terrain feature properties A definition of the visible properties of a terrain feature in imagery (e.g., bridges appear bright in SAR images).

SAR image An image formed using Synthetic Aperture Radar (see Appendix A)

top-down A term used in the field of Artificial Intelligence to describe a technique in which a model is used to guide the search through the data (e.g., imagery). Opposite in meaning to *bottom-up*.

5. REFERENCES

- [1] D.H. Ballard and C.M. Brown, *Computer Vision*, Prentice-Hall, Englewood Cliffs, New Jersey, 1982.
- [2] *The Handbook of Artificial Intelligence*, vols. 1-3, A. Barr, E.A. Feigenbaum, and P. Cohen(eds.), William Kaufmann, Stanford, California, 1981.
- [3] T.O. Binford, T.S. Levitt, and W.B. Mann, "Bayesian Inference in Model-Based Machine Vision," *Proceedings AAAI Uncertainty in Artificial Intelligence Workshop*, Seattle, Washington, July 1987.
- [4] P.J. Burt, "Fast Filter Transforms for Image Processing," *Computer Graphics and Image Processing*, vol. 16, pp. 20-51, 1981.
- [5] J.F. Canny, "Finding Edges and Lines in Images," AI-TR-720, MIT Artificial Intelligence Laboratory, Cambridge, Massachusetts, June 1983.
- [6] E. Davis, *Representing and Acquiring Geographic Knowledge*, Morgan Kaufman, Los Altos, California, 1986.
- [7] Defense Mapping Agency, *Product Specifications for Digital Landmass System (DLMS) Data Base*, Second Edition, April 1983.
- [8] J.D. Foley and A. van Dam, *Fundamentals of Interactive Computer Graphics*, Addison-Wesley, Reading, Massachusetts, 1982.
- [9] D. Franks, C. McKee, and G.D. Conner, *Terrain Analysis Techbase 0-Domain Knowledge*, ADRIES Internal Memorandum, Advanced Decision Systems, Mountain View, California, December 1986.
- [10] E.B. Goldstein, *Sensation and Perception*, Wadsworth, Belmont, California, 1980.
- [11] Goodyear Aerospace, *Interpretation Textbook for the AN/UPD-4 Radar System*, GTA-6070A, July 1977.
- [12] P. Kahn, L. Kitchen, and E.M. Riseman, "Real-Time Feature Extraction: a Fast Line Finder for Vision-Guided Robot Navigation," COINS Technical Report 87-57, University of Massachusetts, Amherst, Massachusetts, July 1987.
- [13] L.J. Kitchen and J.A. Malin, "The Effect of Spatial Discretization on the Magnitude and Direction Response of Simple Differential Edge Operators on a Step Edge Part 1: Square Pixel Receptive Fields," COINS Technical Report 87-34, University of Massachusetts, Amherst, Massachusetts, April 1987.
- [14] D.T. Lawton, J. Glicksman, G.D. Conner, and R.J. Drazovich, "Linear Feature Extraction from Radar Imagery," Final Technical Report, ETI-LA-L7-84 1521-E241-83, U.S. Army Engineer Topographic Laboratories, Fort Belvoir, Virginia, August 1985.

- [15] L. D. Erman and V. R. Lesser, "A Multi-Level Organization for Problem Solving using Many Diverse, Cooperating Sources of Knowledge" *IJCAI* 4, pp. 483-490.
- [16] M.D. Levine, *Vision in Man and Machine*, McGraw Hill, New York City, New York, 1985.
- [17] D.G. Lowe, *Perceptual Organization and Visual Recognition*, Kluwer Academic, Boston, 1985.
- [18] D. Marr, *Vision*, W.H. Freeman: New York, 1982.
- [19] Y. Ohta, *Knowledge-Based Interpretation of Outdoor Natural Color Scenes*, Pitman Advanced, Marshfield, Massachusetts, 1985.
- [20] R.F. Pascucci, "An Automated System for the Computer-Assisted Identification of Features on SAR Imagery," Final Technical Report, U.S. Army Engineer Topographic Laboratories, ETL-0415, Jan. 1985 -- Dec. 1985.
- [21] A. Rosenfeld and A.C. Kak, *Digital Picture Processing*, Academic Press, New York City, New York, 1976.
- [22] K.J. Schmucker, *Fuzzy Sets, Natural Language, Computations, and Risk Analysis*, Computer Science Press, Rockville, Maryland, 1984.
- [23] D.S. Simonett and R.E. Davis, "Image Analysis-Active Microwave," *Manual of Remote Sensing*, vol. I, American Society of Photogrammetry, 1983.
- [24] M.A. Vogel and G.S. Zabele, "SAR DTED/DFAD Auto Extraction," Final Technical Report, Rome Air Development Center (Air Force Systems Command), RADC-TR-86-18, March 1986.
- [25] T.E. Weymouth, "Using Object Descriptions in a Schema Network for Machine Vision" COINS Technical Report 86-24, University of Massachusetts, Amherst, MA May 1986.
- [26] G.D. Conner, D.T. Lawton, C.C. McConnell, and D.L. Milgram, "Linear Feature Extraction from Radar Imagery, SBIR Phase II, Base Contract", Final Technical Report, ETL-0469, U.S. Army Engineer Topographic Laboratories, Fort Belvoir, Virginia, July 1987.
- [27] G.D. Conner, D.L. Milgram, D.T. Lawton, and C.C. McConnell, "Linear Feature Extraction from Radar Imagery, SBIR Phase II, Option I", Final Technical Report, ETL-0469, U.S. Army Engineer Topographic Laboratories, Fort Belvoir, Virginia, April 1988.