

TEC-0031

This technical report originally
contained COLOR ILLUSTRATIONS.
For more information, contact the
TEC COR.

AD-A276 817



DFAD PLUS System Development

Lawrie Jordan Nick Faust
ERDAS, Inc.
2801 Buford Highway
Atlanta, GA 30329

DTIC
ELECTE
MAR 18 1994
S B D

John Gilmore
Gilmore Aerospace Corporation
1225 Johnson Fairy Road, NE
Suite 460A1
Marietta, GA 30068

March 1992

Approved for public release; distribution is unlimited.

Prepared for:
U.S. Army Corps of Engineers
Topographic Engineering Center
Fort Belvoir, Virginia 22060-5546

94 3 18 007

61
pgs

94-08837

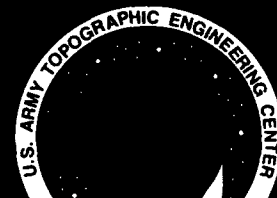


US Army Corps
of Engineers
Topographic
Engineering Center

T

E

C



1

**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

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503

1. AGENCY USE ONLY (Leave blank)		2. REPORT DATE March 1992	3. REPORT TYPE AND DATES COVERED Final Technical Sep. 1987 - Mar. 1992	
4. TITLE AND SUBTITLE DFAD PLUS System Development			5. FUNDING NUMBERS DACA72-87-C-0007	
6. AUTHOR(S) Lawrie Jordan ¹ John Gilmore ² Nick Faust ¹				
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) ¹ ERDAS, Inc. 2801 Buford Highway Atlanta, GA 30329			8. PERFORMING ORGANIZATION REPORT NUMBER ² Gilmore Aerospace Corporation 1225 Johnson Fairy Road, NE, Suite 460A1 Marietta, GA 30068	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) U.S. Army Topographic Engineering Center Fort Belvoir, VA 22060-5546			10. SPONSORING/MONITORING AGENCY REPORT NUMBER TEC-0031	
11. SUPPLEMENTARY NOTES				
12a. DISTRIBUTION/AVAILABILITY STATEMENT Approved for public release; distribution is unlimited.			12b. DISTRIBUTION CODE	
13. ABSTRACT (Maximum 200 words) Between 1987 and 1992, ERDAS, Inc., and Gilmore Aerospace Corporation developed a prototype expert system for the update of Digital Feature Analysis Data (DFAD) based on multispectral imagery and a number of commonly available Geographic Information System (GIS) layers. A method was developed by which ERDAS GIS Modeling tools were used to simulate and validate decision rules that were then downloaded to the Ada Based Expert System Testbed (ABEST) for execution. ERDAS developed data acquisition programs for acceptance of DFAD point, line and polygon data and conversion of DFAD into the ERDAS internal polygon file structure. The information is subsequently gridded to produce a raster GIS layer. ERDAS also made improvements to a semi-automatic Digital Terrain Elevation Data (DTED) loader.				
14. SUBJECT TERMS Geographic Information System, Feature Extraction, Expert System, Digital Feature Analysis Data, Multispectral Imagery			15. NUMBER OF PAGES 60	
			16. PRICE CODE	
17. SECURITY CLASSIFICATION OF REPORT UNCLASSIFIED	18. SECURITY CLASSIFICATION OF THIS PAGE UNCLASSIFIED	19. SECURITY CLASSIFICATION OF ABSTRACT UNCLASSIFIED	20. LIMITATION OF ABSTRACT UNLIMITED	

CONTENTS

TITLE	PAGE
FIGURES	iv
PREFACE	v
ABSTRACT	1
INTRODUCTION	1
BACKGROUND	2
DEVELOPMENT STRATEGY	5
GISMO MODELING	10
TEST SITE RULE DEVELOPMENT	13
ABEST EXPERT SYSTEM IMPLEMENTATION	19
CONCLUSION	23
APPENDIX A	25
APPENDIX B	30
APPENDIX C	35
ILLUSTRATIONS	41

Accession for	
NETZ 6001	<input checked="" type="checkbox"/>
DEPT 100	<input type="checkbox"/>
UNIVERSITY	<input type="checkbox"/>
STATE	<input type="checkbox"/>
Date received	
Disposition	
Remarks	
Port	
A-1	

FIGURES

	TITLE	PAGE
1	Expert System Flow	42
2	DFAD Loading Process	43
3	SPOT XS and PAN Merge for Denver Test Site	44
4	Roads and Hydrology for Denver Site	45
5	Decision Tree	46
6	Tasseled Cap for Spring Lanier	47
7	Water Logic	48
8	Preliminary Classes	49
9	High Density Urban Classification	50
10	DFAD PLUS Classification Comparison for Spring Lanier	51
11	DFAD PLUS Confidence	52
12	DFAD PLUS Classification Comparison for Winter Lanier	53
13	DFAD PLUS Confidence	54
14	DFAD PLUS Classification for Denver Test Site	55

PREFACE

This report was prepared under Contract DACA72-87-C-0007 for the U.S. Army Topographic Engineering Center, Fort Belvoir, Virginia 22060-5546 by ERDAS, Inc., Atlanta, Georgia 30329, and Gilmore Aerospace Corporation, Marietta, GA 30068. The Contracting Officer's Representative was Mr. Richard Tynes.

I. Abstract

Between 1987 and 1992, ERDAS, Inc. and Gilmore Aerospace Corporation developed a prototype expert system for the update of Digital Feature Analysis Data (DFAD) based on multispectral imagery and a number of commonly available Geographic Information System (GIS) layers. A method was developed by which ERDAS GIS Modeling tools were used to simulate and validate decision rules that were then downloaded to the Ada Based Expert System Testbed (ABEST) for execution. ERDAS developed data acquisition programs for acceptance of DFAD point, line and polygon data and conversion of DFAD into the ERDAS internal polygon file structure. The information is subsequently gridded to produce a raster GIS layer. ERDAS also made improvements to a semi-automatic Digital Terrain Elevation Data (DTED) loader.

II. Introduction

The Small Business Innovative Research (SBIR) Program is one method by which innovative ideas from university and industry may be developed into practical products. The SBIR program is funded in several steps with an evaluation of progress occurring at each step. Initial funding normally leads to a detailed design that can be implemented in later phases. Phase II is generally a development phase in which software and/or hardware are developed that have development potential towards a commercial product. Phase III is focused on productizing the work of Phase II.

ERDAS, Inc. and Gilmore Aerospace (GA) decided to team on a SBIR proposal which would combine the strengths of the two companies' commercial software products. ERDAS produces commercial software for the integration of image data into Geographic Information Systems (GIS). ERDAS has its own raster based GIS which provides a large variety of analysis functions for many types of information that are tied together by spatial location. In addition, ERDAS provides sophisticated image processing tools for analysis of single and multi-channel image data sets.

Gilmore Aerospace is an Atlanta, Georgia based company that has developed a product called the ADA Blackboard Expert System Tool (ABEST). This tool has been developed in ADA for eventual implementation into complex development environments that require the ADA language. An expert system tool is used to develop and exercise semantic rules that govern how a particular process is to operate. An expert system

combines knowledge from experts in a particular field into a set of rules that then can be queried to make or help make decisions.

The problem that was selected was the integration of image processing, GIS, and expert systems techniques and the application of those techniques toward the development of a system that would allow for the creation and update of Digital Feature Analysis Data (DFAD) from imagery and existing GIS information. Current systems utilize imagery extensively and use map information as supplementary information, but do not directly use GIS information in the decision making process. The ERDAS team proposed the concept of a digital classifier that utilized GIS information within the decision making process using an expert system implementation.

III. Background

Digital Feature Analysis Data (DFAD) is produced by the Defense Mapping Agency (DMA) for use by the U.S. Government and approved contractors. This data is digital information coded into geographically based points, lines, and polygons. Associated with these graphic entities are multiple attributes. For example a particular class of low density urban may consist of all areas that have a certain percentage coverage of roof materials, grass, and concrete. The data set may be thought of as a detailed land use/land cover classification that has more detail in urban regions than in rural areas. The original intent of the DFAD data set was to model radar return which has high response to manmade objects due to the sharp corners and radar reflecting materials used. More detail was thus coded in the urban regions. The rural regions with predominantly natural features was less accurately modeled. Current requirements for DFAD data also required accurate information for natural regions so that analyses such as cross-country mobility may be performed. Image data acquired from satellite and aircraft may be easily used within a GIS structure to update the general classification scheme for rural areas. However, to use this functionality, the DFAD data set must be converted to a standard GIS format. One of the goals of this project is to accomplish this transition. The structure of the DFAD data is documented in DMA publications (reference 1). DFAD data currently exists in two levels of resolution, with DFAD Level I having a positional accuracy of approximately 100 meters and DFAD Level II having a positional accuracy of approximately 30 meters. These levels of resolution roughly correspond to the Level I and II of Digital Terrain Elevation Data (DTED), also produced by DMA, showing the shape of the local terrain. Additional detail in Level I is given by DFAD Level I C which provides additional information on lines of

communication. This data set has more information on the position and types of road networks, power lines, power generation stations, etc. Since the existing DFAD structure is not compatible with most of the common GIS commercial systems currently in existence, a conversion utility must be developed to accept DFAD data in its raw form and provide it in a form that can be shared with common GIS systems.

ERDAS was founded in 1979 as a spin-off of research efforts at the Georgia Institute of Technology and Harvard University. ERDAS develops multipurpose computer systems for commercial sale with a particular focus on the integration of satellite image information with traditional raster based GIS. ERDAS currently has approximately 3000 systems in the field, and has an extensive world wide distribution system. The principal operating office of ERDAS is in Atlanta, Georgia, with field offices in California, Wisconsin, Illinois, and South Carolina. To support its software development with thorough testing, ERDAS has extensive quality assurance and quality control checking. In addition an experienced staff conducts training at ERDAS and in certain cases on a customer's site. The ERDAS Production Department provides extensive software checking and utilization on large scale projects such as a current project for the Georgia Department of Natural Resources in which land cover maps are provided for the entire state of Georgia on a 7-1/2 minute quadrangle basis in an Arc/Info vector format. Landsat Thematic Mapper (TM) data were geographically referenced, classified using supervised and unsupervised classification, and converted into a topologic data structure. The Production Department uses the latest ERDAS tools in its projects, thereby finding software problems before they get to ERDAS's users.

ERDAS software is specifically designed to be hardware independent, since ERDAS supports numerous vendors' products. In the workstation environment, ERDAS currently supports Sun, Data General, Hewlett Packard, Silicon Graphics, DEC, and IBM platforms. ERDAS also supports IBM compatible personal computer (PC) systems with external image processors, as well as a low cost option using a standard VGA graphics card.

One of the primary strengths of ERDAS software is the ERDAS software Toolkit, which allows users to design and implement their own software modules using ERDAS low level functions for accessing disk files and writing to an image processor screen. This Toolkit is well documented and easy to use. The ERDAS menu structure is built upon text menu files that can be edited and created by users who wish to configure their own system.

Existing ERDAS applications may be renamed in a menu that pulls together applications that are often used into one basic menu. The user has the ultimate flexibility to completely revise the ERDAS menu system into a system which may be more responsive to a users needs.

Error checking for input/output is supported in the software toolkit with prompts that only allow specific answers to questions, as well as other functions that only allow a specified range of user responses to questions requiring numerical answers. This error checking capability is a trademark of ERDAS that makes the system easy to use and flexible.

The ERDAS GIS Modeling function (GISMO) is an extremely powerful function which allows arithmetic and logical functions to be applied to multiple GIS and image raster layers. A function is defined by a script language that opens files, provides a data flow structure, and outputs multiple results to an image screen or to output files. Many traditional GIS functions can be implemented in a GISMO file structure and saved for later retrieval as a named function. GISMO applies the specified function to every pixel in each selected input file and one or more calculated values to user specified output files. GISMO manages a library of functions that have been developed by ERDAS and accommodates new functions that can be easily developed by ERDAS users.

Gilmore Aerospace was founded in 1985 also as a spin-off of research at Georgia Tech. An initial software product of GA is the Ada Blackboard Expert System Testbed, or ABEST. GA employees have extensive experience in the development and use of expert systems for applications ranging from knowledge-based autonomous vehicle systems to manufacturing design. ABEST provides a window based user interface which allows the user to interact and query the expert system throughout its processing. The model supports forward and backward inferencing engines as well as debugging tools. The implementation of ABEST in ADA is a forward looking thrust into a growing market place in systems developed for the government.

The Ada Blackboard Expert System Tools (ABEST) was developed by Gilmore Aerospace to address the need for artificial intelligence capabilities in Ada-based military applications. ABEST is a generic tool that incorporates forward reasoning inferencing with multiple knowledge representation schemes in a distributed parallel blackboard problem

solving paradigm. The repertoire of functionality provided by ABEST allows users to tailor the tool to their specific applications.

In a typical expert system application, problem solving algorithms are applied to a global database of knowledge to deduce a solution. The DFAD PLUS system used expert system technology to classify pixels given a set of satellite images. The design of such a system consists of a knowledge base of rules concerning the domain of the problem, a control structure for applying those rules, and problem specific data and knowledge.

The ABEST system provided the framework for creating, editing, storing, executing, testing, and explaining rules. Supporting the rules are data related elements, facts and frames, and control related structures, knowledge sources and plans.

IV. Development Strategy

The basic strategy for implementation of DFAD PLUS involved the use of both ERDAS and Gilmore Aerospace commercial code. Major new development of software was not anticipated, even though necessary changes would be made to make the process more efficient and to be able to read in standard DMA DFAD data sets. The goal was to develop an expert system implementation of the DFAD PLUS algorithms by providing methods for rule development and testing and final implementation of those rules in the expert system.

The development strategy for the DFAD PLUS project involved a number of steps which were followed sequentially:

- 1) Develop overall logic flow
- 2) Choose initial development platform
- 3) Evaluate separation of tasks between image processing/GIS and expert systems
- 4) Develop DFAD and DTED loading programs
- 5) Select candidate test sites based on contractor owned and government provided data sets
- 6) Select raw and interpreted variables for detailed analysis
- 7) Use ERDAS knowledge to define a layered classification flow
- 8) Determine needs for intermediate data products

- 9) Develop needs for layered classification in ERDAS GISMO
- 10) Iterate on GISMO classification scheme
- 11) Create ADA rule set based on GISMO rules
- 12) Create efficient implementation of ABEST for classification
- 13) Create ADA structure for post classification analysis
- 14) Select final hardware for implementation
- 15) Evaluate ABEST classification results versus GISMO results

The initial logic for the DFAD PLUS classification system was derived from the extensive experience in house at ERDAS in land/cover - land/use classifications. ERDAS has performed many large scale classifications using satellite multispectral data along with GIS data layers which have been digitized from existing maps or were already available digitally. An initial expert system approach was developed and formulated into a detailed flow chart for classifier logic flow (figure 1). The initial logic involved a number of derived data layers which may or may not be available in the base data layers for a particular site. The original logic involved a number of separate expert systems which were tied together by the ABEST blackboard logic with specific processing programs to develop intermediate products that could be directly involved in the classification. If the classification system needed extra data to complete a classification, it could query the expert systems through the blackboard structure to see if the data existed, and if not, a message would be sent to one of the subservient expert systems to generate the intermediate product. This logic represented the 'perfect world' scenario in which all data needed are available at high resolutions and high detail.

Soon after the project started in 1987, a choice was made as to the initial development platform. The Sun 386i was chosen because there was an existing ADA compiler for it and both ERDAS and ABEST operated on that system. In addition, the Sun system supported Sunview, a sophisticated windowing system which would provide an improved interface on a UNIX platform.

A natural division of labor between the ERDAS and the ABEST components of the DFAD PLUS classifier was then made which allocated all data ingest, preprocessing, and geometric correction steps for all potential data layers to the ERDAS system. Initial rules would also be developed using the then current ERDAS 7.3 version GIS and image processing tools. At this point in the development, a number of different steps were

required using many different ERDAS executables. The complex logic could be captured in an ERDAS Audit Trail file, and could be rerun by simply changing input file names.

To achieve the goals of the development of an expert system based DFAD creation and update system, the need was seen to develop a capability for the reading and reformation of DFAD data into the ERDAS system. The ERDAS system uses a raster based GIS format that has each cell of a raster image coded as to color and class type. The DFAD data set on the other hand is vector in nature and represents entities in point, line, or polygon form. ERDAS also supports its own vector structure but the vectors are used only for graphic overlay or as a source for developing a raster GIS layer. No analysis is done with vector data in ERDAS.

The initial task involved the importing of the DFAD data and conversion into the ERDAS vector format (.DIG file). A DFAD loading capability was developed as a part of the ERDAS Data Conversion Module, which also handles a number of other vector formats (Figure 2). Once the data has been converted into a .DIG file, it can be plotted on the screen using ERDAS program DISPOL.

The next step is to determine the cell size of the desired raster layer. This is usually driven by the existing cell sizes of multispectral imagery (TM = 30m) or of other existing GIS layers. The ERDAS program GRDPOL is run next with the geographic extent made similar to other GIS layers. The output of GRDPOL is a raster GIS file of DFAD classes (Figure 2).

To load the corresponding DTED elevation data set, improvements were made to existing raster data loading programs to create a program LDDTED. LDDTED provides the capability of initial scan of the data tape to determine the DTED cells resident on the tape. Once the desired cell is located, the program allows direct loading of the elevation data into an ERDAS 16-bit .LAN file.

DTED data are organized into either 3 arc second (Level I) or 1 arc second (Level II) raster images. This is appropriate for DMA provided data since DMA has responsibility for worldwide coverage; however, for the data to be used directly in a GIS for a local area, a conversion must be applied to transform the arc second data into the selected map projection. At that point, the elevation data are interpolated into a square pixel in meters or feet.

To provide the transformation matrix for this conversion, ground conversion control points must be found with coordinates both in geocentric units and map projection units. Since each pixel in a DTED cell implicitly has a geo-coordinate, a subset (approximately 25) of the cells are selected in a coarse grid and run through the ERDAS CCVRT program to get the projection coordinates. Next, the COORDN program is run with a selection of a third order least squares fit. A third order transformation is accurate enough for transformation of a 1 degree by 1 degree cell.

Finally the ERDAS program NRECTIFY is run to create the appropriate digital elevation model (DEM).

The next step in the process was the selection of test sites for the DFAD PLUS strategy. The concept was developed in a manner such that rules for differing environments and times of the year could be included into an overall classification scheme, with the available data sets dictating which detailed logic was to be implemented. ERDAS has extensive information over an area just north of the city of Atlanta in the Georgia Piedmont Geomorphological area. In addition equivalent sets of data were selected as one of the test sites. TEC provided imagery (Figure 3) and terrain information over a second test site near Denver, Colorado where vegetation patterns are significantly different than those encountered in the southeastern United States (Figure 4). Level I DFAD data was provided for both sites. A general description of the data required for the analysis is given below, with a detailed description of the data sets used in rule development given later in this report.

For each site multispectral data (Thematic Mapper Digital Imagery and SPOT 10 meter Panchromatic and 20 meter color infrared XS data) were acquired and processed. DFAD and DTED Level I data were acquired and processed for the study areas. Soils information was digitized and included in the data set with as much detail as was warranted by the resolution of the source data sets. Transportation variables were collected from U.S. Geological Survey maps at a scale of 1:24,000. Hydrology information was also digitized from the same scale maps. Geopolitical information such as city boundaries and location of public lands was also digitized. Initial intermediate variables for analysis were determined to be slope/aspect (generated from DTED), pattern recognition classifications of multispectral data, and a number of proximity variables having to do with distance from features such as water and transportation.

Using the raw (image and feature) data and the derived intermediate variables, an initial layered classification system began to be developed. The first step in the layered classifier was to define regions of water, vegetation, and bare ground. Initially, the original TM and SPOT bands were used to perform this separation; however, a color transformation into a Tasseled Cap (reference 2) image representation gave much better breakout of the three basic categories. The Tasseled Cap transformation is an empirically derived set of coefficients that can be used with the original TM and MSS bands to give resulting transformed band which represents brightness, wetness, and greenness. Greenness is often used as a measure of the vegetation a scene, and high brightness often corresponds to bare ground and urban areas. Since our three desired categories parallel the Tasseled Cap channels, a simple interpretation of the transformed channels could be used directly. An equivalent transformation of the SPOT XS data was not available, so TM data was used with a later panchromatic merge to improve spatial resolution.

Now, armed with a general classification, we proceeded to define functional 'rules' that would allow the extraction of more detailed classes from the general classes. The rules could involve raw image data values, transformed image data values, GIS values, and derived GIS or image values. The 'rules' were developed with the ERDAS Production Department's assistance which brought a wealth of practical experience in image classification and GIS development and analysis into the project. Steps that are normally followed in a large area land cover classification were included in the definition of potential intermediate data variables that might allow a classification to achieve a better accuracy. The initial rules were diagrammed (Figure 5) into a decision tree that sequentially made finer and finer discriminations as the tree was traversed.

The process for defining a potential class for each database pixel could follow the decision tree, but at that point there was no ancillary knowledge as to the perceived accuracy of that classification. A method needed to be developed that would consider the process throughout the decision tree to decide a relative confidence in the output pixel's classification. An expert system has the capability of assigning confidence factors, so an analog to this process was needed in the ERDAS procedure for developing rules for the expert system.

ERDAS has an existing modeling tool called GISMO (GIS Modeling) that would allow interactive rule definition within the ERDAS framework. Once the rules are

developed, tested, and validated, the rules may be passed on to an expert system for implementation. GISMO is an extremely powerful function that allows extensive models to be developed using numerous GIS variables and almost any algebraic and logical function. An analogous function for image data is in the ERDAS ALGEBRA function. Since DFAD PLUS rules were designed to incorporate raw data values as well as GIS data values, the multichannel image data sets were converted into single layer GIS variables.

Initial implementation of the GISMO rules into the standard ABEST system resulted in extremely long computation times for an image classification. Extensive work was done with ABEST to create a streamlined version of the ABEST code which did not implement all expert system functions, but did satisfy the above requirements. This allowed the desired classifications and backtrack logic, while still satisfying the desire for reasonable computation times through the development of a rule compiler. ERDAS worked with Gilmore Aerospace in development of an accurate implementation of the GISMO rules.

As development proceeded on the prototype system for DFAD PLUS, the technology of workstations bypassed the initial test system. Sun no longer supported the 386i system, and had moved into the Sun 4 series with a SPARC implementation. A decision was made jointly by TEC and the ERDAS team that the system should be moved to the SPARC architecture. This change caused extensive modifications in the user interface and data structures of the DFAD PLUS system even though both systems were made by the same vendor. Since ABEST is written in ADA and requires an ADA compiler to run, a new ADA compiler was required for ABEST development.

V. GISMO Modeling

GISMO has the capability of defining a text script that controls the processing of multiple GIS files and outputs to the screen or to data files. ERDAS has written the GISMO interpreter to be flexible and easy to use with specific sections for the definition of input variable names, and algebraic functions or logical operations. The GISMO control structure allows looping, logical decisions, and numeric decisions. A simple example of a GISMO function for the creation of a general vegetation index from image data channels is shown below:

```
#  
# This is an example GISMO file
```



```

#
data
#   TM image data is separated into separate files
#
input ch1    file "TM1"
input ch2    file "TM2";
input ch3    file "TM3";
input ch4    file "TM4";
#   Output vegetation index
output DVI   file "DVI";
#   define program variables
integer DVI;
integer numer;
integer denom;
start
numer = ch4 - ch2
denom = ch1 + ch3
DVI = numer/denom
end

```

For this example, the first four TM channels have been separated into 4 files: TM1.GIS, TM2.GIS, TM3.GIS, and TM4.GIS. An output file called DVI.GIS is also created. Numer and denom are intermediate variables, and DVI is the result. Given this model, GISMO will process each pixel in the input files and create a pixel in the output file. The file size is taken care of automatically such that all data are processed.

Initially a set of rules were defined for each of the classes that we wished to discriminate. These were encoded in GISMO models to provide a 1 class classifier. After successful implementation of the full individual classification models, the models were combined into a general set of rules that included dependencies between previous rules. For example, a low density urban classification might require that a pixel be previously categorized into a general urban class before it could be considered for the low density class. Because of this factor, it became important to correctly order the rule satisfaction such that a prerequisite class designation would have been assigned before a detailed rule that requires it.

GISMO allows the interactive updating and changing of the rules for a given model. The model is kept in a library that is managed by the GISMO program. A user can define his own model, save it, and transfer it to other ERDAS users on other systems. The library system allows the user to look at selected models, edit the models, and print the models.

Once the initial GISMO model for tree classification has been developed, the process normally involves extensive interaction by the user to tailor his rules to provide more accurate classifications. As the rules are changed, more intermediate variables may be required to be derived from the initial image and GIS data sets. Proximity variables must also be analyzed carefully. One of the challenges to the use of such a decision structure is the choice of weights for each of the raw and derived data variables. For example, is it more important that an image pixel satisfy certain grey scale constraints than it is for that pixel to be within a neighborhood of another class? If the constraints are more important, then the decision rules should weight the decisions based on raw data values greater than that based on a proximity variable. Estimating these weights for our test areas was not attempted for this project since a great deal of subjective judgement is often necessary for adequate definition. The weights for each data source were established as equal for our test cases. GISMO, however, does not require equal weights. These weights could be implemented as a measure of the relative accuracy of each data source.

For each decision point in the decision tree, a number of rules may be satisfied. The terrain slope and elevation at a point may assist in the determination as to whether a pixel represents upland softwoods or lowland hardwoods. The image data values from either the transformed Tasseled Cap or the raw TM data may be used to estimate vegetation cover, and soils data may be considered to determine if such a vegetation type is normal for the soil type associated with each database pixel. If a pixel has an appropriate data value to belong to a hardwood class for a particular season, if the area is near water, if the area is on a level slope in a valley, and is on a wetland soil, then the classifier will assign the output pixel to that class. GISMO allows that another variable may be defined to be the number of rules satisfied in that decision. If all are satisfied, then a certainty of 100% is given. If only half of the rules are satisfied, a confidence of only 50% is given. If a low confidence factor is given, the pixel might reasonably belong in another class, and decision rules in the decision tree above should be examined. For each classification GISMO provides a resulting classification and a confidence factor. By rearranging the decision process, rules might be developed to find the second most likely class for each pixel, given that the first classification is uncertain.

The processing speed for GISMO models is very fast, providing quick feedback to the rule developer. By continuous iteration, the developer finds the set of rules that generally provide a classification with high confidence. The next step is the implementation of these rules in an expert system. The ABEST system is written entirely in ADA and requires the rules be defined in an ABEST readable fashion. Rules developed in GISMO were then coded into the ABEST format by Gilmore Aerospace and installed into ABEST. Requirements for the classification were developed that required computation of an output classification, computation of a confidence factor for each pixel, and the expert system implementation of a backtracking logic that would allow the detailed evaluation of the decision process for each process. This system would show each rule satisfied along with its relative confidence factor.

It is possible in the GISMO Modeling scheme that a pixel might have the same confidence factor for several class types. In this case the GISMO model will assign the output pixel to the class that occurred first in the decision tree. The order of the rules in the GISMO file will thus affect the resulting classification if equally likely classes have been encountered. In general, the class category rules are listed in the order of water/wet classes first, vegetation classes second, and bare ground/urban classes last.

VI. Test Site GISMO Rule Development

A separate set of rules was developed for each of the test sites and imagery dates. The detailed rule set for all sites are given in Appendices to this report.

The initial image data set considered (WINTER-LANIER) was a winter TM scene for an area around Lake Lanier, a large Corps of Engineers reservoir near Gainesville, Georgia. This test site is approximately 50 miles northeast of the city of Atlanta, and was available for field checking of classification results. The image data gathered for the winter test classification included six channels of the Landsat Thematic Mapper data with the thermal data channel not being used. GIS variables included Digital Elevation Model (DEM) data at a 1:24,000 scale, previously gathered land use information, DFAD data, detailed soil series data, hydrology and flood plain data, town boundaries, and the location of transportation and utility corridors.

Derived GIS layers were slope, derived from the DEM; and proximity data showing distance to hydrology and transportation features. Derived image data variables were computed by applying a Tasseled Cap (reference 2) transformation to the raw TM data values (Figure 6).

Much of the GIS data were digitized using the Polygon Digitizing Module of the ERDAS software. The data were digitized in a UTM projection from U.S. Geological Survey 7-1/2 minute topographic quadrangles. Soils data were digitized from varying scale county soil maps. The polygon and vector information were immediately converted into a raster GIS file at a cell size equal to the 30 meter spacing of the TM data that had been rectified to the same projection using the GRDPOL program in the same manner as it was used for DFAD data.

Appendix A shows the GISMO model for the winter DFAD PLUS classification. For this test site the available DFAD Level I data that were available for the site were not used in the classification. The objective of this test site was to build a DFAD like classification assuming no DFAD was available.

The initial file opening part of the GISMO file opens a number of GIS files for processing as well as two output GIS files. The raw TM data had previously been processed into a Tasseled Cap transformation and the resulting 3 channels for wetness, greenness, and brightness loaded into separate GIS files. The proximity data sets and the derived data layers were loaded next. The last data set loaded is a further processed resultant of the Tasseled Cap transformation. A GIS contiguity model was run on the vegetation variable, with contiguous areas of vegetation identified.

Next in the model is the definition of a number of intermediate variables that will exist only within the model. Class names representing the major DFAD categories were then defined.

Between the start and end statements in the model the logic is defined for a layered classifier. This logic is to be applied to each pixel in the co-registered data sets. As mentioned above, the logic proceeds top to bottom within the model. All options within the structure are evaluated with the result being the one with the highest level of confidence. Comment statements within the model are preceded by a pound sign (#).

For the first category, water, the following logic is applied. If a pixel's value in the wetness variable is greater than the mean of the wetness plus one standard deviation; the wetness value is greater than the corresponding value in the greenness variable; and if the brightness variable is between the minimum and the mean value minus one standard deviation; then the mask variable is assigned a value of 1 (water class). The values in the GISMO model have been computed from the statistics associated with each layer (Figure 7).

If, on the other hand, the brightness variable has a value greater than the brightness mean; if the greenness variable has low values, and the brightness is greater than the greenness; then the mask variable is assigned a value of 2 (soil-like class).

Next, if the greenness variable has a value that is greater than the mean of the greenness minus 1 standard deviation, and none of the other two classes have been chosen, the mask is assigned to 3 (vegetation class) (Figure 8).

Now, once the general classifications have delegated each pixel into one of the three basic classes (water, soil-like, and vegetation), specific rules may be applied for each of these categories. The water class is the simplest of the modeled DFAD classes. If the mask variable is 1 for water, and the pixel is within a close proximity to existing hydrology, a combined value of the HVAL variables may be 10. If the mask is 1 and the proximity to existing hydrology is greater than a small distance (2), then the value would be 6. If the preliminary classification declares that it is not a water class, but the proximity criteria is met, the combined value is set to 0. This combined value is set to the variable name and represents the confidence (0->10) that the pixel is of a water class. Note that the values can be negative 4 if neither criteria is met.

The next class to be considered is the general urban class. Five intermediate variables are used in the general urban rule in addition to the mask variable defined above. For general urban, the value of TVAL is assigned to 1 if the proximity of the pixel to existing transportation is less than 40 pixels (1.2 kilometers), or 0 if the distance is greater. The value of SLOV is defined to be 1 if the local slope at a pixel is less than 20 percent and 0 if it has a greater slope. The value of the third variable, MVAL is equal to 1 if the prior classification has placed the pixel in the soil-like category. If the greenness variable has a very low value, the variable GVAL is set to one. If it is not, the value is again 0. Lastly,

the wetness variable is tested to see if the pixel has a low wetness. If so, a 1 is assigned to the WVAL variable. If the pixel wetness value is high, the WVAL variable is set to zero.

The weight associated with the general urban category is simple sum of the above intermediate variables scaled on a 1 to 10 basis (urban). In all cases, for this model the output weights will be in the 1 to 10 range.

Now, sub categories of the urban class can be considered. For high density urban, some of the criteria applied in a classification of urban are tightened. Distance to existing transportation now must be less than ten pixels, or .3 kilometers, and allowed terrain slope is mandated to be less than 10 percent. If the pixel is already classified as general urban, if it has a very high brightness (greater than the mean of the soil-like variable plus 2 standard deviations), and if it has a very low greenness value, the HDU variable will be assigned a 10 (Figure 9).

For low density urban, similar criteria are considered with less emphasis on the high brightness and low wetness values. The LDU variable is assigned based on whether the general class is urban, the brightness is between the mean and mean plus one sigma range, and the wetness is between the mean minus 1 standard deviation and the mean. For a pixel satisfying all those criteria, the LDU variable is set to 10.

Bare urban is a class that identifies bare land within a city. If a pixel has very high brightness values and very low wetness values and is in the general urban class, the BARE_U variable is set to 10.

The bare field class is considered next. The bare field class includes bare dirt areas that are not within a city. Thus, the general urban category is not considered. Bare fields are those with a low percent of vegetative cover. The criteria for the bare field category are defined below:

- 1) Distance to existing transportation less than 1.5 kilometers
- 2) Slope less than 10 percent
- 3) Soil-like general classification
- 4) Very low wetness
- 5) Low greenness
- 6) High value in brightness and contiguity (large areas)

The value of the BARE_F will represent how well these criteria are satisfied.

Next in the classification are the rules which decide vegetation classes. The first category is growing fields. The growing fields class will contain pixels which meet all or some of the following criteria:

- 1) Less than 1.5 kilometers from existing transportation
- 2) Slope less than 10 percent
- 3) Vegetation mask (3)
- 4) Very high brightness value
- 5) Very high greenness value
- 6) Low wetness value

The growing variable is assigned a value of 10 if all criteria are met.

Next in the vegetation class is the general hardwood category. This category considers appropriate data from the soils GIS layer as well as the other variables used before. For this case a detailed soils series map was available with some information as to whether the soil would support various tree types. The criteria for general hardwood are as follows:

- 1) Appropriate soil for hardwoods
- 2) Soil-like general classification
- 3) High brightness
- 4) Low wetness

The hardwood variable is assigned values depending on the criteria satisfaction. A more detailed hardwood classification may be developed by following slope and hydrology information. Upland hardwoods should be able to be distinguished from lowland hardwoods.

Pine forest classification is also weighted by the soil variable. The criteria for pine are:

- 1) Pine supporting soil
- 2) Vegetation general classification
- 3) Greater value in wetness than in greenness
- 4) Medium value in brightness

The last category considered is the wetland class. For the initial test site, this classification was based almost entirely on soils (poorly drained) data.

At the very bottom of the GISMO file is the decision rule for determining the unique class to be assigned to the candidate pixel. A search is made through the class variables to find the class with the highest certainty. If several class variables have the same largest certainty, the class is assigned to the last category in the decision tree. The ordering of the classes in the decision rule should represent the best estimate of the likelihood of the class occurring in the scene. If desired, each of the class variables could have been saved to a file and a more complex logic could have been used later to decide on class conflicts. While this approach would allow the user to look at how probable the classification scheme considered each class, the disk storage needed might be prohibitive.

Figure 10 shows the results of the DFAD PLUS classification for the Spring Lanier data set. At top left is an image classification based on interpretation of ISODATA clustering results. At the top right is the DFAD PLUS classification without the DFAD GIS layer. The bottom right image shows the adjusted DFAD PLUS classification using the DFAD GIS layer. Figure 11 shows the confidence factors attributed to the classification. Areas in green demonstrate high confidence regions while red areas represent only moderate confidence levels.

Classification rules were developed for each of the other two test sites, and finally, all three classifications were combined to prove a general classifier that executed certain sets of rules based on the time of the year, geographic location, and type of geomorphology. The main difference between the winter and the spring test sets (Appendix B) for the Lake Lanier area involved the rules for hardwood, pine, and growing fields. Most of the other rules used values determined from the image data sets statistics so that new discrete rules were not necessary.

Figure 12 shows the results of the DFAD PLUS classification for the winter Lanier data set. This figure also shows results for clustering DFAD PLUS without DFAD data, and DFAD PLUS using DFAD data. Figure 13 shows the confidence levels of the DFAD PLUS classification using DFAD data. Green shows high confidence and red is moderate confidence.

The classification rules for the Denver test site (Appendix C) were somewhat different from those used in the Lake Lanier area. This is not surprising since Denver is in a totally different geologic and geomorphic region. In addition the climate for the West is substantially different than that for the southeast. New classes such as rangeland, snow, and high plains vegetation had to be incorporated into the Denver model. The structure of the GSIMO model remained relatively constant, however, indicating that models might be developed for geomorphic regions and ecoregions that might be able to be applied on a wider basis. By using logic switches within GISMO, a library of models could be build and updated that would allow general application with only minimal information and knowledge required from the user. Figure 14 shows the DFAD PLUS classification for the Denver Test site.

VII. ABEST Expert System Implementation

A. Knowledge Acquisition

Gilmore Aerospace consulted with engineers from ERDAS in quantifying rules used by human experts to classify image data. Initial efforts produced a hierarchical decision tree for classifying coarse image features. This decision tree was incrementally refined to include more and more detailed classifications. For example, the initial high-level objects consisted of coarse groupings such as water, vegetation, and urban areas. Successive refinements and additions of the decision tree added finer grained classifications. For example, vegetation was split into grass, pine, hardwood, fields, etc. Urban areas were refined into high density and low density urban areas. Additional knowledge acquisition was obtained through feedback from ERDAS engineers who were hand analyzing the image data sets and developing the GISMO rule system.

B. Ada Package Development

In order to encode the knowledge acquired in the Knowledge Acquisition phase, a number of utilities were implemented. These utilities include Ada packages for reading and writing ERDAS image and data files, examining input files and result files, and the main program to run the compiled rules generated by RULECOMP. These capabilities enabled the knowledge base system to access satellite image, feature data, statistical image information, and generate result files in standard ERDAS image format for convenient display.

ERDAS_IMAGE is an Ada package that provides ABEST with the capability to read, write, and create ERDAS image files and to modify individual pixels in them. It contains data types for the various aspects of ERDAS images as well as functions and procedures to manipulate them.

The DFAD PLUS program runs the compiled rules generated by the RULECOMP rule compiler. These rules are identical to the rules used in the interpreted ABEST development system, but are compiled into Ada procedures. These procedures are included in the DFAD PLUS via a 'separate' clause, allowing separate compilation of the main program and the rules. DFAD PLUS also contains adjunct procedures to initialize, read, and write images. DFAD PLUS loops over each of the pixels in the input images, executes the rules in order to generate a classification, then writes out the classification results.

An auxiliary program called WHY was developed to provide an explanation capability to the compiled rule generated results. WHY takes the same input data as DFAD PLUS and analyzes a single pixel and allows the user to browse the justifications for that pixel's classification. WHY runs a functional front-end to ABEST and thus provides the same explanation capability without the full fledged window development environment.

XPIX (eXamine PIXEL) is a utility that allows a developer to simultaneously display all input (spectral, hydrology, transportation, soils, tasseled cap transformation, etc.) values as well as any output classification and confidence values generated by the DFAD PLUS system.

C. Rule Development, Testing, and Generalization

The decision tree produced during the Knowledge Acquisition phase was implemented as a set of rules for ABEST. Initial efforts centered on achieving correct coarse classifications. Once acceptable results were attained, more specific classifications were attempted. In consultation with ERDAS, Gilmore Aerospace experimented with additional various input data sources. The knowledge base development closely paralleled the GISMO rule development as described in Section V. As refinements were made to the GISMO rules, additional insights were gained which allowed enhancements to be made to the ABEST knowledge source. By the end of the project, the use of tasseled cap transformation data significantly improved classification results.

Input to the DFAD PLUS program consists of the ABEST knowledge base (rules, facts, and frames), spectral image data, feature image data (proximity to transportation, proximity to hydrology, previous DFAD classification, etc.), and image condition data (season, general geographic area, image resolution). Output of the program consists of primary and secondary classification for each pixel, the confidence values for each classification, and ambiguity analysis.

In DFAD PLUS each image pixel's spectral and feature data was stored in a frame. A second frame was used to store deductions and supporting evidence for various classifications. Due to image size (512 x 512 pixels), the image was analyzed in small sections without losing neighboring image pixels' classification influence.

Initial knowledge bases centered around the Winter Lake Lanier data set. Based upon the evaluation of intermediate DFAD PLUS systems, new rules were created and implemented for both Denver (spring) and Lanier (spring and winter).

Two distinct rule sets were developed, one utilizing existing DFAD data and one that did not. This allowed testing the hypotheses that the use of DFAD data would provide information to allow better classification in cases where [a] the rules generated multiple classifications, or [b] did not provide sufficient certainty of classification. This indeed turned out to be the case. The inclusion of preexisting DFAD classifications improved DFAD PLUS classification.

While the GISMO rules use hard coded numbers for the tassle cap transformation (brightness, greenness, and wetness) statistics (means and standard deviation), the ABEST rules use information read from the ERDAS statistical (".sta") data files to allow a single rule set to be able to intelligently operate on different seasonal parameters for mean and standard deviation.

Comparison of ABEST rule results with the ground truth file indicated excellent classification results for some categories (90%+ for water, 90%+ for fields, 99% for some types of forests) and poorer results for some categories (50-80% for urban) and faulty results for wetlands. For example, it appeared that any rules checking for high 'wetness' caused classification problems. In order to analyze the dependency of certain classifications on particular input data, Gilmore Aerospace designed a program, STAT, to analyze the correlation between pixel classification and input data type (e.g., the dependency of PINE classification on TM NEAR-IR value). The results of this analysis led to enhancement of the knowledge base to reduce wetland and urban classifications and results improved.

The Lanier and Denver data sets are imaged at different resolutions. This required the rules in the knowledge base which referred to proximity to be generalized based on the global image condition data.

D. Rule Compiler

Due to the large number of pixels in an image, the full analysis of an entire image required several hours of processing time. ABEST provides powerful functionality (trace history, explanations, justifications, etc.) that assists the developer during the system development phase. These features, however, are not without cost in terms of execution time. In order to provide quick batch classification, Gilmore Aerospace scientists analyzed the specific requirements of the DFAD PLUS knowledge base and developed a rule compiler. RULECOMP (RULE COMPiler) implements a subset of ABEST functionality by translating knowledge bases into compiled Ada code. During the development phase, rules are loaded into ABEST and interpreted. The ABEST framework computes explanation and justifications for each decision made. In addition, tracing information is available. This provides enormous amounts of information which can be used to determine how decisions were made. Once rules have been developed and debugged, the knowledge base can be translated by RULECOMP into an efficient Ada program to quickly classify a

large image. The generated program can then be used on any user specified image data sets.

E. Machine Implementation Issues

Initial versions of DFAD PLUS were developed on an IBM PC-AT with 4 Mbytes of memory using the Alsys Ada compiler. Later versions were implemented on a Sun 386i machine (80386 processor) running UNIX with the Alsys Ada compiler. The current version is implemented on a Sun 4 (SPARC architecture) compiled with the Verdex Ada system. Switching from the Sun 386i to the Sun 4 also involved switching from the ERDAS 7.3 system to the ERDAS 7.4 system. This entailed rewriting portions of the image support packages for compatibility with the new ERDAS image file formats.

VIII. Conclusions

ERDAS and Gilmore Aerospace have developed a prototype system that would allow multisource information to be included in a detailed rule based classification of multispectral satellite information. Decision rules were developed for two sites in two different geomorphic regions. Two seasons of imagery data were processed for the east coast test site as a third test case. Rules were developed based on experience in classification of imagery data and its merger into GIS systems. Rules were developed interactively using the ERDAS GISMO program. Once the rules were tried and tested, they were passed on to the ABEST system where they were implemented directly and improved upon. The classification system provided good results versus unsupervised and supervised direct classification of TM data. The classification system provided results more consistent with requirements for DFAD data categories. While intensive interactive control is necessary for initial rule development, it seems that new rules for differing regions may be devised as simple modifications of an existing rule library.

One of the requirements for this study was the acceptance of DFAD vector data into the ERDAS Image Processing and GIS system. This capability was provided in the ERDAS Data Conversion Module in which conversion of DFAD data was accommodated along with vector data from a number of other sources.

For areas with existing DFAD data, the classification system may be modified to focus on extracting detailed land cover information within previously defined DFAD class

areas. If no DFAD data are available, the classification system can provide a DFAD type classification by multi-source data analysis. New regions with differing climates and geomorphologies, however, will require user experience for rule modification.

This type of classification system may be very useful for determining change that occurs over time, since the same rules may be applied to a new scene at approximately the same season with little or no modification.

References:

- 1) Defense Mapping Agency Product Specification for Digital Feature Analysis Data (DFAD), Level I and Level II.
- 2) Crist and Kauth, 1986, "The Tasseled Cap De-Mystified", Photogrammetric Engineering and Remote Sensing, Vol 52, pp 81-86.

IX. Appendix A

```
# DFAD_PLUS classifier - WINTER LANIER
```

```
# This rule set uses image statistics and a high-level mask  
# for classification. NO use of COVER GIS
```

```
data
```

```
# Imagery data - Tassle Cap transformation of TM data
```

```
input bright file "bright";  
input green file "green";  
input wetness file "wetness";
```

```
# Proximity data - using ERDAS SEARCH program
```

```
input prxtran file "../proxtrans";  
input prxhydr file "../proxhydro";
```

```
# Other data
```

```
input soil file "../tdsoils5";  
input slope file "../slope";
```

```
# Data derived from tassle cap data
```

```
input cont_bri file "contig-bri";
```

```
# Output DFAD_PLUS classification and certainty files
```

```
output dfad file "dfad2new";  
output cert file "cert2new";
```

```
# Program variables
```

```
integer bval;  
integer gval;  
integer wval;  
integer tval;  
integer hval;  
integer soiv;  
integer slov;  
integer cval;  
integer mval;  
integer mask;  
integer total;
```

```
# DFAD class DFAD code  
# -----  
integer water; # WATER 1  
integer urban; # GENERAL_URBAN 2  
integer hdu; # HIGH_DENSITY_URBAN 3  
integer ldu; # LOW_DENSITY_URBAN 4  
integer bare_u; # BARE_URBAN 5  
integer bare_f; # BARE_FIELD 8  
integer growing; # GROWING_FIELD 9  
integer hardwood; # GENERIC_HARDWOOD 12  
integer pine; # PINE 13  
integer wetland; # WETLAND 16
```

```
start
```

```
# Generate high-level mask  
# Separate image into water, soil-like, and vegetation
```

```
mask = conditional {  
# * Water *  
# High wetness (mean + 1sigma, max)  
# Wet > green  
# Very low bright (min, mean - 1sigma)  
(wetness >= 6 and wetness > green and bright <= 78) 1  
  
# * Soil-like rule *  
# High bright (mean, max)
```



```

# Lower green (min, mean + 1sigma)
# Bright > green
(bright >= 102 and green <= 4 and bright > green) 2

# * Vegetation rule *
# High green (mean - 1sigma, max)
# Everything else that is not classified
(green >= 0) 3

(default) 0 );

#----- WATER -----
hval = either -4 if (prxhydr > 2) or 0 otherwise;
mval = either 10 if (mask eq 1) or 0 otherwise;

water = mval + hval;

#----- GENERAL_URBAN -----
tval = either 1 if (prxtran < 40) or 0 otherwise;
slov = either 1 if (slope <= 20) or 0 otherwise;
mval = either 1 if (mask eq 2) or 0 otherwise;

# Low green (min, mean)
gval = either 1 if (green <= 2) or 0 otherwise;

# Low wetness (min, mean)
wval = either 1 if (wetness <= 2) or 0 otherwise;

total = tval + slov + wval + gval + mval;
urban = (total / 5.0) * 10.0 + 0.5;

#----- HIGH_DENSITY_URBAN -----
tval = either 1 if (prxtran < 10) or 0 otherwise;
slov = either 1 if (slope <= 10) or 0 otherwise;

# Very high bright (mean + 2sigma, mean + 4sigma)
bval = either 1 if (bright >= 152 and bright <= 200) or 0 otherwise;

# Very low green (min, mean - 1sigma)
gval = either 1 if (green eq 0) or 0 otherwise;

hdu = total + tval + bval + slov + gval;
hdu = (hdu / 9.0) * 10.0 + 0.5;

#----- LOW_DENSITY_URBAN -----
# Medium bright (mean, mean + 1sigma)
bval = either 1 if (bright >= 102 and bright <= 127) or 0 otherwise;

# Low-medium wetness (mean - 1sigma, mean)
wval = either 1 if (wetness <= 2) or 0 otherwise;

ldu = total + bval + wval;
ldu = (ldu / 7.0) * 10.0 + 0.5;

#----- BARE_URBAN -----
# Very high bright (mean + 4sigma, max)
bval = either 1 if (bright >= 200) or 0 otherwise;

# Very low green (min, mean - 2sigma)
gval = either 1 if (green <= 3) or 0 otherwise;

bare_u = total + bval + gval;
bare_u = (bare_u / 7.0) * 10.0 + 0.5;

#----- BARE_FIELD -----

```

```

# This class contains fields which have low % vegetative cover

tval = either 1 if (prxtran <= 50 and prxtran > 0) or 0 otherwise;
slov = either 1 if (slope <= 10) or 0 otherwise;
mval = either 1 if (mask eq 2) or 0 otherwise;

# Very low wetness (min, mean - 2sigma)
wval = either 1 if (wetness eq 0) or 0 otherwise;

# Low green (min, mean)
gval = either 1 if (green <= 2) or 0 otherwise;

# WINTER - high brightness (mean + 1sigma, mean + 2sigma) - contiguous
bval = either 1 if (cont_bri >= 6) or 0 otherwise;

bare_f = tval + slov + mval + bval + gval + wval;
bare_f = (bare_f / 6.0) * 10.0 + 0.5;

#----- GROWING_FIELD -----
# This category will contain any type of vegetation which is in the
# process of "greening up"

tval = either 1 if (prxtran <= 50 and prxtran > 0) or 0 otherwise;
slov = either 1 if (slope <= 10) or 0 otherwise;
mval = either 1 if (mask eq 3) or 0 otherwise;

# Very high brightness (mean + 1sigma, max)
bval = either 1 if (bright >= 127) or 0 otherwise;

# Very high greenness (mean + 1sigma, max)
gval = either 1 if (green >= 4) or 0 otherwise;

# Low wetness (min, mean)
wval = either 1 if (wetness <= 2) or 0 otherwise;

growing = tval + slov + bval + gval + mval + wval;
growing = (growing / 6.0) * 10.0 + 0.5;

#----- GENERIC_HARDWOOD -----
# Will have different rules depending on season

# Must be a hardwood-supporting soil
soiv = conditional {
    (soil eq 1)                1
    (soil >= 5 and soil <= 18) 1
    (soil >= 21 and soil <= 23) 1
    (soil >= 26 and soil <= 30) 1
    (soil eq 33)              1
    (default)                  0 };

mval = either 1 if (mask eq 3) or 0 otherwise;

# High bright (min, mean + 1sigma)
bval = either 1 if (bright <= 127) or 0 otherwise;

# SPRING rule
# High green (mean + 1sigma, max)
#gval = either 1 if (green >= 4) or 0 otherwise;

# WINTER rule
# Low wetness (min, mean + 1sigma)
wval = either 1 if (wetness <= 4) or 0 otherwise;

hardwood = soiv + mval + bval + wval;
hardwood = (hardwood / 4.0) * 10.0 + 0.5;

```

```

#----- PINE -----
# This rule attempts to pick out coniferous stands; characterized
# chiefly by lower brightness and greenness values than deciduous

# Must be pine-supporting soil
soiv = conditional {
    (soil >= 1 and soil <= 4) 1
    (soil >= 7 and soil <= 10) 1
    (soil >= 13 and soil <= 18) 1
    (soil >= 24 and soil <= 30) 1
    (soil >= 32 and soil <= 33) 1
    (default) 0 };

mval = either 1 if (mask eq 3) or 0 otherwise;

# Lower green
gval = either 1 if (wetness > green) or 0 otherwise;

# Lower bright (min, mean)
bval = either 1 if (bright <= 102) or 0 otherwise;

# High wetness (mean, max)
wval = either 1 if (wetness >= 2) or 0 otherwise;

pine = soiv + mval + gval + bval + wval;
pine = (pine / 5.0) * 10.0 + 0.5;

#----- WETLAND -----
# Bottomland soils
soiv = conditional {
    (soil eq 1) 1
    (soil eq 5) 1
    (soil eq 11) 1
    (soil eq 12) 1
    (soil eq 32) 1
    (default) 0 };

#-----
# Leave out wetland

cert = max(urban, hdu, ldu, bare_u, bare_f, growing, hardwood, pine, water);

dfad = conditional {
    (cert eq water) 1
    (cert eq hardwood) 12
    (cert eq pine) 13
    (cert eq growing) 9
    (cert eq bare_f) 8
    (cert eq hdu) 3
    (cert eq ldu) 4
    (cert eq bare_u) 5
    (cert eq urban) 2 };

end

```

X . Appendix B

DFAD_PLUS classifier - SPRING LANIER

This rule set uses image statistics and a high-level mask
for classification. No COVER GIS

data

Imagery data - Tassle Cap transformation of TM data
input bright file "bright";
input green file "green";
input wetness file "wetness";

Proximity data - using ERDAS SEARCH program
input prxtran file "../proxtrans";
input prxhydr file "../proxhydro";

Other data
input soil file "../tdsoils5";
input slope file "../slope";

Data derived from tassle cap data
input cont_bri file "contig-bri";

Output DFAD_PLUS classification and certainty files
output dfad file "dfad2new";
output cert file "cert2new";

Program variables

integer bval;
integer gval;
integer wval;
integer tval;
integer hval;
integer soiv;
integer slov;
integer cval;
integer mval;
integer mask;
integer total;

#		DFAD class	DFAD code
#	-----	-----	-----
integer water;	#	WATER	1
integer urban;	#	GENERAL_URBAN	2
integer hdu;	#	HIGH_DENSITY_URBAN	3
integer ldu;	#	LOW_DENSITY_URBAN	4
integer bare_u;	#	BARE_URBAN	5
integer bare_f;	#	BARE_FIELD	8
integer growing;	#	GROWING_FIELD	9
integer grass;	#	GRASS_FIELD	10
integer hardwood;	#	GENERIC_HARDWOOD	12
integer pine;	#	PINE	13
integer wetland;	#	WETLAND	16

start

Generate high-level mask
Separate image into water, soil-like, and vegetation

mask = conditional {

* Water *
High wetness (mean + 1sigma, max)
Wet > green
Very low bright (min, mean - 1sigma)
(wetness >= 187 and wetness > green and bright <= 71) 1

* Soil-like rule *

```

# High bright (mean, max)
# Lower green (min, mean + 1sigma)
# Bright > green
(bright > 126 and green <= 190 and bright > green) 2

# * Vegetation rule *
# High green (mean - 1sigma, max)
# Everything else that is not classified
(green >= 65) 3

(default) 0 );

#----- WATER -----
hval = either -4 if (prxhydr > 2) or 0 otherwise;
mval = either 10 if (mask eq 1) or 0 otherwise;

water = mval + hval;

#----- GENERAL_URBAN -----
tval = either 1 if (prxtran < 40) or 0 otherwise;
mval = either 1 if (mask eq 2) or 0 otherwise;
slov = either 1 if (slope <= 20) or 0 otherwise;

# Low green (min, mean)
gval = either 1 if (green <= 128) or 0 otherwise;

# Low wetness (min, mean)
wval = either 1 if (wetness <= 130) or 0 otherwise;

total = tval + gval + wval + mval + slov;
urban = (total / 5.0) * 10.0 + 0.5;

#----- HIGH_DENSITY_URBAN -----
tval = either 1 if (prxtran < 10) or 0 otherwise;
slov = either 1 if (slope <= 10) or 0 otherwise;

# Very high bright (mean + 2sigma, mean + 4sigma)
bval = either 1 if (bright >= 235) or 0 otherwise;

# Very low green (min, mean - 1sigma)
gval = either 1 if (green <= 65) or 0 otherwise;

hdu = total + tval + bval + slov + gval;
hdu = (hdu / 9.0) * 10.0 + 0.5;

#----- LOW_DENSITY_URBAN -----
# Medium bright (mean, mean + 1sigma)
bval = either 1 if (bright >= 126 and bright <= 181) or 0 otherwise;

# Medium wetness (mean - 1sigma, mean)
wval = either 1 if (wetness >= 71 and wetness <= 130) or 0 otherwise;

ldu = total + bval + wval;
ldu = (ldu / 7.0) * 10.0 + 0.5;

#----- BARE_URBAN -----
# Very high bright (mean + 4sigma, max)
bval = either 1 if (bright >= 255) or 0 otherwise;

# Very low green (min, mean - 2sigma)
gval = either 1 if (green <= 3) or 0 otherwise;

bare_u = total + bval + gval;
bare_u = (bare_u / 7.0) * 10.0 + 0.5;

```

```

#----- BARE_FIELD (BARE SOIL, BARE GROUND) -----
# This class contains fields which have low % vegetative cover

tval = either 1 if (prxtran <= 50 and prxtran > 0) or 0 otherwise;
slov = either 1 if (slope <= 10) or 0 otherwise;
mval = either 1 if (mask eq 2) or 0 otherwise;

# Low green (min, mean)
gval = either 1 if (green <= 128) or 0 otherwise;

# Very low wetness (min, mean - 2sigma)
wval = either 1 if (wetness <= 14) or 0 otherwise;

bare_f = tval + slov + mval + gval + wval;
bare_f = (bare_f / 5.0) * 10.0 + 0.5;

# SPRING only
#----- GROWING_FIELD (DEVELOPING, "GREENING UP") -----
# New category for spring

tval = either 1 if (prxtran <= 50 and prxtran > 0) or 0 otherwise;
slov = either 1 if (slope <= 10) or 0 otherwise;

# Low wetness (min, mean - 1sigma)
wval = either 1 if (wetness <= 71) or 0 otherwise;

# Mid green (mean - 1sigma, mean)
gval = either 1 if (green >= 65 and green <= 128) or 0 otherwise;

# Soil-like (predominately soil)
# mval = either 1 if (mask eq 2) or 0 otherwise;

# Medium-high bright (mean, mean + 1sigma) - contiguous
bval = either 1 if (cont_bri >= 6) or 0 otherwise;

growing = tval + slov + wval + gval + bval;
growing = (growing / 5.0) * 10.0 + 0.5;

#----- GROWING_FIELD (GRASS, DEVELOPED) -----
# This category will contain any type of vegetation which is
# in full canopy

tval = either 1 if (prxtran <= 50 and prxtran > 0) or 0 otherwise;
slov = either 1 if (slope <= 10) or 0 otherwise;
mval = either 1 if (mask eq 3) or 0 otherwise;

# Very high brightness (mean + 1sigma, max)
bval = either 1 if (bright >= 181) or 0 otherwise;

# Very high green (mean + 1sigma, max)
gval = either 1 if (green >= 190) or 0 otherwise;

# Low wetness (min, mean)
wval = either 1 if (wetness <= 130) or 0 otherwise;

grass = bval + gval + wval + mval + tval + slov;
grass = (grass / 6.0) * 10.0 + 0.5;

#----- GENERIC_HARDWOOD -----
# Must be a hardwood-supporting soil
soiv = conditional {
    (soil eq 1) 1
    (soil >= 5 and soil <= 18) 1

```

```

(soil >= 21 and soil <= 23) 1
(soil >= 26 and soil <= 30) 1
(soil eq 33) 1
(default) 0 };

mval = either 1 if (mask eq 3) or 0 otherwise;

# High bright (mean, mean + 1sigma)
bval = either 1 if (bright <= 181) or 0 otherwise;

# SPRING rule
# High green (mean + 1sigma, max)
gval = either 1 if (green >= 190) or 0 otherwise;

hardwood = soiv + mval + gval + bval;
hardwood = (hardwood / 4.0) * 10.0 + 0.5;

#----- PINE -----
# This rule attempts to pick out coniferous stands; characterized
# chiefly by lower brightness and greenness values than deciduous
# (exception: not easily distinguished in winter); also has high wetness
# values compared to greenness and brightness, due to tree shadow

# Must be pine-supporting soil
soiv = conditional {
  (soil >= 1 and soil <= 4) 1
  (soil >= 7 and soil <= 10) 1
  (soil >= 13 and soil <= 18) 1
  (soil >= 24 and soil <= 30) 1
  (soil >= 32 and soil <= 33) 1
  (default) 0 };

mval = either 1 if (mask eq 3) or 0 otherwise;

# Lower green
gval = either 1 if (wetness > green) or 0 otherwise;

# Lower bright (min, mean)
bval = either 1 if (bright <= 126) or 0 otherwise;

# High wetness (mean, max)
wval = either 1 if (wetness >= 130) or 0 otherwise;

pine = soiv + mval + gval + bval + wval;
pine = (pine / 5.0) * 10.0 + 0.5;

#-----
# Leave out wetland

cert = max(urban,hdu,ldu,bare_u,bare_f,growing,grass,hardwood,pine,water);

dfad = conditional {
  (cert eq water) 1
  (cert eq hardwood) 12
  (cert eq pine) 13
  (cert eq grass) 10
  (cert eq growing) 9
  (cert eq bare_f) 8
  (cert eq hdu) 3
  (cert eq ldu) 4
  (cert eq bare_u) 5
  (cert eq urban) 2 };

end

```


XI. Appendix C

```
# DFAD_PLUS classifier - DENVER
```

```
# This rule set uses image statistics and a high-level mask  
# for classification. No COVER GIS
```

```
data
```

```
# Imagery data - Tassle Cap transformation of TM data
```

```
input bright file "bright";  
input green file "green";  
input wetness file "wetness";
```

```
# Proximity data - using ERDAS SEARCH program
```

```
input prxtran file "../proxtrans";  
input prxhydr file "../proxhydro";
```

```
# Other data
```

```
input soil file "../soils";  
input slope file "../slope";
```

```
# Data derived from tassle cap data
```

```
input cont_bri file "contig-bri";
```

```
# Output DFAD_PLUS classification and certainty files
```

```
output dfad file "dfad2new";  
output cert file "cert2new";
```

```
# Program variables
```

```
integer bval;  
integer gval;  
integer wval;  
integer tval;  
integer hval;  
integer soiv;  
integer slow;  
integer cval;  
integer mval;  
integer mask;  
integer total;
```

```
# ----- DFAD class DFAD code -----  
integer water; # WATER 1  
integer urban; # GENERAL_URBAN 2  
integer hdu; # HIGH_DENSITY_URBAN 3  
integer ldu; # LOW_DENSITY_URBAN 4  
integer bare_u; # BARE_URBAN 5  
integer bare_f; # PASTURE/BARE_FIELD 8  
integer growing; # GROWING_FIELD 9  
integer hardwood; # GENERIC_HARDWOOD 12  
integer pine; # PINE 13  
integer wetland; # WETLAND 16  
integer brush; # BRUSH/SHRUB 11  
integer shadow; # SHADOW 6
```

```
start
```

```
# Generate high-level mask  
# Separate image into water, soil-like, and vegetation
```

```
mask = conditional {  
# * Water *  
# High wetness (mean + 1sigma, max)  
# Wet > green  
# Very low bright (min, mean - 1sigma)  
(wetness >= 6 and wetness > green and bright <= 140) 1
```

```

# * Soil-like rule *
#   High bright (mean, max)
#   Lower green (min, mean + 1sigma)
#   Bright > green
#   (bright >= 182 and green <= 20 and bright > green) 2

# * Vegetation rule *
#   High green (mean - 1sigma, max)
#   Everything else that is not classified
#   (green >= 0) 3

#   (default) 0 );

#----- WATER -----
hval = either -2 if (prxhydr > 2) or 0 otherwise;
mval = either 10 if (mask eq 1) or 0 otherwise;

water = mval + hval + slov;

#----- GENERAL_URBAN -----
tval = either 1 if (prxtran < 40) or 0 otherwise;

#   Low green (min, mean)
gval = either 1 if (green <= 8) or 0 otherwise;

#   Soil-like
mval = either 1 if (mask eq 2) or 0 otherwise;

total = tval + gval + mval;
urban = (total / 3.0) * 10.0 + 0.5;

#----- HIGH_DENSITY_URBAN -----
tval = either 1 if (prxtran < 10) or 0 otherwise;
slov = either 1 if (slope <= 10) or 0 otherwise;

#   Very high bright (mean + 2sigma, mean + 4sigma)
bval = either 1 if (bright >= 254) or 0 otherwise;

hdu = total + tval + bval + slov;
hdu = (hdu / 6.0) * 10.0 + 0.5;

#----- LOW_DENSITY_URBAN -----
#   Medium bright (mean, mean + 1sigma)
bval = either 1 if (bright >= 102 and bright <= 223) or 0 otherwise;

slov = either 1 if (slope <= 20) or 0 otherwise;

ldu = total + bval + slov;
ldu = (ldu / 5.0) * 10.0 + 0.5;

#----- BARE_URBAN -----
#   Very high bright (mean + 4sigma, max)
bval = either 1 if (bright >= 254) or 0 otherwise;

#   Very low green (min, mean - 2sigma)
gval = either 1 if (green <= 0) or 0 otherwise;

bare_u = total + bval + gval;
bare_u = (bare_u / 5.0) * 10.0 + 0.5;

#----- PASTURE (BARE_FIELD) -----
#   This class contains fields which have low vegetative cover
#   no slope requirement

```

```

mval = either 1 if (mask eq 2) or 0 otherwise;

# High brightness (mean, mean + 1sigma)
bval = either 1 if (bright >= 182 and bright <= 223) or 0 otherwise;

# Wetness is zero
wval = either 1 if (wetness eq 0) or 0 otherwise;

# Low green (min, mean)
gval = either 1 if (green <= 2) or 0 otherwise;

# Not to be confused with roads
tval = either 1 if (prxtran > 1) or 0 otherwise;

bare_f = mval + gval + wval + bval + tval;
bare_f = (bare_f / 5.0) * 10.0 + 0.5;

#----- GROWING_FIELD -----
# This category will contain any type of vegetation which is in the
# process of "greening up" or is in full canopy

tval = either 1 if (prxtran <= 50) or 0 otherwise;
slov = either 1 if (slope <= 10) or 0 otherwise;

# Cannot be on soils unsuitable for agriculture; should be good soils
soiv = conditional {
  (soil eq 50) -1
  (soil eq 110) -1
  (soil eq 111) -1
  (soil eq 139) -1
  (soil eq 148) -1
  (soil eq 1) 1
  (soil eq 17) 1
  (soil eq 18) 1
  (soil eq 41) 1
  (soil eq 60) 1
  (soil eq 61) 1
  (soil eq 70) 1
  (soil eq 88) 1
  (soil eq 90) 1
  (soil eq 97) 1
  (soil eq 103) 1
  (default) 0 };

# High brightness (mean, max)
bval = either 1 if (bright >= 182) or 0 otherwise;

# Very high greenness (mean + 1sigma, max)
gval = either 1 if (green >= 20) or 0 otherwise;

# Vegetation
mval = either 1 if (mask eq 3) or 0 otherwise;

growing = tval + slov + bval + gval + mval + soiv;
growing = (growing / 6.0) * 10.0 + 0.5;

#----- BRUSH/SHRUB -----
# Not using mask

# Mid green (mean, mean + 1sigma)
gval = either 1 if (green >= 8 and green <= 20) or 0 otherwise;

# High bright (mean - 1sigma, max)
bval = either 1 if (bright >= 140) or 0 otherwise;

```

```
# Low wetness (min, mean)
wval = either 1 if (wetness <= 2) or 0 otherwise;
```

```
brush = gval + bval + wval;
brush = (brush / 3.0) * 10.0 + 0.5;
```

```
#----- PINE -----
# Changes a lot from Lanier;
```

```
# Must be pine-supporting soil
soiv = conditional {
    (soil eq 3) 1
    (soil eq 123) 1
    (soil eq 124) 1
    (soil eq 75) 1
    (default) -1 };
```

```
slov = either 1 if (slope < 60) or 0 otherwise;
```

```
# Vegetation
mval = either 1 if (mask eq 3) or 0 otherwise;
```

```
# Lower green (min, mean)
# DENVER - needs to be high (mean + 1sigma, max)
gval = either 1 if (green >= 20) or 0 otherwise;
```

```
# Lower bright (min, mean)
bval = either 1 if (bright <= 182) or 0 otherwise;
```

```
pine = soiv + slov + mval + gval + bval;
pine = (pine / 5.0) * 10.0 + 0.5;
```

```
#----- WETLAND -----
# Bottomland soils
```

```
soiv = conditional {
    (soil eq 1) 1
    (soil eq 50) 1
    (soil eq 60) 1
    (soil eq 61) 1
    (soil eq 88) 1
    (soil eq 90) 1
    (default) -1 };
```

```
slov = either 1 if (slope < 4) or 0 otherwise;
hval = either 1 if (prxhydr <= 10) or 0 otherwise;
```

```
# Vegetation
mval = either 1 if (mask eq 3) or 0 otherwise;
```

```
# Medium high wetness (mean, mean + 1sigma)
wval = either 1 if (wetness >= 2 and wetness <= 6) or 0 otherwise;
```

```
# Low brightness (min, mean)
bval = either 1 if (bright <= 182) or 0 otherwise;
```

```
wetland = soiv + slov + hval + mval + wval + bval;
wetland = (wetland / 6.0) * 10.0 + 0.5;
```

```
#----- SHADOW -----
```

```
# Low bright, green, wetness (min, mean)
bval = either 1 if (bright <= 182) or 0 otherwise;
gval = either 1 if (green <= 8) or 0 otherwise;
wval = either 1 if (wetness <= 2) or 0 otherwise;
```

```
# Usually on steep slopes
slov = either 2 if (slope >= 20) or 0 otherwise;
```

```
shadow = bval + gval + wval + slov;
shadow = (shadow / 5.0) * 10.0 + 0.5;
```

```
#-----
# Not using wetland
```

```
cert = max(urban, hdu, ldu, bare_u, bare_f, growing, brush, pine, water, shadow);
```

```
dfad = conditional {
  (cert eq water)      1
  (cert eq pine)      13
  (cert eq growing)   9
  (cert eq brush)     11
  (cert eq bare_f)    8
  (cert eq hdu)       3
  (cert eq ldu)       4
  (cert eq bare_u)    5
  (cert eq urban)     2
  (cert eq shadow)    6 };
```

```
end
```

XII. Illustrations

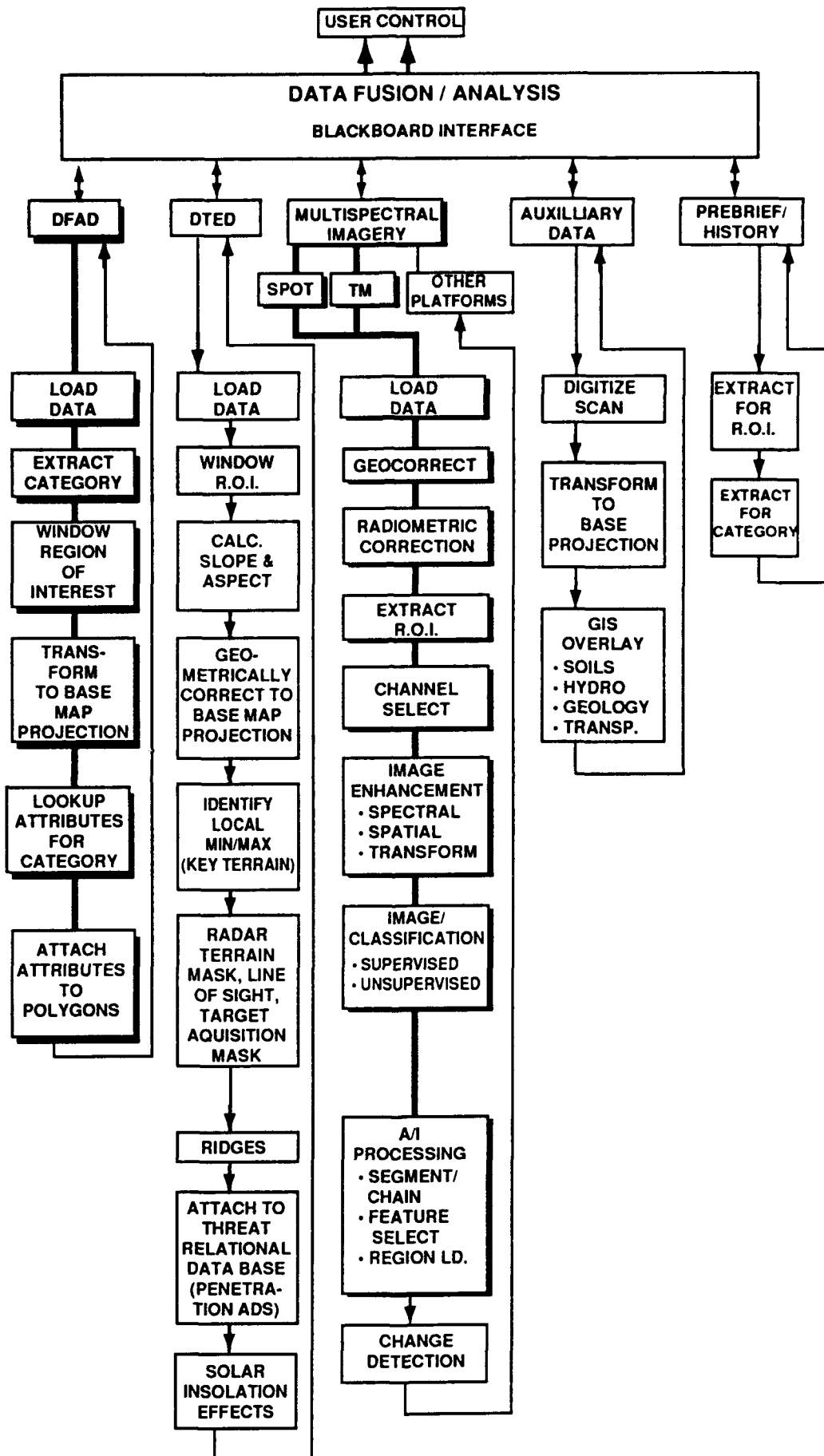


Figure 1 - Expert System Flow

DFAD TO GIS PROCESS

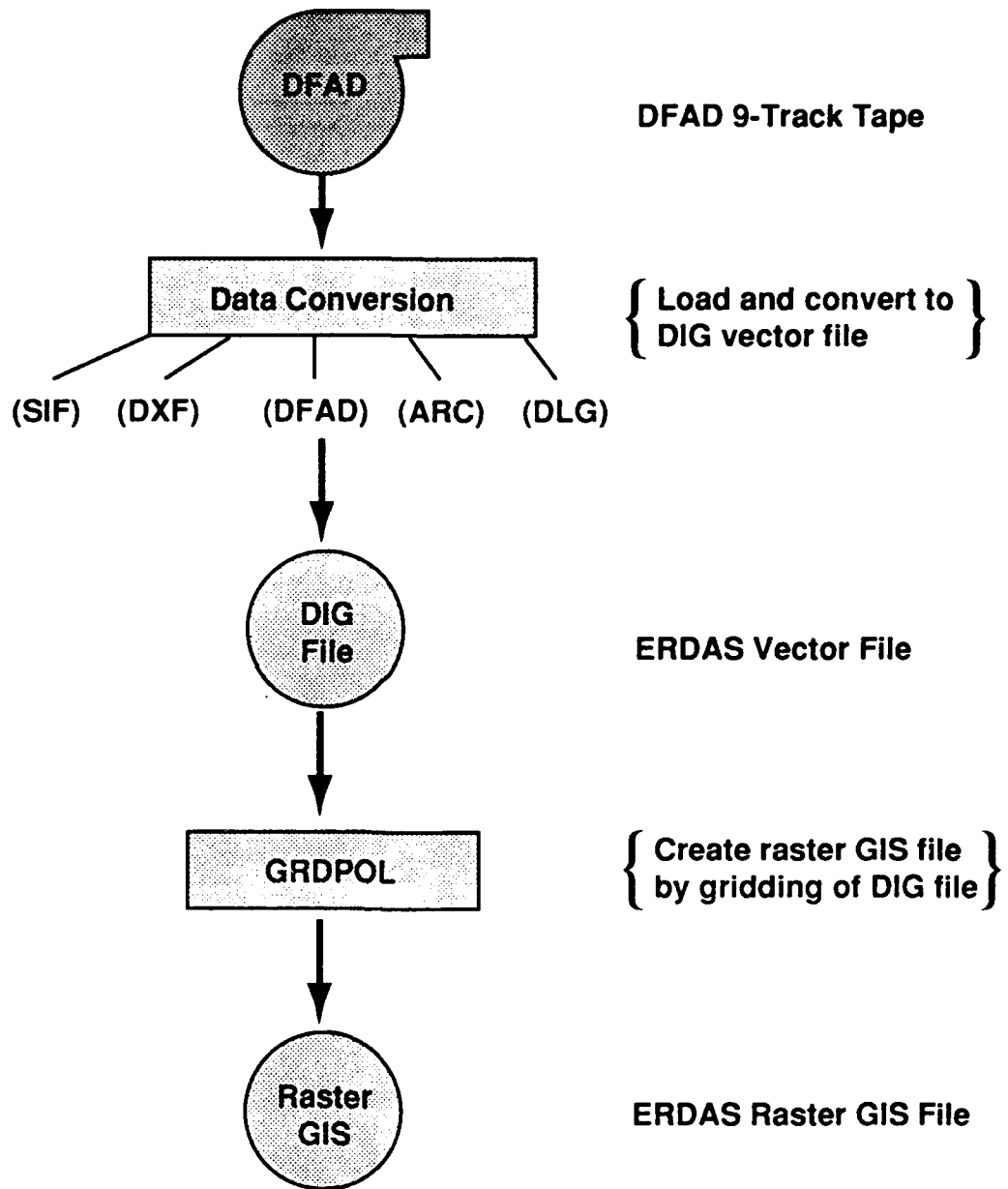


Figure 2 - DFAD Loading Process



Merged SPOT XS and PAN

Figure 3
SPOT XS and PAN merge for Denver
Test Site

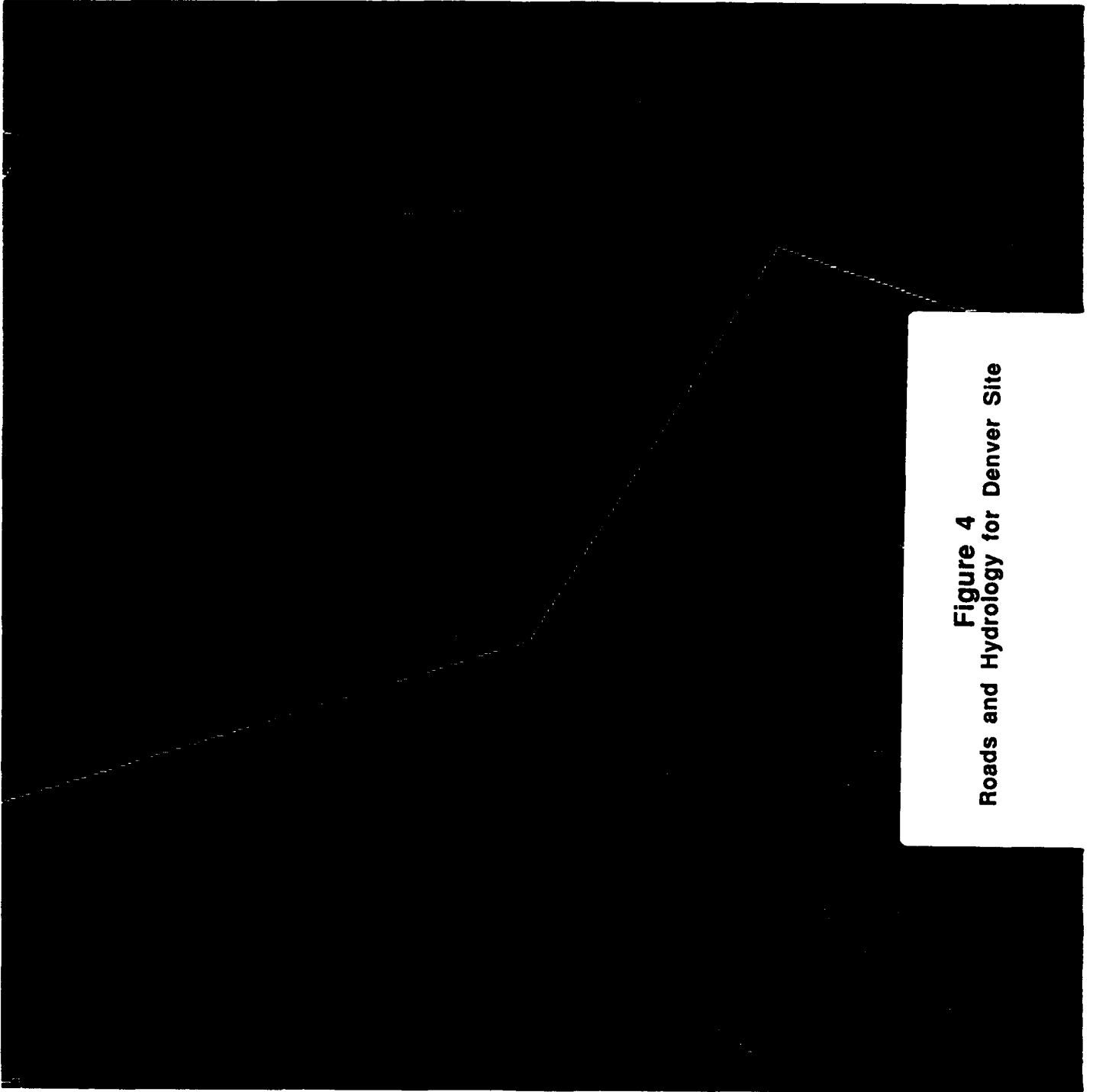


Figure 4
Roads and Hydrology for Denver Site

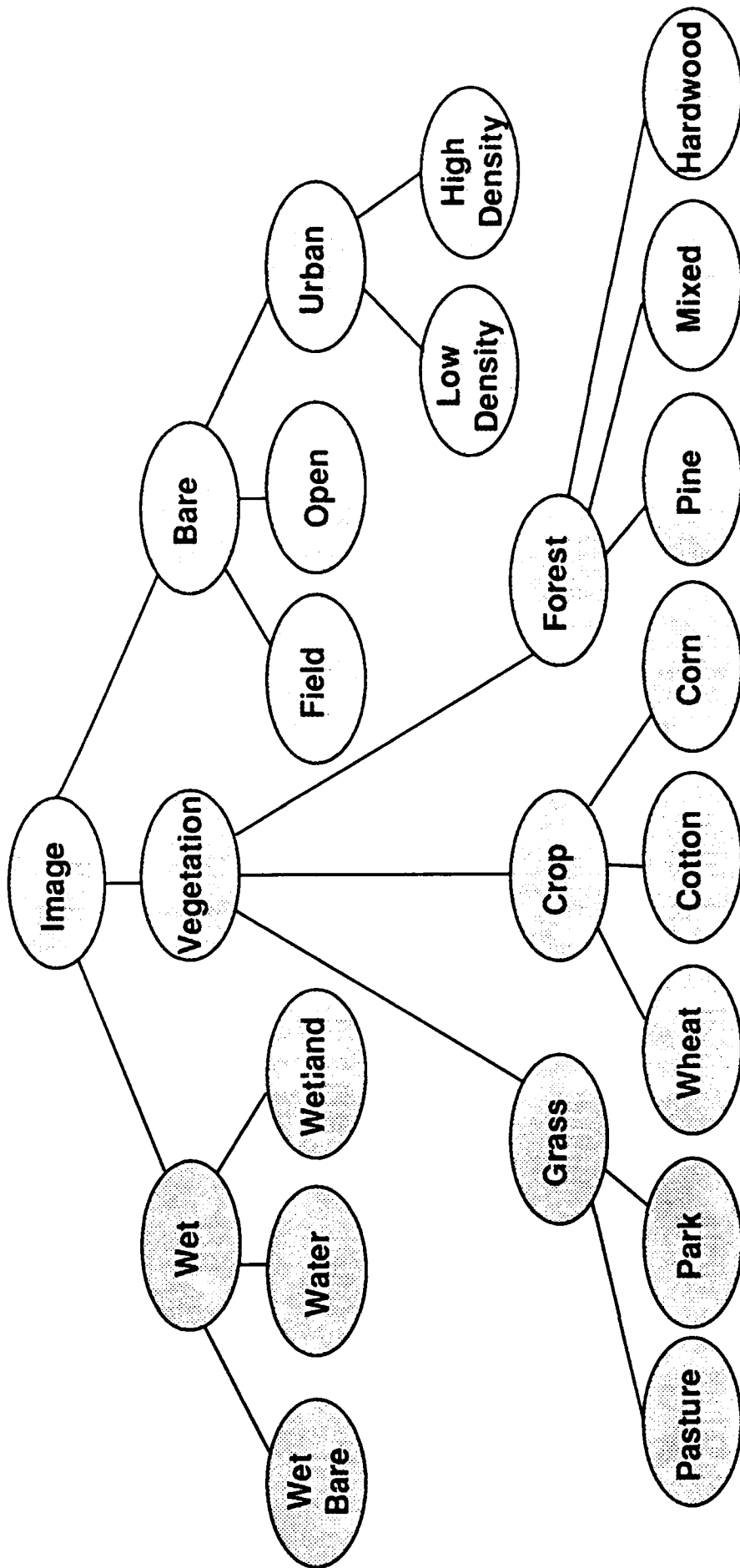


Figure 5 - Decision Tree

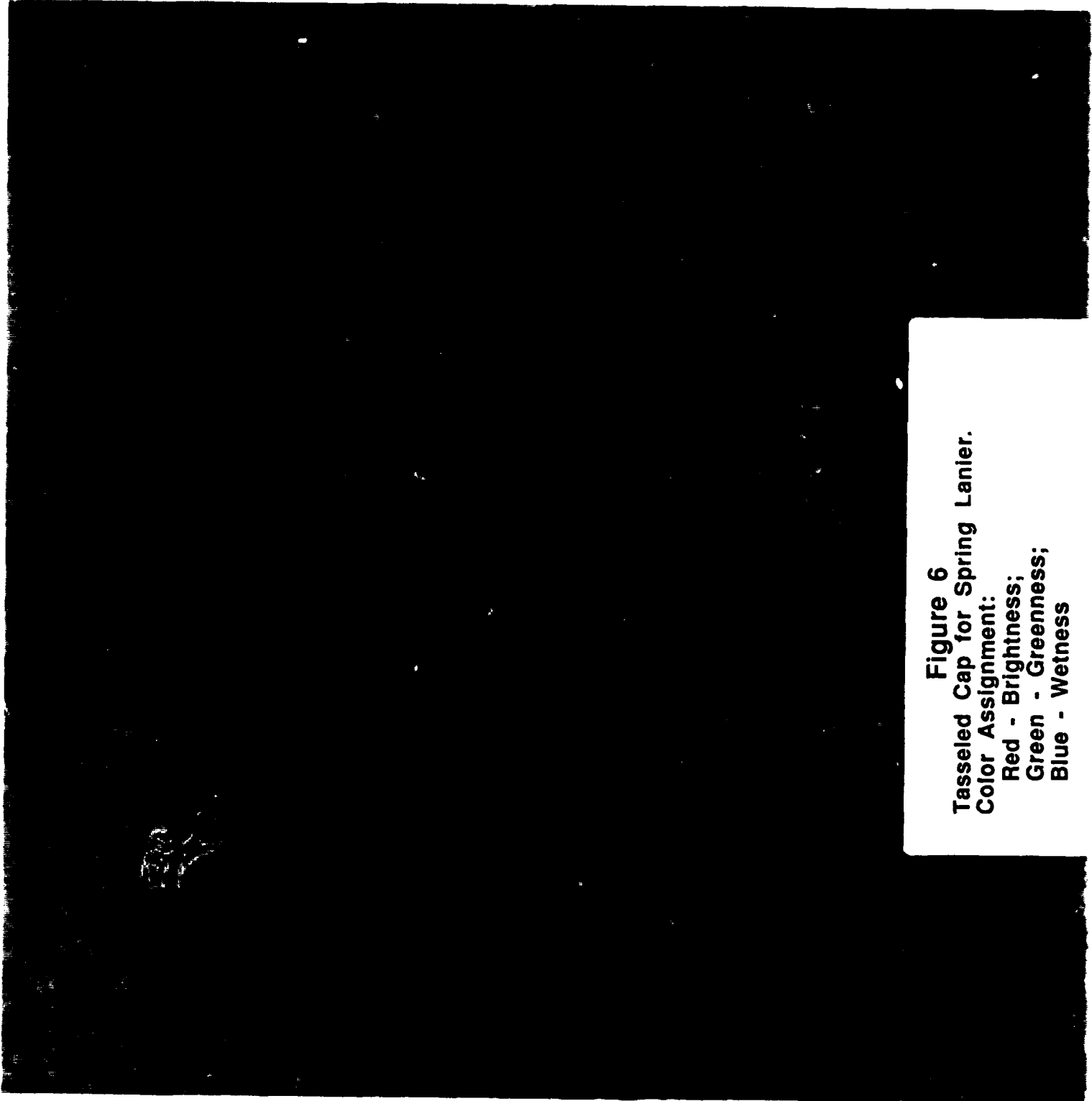


Figure 6
Tasseled Cap for Spring Lanier.
Color Assignment:
Red - Brightness;
Green - Greenness;
Blue - Wetness

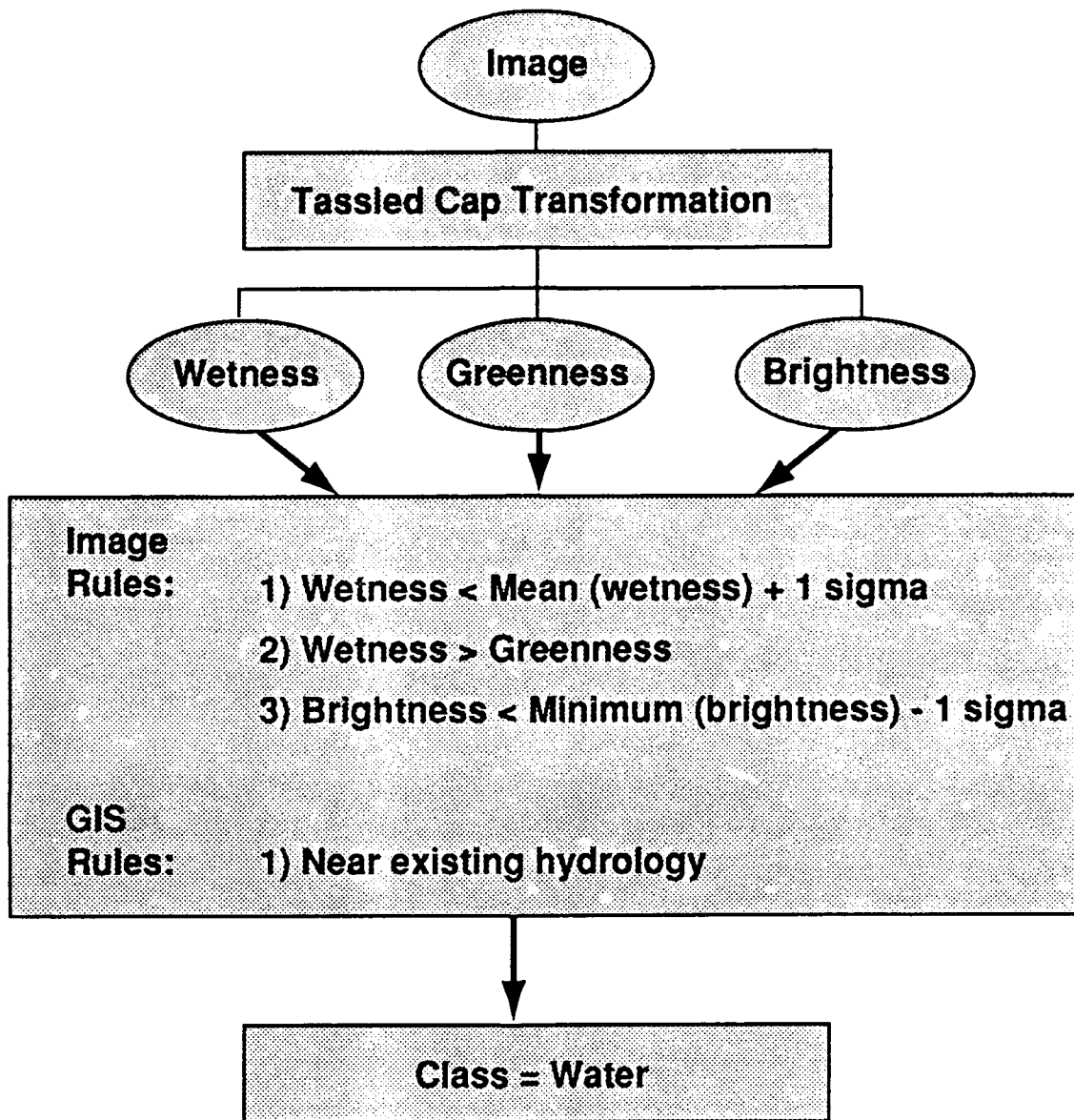


Figure 7 - Water Logic

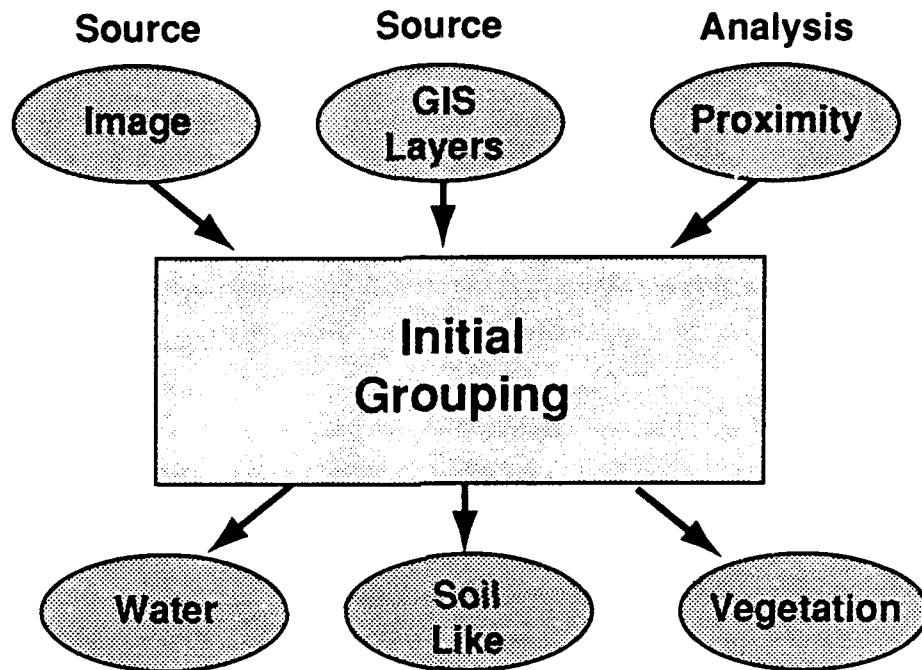


Figure 8 - Preliminary Classes

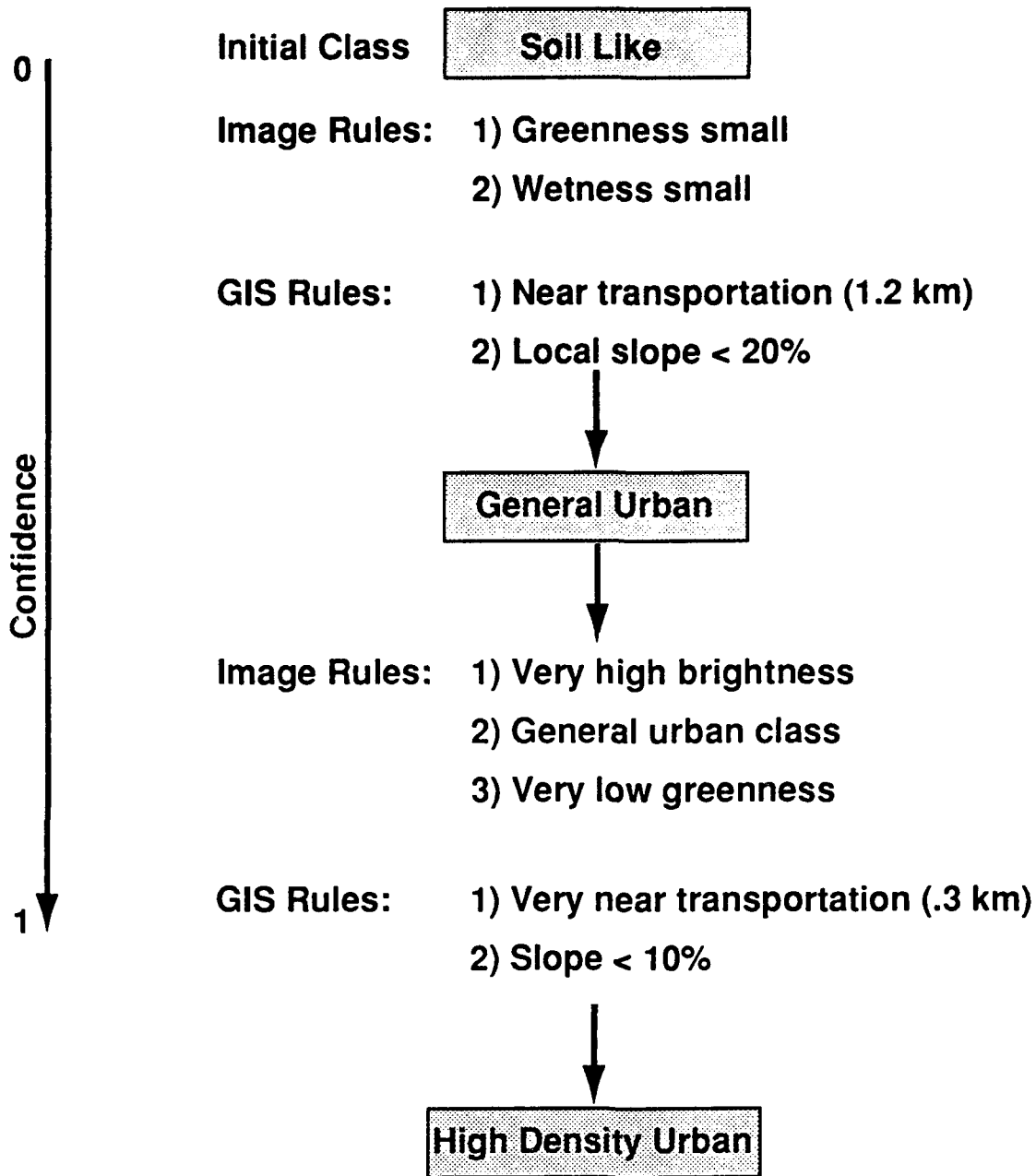


Figure 9 - High Density Urban Classification



Figure 10
 DFAD PLUS classification comparison
 for Spring Lanier

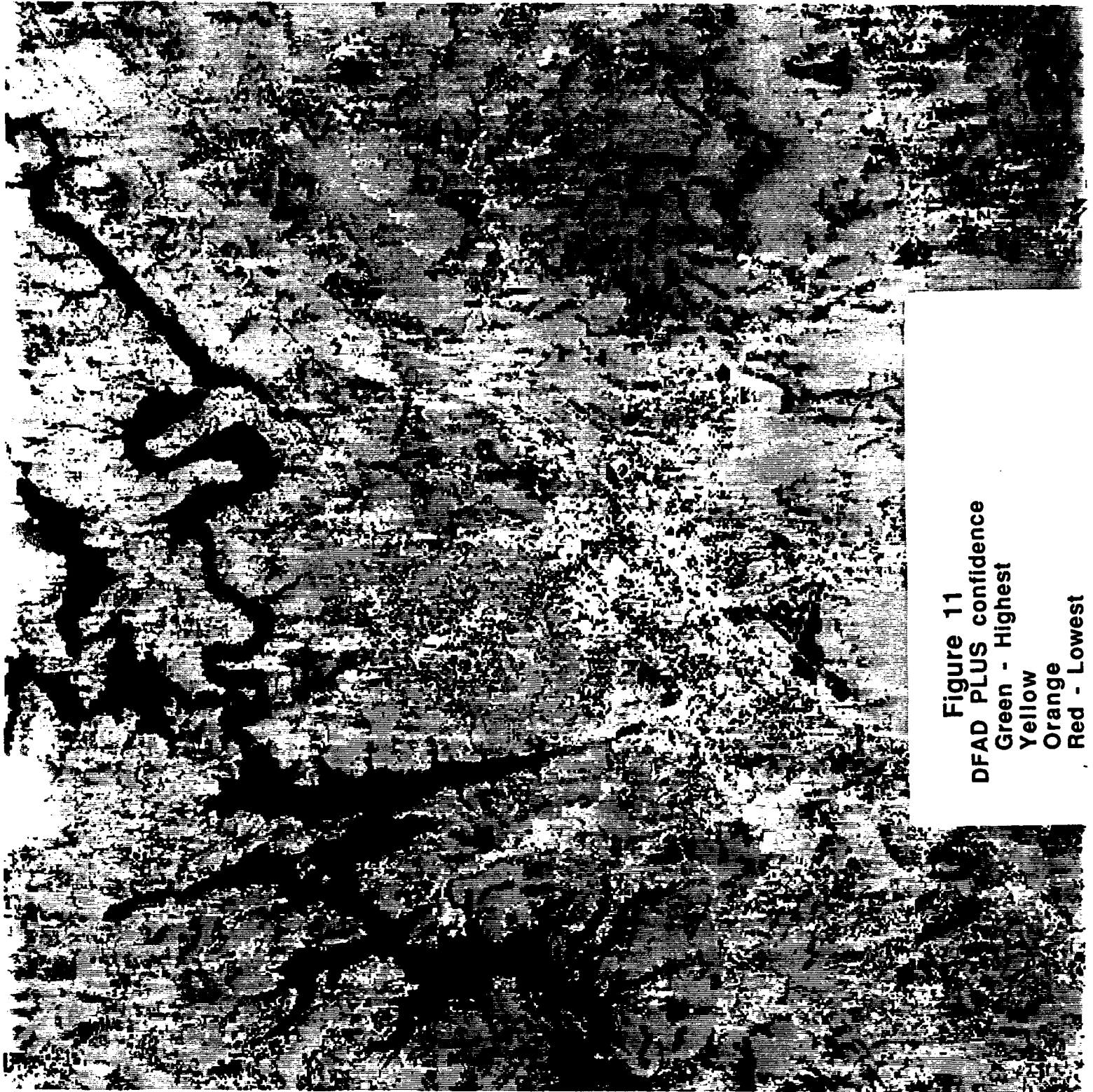


Figure 11
DFAD PLUS confidence
Green - Highest
Yellow
Orange
Red - Lowest



Figure 12
 DFAD PLUS classification comparison
 for Winter Lanier

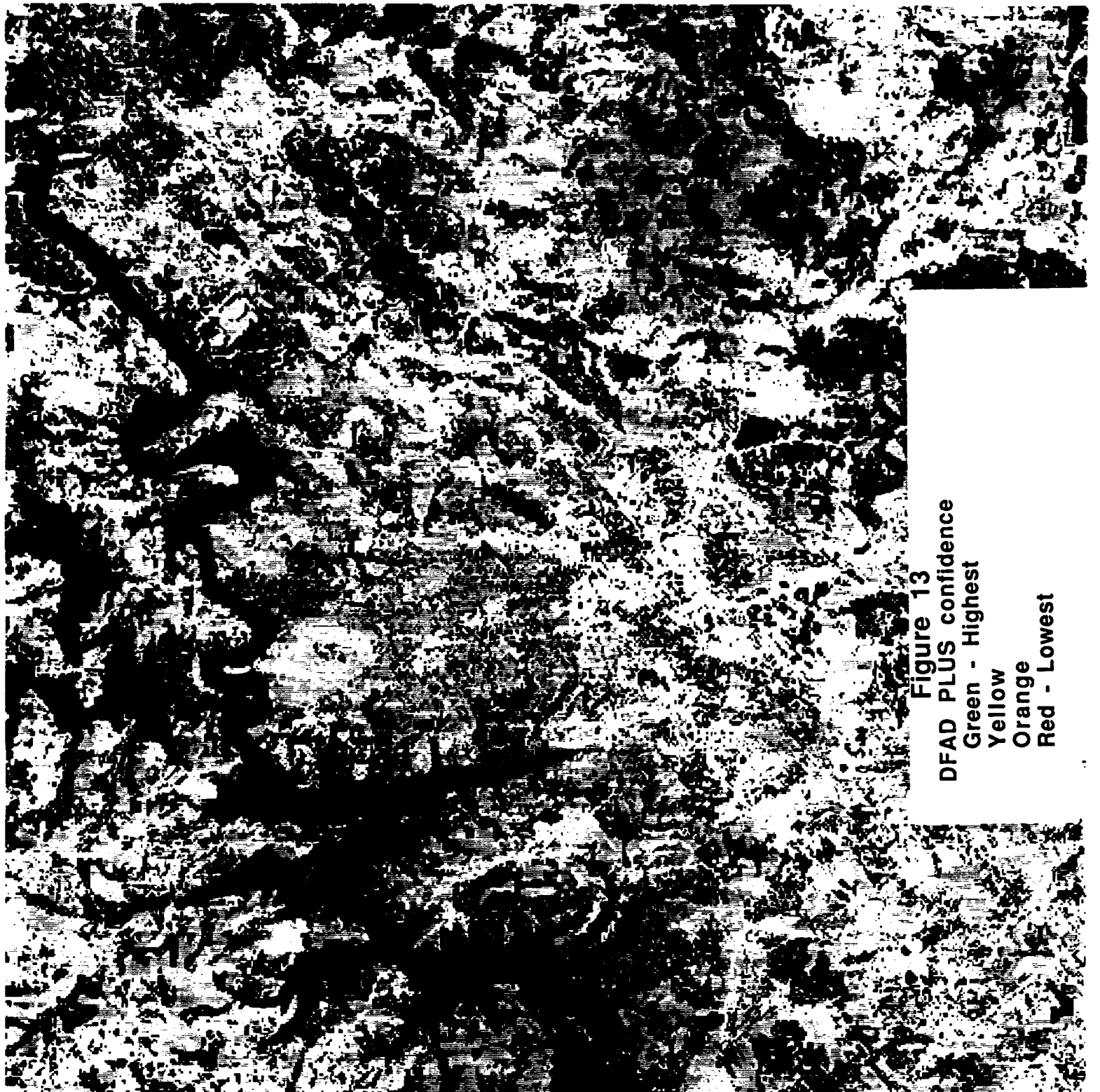


Figure 13
DFAD PLUS confidence
Green - Highest
Yellow
Orange
Red - Lowest



- 1 Water
- 2 General urban
- 3 High density urban
- 4 Low density urban
- 5 Bare urban
- 6 Shadow
- 7
- 8 Bare field
- 9 Growing field
- 10
- 11 Shrub sparse vegetation
- 12
- 13 Pines

Figure 14
 DFAD PLUS classification for
 Denver Test Site