LARS Internal Report 033082

COMTAL User's Note No. 4

Clustering and Calculation of Statistics

Shirley M. Davis
Philip H. Swain
Jie-Yong Juang

Laboratory for Applications of Remote Sensing
Purdue University West Lafayette, Indiana 47906 USA
1982

PREFACE

Under sponsorship of the COMTAL/3M Corporation, the Laboratory for Applications of Remote Sensing (LARS) implemented an interactive clustering algorithm for multispectral remote sensing data on the COMTAL Vision One. The processor, an ISODATA-type iterative clustering algorithm, allows the analyst to interact with the algorithm and the remote sensing data to achieve expediently the desired unsupervised classification results.

The documentation contained in this User´s Note is a slightly modified version of the documentation originally prepared as a part of LARS Contract Report 022882, *Remote Sensing Image Processing on the COMTAL Vision One/20* by P.H. Swain and S.M. Davis. Specifically, the following items are included:

- *User´s Guide to Cluster on the COMTAL Vision One*, starting on page 5, a user-oriented description of the processor
- *A Hands-on Experience with Cluster on the COMTAL Vision One*, starting on page 17, a self-taught hands-on tutorial module designed to demonstrate effective use of the Cluster processor with a sample multispectral data set.

Appendix A contains preliminary documentation for a program named CSTATS which provides the link between the results of the Cluster processing and further data processing using LARSYS. Based on the cluster map produced by Cluster and the corresponding multispectral image data, CSTATS computes and displays on the terminal the statistics (mean vector and covariance matrix) for each cluster class. On request it also produces a standard-format LARSYS statistics file (sometimes called a statistics "deck") which may be transmitted back to the IBM computer for input to the various LARSYS processors.

Page 2 is blank

TABLE OF CONTENTS

Page 4 is blank

USER´S GUIDE TO CLUSTER ON THE COMTAL VISION ONE

An essential aspect of any pattern-recognition based approach to classification (supervised or unsupervised) is the development of training statistics. These statistics characterize the training classes and enable the classifier to assign each point in the data to one of the classes.

The Cluster processor provides one method of determining the sets of multidimensional data vectors to be used to represent the training classes. When applied to the training samples (usually a subset of the data to be classified), this processor isolates spectrally similar "clusters" of data vectors from which the training statistics may be computed. When the training statistics are determined in this manner, based on spectral similarity, the training classes are often referred to as "spectral training classes" or simply "spectral classes."

Implemented on the COMTAL Vision One, the Cluster processor allows the data analyst to interactively develop training classes by applying an iterative clustering algorithm to multivariate image data. The area processed may consist of up to three channels of data (on a four-image-plane system), 512-by-512 pixels in extent. The results are stored in an image plane as a classification map (also called a cluster map) which may be viewed in black-and-white or pseudocolor, permitting the analyst to make a visual evaluation of the results. The classification map may also be transferred back to the host computer for further processing.

GENERAL DESCRIPTION OF THE CLUSTERING PROCESS

The first step in defining spectral training classes in a data set is to identify the natural spectral groupings of pixels in a sample of those data, that is, to find the clusters or groups of pixels that occur together in measurement space. Pixels derived from similar materials on the ground tend to group together in measurement space; for example, data vectors from clear water will usually group together in measurement space, as will those from turbid water, deciduous forest, urban areas, etc. The procedure for locating these clusters is implemented in the processor named "Cluster."

How can a computer find clusters in the training sample? Basically, the Cluster processor used a "guided trial-and-error" approach to assign the pixels in the image to disjoint classes. The objective is to make the assignment in such a way that pixels within any given class or "cluster" are as similar as possible while the pixels in different clusters are as different as possible. Many iterations of the assignment process are made and on each the objective is more closely met until on two successive iterations no change occurs, i.e., the process reaches "convergence."

The Cluster processor first requires the following inputs: the analyst specifies which pixels are to be submitted to the processor for clustering, how many clusters are desired, and how many iterations the processor should run through. The final input is several pixels (one per cluster), selected so that together they are representative of the spectral variations in the scene. This initializes the cluster centers.

The processing then follows a sequence of operations which is repeated until the stated maximum number of iterations or convergence is reached, whichever occurs first. The sequence, summarized in Figure CLU-1, can be described as follows:

1) the processor calculates the multidimensional distance between each pixel in the sample and each specified cluster center and assigns each pixel to the cluster whose center is nearest in measurement space;

2) for each cluster, the mean vector of measurements for all the pixels currently assigned to that cluster is calculated, and the mean vector becomes the new cluster center;

3) if all the new means are identical with the previous means (meaning that convergence has been reached), the Clustering operation halts; otherwise, the processor sets the cluster centers equal to the new means, checks to see if the maximum number of iterations has been reached and either stops or repeats steps 1) and 2).

This process of distance calculation, point assignment, and migration of the means can be demonstrated graphically (Figure CLU-2). The shaded area in Figure CLU-2(a) is the location in two-dimensional measurement space (coordinate axes x_1 and x_2) of the measurements in the training sample. The two crosses mark the initial locations of the cluster centers, with a separating boundary equidistant from the centers. Figure (b) shows the new locations for the cluster centers, each determined by calculating the mean of all the data points in the corresponding cluster. The boundary between the clusters again lies equidistant from the new centers. With the second iteration, the cluster centers again move to a new location in the measurement space, shown in Figure (c), and the boundary is re-positioned. The final iteration, Figure (d), shows no change in the location of the cluster centers from the previous iteration and no change in the position of the decision boundary. Convergence has been reached.

The distance measure used for determining cluster assignments is L_1 distance. In essence, L_1 distance is the sum of the individual component distances. As shown in Figure CLU-3, the L_1 distance between points A and B in two-dimensional feature space is calculated by determining the difference between a_1 and b_1, the difference between a_2 and b_2, and adding the two values. The procedure can easily be extended to accommodate higher-dimensional situations.

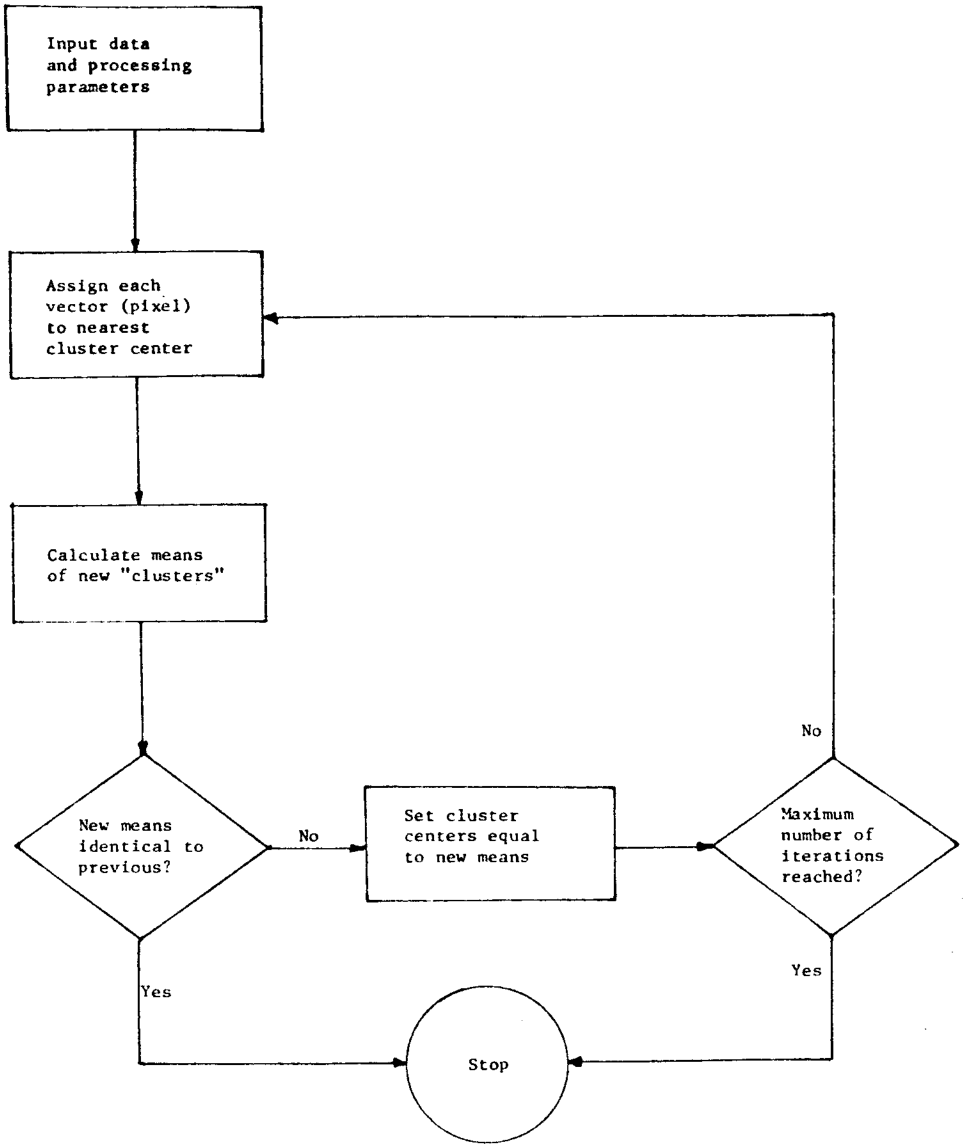


Figure CLU-1. Basic flowchart of clustering algorithm.

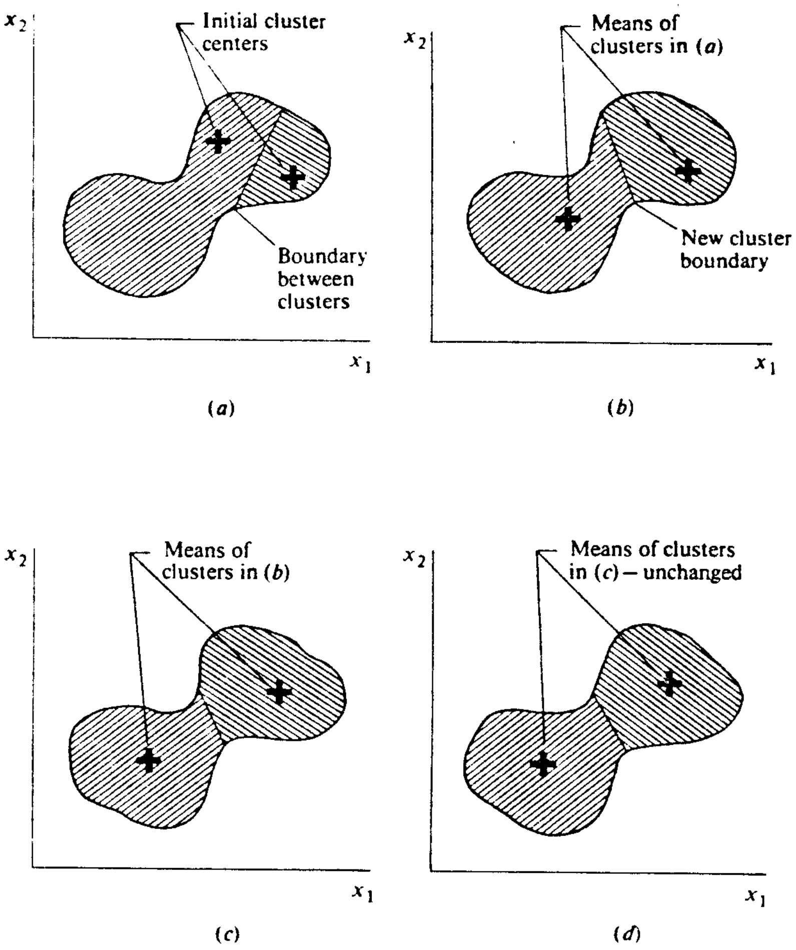
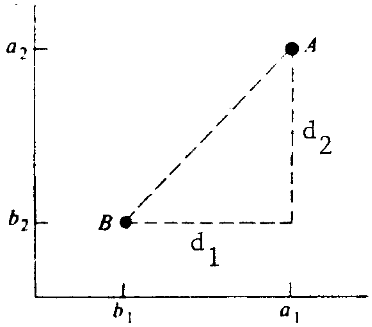


Figure CLU-2. A sequence of clustering iterations. (from Swain and Davis)



L_1 distance

$$D_{AB} = \sum_{i=1}^{n} |a_i - b_i|$$

In two-dimensional space, n = 2 and $D_{AB} = d_1 + d_2$.

Figure CLU-3. Interpoint distance measure using L_1 distance.

The clustering algorithm is called a nonsupervised classifier because it groups or classifies pixels strictly on the basis of their multidimensional data values. Neither the locations of the pixels relative to one another (spatial information) nor the actual surface materials that the pixels represent are considered when the algorithm determines the clusters. The result of the processing is a classification of each data vector in the sample into one of the clusters, with the decision stored as a classification map. Figure CLU-4 shows an example of three channels of input spectral data and a classification map derived from those data.

USER INPUTS TO CLUSTER

To run the Cluster processor, the analyst must first enter the image data set into the COMTAL image memory, call the program and then provide the required information: (a) the area in the image data on which the processor is to operate, generally a subset of pixels from a larger scene; (b) the maximum number of clusters that the processor should produce from the data; (c) the locations of pixels to be used as initial cluster centers, one per cluster; and (d) the maximum number of iterations through which the program should run.

Procedures for entering the data, setting up the display and running the processor are discussed in detail below.

1. Enter the image data

   Data to be clustered must be in the image memory of the COMTAL. If the system has three image planes, then two channels of data may be clustered; if there are four image planes available, then up to three channels of data may be clustered. One image plane must be available to receive a new image, the results of the classification performed by the Cluster processor.

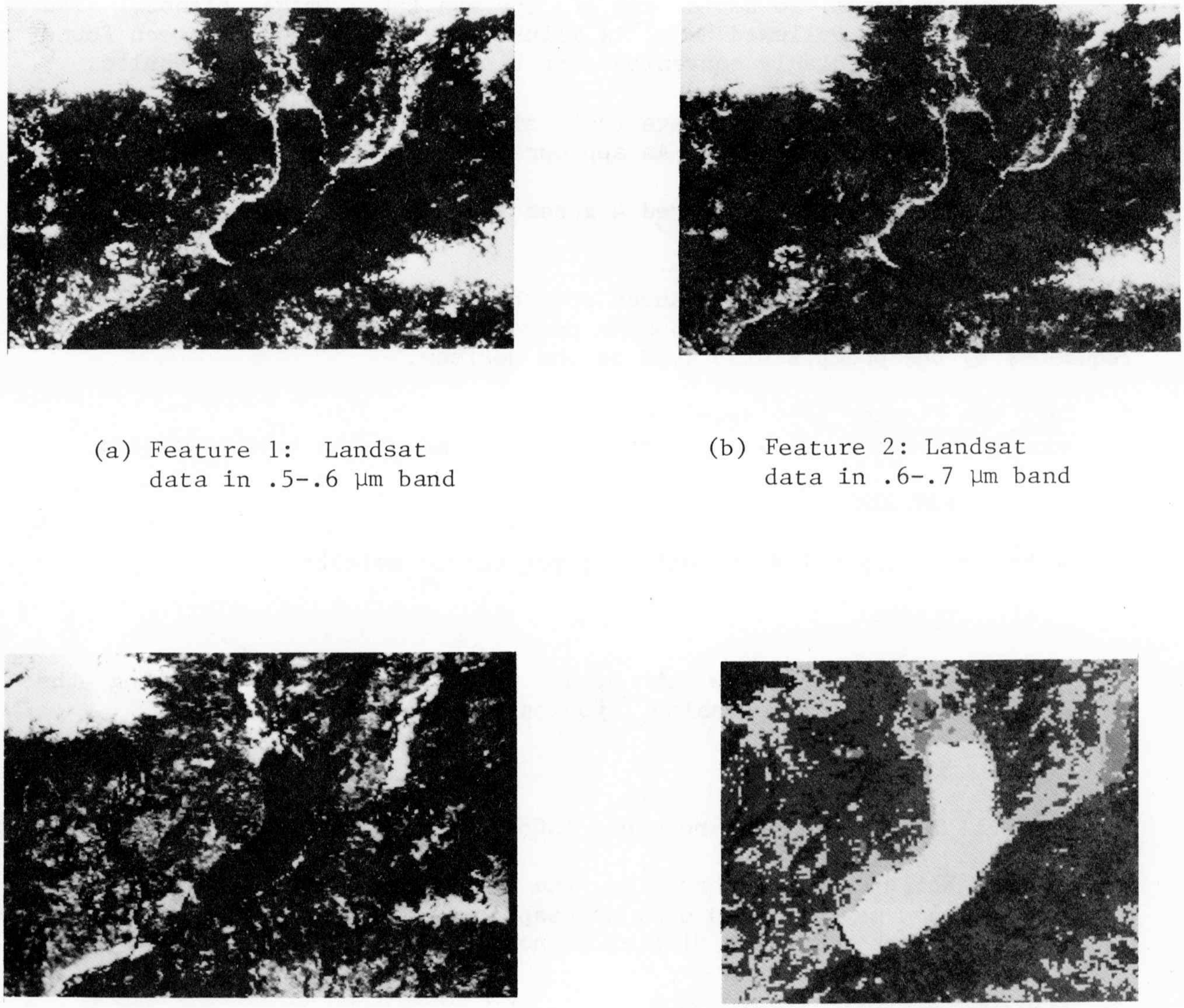
   The following sequence of operations will set up the image data in an optimal way:

   A. Initialize the COMTAL: type (SHIFT RS) R (see COMTAL Notation Conventions on page 16 for interpretation of the notation conventions adopted.)

   B. Load the first image to be used in the clustering into Image Plane 2; enhance the displayed image by histogramming the image and using the equalized histogram as Function Memory 2. Display Image 2 with Function Memory.

   C. Load the second image to be used in the clustering (e.g., the second channel of a multispectral image) into Image Plane 3; enhance and display Image 3 as above.

   At this point two channels of data have been specified for viewing on the monitor. If the system has four image planes a third image



(a) Feature 1: Landsat data in .5-.6 μm band

(b) Feature 2: Landsat data in .6-.7 μm band

(c) Feature 3: Landsat data in .8-1.1 μm band

(d) Classification of three-channel image data into 7 classes using the Cluster processor.

Figure CLU-4. Display of Landsat data in three channels and an example of a Cluster results image for the same area.

may be entered in the same way for display and processing if desired.

Note: The Cluster processor allows the input images and the results image to be in any of the available image planes. The arrangement followed here is illustrative but also has been found to be particularly convenient for viewing both data and results.

D. Set up a truecolor image that allows you to look at the input images simultaneously. An appropriate sequence of commands is:

```
Assign Truecolor 8 red 4 green 3 blue 2
ck N
```

After the input images are enhanced and the truecolor image is displayed, the next step is to call the Cluster processor and provide the information requested by the prompts displayed on the monitor.

2. Call the Cluster processor

Access the COMTAL Utilities Program by typing on the host computer:

```
RUN COMTAL
```

A menu provides a list of options; you should select:

```
CLUSTR
```

If the image display is set up as you wish to have it during the following interactive session, type on the COMTAL:

```
EXecute Code
```

You will be prompted for required information.

Note: While Cluster is running, the command keys are inoperative, and the keyboard may be used only to respond to the questions asked or to end the processing. The display cannot be changed.

3. Provide requested information

Prompts appear on the COMTAL requesting you to supply information as needed.

A. Specify the input and output image planes

The image plane in which the results are to be stored should be different from the image planes containing the data to be clustered. The input images ("features") may be all or any subset of the image planes loaded previously.

Note: The processing time required per iteration is proportional to the number of input images.

B. Specify the area to be processed

The target appears on the screen and you are prompted to specify the area to be clustered. Use the trackball to position the target at the upper-left corner of the rectangular area to be clustered and depress function switch 1 (fs 1). Position the target at the lower-right corner of the area and depress (fs 2). The rectangle shown on the screen outlines the area to be clustered. It may be modified by moving the target and depressing the appropriate function switch (upper-left corner: fs 1; lower-right corner: fs 2). To accept the area defined by the rectangle, hit the space bar.

Note: The processing time required per iteration is proportional to the size of the area clustered.

C. Enter the maximum number of clusters

When prompted, type an integer between 2 and 16 (inclusive), followed by a space bar.

Estimating the right number of clusters is a critical part of the analyst´s job. To make a good estimate, first look at the imagery and at maps and photographs of the area to decide how many information classes occur in the outlined training area. Information classes to count are those whose identity is important to the analysis, such as all primary earth surface materials, perhaps with sub-classes for features of particular importance. Next, look closely at the displayed truecolor image to find characteristic samples of each of the classes. Based on the appearance of the imagery, you may need to adjust the number of classes, decreasing it to account for the possible spectral similarity of different materials and increasing it when obvious subclasses of materials are present. If too few clusters are requested, spectrally different ground covers may be lumped into the same cluster, making the clusters ineffective as training classes. If too many clusters are requested, the spectral classes may be difficult to correlate with information classes, an important step later in the process. When uncertain, it is preferable to ask for too many rather than too few clusters since later some of the clusters may be deleted or combined.

D. Select initial cluster centers

The target will appear on the screen and a prompt will ask you to select the initial cluster centers. Move the target to the first pixel and then hit the space-bar to input the pixel. Pixels may be specified anywhere on the image, either inside or outside of the rectangle surrounding the area to be clustered.

Processing will be more efficient if the initial cluster centers are chosen carefully to represent each spectral class expected. While the processor will run even if the initial cluster centers are not chosen in this way, convergence will be reached sooner if the initial cluster-center locations approximate the final ones. Any identical cluster centers will be replaced by a single cluster center, thereby reducing the total number of clusters in the result.

E. Request number of iterations
You are prompted to specify the maximum number of iterations for the clustering sequence, any integer from 1 through 99, followed by a space. If the processor reaches 100% convergence before it reaches the specified number of iterations (that is, if the cluster centers do not change location in feature space and there are no shifts in pixel assignment on two successive iterations), the processor will stop.

RUNNING CLUSTER

Hitting the space bar after typing the desired number of iterations starts the actual clustering operations. The number of the iteration in progress is displayed on the monitor as well as the number of pixels that have changed class assignment during the previous iteration. (The value that appears there during the second iteration is the total number of pixels in the area being clustered.)

The length of time required to reach the maximum number of iterations is a function of the number of clusters requested, the number of pixels submitted to the processor, and the number of input images; as any of these quantities increases, processing time increases. Moveover increasing any of these quantities tends to increase the number of iterations required to reach 100% convergence.

If you want to stop the processor, depressing fs 1 will cause the Cluster processor to complete the iteration in progress and then return to the point at which the number of clusters is requested; alternatively, depressing ESC will stop the clustering immediately and return to the same point. If no further clustering is desired, you may exit Cluster by simply hitting the space bar in response to the question.

After exiting Cluster, you may run another cluster job on the same data set by typing EXecute Code on the COMTAL; this is possible as long as system reset has not been executed and no other processing code has been loaded from the host computer.

OUTPUTS FROM CLUSTER

With each iteration, the Cluster processor creates a new image of the processed portion of the input image, storing the results in the results image plane you previously designated for that purpose. Each new results image will be written over the previous results image. The results image will appear in the same screen location as the input data and will contain data values from 1 to the number of clusters; all other pixels in that image plane are set to zero.

The results image may be enhanced in several ways for further visual analysis; for example, you could histogram the sub-image and equalize the related function memory, or you could use an integer function to create a suitable function memory as follows:

```
Set INteger function 1
INteger function 1 = Integer (X * Constant 255 / Constant NC #)
Function memory 1 = Integer (integer function 1 #) #
```

where NC equals the number of spectral classes in the clustered data. The cluster image can also be colored effectively by using pseudocolor memory.

The final results image may also be read back to the host computer by means of the COMTAL Utilities Program.

A hands-on self-guided tutorial introduction to Cluster is available. See "A Hands-on Experience with Cluster on the COMTAL Vision One."

References

P.H. Swain and S.M. Davis, 1978. Remote Sensing, The Quantitative Approach, pp. 177-184, "Clustering."

COMTAL Notation Conventions

Notation	Connotation
Display Graphic 1 INteger function 1 = Integer (X + Y #) R	Commands you enter by typing only the capital letters, numerals, and operands, each followed by a space. Lower case letters, equal signs, and parentheses are added by the system.
#	Add an extra space. Ex: for INItialize PSeudocolor memory #, user would type INI, space, PS, space, space.
___	(Underline) A single key represented by a group of letters or numbers; e.g., ESC Escape Key fs 1 Function Switch 1 ck A Command Key A
n,g	(Lower case letters) A number that can vary, depending on the size of the system. n refers to an image number or a memory area related to an image; g refers to a graphic number.
* /	multiplication sign division sign

A HANDS-ON EXPERIENCE WITH CLUSTER ON THE COMTAL VISION ONE

The following exercise leads you through a hands-on, tutorial session with the Cluster processor using a multispectral data set from the Landsat satellite.

Before doing this exercise, you should have read the User's Guide to Cluster on the COMTAL Vision One, beginning on page 5. It is also assumed that you are already familiar with the steps involved in remote sensing analysis based on pattern recognition, with the concepts of COMTAL-based image processing, with loading data into the COMTAL image memory and issuing basic commands on the COMTAL. (See Image Processing on the COMTAL Vision One Series - A Beginner's Guide by S.M. Davis, D.M. Freeman, and P.H. Swain, 1981, to acquire experience using the COMTAL.)

When you have finished this exercise (including the user documentation listed above), you should be able to successfully carry out the following:

1. Describe the role of clustering in the analysis of multispectral data.

2. Explain with the aid of a sketch how the clustering algorithm works; include in your explanation the following terms: initial cluster centers; migration of cluster centers; iterations; class means; L_1 distance; and convergence.

3. Define the following terms: feature space, cluster class, cluster map, and convergence.

4. Call and run the Cluster processor, selecting an area to be clustered, specifying the number of clusters to be formed, providing the locations of the initial cluster centers, and stating the number of iterations to be used.

5. Display the results of the Cluster processor, and, with the aid of reference data, infer the information class most closely associated with each spectral or cluster class.

6. Transfer Cluster results back to the host computer for storage or further processing.

Step 1 - Load the data into the COMTAL image memory

The loading instructions given here are for use with the COMTAL Utilities Program installed on the PDP-11/34 system at Purdue/LARS and may require modification for other systems. See the COMTAL Notation Conventions at the conclusion of the "User´s Guide to Cluster," page 16.

Initialize the COMTAL by typing on the keyboard:

```
        [SHIFT RS] R
```

Load image data from the PDP using the COMTAL Utilities Program. On the PDP, enter the following commands, each followed by CAR RET:

```
        RUN COMTAL      (to call the COMTAL Utilities Program)
        DATA            (to request the routine for transferring data)
        DATATO          (for subroutine that sends data to the COMTAL)
```

When requested, load the following images into the image planes shown:

```
        DB1:[120,10]VALLECITO.CH1     into Image 2
        DB1:[120,10]VALLECITO.CH2     into Image 3
        DB1:[120,10]VALLECITO.CH4     into Image 4
        ESC                           (to return to sub-menu)
        END                           (to return to main menu)
```

Image 1 is left unused so that it can be used later to receive the results of the clustering, an arrangement that will be helpful in interpretation and comparison of the images.

Step 2 - Call the Cluster processor

The main menu of the COMTAL Utilities Program gives you access to the Cluster processor. On the PDP, type:

```
        CLUSTR
```

With this command, the code for running Cluster is transferred to the COMTAL, and you may log off the PDP.

The PDP screen will prompt you to type "EXECUTE CODE"; however, before doing this you should enhance the image data and set up a truecolor image made up of the three images. (If you wish to do so now, you may alter Function Memory 1 to enhance the Cluster results; specific instructions for doing this are given as part of Step 6, a more convenient time to perform this operation.) These enhancement steps may not be carried out while Cluster is running; the command keys are inoperative then, and the keyboard may be used only to respond to specific prompts. Enhancements must therefore be done before you type "execute code" or after the processing is complete.

Step 3 - Enhance the image data

Begin this operation by histogramming each of the three images containing data and equalizing the histograms to create the respective function memories. You may do this by entering the following commands on the COMTAL:

```
Function memory 2 = Histogram of image
Equalize Function memory 2
ck H
```

Repeat this sequence for Images 3 and 4, and then display each image in turn to become familiar with the general characteristics of the data. Remember that there is no command key that allows you to display Image 4. You must instead use the commands:

```
Display Image 4
Add Function memory 4
```

Create a truecolor image now, using images 2, 3, and 4. This will allow you to see the three channels of data simultaneously as a color composite. If you wish to display the scene with colors that approximate those of a color-IR photograph, make the following color assignments:

```
ASsign Truecolor 8 red 4 green 3 blue 2
```

Use ck N to see the truecolor image with function memories added.

Step 4 - Become familiar with the data and region

Look at the three enhanced images separately and also together in a truecolor image to gain a general sense of the quality of the data and the primary features of the scene. Answer the following questions:

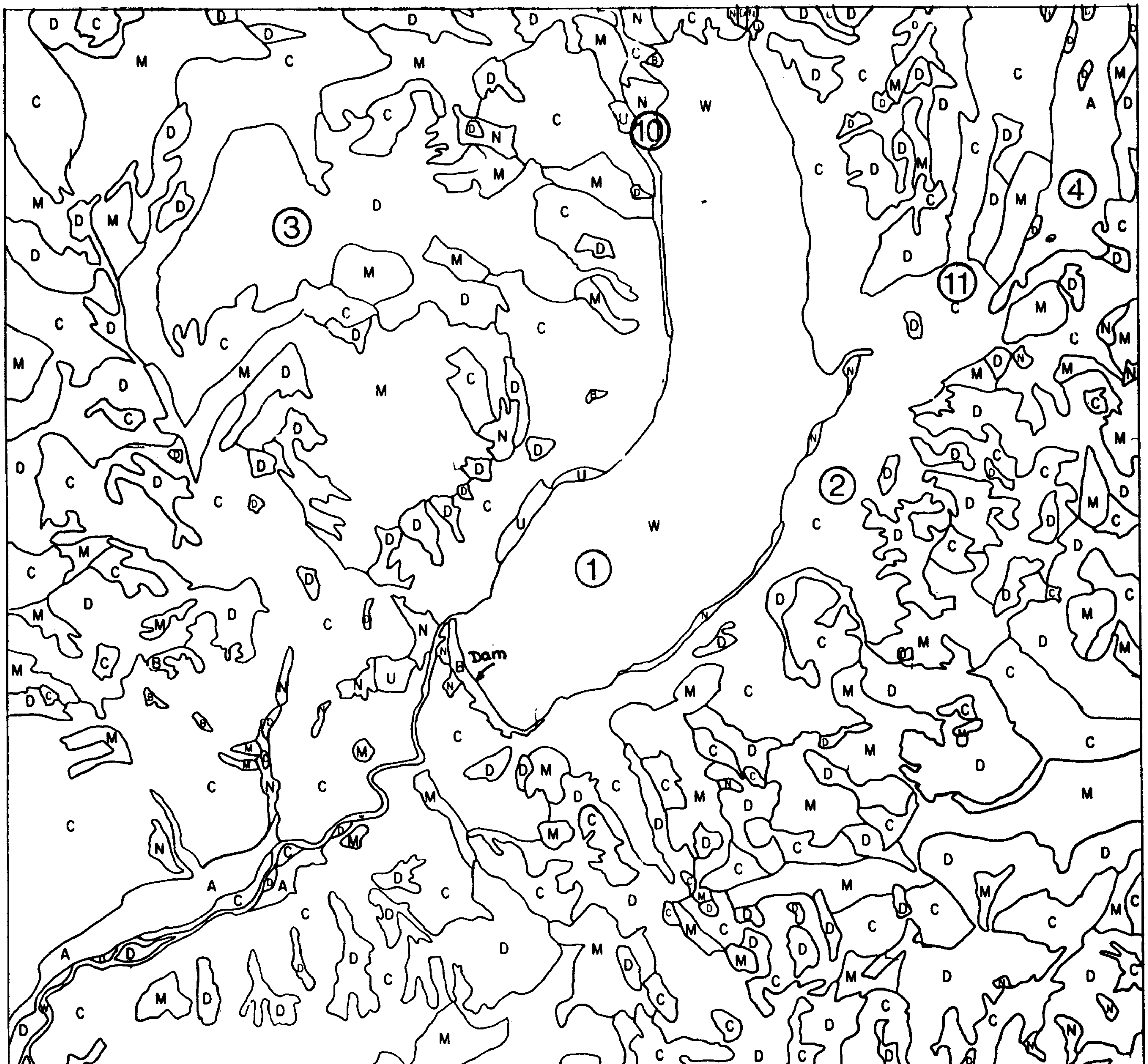
a) Is there any obvious noise in any of the images?
Look for both linear patterns that may have been caused by the sensor and for other abberations in the data, including clouds and cloud shadows. Any significant amount of "noise" may degrade the clustering results.

b) What are the major geographic features of the imaged area?
Several kinds of reference data are included to help you answer this question. A portion of a U.S.G.S. topographic map, Figure 1, shows the part of the scene near Vallecito Reservoir, the largest body of water. Figure 2 is a Level 1 Cover Type map of the same area showing the primary materials

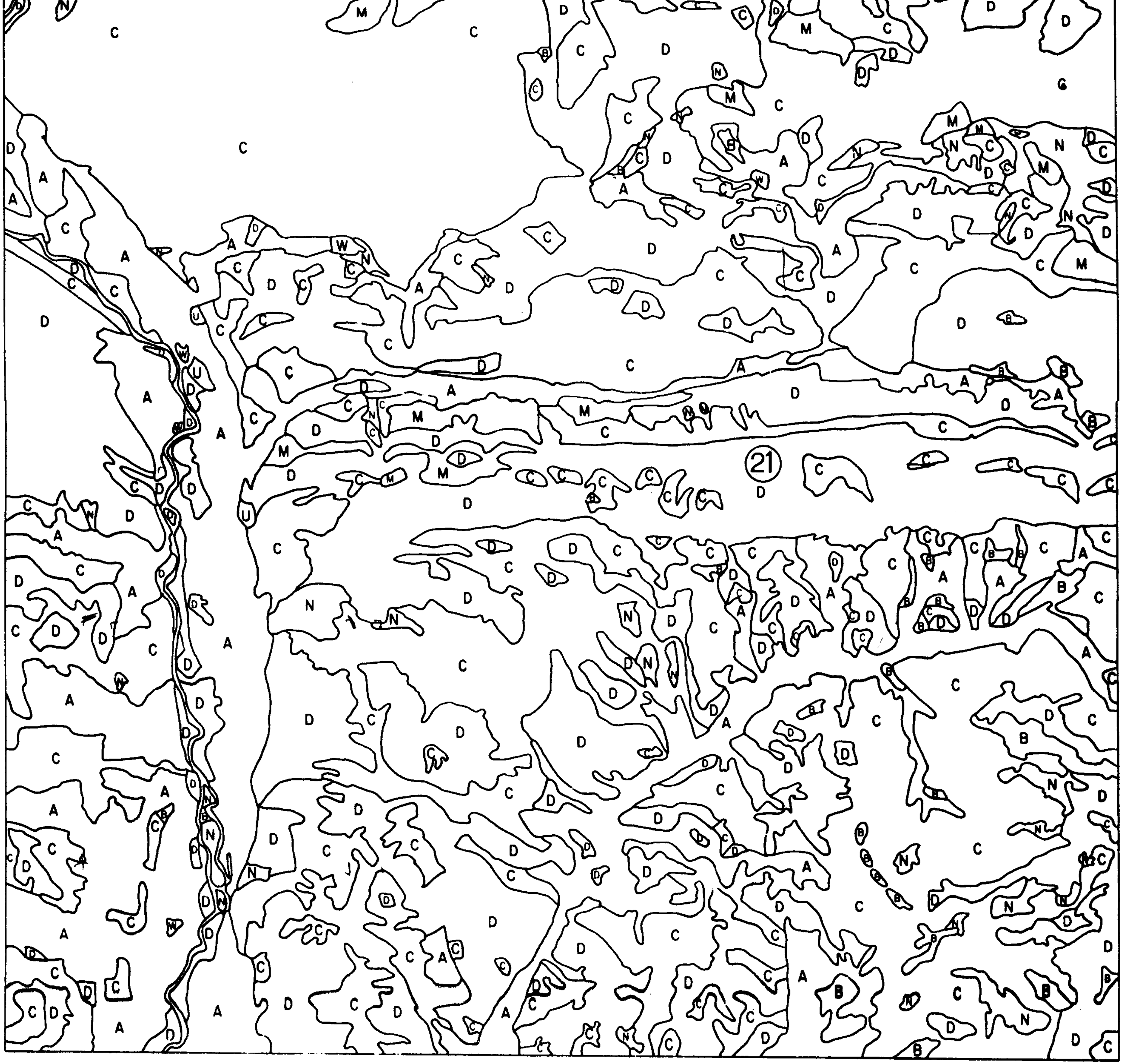
present, e.g., deciduous forest, coniferous forest, agriculture, urban, water, etc. A color IR aerial photograph of the reservoir area is available as a slide.

To enhance your familiarity with the scene, sketch the major geographic features of the area shown on the screen, providing general labels for features you can identify.

c) Are there any features in the image data which are transitory or seasonal and therefore may not appear on maps or on photographs taken at a different time? Examples may include floods, clouds and cloud shadows, seasonal vegetation changes, and so on. List here as many as you can identify.


W
1
2
3
4
10
11
Dam
A
B
C
D
M
N
U

21b



VALLECITO STUDY AREA

21c

Figure 2. Level I cover type m
the Vallecito Study
produced through pho
interpretation.

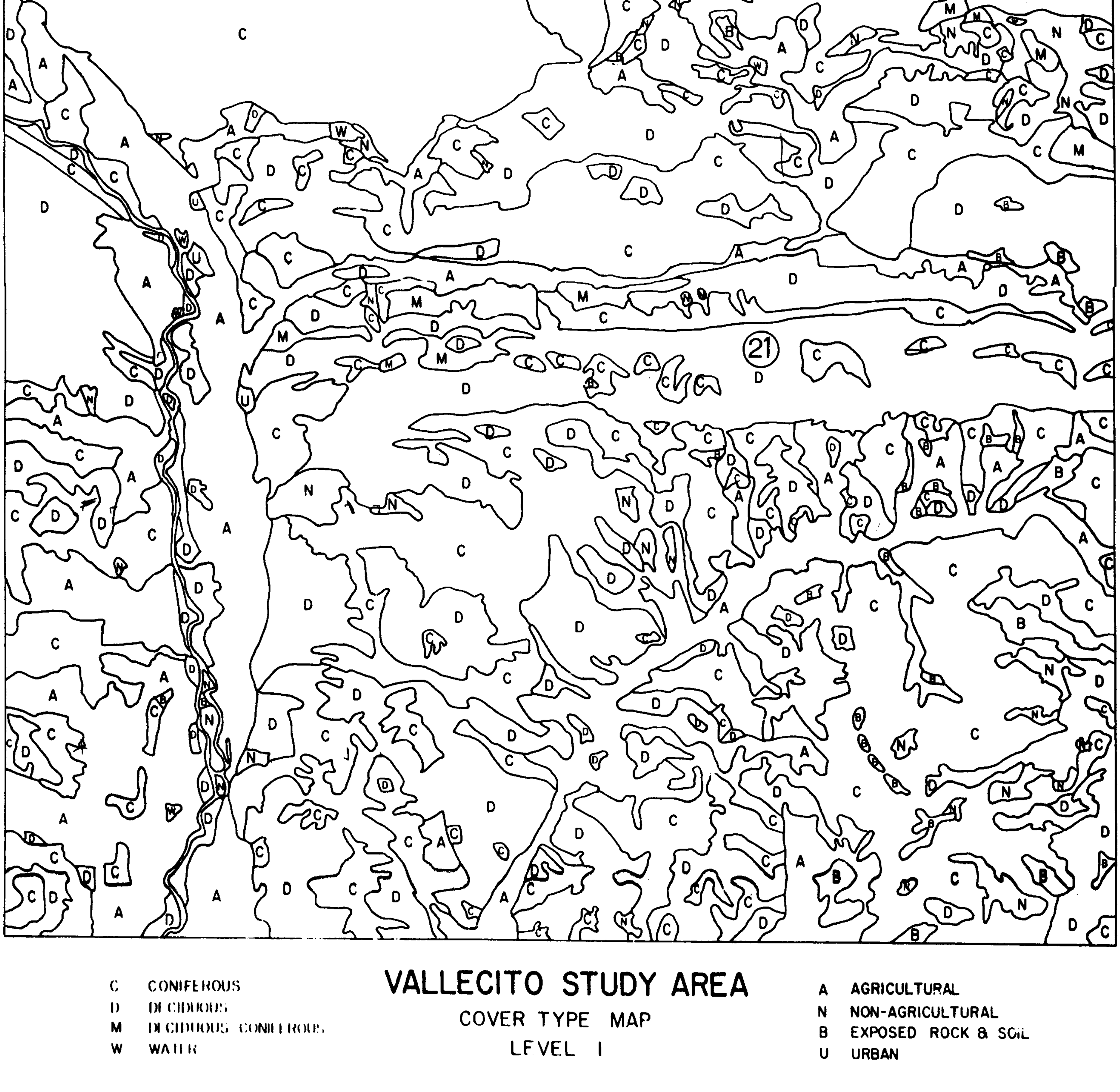


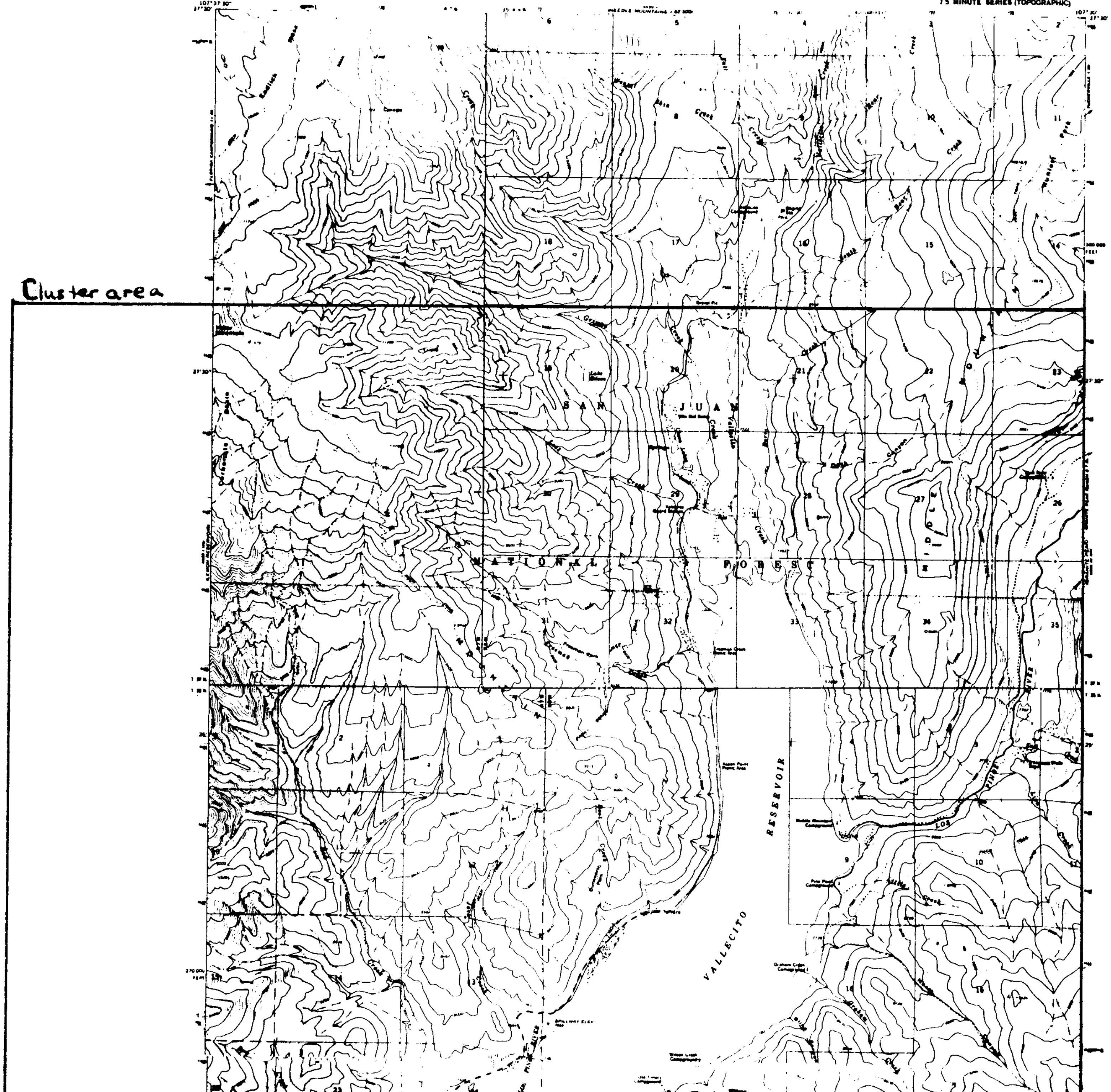
Figure 2. Level I cover type map of the Vallecito Study Area, produced through photo-interpretation.

21c

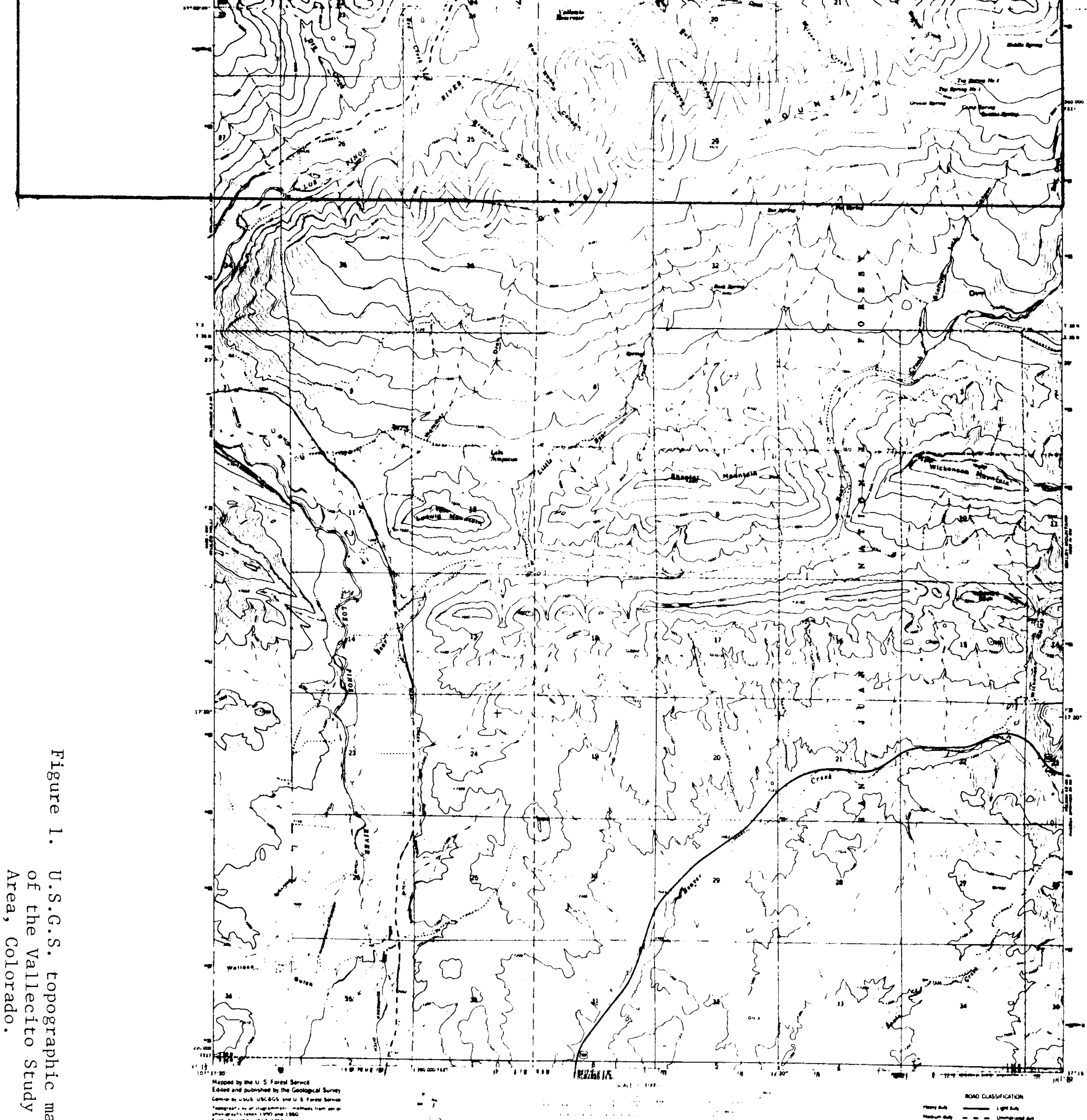
22a

UNITED STATES
DEPARTMENT OF THE INTERIOR
GEOLOGICAL SURVEY

VALLECITO RESERVOIR QUADRANGLE
COLORADO - LA PLATA CO.
7.5 MINUTE SERIES (TOPOGRAPHIC)



22b


Shearer Mountain
Mapped by the U. S. Forest Service
Edited and published by the Geological Survey
Control by USGS, USC&GS, and U. S. Forest Service
Topography by photogrammetric methods from aerial photographs taken 1950 and 1960
ROAD CLASSIFICATION
Heavy duty
Light duty
Medium duty
Unimproved dirt

22c

Figure 1. U.S.G.S. topographic ma
of the Vallecito Study
Area, Colorado.

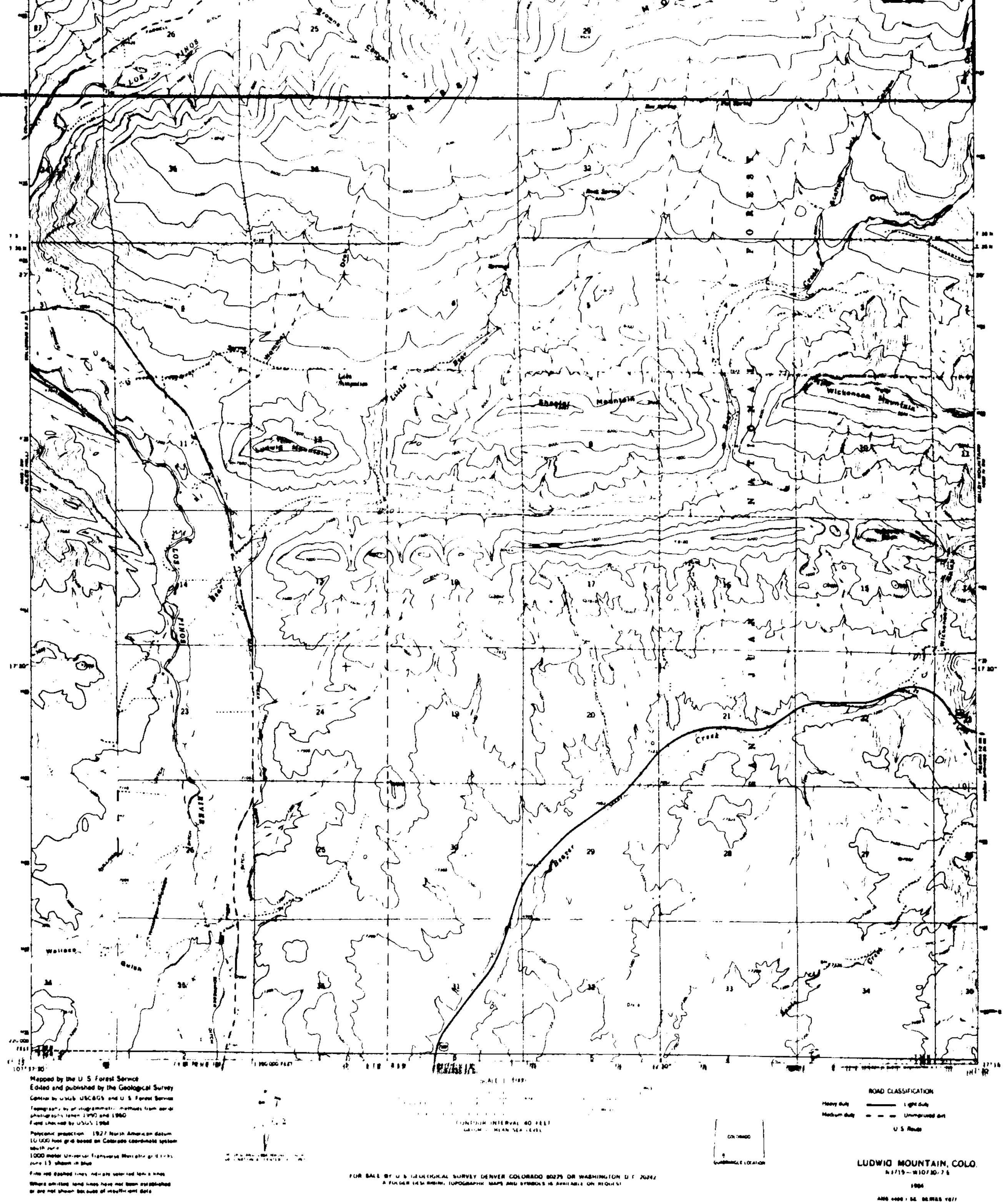

Los Pinos
Shearer Mountain
Wickensen Mountain
Ludwig Mountain
Little Bear
Los Pinos River
Beaver Creek
Wallace Gulch
Mapped by the U.S. Forest Service
Edited and published by the Geological Survey
Control by USGS, USC&GS and U.S. Forest Service
Topography by photogrammetric methods from aerial photographs taken 1950 and 1960
Field checked by USGS 1964
Polyconic projection. 1927 North American datum
10,000-foot grid based on Colorado coordinate system south zone
1000-meter Universal Transverse Mercator grid ticks, zone 13, shown in blue
Fine red dashed lines indicate selected fence lines
Where omitted, land lines have not been established or are not shown because of insufficient data
CONTOUR INTERVAL 40 FEET
DATUM IS MEAN SEA LEVEL
FOR SALE BY U.S. GEOLOGICAL SURVEY, DENVER, COLORADO 80225 OR WASHINGTON, D.C. 20242
A FOLDER DESCRIBING TOPOGRAPHIC MAPS AND SYMBOLS IS AVAILABLE ON REQUEST
COLORADO
QUADRANGLE LOCATION
ROAD CLASSIFICATION
Heavy duty
Light duty
Medium duty
Unimproved dirt
U.S. Route
LUDWIG MOUNTAIN, COLO.
N3715—W10730/7.5
1964
AMS 4460 I SE—SERIES V877

Figure 1. U.S.G.S. topographic map of the Vallecito Study Area, Colorado.

22c

Step 5 - Running Cluster

Now you are ready to run the Cluster processor. With the truecolor image displayed, type on the COMTAL keyboard:

EXecute Code

You will then be asked a series of questions that will control the way Cluster is to be run. For the clustering, you will use all three images you entered into COMTAL memory: Landsat Band 4 (.5-.6 µm) in Image 2; Band 5 (.6-.7 µm) in Image 3; and Band 7 (.8-1.1 µm) in Image 4. Each of these images is referred to as a feature, a pattern recognition term used to refer to the individual elements that make up a multivariate data element.

Enter the following responses to the prompts, followed by a space:

Prompt	Response
How many channels of input data	3
What image for results image?	1
What image plane contains feature 1?	2
What image plane contains feature 2?	3
What image plane contains feature 3?	4

The next prompt asks you to select the cluster area. You do this by moving the target with the trackball and hitting function switches 1 and 2 to identify the upper left and lower right corners respectively.

The first area you will cluster includes the Vallecito Reservoir, the surrounding forested area, some snow and an agricultural area. The area is outlined in Figure 1.

Move the target to the upper left corner of the area and hit <u>fs 1</u>; then move it to the lower right corner and hit <u>fs 2</u>. A rectangle will appear on the screen outlining that area. If you want to reposition either of the two corners, move the target and hit either <u>fs 1</u> or <u>fs 2</u>, as appropriate. Hit the space bar to accept for clustering the area enclosed by the resulting rectangle.

The next prompt asks you to state the number of cluster centers you will input, in other words, how many classes should the data be divided into? There is no "right" answer to this question, but you can begin to answer this question by looking at the image and the reference data. Correlate the representation of this area on the display with its appearance on the topographic map and the cover type map, Figures 1 and 2.

Approximately how many different major classes of materials do you find in the scene? Count the identifiable major cover types; if any of the classes appear in areas exhibiting significantly different spectral responses, count them as additional classes. The snow-covered area would be an example of this. You should be able to identify between five and nine potential classes. List the ones you find.

1.
2.
3.
4.
5.
6.
7.
8.
9.

While you are looking at the classes, try to identify at least one fairly homogeneous sample of each of the classes. These samples will be used later for identifying cluster centers.

There are several ways to draw up the list, but for the sake of this exercise, we´ll use these six classes:

1. water
2. coniferous forest
3. deciduous forest
4. agriculture
5. snowedge
6. snow

The location of a representative sample for the first four of these classes is shown on Figure 2.

In response, now, to the prompt, type 6 and then specify the six initial cluster centers by placing the target on a representative pixel, then hitting space. (Remember, the upper left corner of the target is the pixel you are pointing to.) Provide a representative pixel for each of the clusters by moving the target to the pixel and hitting space.

The last prompt asks you to set a maximum number of iterations. Set your limit at 99, the maximum allowed; in that way the processor will not stop prematurely but in all likelihood will run to 100% convergence, unless you interrupt it.

At this time the processor will begin to run. A message on the screen shows the number of the iteration currently underway and the number of points that had changed class assignment during the previous iteration. During the first iteration the number is, of course, zero, but during the second iteration the number that appears is the total number of points being processed.

Make note of that number here: _______________

If you wish, you may stop Cluster in either of two ways: hitting fs 1 will stop the processor when the current iteration is complete; ESC will stop it immediately. Both actions return you to the point in the program where the prompt occurs "How many cluster centers?"

The ideal in clustering, of course, is to reach convergence, but, to save processing time, 99% convergence is quite adequate for most analysis of multispectral data. If you made note above of the total number of points being processed, figure out what 1% of that amount would be. When the number of points that changed class membership falls below that number, hit fs 1 to complete the current iteration and halt processing. Most likely this will occur in less than ten iterations. Alternatively, you can let the processor run to convergence. Don´t be dismayed if the number of changes increases on a few iterations; the trend will continue toward convergence.

In either case, when the clustering is complete, you will be prompted: How many cluster centers? If you are not satisfied with the results, you may run Cluster again by typing in the desired number of clusters and responding to the subsequent prompts as before. If you are satisfied and want to proceed to the next step, simply hit the space bar.

Step 6 - Interpreting the results

When the clustering is complete, a classification image (or cluster map) has been created in Image 1, which you left clear for that purpose. Since there are six classes in that cluster map, the data values in Image 1 range from 1 through 6, with 0 for all parts of the image outside the map. For the six cluster classes to be distinguishable from each other on the screen, you may use an integer function to create Function Memory 1 with the best attributes:

```
Add Annotation characters
Set INteger function 1
INteger function 1 = Integer (X * Constant 255 / Constant 6 #)
Function memory 1 = Integer (integer function 1 #) #
```

The constant 6 was entered in the third line to correspond to the six classes used in the clustering. (If a different number of cluster classes had been requested, the constant should have corresponded to that number.)

To see what this step accomplished, hit ck J to display Function Memory 1. (Subtract the bottom line -- SUbtract Bottom line -- to allow you to see the whole screen.) The first seven values of the image (shown across the horizontal axis) will be displayed using the full brightness range of the display (shown along the vertical axis). You should be able to count the values from 0 through 6; see Figure 3 for further explanation.

Another way to enhance the classification image is to histogram all the values in Image 1 and equalize the histogram to create a new function memory; however, the large area of the image outside the clustered areas contains all zeros and would yield a less-than-ideal image function memory.

Add the annotation characters; hit ck G to see the enhanced cluster map.

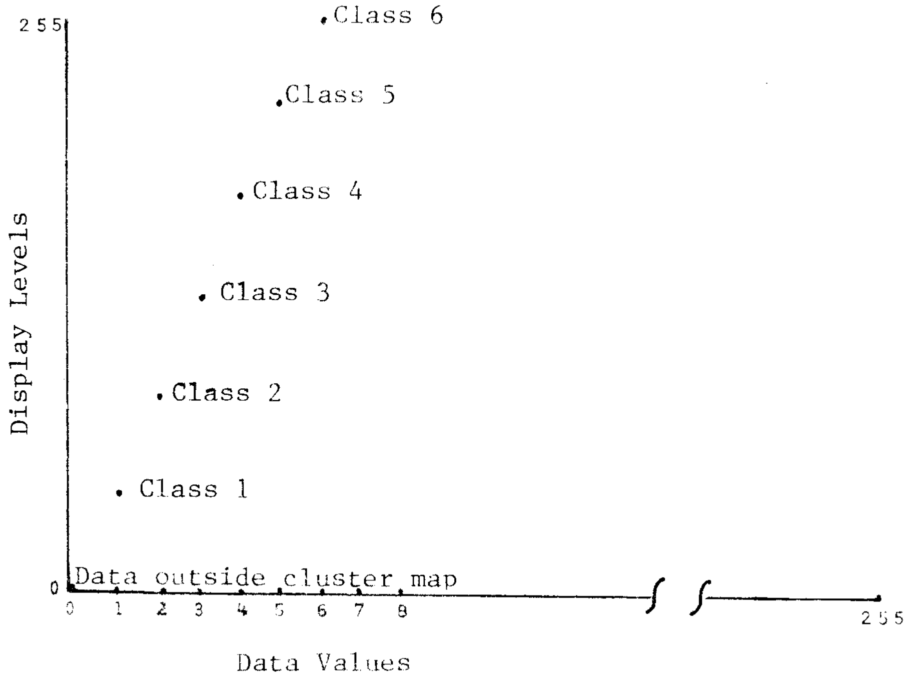


Figure 3. A portion of Function Memory 1, calculated to enhance a 6-class classification image. (Horizontal axis is exaggerated for clarity.)

Once the Cluster results image is enhanced and displayed, you are ready to begin the interpretation of the classes. Each of the six cluster classes will need to be identified with a descriptive label that relates it to cover type. To do this more easily, zoom the cluster map (Image 1) by a factor of four:

ROam Image; fs 1

Use the track ball to move the image on the screen to the upper left corner, then strike the Escape Key.

Display Zoom Image by a factor of 4

Now, with the cluster map displayed, put the COMTAL in the Dump Image mode by typing DUmp Image. The values that are shown make it easy to read the class assignment of each pixel in the map. Use the Level 1 Cover Type Map (Figure 2) and the aerial photograph (slide format) to help label each of the clusters in the chart below. It may help to compare the results image with the truecolor image. If you enlarge the truecolor image (Image 8) by the same factor and roam it so that the features have the same screen location as in the cluster results image, you can alternate between the two using ck N and ck G.

Another aid in viewing the cluster image is to display it with pseudocolor. Roll the three pseudocolor memories so that they are distributed along the vertical axis in the 0 to 6 range. Use ck 0 to display the pseudocolor memory, roll the pseudocolor memories to appropriate locations, then type:

```
ck D
Add Function memory 1
```

With the pseudocolor display of the results image, it is easier to locate individual occurrences of a given class. You may also use DUmp Image here with the pseudocolor image to assist in class definition, or you may use it with the truecolor image displayed. In the latter case, it allows you to read the relative spectral response in each channel and make judgments about cover materials based on this information. Fill in this chart:

Cluster Number	Cover type of input pixel	Class labels you assigned
1	Water	
2	Coniferous	
3	Deciduous	
4	Agriculture	
5	Snow Edge	
6	Snow	

Depending on where precisely you located the area to be clustered and which pixels you used to initialize the cluster centers, you should have found that the new class labels were the same or nearly the same as the original labels.

Are there any areas where the results are surprising, where something on the ground appears to be misclassified? If so, can you explain why it was classified as it was? Three specific areas that may have caused some problems require additional study:

1. How did the dam get classified? Most likely it came out as snow or snow edge. Recall that you input no cluster center to represent the dam and the highly reflective bare rock along side it. Can you explain why it was put into the same class as the snow?

2. On the west edge of the reservoir near the top there is a triangular area (#10) identified on the Cover Type Map as "non-agricultural." Which class was it assigned to? Look at the topographic map and the aerial photograph to see if you can explain why it was classified as it was.

3. According to the Cover Type Map, an area to the east of the reservoir (#11) contains coniferous forest. How was the area classified by Cluster? Look at the area on the aerial photograph and see if you can understand why it was so classified.

In comparing the image on the screen with the maps and photograph, did you notice that the geometric proportions of large recognizable features are not the same? For example, Vallecito Reservoir is more elongated on the maps than on the screen. This geometric distortion is caused by the fact that each pixel represents not a square but a rectangular area on the ground, as shown in Figure 4. This same pixel is displayed on the monitor by means of a screen area having equal length and width. The result is the vertical foreshortening of features on the display screen as in Figure 4.

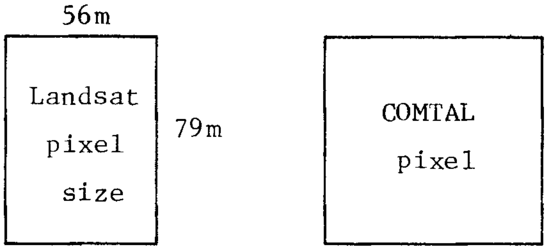


Figure 4. Area of the earth represented by each pixel of Landsat data.

Step 7 - Run Cluster a second time

To complete this exercise, run Cluster again using another portion of the scene.

You will first need to restore the images to their original size and position by zooming both Image 1 and Image 8 by a factor of 1 and then repositioning them with Roam Image and fs 5. (With the present version of the program, the area to be clustered and the initial cluster centers cannot be properly defined if the image is zoomed and/or roamed.)

Choose an area you would like to work with and select a specific rectangle to cluster. You may want to choose an area to the south of the area we just used since the maps that are available cover that area. In general this area is much brighter than the previous area; this change is especially evident when you contrast the appearance of the same cover types. For example the deciduous forests (#21) are quite a bit lighter in tone.

To begin Cluster again, display the truecolor image of the area and type:

```
EXecute Code
```

You may respond to the prompts as you did before, using up to three channels of data for the input data and placing the Cluster results in Image 1. (This will destroy the Cluster results currently there.)

Estimate the number of clusters needed for this area by looking at the image along with the reference data, as you did before. Up to 16 clusters may be requested. Again, be prepared to specify a cluster center for each cluster, and set the number of iterations at 99. Watch for the number that tells you how many pixels are in the area being clustered and hit fs 1 when the number of changed pixels falls below 1% of that total. (Note: Ignore the number of changed pixels displayed during the first iteration.)

If you vary the number of clusters, make the area larger or smaller, or use two channels of input data instead of three, you will gain some understanding of relative differences in processing times with different parameters.

Step 8 - Store Cluster results

Now that you have completed the Cluster operation, you need to store the results for future use or for further processing. This can be done through the COMTAL Utilities Program which is implemented on the PDP-11/34 at Purdue/LARS.

The processor for doing this is DATAFR, a subroutine that sends data from the COMTAL to the host computer. Log on the PDP and enter the following commands:

```
RUN COMTAL     (to call the COMTAL Utilities Program)
DATA           (to request the routine for transferring data)
DATAFR         (for subroutine that sends data from the COMTAL to
                the host)
```

You will be prompted to provide a file name for the new image and to state the COMTAL image plane number where the cluster results are located.

```
Enter your image file name:
```

The format expected is filename.filetype where up to nine characters may be specified for the filename and up to three for the filetype. The next prompt will be:

```
Enter image number 1, 2, 3:
```

Since the Cluster results are in Image 1, respond by typing 1. The sign VIS@VIS1 indicates that the transfer is taking place. You may verify that the file is saved by listing all the files on your portion of the disk; first hit ESC and then type END twice to exit the COMTAL Utilities Program. To see a list of the files, type the following on the PDP:

```
        PIP DB1:/LI
```

The list that appears should contain the name you just entered. If you wish to delete the file, type:

```
        PIP DB1:FILENAME.FILETYPE;*/DE
```

This ends the hands-on portion of this exercise. If the concept of clustering was new to you, you may benefit from reviewing some of the user documentation that you read earlier, prior to this hands-on work.

Step 9 - Review portions of the user documentation

Now that you have completed the hands-on activity, take a few minutes to review portions of the user documentation to reinforce your understanding of the clustering process and the significance of its output results. Especially recommended is "General Description of the Clustering Process," pages 5-10 in the *User´s Guide to Cluster*.

When you have completed this reading, do the self-check questions that follow.

Page 31 is blank

Self-Check Questions

The questions below are based on some of the objectives stated at the beginning of this hands-on exercise. Please test your understanding of the material by writing your answers in the space provided. You may then check your answers against the sample answers on the following page.

1. Describe the role of clustering in the classification of multispectral data.

2. Using Figure CLU-2 on page 8, explain the following terms in relation to the clustering processor: initial cluster centers; migration of cluster centers; iteration; class means; L_1 distance; convergence.

3. Describe briefly the steps you followed in running Cluster.

(Answers to Self-Check Questions are on page 33.)

Answers to self-check questions

1. The clustering algorithm finds natural clusters or groupings within the data, based on their values in two or more channels. Statistical parameters of the resulting spectral classes, such as the means and covariances of the classes, are then calculated and passed to the classifier to characterize the training classes used in the classification.

2. Initial cluster centers are the multi-channel values of points which have been submitted by the user to start the clustering process. Migration of cluster centers is the movement, in feature space, of the cluster centers as the mean of each class is re-calculated to represent the new clusters of pixels. An iteration in Cluster is the process of assigning each point to the nearest cluster center and then re-calculating the mean of each cluster. The class mean is the mean value of all the points assigned to a single cluster class. L_1 Distance is a statistical distance measure that is the sum of the distances between two points measured for each channel (or feature). Convergence is the condition that occurs when newly calculated cluster centers are identical with previously calculated ones, indicating that no pixels were assigned to different classes during that iteration.

3. Your answer should contain at least the following: sent data to the COMTAL image memory; enhanced the images; sent the Cluster code to the COMTAL; responded to prompts to provide information on ths features to be used in processing, the desired location of the results image, the area to be clustered, the number of clusters, the locations of initial cluster centers, and the number of iterations; and stored the results.

Page 34 is blank

APPENDIX A

Statistics Calculation Using CSTATS

This note describes the use of the CSTATS program which computes class statistics based on a cluster map generated by the COMTAL image display system (hence, Cluster STATisticsS). The program displays the statistics on the PDP terminal screen and, on request, generates a LARSYS statistics file which can be transmitted to the IBM computer for use in further processing steps.

After running the CLUSTER processor on the COMTAL, a "cluster map" resides in one of the COMTAL image planes. The cluster map may be moved to a PDP disk file using the COMTAL utilities´ DATA/DATAFROM function.

Once the cluster map is stored on the PDP system, the CSTATS program can be invoked and run interactively as follows (user inputs are underlined):

```
>RUN CSTATS                                           (invoke program)
> RUN CSTATS
PLEASE ENTER FILE SPECIFICATION OF CLUSTER MAP:
CLU.CHX                                       (name of cluster map file)
ENTER NUMBER OF FEATURES (CHANNELS):
2                                                     (maximum of 3)
ENTER FILE SPECIFICATION OF FEATURE 1:
TIPPE.CH1                                             (name of data file)
ENTER FILE SPECIFICATION OF FEATURE 2:
TIPPE.CH2                                             (name of data file)
```

After the cluster map name and the original data file names have been supplied, CSTATS computes the statistics based on the CLUSTER map. It may take 7-10 minutes for computation. In order to inform the user about the progress of computing, it writes a message on the screen for each 16 scan lines.

```
PROCESSING COMPLETE TO LINE 16
PROCESSING COMPLETE TO LINE 32
    '          '        '
    '          '        '
    '          '        '
PROCESSING COMPLETE TO LINE 512
```

The cluster class statistics are then written out on the screen, one class at a time:

```
CLUSTER CLASS 1
  NO. OF PIXELS = 2586
  MEAN VECTOR = 49.4  51.2
  COVARIANCE MATRIX:
         15.2   20.5
         20.5   33.4
--------------------------------------------------
TYPE -RETURN- TO SEE STATISTICS OF NEXT CLASS
CAR RET

CLUSTER CLASS 2
  NO. OF PIXELS = 1294
  MEAN VECTOR = 35.9  28.2
  COVARIANCE MATRIX:
         4.7   5.5
         5.5  12.3
--------------------------------------------------
TYPE -RETURN- TO SEE STATISTICS OF NEXT CLASS
CAR RET

CLUSTER CLASS 3
  NO. OF PIXELS = 2728
  MEAN VECTOR = 41.7  38.7
  COVARIANCE MATRIX:
         3.9   4.5
         4.5  10.7
```

The above procedure will continue until statistics of all classes have been written out. The program then will ask the user whether a statistics file for LARSYS is desired:

```
DO YOU WANT TO CREATE A STATISTICS FILE? (YES/NO)
YES
```

If the answer is yes, information which must be supplied by user will be requested as followed:

```
PLEASE ENTER STATISTICS FILE NAME
TIPPE.DAT                                          (output file name)
PLEASE ENTER TRAINING FIELD RUN NUMBER
73077777                                   (original data run number)
PLEASE ENTER WAVELENGTHS OF SPECTRAL BAND OF FEATURE: 1
- LOWER END: (IN MICROMETERS) 0.5
- UPPER END: (IN MICROMETERS) 0.6
PLEASE ENTER WAVELENGTH OF SPECTRAL BAND OF FEATURE: 2
- LOWER END: (IN MICROMETERS) 0.6
- UPPER END: (IN MICROMETERS) 0.7
```

```
PLEASE ENTER CALIBRATION CODE: 1

CALIBRATION VALUES FOR CHANNEL 1
PLEASE ENTER VALUE OF CO
0
PLEASE ENTER VALUE OF C1
2.48
PLEASE ENTER VALUE OF C2
0

CALIBRATION VALUES FOR CHANNEL 2
PLEASE ENTER VALUE OF CO
0
PLEASE ENTER VALUE OF C1
2.00
PLEASE ENTER VALUE OF C2
0

TT3 -- STOP
```

The CSTATS program finishes at this step, and the contents of file TIPPE.DAT (name supplied above by user) look like Figure 1. To display your file contents, type PIP TI:=filename.filetype. The first line is inserted to tell CMS ´READCARD *´ COMMAND that the statistics file name is CLUSTER STATS. This name can be edited either on the PDP or later on the IBM.

```
:READ CLUSTER STATS                                                                  0
LARSYS VERSION 3 STATISTICS FILE          0                                          1
CLASS NS-    1    3                                                                  2
73077777             9999 9999    9 9999 9999    9                                   3
CLASS NS-    2    3                                                                  4
73077777             9999 9999    9 9999 9999    9                                   5
CLASS NS-    3    3                                                                  6
73077777             9999 9999    9 9999 9999    9                                   7
    3 CLASS     3 FIELD     2 CHANNELS                                               8
CHAN  1 WAVELENGTH 0.50- 0.60 CODE  1 CO    0.00 C1    2.48 C2    0.00               9
CHAN  2 WAVELENGTH 0.60- 0.70 CODE  1 CO    0.00 C1    2.00 C2    0.00              10
NO. PTS.     2586      1294      2728                                               11
MN 0.4943349E+02 0.5115661E+02 0.0000000E+00                                        12
MN 0.3587867E+02 0.2818315E+02 0.0000000E+00                                        13
MN 0.4165396E+02 0.3871701E+02 0.0000000E+00                                        14
CV 0.1522392E+02 0.2047387E+02 0.3343603E+02                                        15
CV 0.4743394E+01 0.5517585E+01 0.1229335E+02                                        16
CV 0.3947704E+01 0.4522308E+01 0.1066625E+02                                        17
EOS           *****   LAST CARD OF STATISTICS DECK   *****                          18
```

Figure 1. Contents of File TIPPE.DAT

To transfer the statistics file to the LARSYS system, the PDP-IBM interface program LITER can be used.

Note that TEL interface may not be active at all times. A command to test the status of it is:

```
> TEL STA
```

If it is not ´ON,´ the INACTIVE message will appear on the screen; if it is, the system returns to MCR mode without any further response.

```
>@DBØ:[1,54]LITER                                (invoke the program)
>*LITER[S]: XFER
>*IBMID[S]: MADDEN                                         (user ID)
>*DO YOU WISH A HEADER READ CARD?[Y/N]: N
>*FILE NAME[S]: TIPPE.DAT                           (statistics file)
>*IS THIS A BINARY FILE = [Y/N]: N                             (no)
>*PIP LITER.TMP/NV = LITER.TMP.TIPPE.DAT
  DB1:[261,6]TIPPE.DAT;
> TEL Q = LITER.TMP
TEL -- JOB SUCCESSFULLY QUEUED -- TELPRO IS * ACTIVE
> PIP LITER.TMP;*/DE
>*FILE NAME[S]: CAR RET                             (carriage return)
>*LITER[S]: Q                                                (quit)
>(back in MCR mode)
```

LITER simply puts the statistics file in the virtual card reader of IBM user ID specified. Logon the IBM, invoke CMS and issue the command:

```
READCARD *
```

Under CMS, this will read the statistics file onto the A-disk of the IBM user ID. The file will have the name CLUSTER STATS.