# GRASS Tutorial: Image Processing

*Victoria Harmon*
*Michael Shapiro*

U. S. Army Construction Engineering Research Laboratory

*ABSTRACT*

This tutorial guides the reader through a step by step procedure using the imagery commands available in the Geographic Resource Analysis Support System (GRASS). The procedure describes the GRASS image-processing programs in a sequence that is often used in image processing, starting with loading imagery data into GRASS, classifying the data, and rectifying it. The procedure is therefore linear, but does branch occasionally to describe alternate directions when alternate paths are available. During the tutorial the reader is encouraged to use his or her imagination and knowledge of GRASS to identify non-imagery GRASS programs that may be applied to raw digital imagery or classified maps.

## 1. INTRODUCTION

Within the last decade geographic information systems (GIS) and image-processing systems have undergone an evolutionary development. The challenges of the future require that these two technologies be integrated. The Geographic Resource Analysis Support System (GRASS) integrates these two technologies resulting in numerous advantages. Some of these advantages are described below.

A GIS has four components: a data input system, a data storage and retrieval system, a data manipulation and analysis system, and a data reporting system. The data inputs include spatial and thematic data derived from a combination of existing maps, aerial photographs, and manual interpretations of remotely sensed imagery. Remotely sensed imagery is imagery acquired at some distance using a camera or a sensor. With a GIS an analyst can define spatial procedures to generate new information, such as the best location to build a road, preserve a wildlife habitat, or harvest timber. A problem exists, however, in that the input data on which a GIS is based become obsolete quickly. Remote sensing data are an essential and cost-effective means to update the spatial and thematic information in a GIS. Remote sensing products are also valuable in themselves, as a cost effective source of geographic information such as vegetation type, vegetation productivity, archaeological feature location, water quality, and land use change.

An image-processing system has at least five elements: image input, image storage, image analysis, accuracy assessment, and information reporting. Note the similarity of these two systems:

| GIS | IMAGE PROCESSING |
|---|---|
| input | input |
| storage | storage |
| analysis | analysis |
| | accuracy assessment |
| reporting | reporting |

Only the input and analysis elements require different functions for each system. If the GIS is a raster-based system, the storage and reporting functions are identical. Some of the analysis functions may overlap. For example, the image accuracy assessment programs are actually standard GIS analysis routines. It seems natural then, that these two systems should be integrated.

When geographic information systems and image-processing systems are integrated, not only can the remote sensing data be used to update the GIS, but the GIS thematic data and attributes can be used to guide image classification. By integrating GIS and image-processing capabilities, GRASS offers the analyst these advantages.

This tutorial guides the reader through a step-by-step procedure using the imagery commands available in GRASS 4.0. Where applicable, non-imagery GRASS commands are also described. It is assumed that the reader has a knowledge of the basic concepts of GRASS such as its raster (grid cell) format, its map layers, LOCATION, MAPSET, and regions. For a description of these concepts, refer to *An Introduction to GRASS*.

## 2. THE IMAGERY COMMANDS

All of the GRASS 4.0 imagery commands begin with a lower case *i* followed by a dot. These programs are presented in this tutorial in a sequence that is often used in image processing, but this is not the only order in which they can be executed. Please refer to section 14, **A NOTE ABOUT PROGRAM SEQUENCE**, and section 15, **PATCHING IMAGES**. The imagery commands are described in this tutorial in the following sections.

### Section 7. Tape Extraction

*i.tape.mss.h*
*i.tape.mss*
*i.tape.tm*
*i.tape.other*

**Section 8. The Contrast Stretch**

*i.grey.scale*

**Section 9. The Color Composite**

*i.composite*

**Section 10. Imagery Groups**

*i.group*

**Section 11. Targeting Imagery Groups**

*i.target*

**Section 12. Image Classification**

*i.cluster*
*i.maxlik*

**Section 13. Image Rectification**

*i.points*
*i.rectify*

## 3. DESCRIPTIONS OF THE IMAGERY COMMANDS

Below, a brief description of each imagery command is provided. These descriptions are also available by running the command *g.manual* from within GRASS. Typing the word *g.manual* will display the manual entry for any GRASS program and give the user the option to print it out. All GRASS programs mentioned in this tutorial, other than the imagery programs, are referenced with their manual entry section in brackets, for example, *g.region*.

IMAGERY FUNCTION — DESCRIPTION

i.cluster – An imagery function that generates spectral signatures for land-cover types in an image using a clustering algorithm. The resulting signature file is used as input for *i.maxlik*.

i.composite – An imagery function that creates a color composite image from 3 band files specified by the user.

i.colors – An imagery function that creates colors for imagery groups. This program is not yet complete and is not covered in this tutorial. When it is completed it will replace *i.composite* and the color option in *i.group*.

i.grey.scale – An imagery function that assigns a histogram contrast stretch grey scale color table to a map layer.

i.group – An imagery function that creates and edits groups and subgroups of imagery files.

i.maxlik – An imagery function that classifies the pixel spectral reflectances in imagery data based on the spectral signature information generated in *i.cluster*.

i.points – An imagery function that enables the user to mark coordinate system points on an image to be rectified and then input the coordinates of each point for creation of a transformation matrix. The transformation matrix is needed as input for the GRASS program i.rectify.

i.rectify – An imagery function that rectifies an image by computing a coordinate transformation for each pixel in the image using the transformation coefficient matrix created by the GRASS program i.points.

i.tape.mss – An imagery function that extracts Multispectral Scanner Imagery from half-inch tape.

i.tape.mss.h – An imagery function that extracts header information from Landsat Multispectral Scanner imagery data stored on half-inch tape.

i.tape.other – An imagery function that extracts scanned aerial imagery (NHAP, etc.) and SPOT imagery from half-inch tape.

i.tape.tm – An imagery function that extracts Thematic Mapper imagery from half-inch tape.

i.target – An imagery function that establishes a GRASS target LOCATION for an imagery group.

## 4. WHAT IS DIGITAL IMAGERY?

To perform digital image processing of remotely sensed imagery the imagery must be in a digital format. Imagery that is initially in analog format (such as an aerial photograph) can be converted to digital format in a digitization process called optical-mechanical scanning. Film density of the blue, the green, and the red emulsion layers is converted to a digital number (DN) that ranges from 0 to 255 for the chosen sampling interval.[1] The DN value 0 can be assigned to represent the lack of reflected or emitted electromagnetic radiation and the value 255 to represent maximum reflectance. An example of a sampling interval is 100 micrometers. The resulting digital image has three bands: one for the blue emulsion layer, one for the green emulsion layer, and one for the red emulsion layer. The sampling interval becomes one 8-bit pixel in the digital image which is stored in raster format. Raster format is a two-dimensional matrix composed of cells in which each cell has the DN integer value. The DNs of the digital image (representing film density) closely approximate the relative radiance that is reflected and emitted from the features on the ground within the sampling interval.

The resolution of the image is a function of the sampling interval and the scale of the diapositive being scanned. If the sampling interval is 100 micrometers and the scale of the aerial photo is 1:50,000, then the resolution is 5 meters:

100 micrometers * 50000 = 5 meters

Other types of remotely sensed imagery are stored directly in digital format as they are collected. The storage medium is usually half-inch magnetic tape in raster format. (Video imagery is also stored on half-inch tape, but it is stored in a special video format. At present there are no imagery commands in GRASS to read video data stored on video tape.) Examples of these other types of remotely sensed imagery are: Landsat Multispectral Scanner data (MSS), Landsat Thematic Mapper data (TM), and Systems Probatoir d'Observation de la Terre data (SPOT), to name a few. Each of these types of data are satellite data. Each has a specific resolution and an area of coverage that is a product of the sensor, the height of the sensor above the earth, and the focal length. Like aerial imagery, the digital numbers for these data also range from 0 to 255 and represent the reflected and emitted radiation from features on the ground.

The imagery programs in GRASS are able to read and analyze both digitized aerial imagery and satellite data. The program sequence for reading and analyzing these types of digital imagery is described in the following sections.

## 5. THE IMAGERY LOCATION

To use the imagery programs, GRASS 4.0 must be invoked by typing:

**grass4.0**

[1] Imagery data is usually 0 to 255 but GRASS is not restricted to this data range.

After typing this command and hitting <RETURN>, a screen is displayed that requires you to choose a GRASS LOCATION and MAPSET.

Before choosing a LOCATION in which to extract imagery, it is important to realize that raw digital imagery on tape is not geo-referenced with a standard coordinate system such as a UTM coordinate system or a State Plane coordinate system when it is captured or digitized. The imagery on tape is referenced using the rows and columns of the format in which it is stored. Therefore, for image tape extraction, a GRASS LOCATION must be chosen that has an x,y (row, column) coordinate system. The 4.0 version of GRASS comes with an x,y coordinate LOCATION named *imagery*. You can choose this LOCATION for imagery extraction, or you can create your own x,y coordinate LOCATION. You may want to create an x,y LOCATION for every image or collection of images that you extract, and/or you may want to use the x,y LOCATION, *imagery*, for general imagery tape extraction.

**Note:** Only the *i.tape* commands must be run in a LOCATION with an x,y coordinate system. All of the other GRASS imagery commands can be run in a LOCATION having any GRASS coordinate system.

### 5.1. Creating an X,Y Coordinate LOCATION

The procedure to create a LOCATION with an x,y coordinate system in GRASS is the same as that for creating any LOCATION in GRASS, however, there are a few differences that merit special attention.

The first step is the same for any LOCATION: enter the new LOCATION name and respond affirmatively to the question of whether you would like to create a new LOCATION. (While reading this tutorial, you may want to follow along, implementing some of the commands, using imagery of Spearfish, South Dakota, if available, that will eventually reside in the UTM coordinate LOCATION, **spearfish**. **spearfish** is the sample database for GRASS. Therefore, for demonstration purposes, we will name the new x,y LOCATION *spear_xy*.)

In the second step, you will be asked whether you have the information to create a new LOCATION. Answer yes to this question too. Unlike creating a UTM or State Plane LOCATION, you will not need to have available information from maps or other reference materials.

Next, you will be asked to specify the coordinate system for the LOCATION:

**Please enter the following coordinate system information for location** <*spear_xy*>

**0 x,y**
**1 UTM**
**2 State Plane**
>

The code number for an x,y coordinate system in GRASS is **0**. The code number for the UTM coordinate system in GRASS is **1**, and the code number for the State Plane coordinate system is **2**. Enter **0**,

for the x,y coordinate system.

Then, the following screen will be displayed:

**Please enter the following information for location** <*spear_xy*>

**TITLE:**

**DEFAULT REGION**

| | | |
|---|---|---|
| | **NORTH EDGE:0.00** | |
| **WEST EDGE 0.00** | | **EAST EDGE 0.00** |
| | **SOUTH EDGE:0.00** | |
| | **GRID RESOLUTION** | |
| | **East-West:0.00** | |
| | **North-South:0.00** | |

**AFTER COMPLETING ALL ANSWERS HIT <ESC> TO CONTINUE**
**(OR <Ctrl-C> TO CANCEL)**

When imagery data are extracted from tape into an x,y coordinate LOCATION in GRASS 4.0, the map-set region setting will be changed to match the extracted rows and columns from the tape. The initial start-up coordinates, then, are of little significance, except as a default region for the x,y LOCATION, which is a requirement for every LOCATION in GRASS. Any general coordinates, then, may be entered as the default region, as long as:

1 The north and south coordinates are negative;

2 All of the coordinates are offset by .5: the northern edge is 0.5 less than the chosen first row, the southern edge is 0.5 greater than the chosen last row, the west edge is 0.5 less than the chosen first column, and the east edge is 0.5 larger than the chosen last column;

3 The north coordinate is greater than the south coordinate; and

4 The east coordinate is greater than the west.

The reason the north and south coordinates are negative is because GRASS requires that the region values decrease from north to south but the row coordinates on the tape increase from north to south. Assigning negative values to the rows allows them to decrease from north to south and, by ignoring the minus signs, the correct row numbers are reported.

By setting the northern edge to 0.5 less than the first row, the southern edge to 0.5 larger than the last row, the west edge to 0.5 less than the first column, and the east edge to 0.5 larger than the last column, the x,y coordinates (as integers 1 to n) become located at the center of the pixel and are then the same as the row and column numbers on the tape.

### 5.2. Sample X,Y Default Region

An example of a valid x,y coordinate default region would be:

**DEFAULT REGION**

| | | |
|---|---|---|
| | **NORTH EDGE:-0.50** | |
| **WEST EDGE 0.50** | | **EAST EDGE 1000.50** |
| | **SOUTH EDGE:-1000.50** | |
| | **GRID RESOLUTION**<br>**East-West:1.00**<br>**North-South:1.00** | |

**AFTER COMPLETING ALL ANSWERS HIT <ESC> TO CONTINUE**
**(OR <Ctrl-C> TO CANCEL)**

These coordinates can be used to create any x,y LOCATION.

## 6. REGIONS AND MASKS

All of the tape extraction programs (*i.tape.mss, i.tape.tm*, and *i.tape.other*) set the mapset region to the row and column coordinates of the image or image subset being extracted from the tape. If more than one image or image subset is extracted into the same x,y LOCATION, and the images or image subsets have different regions (which are based on the extracted rows and columns), then, to display one of the image raster files in the x,y LOCATION, you will have to check the current MAPSET region to make sure that it matches the raster file to be displayed. A way to do this is to use the GRASS program *g.region* to see the current region setting, and to modify the current region setting. Use the GRASS program *g.region* to see the current region setting, and to modify the current region setting.

The presence of a mask in an x,y LOCATION also requires special attention. If a mask is set to a band raster file of one image in an x,y LOCATION and the user wishes to display a different image or band raster file in the same LOCATION, then the mask will effect the displayed raster file as well, even if the mapset region is set to the raster file you wish to display. The result may be the display of a raster file composed of zeros. To avoid this problem, check to see if there is a mask by typing **r.mask**. See the manual entry for *r.mask* for further details.

## 7. TAPE EXTRACTION

To load or read digital imagery into GRASS, four commands are available:

*i.tape.mss.h*
*i.tape.mss*
*i.tape.tm*
*i.tape.other*

If the image is an MSS image the command *i.tape.mss.h* can be used to read the header on the tape. The command *i.tape.mss* is used to extract an MSS image or a portion of an MSS image from tape. If the image is a TM image, the command *i.tape.tm* is used to extract the image. If the image is a scanned aerial image, NHAP image, NAPP image, SPOT image, or some other imagery type stored on half-inch tape (except for video format), the program *i.tape.other* is used to extract the data from tape. This section explains how to use each of these programs.

### 7.1. Extracting MSS Imagery From Tape

**Note:** Please become familiar with section 6 **REGIONS AND MASKS** before running a tape extrac-

tion.

### 7.1.1. i.tape.mss.h

This program is an imagery function that extracts header information from Landsat Multispectral Scanner imagery stored on half inch tape. The header contains the center latitude and longitude for the image and the number of rows and columns in the scene. This information may be of use in locating the portion of the MSS image of interest, if the whole image is not required. If you are not familiar with MSS imagery, however, the header information format may be difficult to understand. This program is not required prior to running *i.tape.mss*.

To run the program, first mount the tape on the tape drive for the machine you are using and, at the GRASS prompt, type:[2]

**i.tape.mss.h** *tape_drive_name*

The tape drive name will depend on the system being used but it is *sometimes* of the form /dev/rmt0. If so, this command would then become:

**i.tape.mss.h /dev/rmt0**

If entered like this, the tape header information will be displayed on the screen. To read the header information into a file (thereby saving it for future use), the output can be redirected using the command:

**i.tape.mss.h /dev/rmt0** > *headerfile*

The user-specified *headerfile* will contain the header information that occurs on the tape.

### 7.1.2. i.tape.mss

This program is an imagery function that extracts Multispectral Scanner (MSS) Imagery from half inch tape.

**Note:** This program must be run in a LOCATION with an x,y coordinate system (see section 5, **THE IMAGERY LOCATION**).

Before running a tape extraction, it is a good idea to check to see if there is enough disk space in which to store the files to be extracted on your machine. The number of bytes in an MSS image can be calculated by multiplying the number of rows in the image by the number of columns in the image and then multiplying that number by the number of bands to be extracted. Determining the number of rows and columns in an image is explained in subsection 7.1.2.2., **Row and Column Extraction**.

To run *i.tape.mss*, at the GRASS prompt, type:[3]

**i.tape.mss**

The first prompt in *i.tape.mss* asks for the tape device name. This is *sometimes* /dev/rmt0 (for a tape density of 1600), but this varies with each machine.

The next prompt is:

**Please mount and load tape, then hit RETURN -->**

---

[2] MASSCOMP users may first have to assign the tape drive the appropriate density. Frequently, the density of the data stored on tape is 1600 bpi (bytes per inch). The relevant command to assign the tape drive is then usually **assign 1600** or **assign 1600x**.

[3] MASSCOMP users may first have to assign the tape drive the appropriate density. Frequently, the density of the data stored on tape is 1600 bpi (bytes per inch). The relevant command to assign the tape drive is then usually **assign 1600** or **assign 1600x**.

#### 7.1.2.1. MSS Identification Screen

After hitting RETURN the first screen in the program asks for information about the data.

| **please enter the following information** |
|---|
| **Tape Identification:** |
| **Image Description:** |
| **Title for the Extracted Cell Files:** |
| **AFTER COMPLETING ALL ANSWERS, HIT <ESC> TO CONTINUE (OR <Ctrl-C> TO CANCEL)** |

Using information from the tape header, the program automatically enters the scene ID number and the date of the image into the field for Tape Identification. The sun angles are automatically entered into the field for Image Description. This information can be edited by the user, but, left untouched, it is accurate and informative. The user should enter other pertinent information and the title for the extracted raster files. The tape identification and the image description are entered automatically into the history files for the raster files by GRASS.

#### 7.1.2.2. Row and Column Extraction

The second screen is:

| **MSS TAPE EXTRACTION**<br>**please select the desired tape region to extract** |
|---|
| **first row:_____(1-2984)**<br>**last row:_____(1-2984)** |
| **first col:_____(1-3548)**<br>**last col:_____(1-3548)** |
| **AFTER COMPLETING ALL ANSWERS, HIT <ESC> TO CONTINUE (OR <Ctrl-C> TO CANCEL)** |

The numbers in parentheses are the total numbers of rows and columns including filler (zeros) on the tape, which is obtained from the tape. This information and additional information can be obtained by running the *i.tape.mss.h* program, which reads the header information on an MSS tape. The MSS header contains the scene center latitude and longitude and the number of rows and columns in the scene. The *m.examine.tape* program will also read any tape and provide the number of files on a tape, the number of records on a tape, and the record lengths. The number of records in a file is the number of rows in a file, and the record length is the number of columns.

Any subset of the image on the tape can be extracted. To find the row and column numbers needed to identify a subset of an MSS image, several sources of information are helpful:

1 The information from the tape header

2 The corner point latitudes and longitudes of the image from the Landsat search printout

3 A diagram of the position of the 7.5-minute and 15-minute maps of the image

4 The 7.5- and 15-minute maps of the image

5 A photograph of the image if available

6 A printout describing information about the image if it comes with the tape

GRASS display commands are also helpful. If enough disk space is available, one band of an entire image or, one band of a portion of an image known to contain the area of interest, can be extracted and displayed.

The x,y coordinates of the displayed image can be echoed to the screen using the *measurements* option in **d.display**, or the **d.where** command (following the use of **d.rast**). (These coordinates will display negative numbers in the north-south direction but, by ignoring the negative sign, the correct row number is reported.)

To find the area of interest using these display options in GRASS, the features displayed on the monitor can be found on the maps by locating on the maps the known coordinates of the image (latitude and longitude) and looking for the displayed physical features. If the latitude and longitude coordinates contain many maps within their bounds (for example, many 1:24000 maps), one method to help locate the area of interest is described as follows:

1 Draw an approximate square or rectangle representing the entire image stored on tape. Identify the known latitude and longitude corner points (and center point if known) and mark these on the square. Section the "square" into a grid and approximate the latitude and longitude for each grid intersection.

2 Based on the known latitude and longitude corner points for the image, gather together all of the maps (of an appropriate scale) that are contained within the lat/long. Draw another square and section it into a grid in which each cell represents a map. Mark the latitude and longitude of each intersection.

3 Locate your area of interest on the square that represents the grid of maps (step 2).

4 By looking at the lat/long on the "image square", and the lat/long on the "map square", you can determine in what grid cells of the image the features you are interested in lie. You can then zoom-in to the appropriate grid cell equivalents of the image on the display monitor and try to match the features displayed on the monitor with those on the map.

5 Once you have found your area of interest on the image, you can use *d.where* or the *measurements* option in *d.display* to determine the row and column numbers for the four corners of the area. These values are the first row, last row, first column, and last column to be entered in the tape extraction screen.

If a photograph of the digital image is available, the rows and columns to be extracted can also be determined from it by associating inches with the total number of known rows and columns in the scene. For example, if the total length of the photograph is 12 inches, the total number of rows on the tape is 2000, and the northwest corner of the area of interest begins 2 inches from the top of the photo, then:

$$12" / 2000 \text{ rows} = 2" / x \text{ rows}$$
$$x = 333.333$$

The northwest corner of the area of interest starts at row 333. The starting row, ending row, starting column, and ending column can be calculated in this manner.

#### 7.1.2.3. MSS Band Extraction

The next screen is:

**please make an x by the bands you want extracted**

**_____ 1**
**_____ 2**
**_____ 3**
**_____ 4**

**AFTER COMPLETING ALL ANSWERS, HIT <ESC> TO CONTINUE**
**(OR <Ctrl-C> TO CANCEL)**

MSS imagery has 4 bands but you may want to extract only a subset of these bands. Since all of the bands can be extracted from tape now and gathered together in different combinations for different applications later using the command *i.group*, reasons for extracting fewer bands might be:

1 For initial identification of the area of interest; extract one band first to locate the area of interest (see subsection 7.1.2.2. **Row and Column Extraction**), or

2 For application purposes; the application requires only the use of some of the bands.

#### 7.1.2.4. Prefix/Group Name

You are then asked to enter the prefix/group for the band raster files to be created. This name will precede each band file extracted into GRASS. For example, if 3 bands are extracted the following 3 band files will result:

*prefixname.1*
*prefixname.2*
*prefixname.3*

The specified *prefixname* will also automatically become the name for the imagery group file being created. Each MSS image (i.e., each MSS tape set) should be given a unique prefix/group name. Any number of image subsets from one MSS image or tape set can be extracted into the same imagery group (same prefix/group name), even if they all have different row/column coordinates. Only different MSS images (i.e., different MSS tape sets) should be extracted into different imagery groups. The reason for this is that the purpose of an imagery group is to identify raster files for classification and rectification, and classification and rectification properties vary from MSS tape set to MSS tape set.

#### 7.1.2.5. The Extraction Process

The extraction process will begin by first skipping the number of specified files, advancing to the starting row, and then reading the tape. The percent completion of the extraction is displayed on the screen. If more than one tape is required to store the image, the program will pause and inform you to mount the next tape.

**Note:** After extracting an image from tape, the mapset region in the x,y coordinate LOCATION will be set based upon the extracted rows and columns from the tape (see section 5 **THE IMAGERY LOCATION**, subsection 5.1. **Creating an X,Y Coordinate LOCATION**, and section 6, **REGIONS AND MASKS**).

#### 7.1.2.6. MSS Output

The extracted band files will be listed as raster files available in the current MAPSET and may be displayed using either the GRASS command *d.display*, *d.rast* or *i.points*.

**Note:** If more than one image or image subset is extracted into the same x,y LOCATION, and the images or image subsets have different mapset regions (which are based on the rows and columns

extracted), then, to display a raster file in the LOCATION, you will have to check the current MAPSET region to make sure that it matches the raster file to be displayed. For a detailed explanation see section 6, **REGIONS AND MASKS**.

### 7.1.3. About Patching

When more than one MSS image is needed (and therefore extracted) in order to include an entire area of interest for an application, at some point in the image processing sequence you may want to unite the images. In GRASS 4.0 the program to accomplish this is *r.patch*. See section 15, **PATCHING IMAGES** for information about when and how to patch.

## 7.2. Extracting TM Imagery From Tape

**Note:** Please become familiar with section 6, **REGIONS AND MASKS** before running a tape extraction.

This program must be run in a LOCATION with an x,y coordinate system (for further information see section 5, **THE IMAGERY LOCATION**).

*i.tape.tm* is the GRASS program that extracts Thematic Mapper (TM) imagery from tape. When a TM image is captured, it is stored in four quads. If all four quads are present, they will be juxtaposed on the image as follows:

| | |
|---|---|
| **1** | **2** |
| **3** | **4** |

Each quad is stored on tape as a separate image, and therefore, each quad must be extracted separately. Each quad will require a new run of *i.tape.tm*.

Before running a tape extraction, it is a good idea to check to see if there is enough disk space for the extracted files on your machine. The number of bytes in a TM quad can be calculated by multiplying the number of rows in the quad by the number of columns in the quad and then multiplying that number by the number of bands to be extracted. Determining the number of rows and columns in a TM quad is explained in subsection 7.2.2., **Row and Column Extraction**.

To begin the program, at the GRASS prompt type:[4]

**i.tape.tm**

The first prompt in *i.tape.tm* asks for the tape device name. This is *sometimes* /dev/rmt0 (for a density of 1600), but this varies with each machine.

The next prompt is:

**Please mount and load tape, then hit RETURN -->**

### 7.2.1. TM Identification Screen

[4] MASSCOMP users may first have to assign the tape drive the appropriate density. Frequently, the density of the data stored on tape is 1600 bpi (bytes per inch). The relevant command to assign the tape drive is then usually *assign 1600* or *assign 1600x*.

After hitting RETURN the first screen in the program asks for information about the data.

**please enter the following information**

**Tape Identification:**

**Image Description:**

**Title for the Extracted Cell Files:**

**AFTER COMPLETING ALL ANSWERS, HIT <ESC> TO CONTINUE**
**(OR <Ctrl-C> TO CANCEL)**

The program automatically enters the scene ID number into the field for Tape Identification. The mission, path, row, quadrant, date, and whether the image is corrected is automatically entered into the field for Image Description. This information can be edited by the user but, left untouched, it is accurate and informative. The user should enter other pertinent information and the title for the extracted raster files. The tape identification and the image description are entered automatically into the history files for the raster files by GRASS.

### 7.2.2. Row and Column Extraction

The second screen is:

**THEMATIC MAPPER EXTRACT**
**please select the desired tape region to extract**

**first row: _____(1-2984)**
**last row: _____(1-2984)**

**first col: _____(1-4220)**
**last col: _____(1-4220)**

**AFTER COMPLETING ALL ANSWERS, HIT <ESC> TO CONTINUE**
**(OR <Ctrl-C> TO CANCEL)**

The numbers in parentheses are the total number of rows and columns containing image data and filler (zeros) on the tape. The last row of each quad contains zeros. Quads 1 and 2 contain 2983 rows of image data, and quads 3 and 4 contain 2982 rows of image data. It is best not to extract the last row (row 2984 on Quads 1 and 2 and row 2983 on Quads 3 and 4) because, if Quad 1 is patched to Quad 3 or Quad 2 is patched to Quad 4, there will result a line of no data running east and west through the image. If TM quads are to be patched side to side, you will also want to avoid extracting the zeros on the left and right sides of each quad. Zeros, of course, will also take up unnecessary disk space. Methods to identify the row and column coordinates to be extracted from a TM quad are described below.

Any subset of the quad on the tape can be extracted. To find the row and column numbers needed to identify a subset of a TM quad, several sources of information are helpful:

1. The center point and corner point latitudes and longitudes of the image from the Landsat search printout
2. A diagram of the position of the 7.5-minute and 15-minute maps of the image relative to the TM quads

3 The 7.5- and 15-minute maps of the quad

4 A photograph of the image if available

5 A printout describing information about the image if it comes with the tape

GRASS display commands are also helpful. If enough disk space is available, one band of an entire quad or one band of a portion of a quad known to contain the area of interest, can be extracted and displayed. Running the *measurements* option in *d.display* or *d.where* (following the use of *d.rast*) will echo the x,y coordinates of the displayed quad on the screen. (These coordinates will display negative numbers in the north-south direction but, by ignoring the negative sign, the correct row number is reported.)

To find the area of interest using these display options in GRASS, the features displayed on the monitor can be found on the maps by locating on the maps the known coordinates of the quad (latitude and longitude) and looking for the displayed physical features. If the latitude and longitude coordinates contain many maps within their bounds (for example, many 1:24000 maps), one method to help locate the area of interest is described below.

1 Draw an approximate square or rectangle representing the entire quad stored on tape. Identify the known latitude and longitude corner points (using the center point for the TM image) and mark these on the square. Section the square into a grid and approximate the latitude and longitude for each grid intersection.

2 Based on the known latitude and longitude for the quad, gather together all of the maps (of an appropriate scale) that are contained within the lat/long. Draw another square, and section it into a grid in which each cell represents a map. Mark the latitude and longitude of each intersection.

3 Locate your area of interest on the square that represents the grid of maps (step 2).

4 By looking at the lat/long on the "quad square," and the lat/long on the "map square," you can determine in what grid cells of the quad the features you are interested in lie. You can then zoom-in to the appropriate grid cell equivalents of the quad on the display monitor and try to match the features displayed on the monitor with those on the map.

5 Once you have found your area of interest on the displayed quad, you can use **d.where** or the *measurements* option in **d.display** to determine the row and column number for the four corners of the area. These values are the first row, last row, first column, and last column to be entered in the tape extraction screen.

If a photograph of the TM image is available, the rows and columns to be extracted can also be determined from it by associating inches with the total number of known rows and columns in the scene. For example, if the total length of the photograph is 12 inches, the total number of rows on the tape is 2000, and the northwest corner of the area of interest begins 2 inches from the top of the photo, then:

$$12" / 2000 \text{ rows} = 2" / x \text{ rows}$$
$$x = 333.333$$

The northwest corner of the area of interest starts at row 333. The starting row, ending row, starting column, and ending column can be calculated in this manner.

The next screen is:

```
please make an x by the bands you want extracted

_____ 1
_____ 2
_____ 3
_____ 4
_____ 5
_____ 6
_____ 7

AFTER COMPLETING ALL ANSWERS, HIT <ESC> TO CONTINUE
(OR <Ctrl-C> TO CANCEL)
```

TM imagery has 7 bands but you may want to extract only a subset of these bands. Since all of the bands can be extracted from tape now and gathered together in different combinations for different applications later using the command *i.group*, reasons for extracting fewer bands might be:

1 For initial identification of the area of interest; extract one band first to locate the area of interest, see subsection 7.2.2., **Row and Column Extraction**), or

2 For application purposes; the application requires only the use of some of the bands.

### 7.2.3. Prefix/Group Name

You are then asked to enter the prefix/group for the band raster files to be created. This name will precede each band file extracted into GRASS. For example, if 3 bands are extracted the following 3 band files will result:

*prefixname.1*
*prefixname.2*
*prefixname.3*

The specified *prefixname* will also automatically become the name for the imagery group file being created. Each TM quad (i.e., each quad tape set) should be given a unique prefix/group name. Any number of image subsets from one TM quad can be extracted into the same imagery group (the same prefix/group name), even if they all have different row/column coordinates. Only different TM quads (i.e., different quad tape sets) should be extracted into different imagery groups. The reason for this is that the purpose of an imagery group is to identify raster files for classification and rectification, and classification and rectification properties vary from quad tape set to quad tape set.

Examples of prefix/group names for a TM quad might be 37quad1, 37quad2, 37quad3, and 37quad4. The word "quad" indicates that it is a TM image. The number "37" is the TM path or row, whichever might distinguish it from another TM image, if more than one TM image of the same date is extracted into the same x,y LOCATION. The numbers 1, 2, 3, and 4 identify the quad of the image. All quads that will be patched should be extracted into the same x,y LOCATION (see 6.2.6., **About Patching**).

### 7.2.4. The TM Extraction Process

The extraction process will begin by first skipping the number of specified files, advancing to the first band requested, and then reading the tape. After extracting the requested rows and columns for each band, the program creates support files for each band. The percent completion of the extraction is displayed on the screen. Because TM imagery is divided into 4 quads, it is stored in multiple tape sets. The number of tapes required to store one quad depends on the number of bytes per inch (bpi) in which the data are stored. If more than one tape is required to store one quad, the program will pause and inform you to mount the next tape.

**Note:** After extracting an image from tape the mapset region in the x,y coordinate LOCATION will be set based upon the extracted rows and columns from the tape (see section 5, **THE IMAGERY LOCATION**; subsection 5.1., **Creating an X,Y Coordinate LOCATION**; and section 6, **REGIONS AND MASKS**).

#### 7.2.5. TM Output

The extracted band files will be listed as raster files available in the current MAPSET and may be displayed using the GRASS commands **d.display**, **d.rast**, or **i.points**.

**Note:** If more than one image or image subset is extracted into the same x,y LOCATION, regardless of whether they are extracted into the same group, and if the images or image subsets have different mapset regions (which are based on the rows and columns extracted), then, to display a raster file in the x,y LOCATION, you will have to check the current MAPSET region to make sure that it matches the raster file to be displayed. For a more detailed explanation see section 6, **REGIONS AND MASKS**.

#### 7.2.6. About Patching

As was mentioned earlier, each quad requires a separate *i.tape.tm* extraction and each quad should be given a separate prefix/group name (imagery group). When more than one quad is extracted, to include an entire area of interest for an application, at some point in the image-processing sequence you may want to unite the quads. In GRASS 4.0, the program to accomplish this is *r.patch*. See section 15, **PATCHING IMAGES** for information about when and how to patch.

### 7.3. Extracting Other Types of Digital Imagery From Tape

**Note:** Please become familiar with section 6, **REGIONS AND MASKS** before running a tape extraction.

This program must be run in a LOCATION with an x,y coordinate system (for further information, see section 5, **THE IMAGERY LOCATION**.

*i.tape.other* is the GRASS program that extracts all other types of digital imagery from half-inch tape (other than MSS imagery and TM imagery). This program is a generic program that extracts imagery using the tape description that is input by the user. *i.tape.other* cannot extract images that continue from one tape to another utilizing the same header. It can only extract images that are self contained on one tape.

**Note: i.tape.other** must be run in a LOCATION with an x,y coordinate system (see section 5, **THE IMAGERY LOCATION**).

Before running a tape extraction, it is a good idea to check to see if there is enough disk space for the extracted files on your machine. The number of bytes in an image can be calculated by multiplying the number of rows in the image by the number of columns in the image and then multiplying that number by the number of bands to be extracted. For information on how to determine the number of rows and columns in an image, refer to subsection 7.3.5., **Generic Row and Column Screen**.

To begin *i.tape.other*, at the GRASS prompt type:[5]

**i.tape.other**

The first prompt in *i.tape.other* asks for the tape device name. This is *sometimes* /dev/rmt0 (for a density of 1600), but this varies with each machine.

---

[5] MASSCOMP users may first have to assign the tape drive the appropriate density. Frequently, the density of the data stored on tape is 1600 bpi (bytes per inch). The relevant command to assign the tape drive is then usually **assign 1600** or **assign 1600x**.

The next prompt is:

**Please mount and load tape, then hit RETURN -->**

### 7.3.1. Image Identification Screen

After hitting RETURN the first screen in the program asks for information about the data.

```
please enter the following information

Tape Identification:

Image Description:

Title for the Extracted Cell Files:

AFTER COMPLETING ALL ANSWERS, HIT <ESC> TO CONTINUE
(OR <Ctrl-C> TO CANCEL)
```

The user should enter information here to identify the image and, therefore, the extracted raster files. This information will be entered into the support history files for the band raster files automatically by GRASS.

### 7.3.2. Tape Layout Screen

The next screen asks for the physical layout of the tape.

```
GENERIC TAPE EXTRACTION

tape layout
    _0_     number of tape files to be skipped
    _0_     number of records in the remaining files to be skipped
band files
    _0_     number of bands on the tape
data format
    ___     band sequential (BSQ)  |  mark one with an x
    ___     band interleaved (BIL) |
    _0_     if you select BSQ format and all the bands are in a single file
            enter the total number of records in the file. Otherwise enter 0

    _0_     length (in bytes) of the longest record on the tape

AFTER COMPLETING ALL ANSWERS, HIT <ESC> TO CONTINUE
(OR <Ctrl-C> TO CANCEL)
```

The fields in the above screen are explained as follows:

*number of tape files to be skipped*

If there are files at the beginning of the tape which are not image data, they can be skipped. Sometimes information that comes with a tape will indicate the number of header files or records on the tape. The GRASS utility *m.examine.tape* will also provide this information. If a header is present on the tape there will be one or more small files at the beginning of the tape, containing only a small number of records. In contrast, the image file or files are large, containing hundreds or thousands of records. The number of records in an image file depends on the number of bands

in the image, the size of the image, and whether the image is stored in BIL or BSQ format.

NHAP imagery and usually most scanned aerial imagery do not have tape header files but this should be checked. Sometimes this information is written on the outside of the tape.

SPOT imagery, when it comes in a two-tape set, has two files that should be skipped on the first tape, and one file that should be skipped on the second. The number of header files on each tape is available in the printout that accompanies each SPOT tape set.

*number of records in the remaining files to be skipped*

If the file(s) which contain the image begin with non-image data, these records can also be skipped. This is usually 0 for most data types. For SPOT imagery stored in 1600 bpi, there is one header record that should be skipped in the image file on each tape (see subsection 7.3.6, **A Word About SPOT**).

*number of bands on the tape*

Most aerial imagery has 3 bands, but satellite simulator data may have more. SPOT has 3 bands as a standard. The total number of bands on the tape should be specified here, not just the number that will be extracted.

*data format*

The two formats that imagery data are most commonly stored in are band interleaved format (BIL) and band sequential format (BSQ). In BIL format each record on the tape contains one line for one band of data. If the data have three bands, then the first five records will look like this:

band 1, line 1
band 2, line 1
band 3, line 1
band 1, line 2
band 2, line 2

In BSQ format, all lines of one band are stored together on a tape followed by all lines of another band, followed by all lines of the next band, etc. These data are stored as if they were in a one band BIL format:

band 1, line 1
band 1, line 2
band 1, line 3
.
.
.
band 2, line 1
band 2, line 2
.
.
band 2, line 156
band 2, line 157

Each pixel contains one byte and there is one line per record. BSQ format is the format that is usually created by optical scanning devices when they scan photographs, but not all digitized aerial imagery are stored in this format. The format of the data is usually written on the exterior of the tape and this should be checked.

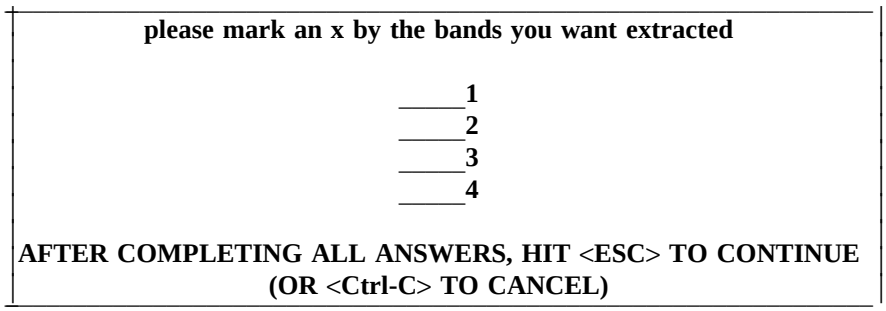
*length (in bytes) of the longest record on the tape*

The length in bytes of the longest record on the tape must be entered here. It is used to determine how large a buffer to use for reading the tape. This value can be obtained using

*m.examine.tape.*

### 7.3.3. Generic Band Extraction Screen

You are then asked to mark an **x** by the bands to be extracted.




SPOT imagery and most aerial imagery have 3 bands but you may want to extract only a subset of these bands. Since all of the bands can be extracted from tape now and gathered together in different combinations for different applications later using the command *i.group*, reasons for extracting fewer bands might be:

1 For initial identification of the area of interest; extract one band first to locate the area of interest (see subsection 7.3.5., **Generic Row and Column Screen**), or

2 For application purposes; the application requires only the use of some of the bands.

### 7.3.4. Prefix/Group Name

Next, you are asked to enter the prefix/group for the band raster files to be created. This name will precede each band file extracted into GRASS. For example, if 3 bands are extracted the following 3 band files will result:

*prefixname.1*
*prefixname.2*
*prefixname.3*

The specified *prefixname* will also automatically become the name for the imagery group file being created. Each SPOT image or each NHAP image (i.e., each individual tape) should be given a unique prefix/group name. Any number of image subsets from one image or tape can be extracted into the same imagery group (same prefix/group name), even if they all have different row/column coordinates. But, *different* images (i.e., different tapes) should be extracted into different imagery groups. This is because the purpose of an imagery group is to identify raster files for classification and rectification, and classification and rectification properties vary from tape to tape.

An example of a prefix/group name for an NHAP image might be NHAP_686, where the number "686" is the month and year of the NHAP image. An example of a prefix/group name for a SPOT image might be 271spot1, where the number "271" is the scene column (K) or row (J) of the image, and the number "1" stands for the first tape of a two-tape set.

### 7.3.5. Generic Row and Column Screen

Finally, the starting row, ending row, starting column, and ending column are required. This allows you to extract any subset from the tape. The generic row and column extraction screen looks like this.

**EXTRACT**
**please select desired tape region to extract**

**start row:_0_**
**end row:_0_**

**start col:_0_**
**end col:_0_**

**AFTER COMPLETING ALL ANSWERS, HIT <ESC> TO CONTINUE**
**(OR <Ctrl-C> TO CANCEL)**

To find the row and column numbers needed to identify a subset of an image, several sources of information are helpful:

1 Information regarding the number of files on the tape, the number of records in each file, and the record lengths. The record length is the number of columns in the image file and the number of records is the number of rows in the image file. The GRASS program *m.examine.tape* will provide this information as well as the presence or absence of a header record in the image file (see subsection 7.3.6., **A Word About SPOT**.

2 A diagram of the position of the 7.5-minute and 15-minute maps of the image.

3 The 7.5- and 15-minute maps of the image.

4 A photograph or diapositive of the image if available

5 A printout describing information about the image if it comes with the tape

The GRASS display commands are also helpful. If enough disk space is available, one band of an entire image or one band of a portion of an image known to contain the area of interest, can be extracted and displayed. Running either the *measurements* option in *d.display*, or *d.where* (following the use of *d.rast*) will echo the x,y coordinates of the displayed image to the screen. (These coordinates will display negative numbers in the north-south direction but, by ignoring the negative sign, the correct row number is reported.)

To find the area of interest using these display options in GRASS, the features displayed on the monitor can be found on the maps by locating on the maps the known coordinates of the image (latitude and longitude, UTM, or State Plane) and looking for the displayed physical features. If the latitude and longitude coordinates contain many maps within their bounds (for example, many 1:24000 maps), one method to help locate the area of interest is described as follows:

1 Draw an approximate square or rectangle representing the entire image stored on tape. Identify the known latitude and longitude corner points (and center point if known) and mark these on the square. Section the "square" into a grid and approximate the latitude and longitude for each grid intersection.

2 Based on the known latitude and longitude corner points for the image, gather together all of the maps (of an appropriate scale) that are contained within the lat/long. Draw another square and section it into a grid in which each cell represents a map. Mark the latitude and longitude of each intersection.

3 Locate your area of interest on the square that represents the grid of maps (step 2).

4 By looking at the lat/long on the "image square," and the lat/long on the "map square," you can determine in what grid cells of the image the features you are interested in lie. You can then zoom-in to the appropriate grid cell equivalents of the image on the display monitor and try to match the features displayed on the monitor with those on the map.

5 Once you have found your area of interest on the image, you can use *d.where* or the *measurements* option in *d.display* to determine the row and column number for the four corners of the area. These values are the start row, end row, start column, and end column to be

entered in the tape extraction screen.

If a photograph of the digital image is available, or the diapositive that was scanned, the rows and columns to be extracted can also be determined from it by associating inches with the total number of known rows and columns in the scene. For example, if the total length of the photograph is 12 inches, the total number of rows on the tape is 2000, and the northwest corner of the area of interest begins 2 inches from the top of the photo, then:

$$12" / 2000 \text{ rows} = 2" / x \text{ rows}$$
$$x = 333.333$$

The northwest corner of the area of interest starts at row 333. The starting row, ending row, starting column, and ending column can be calculated in this manner.

### 7.3.6. A Word About SPOT

Since SPOT imagery is used quite often and has a consistent format, this section will offer specific information about extracting SPOT data from tape. Digital SPOT data are available in both 1600 bpi and 6250 bpi tapes. The 1600 bpi data usually require 2 tapes, and therefore a separate image is stored on each tape. Each tape requires a separate run of *i.tape.other*. The 6250 bpi data usually require only one tape. This section will describe 1600 bpi data.

If the GRASS program *m.examine.tape* is run on the first SPOT tape of a two-tape set, the output will look like that shown below. (These numbers will vary from tape set to tape set.)

| | | |
|---|---|---|
| **file 1** | **len:** | **records** |
| | **360:** | **1 - 5** |
| **file 2** | **len:** | **records** |
| | **3960:** | **1 - 27** |
| **file 3** | **len:** | **records** |
| | **5400:** | **1 - 4495** |

Here, "len" refers to record length. "len" is the number of columns in a file, while "records" is the number of rows. Files 1 and 2 are small, having 5 rows and 27 rows respectively. These are the two header files on the tape. File 3 is the file containing the imagery data.

If *m.examine.tape* is run on the second tape, the output looks like this:

| | | |
|---|---|---|
| **file 1** | **len:** | **records** |
| | **360:** | **1 - 5** |
| **file 2** | **len:** | **records** |
| | **5400:** | **1 - 4498** |
| **file 3** | **len:** | **records** |
| | **1080:** | **1 - 3** |
| **file 4** | **len:** | **records** |
| | **360:** | **1 - 1** |

File 1 is the header file, file 2 is the file containing imagery data, and files 3 and 4 are trailers.

Because SPOT data is in band interleaved format, in order to determine the numbers of rows on each tape, the number of records in the image file (given in *m.examine.tape*) must be divided by the number of bands (3). This gives: 4495/3 = 1498.333 rows or records on tape 1, and: 4498/3 = 1499.333 rows or records on tape 2. Note that there is one odd record in the image file on each tape. This is a header record. It occurs as the first record in the image file and should be skipped during the tape extraction. Not skipping this record may contribute to the occurrence of a no-data line where the two images are patched if they are patched using the GRASS program *r.patch*. The number of rows containing imagery data are then 1498 rows on tape 1 and 1499 rows on tape 2.

Determining the number of imagery columns is not as easy. *m.examine.tape* shows that there are 5400 columns of data in the imagery files on each tape. This includes filler (zeros) on both sides of the image. A printout that comes with every SPOT tape set offers additional information about the columns. The printout provides an item called the number of imagery pixels per monospectral line. For this SPOT tape set, the number of imagery pixels per monospectral line is given as 3510. This is the number of columns of imagery data on each SPOT tape. Determining where these columns fall in the 5400 columns requires use of the GRASS display option *d.where* or the *measurements* option in *d.display*.

The SPOT printout offers other information as well:

1 The scene column (K) and row (J). K is the satellite ground track, and J is the line of latitude.

2 The number of multispectral lines in the scene. A scene is defined as including one 6250 bpi tape or two 1600 bpi tapes, so to get the number of rows of image data on each tape, divide the number of multispectral lines in the scene by the number of tapes. Tape 2 has one more row than tape 1. This result will give the same number of rows as did *m.examine.tape*, but it will not be evident whether or not there is a header record in the image file.

3 The scene latitude and longitude corner points and center point.

4 The scene orientation, incidence, elevation, and more.

### 7.3.7. The Extraction Process

The extraction process will begin by first skipping the number of specified files, advancing to the starting row, and then reading the tape. The percent completion of the extraction is displayed on the screen.

### 7.3.8. The Output

Following the extraction, the extracted band files will be listed as raster files available in the current MAPSET. These files may be displayed individually using either the GRASS command *d.display*, *d.rast*, or *i.points*.

**Note:** After extracting an image from tape the mapset region in the x,y coordinate LOCATION will be set based upon the extracted rows and columns from the tape (see section 5, **THE IMAGERY LOCATION**; subsection 5.1., **Creating an X,Y Coordinate LOCATION**; and section 6, **REGIONS AND MASKS**).

**Note:** If more than one image or image subset is extracted into the same x,y LOCATION, regardless of whether they are extracted into the same group, and if the images or image subsets have different mapset regions (which are based on the rows and columns extracted), then, to display a raster file in the x,y LOCATION, you will have to check the current MAPSET region to make sure that it matches the raster file to be displayed. For a more detailed explanation see section 6, **REGIONS AND MASKS**.

### 7.3.9. About Patching

As was mentioned earlier, SPOT data can be stored in a two-tape set in which each tape is a separate image. In this case, each tape requires a separate *i.tape.other* extraction and each tape therefore creates a separate imagery group (this is the group that the user names with the prefix/group name). When more than one tape is extracted, to include an entire area of interest for an application, at some point in the image processing sequence the user may want to unite the images. In GRASS 4.0 the program to accomplish this is *r.patch*. See section 15, **PATCHING IMAGES** for information about when

and how to patch.

## 8. THE CONTRAST STRETCH

The GRASS imagery command that applies a histogram contrast stretch to a raster map layer is called *i.grey.scale*. *i.grey.scale* is actually a contrast enhancement that assigns a histogram contrast stretch grey scale color table to a map layer. The histogram contrast stretch expands the original range of digital values in the map or image to utilize the full range or sensitivity of the display device. This is done primarily to improve *visual* image analysis, and may be used prior to computer-assisted classification, change detection, or other raw spectral image analysis because it does not change the original pixel values. As each band raster file is extracted from tape using the *i.tape* programs, a histogram contrast stretch grey scale color table is given to each band.

One use for *i.grey.scale*, then, is to improve the visual interpretation of features within a subset of an already extracted raster file, when the subset has low contrast. Low contrast occurs when the range of pixel values is decreased, i.e., when an image is composed of pixels with similar reflectance values or color intensities. The resulting subset image may end up with a pixel color range of say, 30 to 101, while the full range of both the imagery medium and the display device is 0 to 255. The contrast stretch algorithm expands the existing range of pixel values (30 to 101) to the full range (0 to 255), making the visual interpretation of the data easier.

Ways of creating subsets of already extracted files are:

1 Changing the MAPSET region to a smaller region, or

2 Running the GRASS program *r.resample*.

When *i.grey.scale* is run on a subset of an already extracted file in this manner, it will effect the whole extracted raster file. The user will want to run *i.grey.scale* again on the full raster file when visual observation of it is once again desired.

To run *i.grey.scale* simply type:

**i.grey.scale**

You will be asked for the map layer that needs a grey scale:

**Which layer needs a grey scale?**
**Enter 'list' for a list of existing raster files**
**Enter 'list -f' for a list with titles**
**Hit RETURN to cancel request**
>

If you are working in a LOCATION with an x,y coordinate system, all of the raster files residing in that LOCATION will be available to *i.grey.scale* including single-imagery band files and composite images. Keep in mind however, that *i.grey.scale* can be run in any LOCATION having any coordinate system and can be applied to any GRASS map layer.

When the the program is finished, and the map layer is displayed (using the GRASS commands *i.points*, *d.display*, or *d.rast*), it is displayed with a grey scale color scheme.

## 9. THE COLOR COMPOSITE

**Note:** Before running *i.composite* the user should check to see that the rows and columns of the raw band raster files to be composited are the same as those of the mapset region in the current LOCATION. To do this, use the GRASS command *g.region* (see section 6, **REGIONS AND MASKS**).

*i.composite* is an imagery function that creates a color composite image from three band files specified by the user. The user specifies the bands to be used by assigning the bands a red, blue, or green intensity. The resulting image is a single raster file composed of the three bands that were selected. The color composite can be displayed, plotted, or manipulated as would any raster file in GRASS.

This program can be used to create a false color composite. A false color composite is the color assignment that is used in infrared photography. The green band is assigned shades of the primary color blue, the red band is assigned shades of the primary color green, and the near infrared band is assigned shades of the primary color red.

A good use for the color composite is in the program *i.points*, which allows the user to locate points on an image to be rectified. When using *i.points*, if an image is not classified before it is rectified (before *i.points* is begun), the only imagery files available on which to locate and mark points may be the single band files. These are displayed in grey scale color scheme only. When color may enhance the ability to locate points on an image to be rectified, a color composite image can be displayed and marked to provide this advantage.

Other uses of the color composite are in *p.map*, *d.display*, and *d.3d*.

The first *i.composite* prompt asks for the imagery group.

**Select an imagery group file**
**Enter 'list' for a list of existing imagery groups**
**Enter 'list -f' for a verbose listing**
**Hit RETURN to cancel request**
>

This is the group containing the imagery bands to be composited. Then the following screen is displayed:

**Please indicate which files to use for red, green, and blue colors. You may leave any color out. You may specify more than one color per file. However, each color may only be specified once. For example, to get a full color image, specify r,g,b for 3 different files. To get a grey scale image, specify rgb for a single file.**

**b__ spot1.1**
**g__ spot1.2**
**r__ spot1.3**
**___ spotclass1**
**___ spotreject1**

**AFTER COMPLETING ALL ANSWERS, HIT <ESC> TO CONTINUE (OR <Ctrl-C> TO CANCEL)**

You are then allowed to verify the choice of bands:

**Colors assigned as follows:**

**RED: spot1.3 in mapsetname**
**GREEN: spot1.2 in mapsetname**
**BLUE: spot1.1 in mapsetname**

**Look ok? (y/n) [y]**

The color table that is created has 1000 colors or 10 saturation levels (or shades) per primary color (blue, green, red). The number of colors that can be displayed at one time on a color monitor depends on the graphics monitor being used. If the graphics monitor can only display, for example, 512 colors at a time, then you will want to use the GRASS command *Dcolormode fixed* before displaying the file. The colors will be assigned the nearest displayable color, and the file will retain its relative color

accuracy. If the file is displayed in *Dcolormode float* rather than *Dcolormode fixed*, the display on the monitor will not be accurate.

You are then asked to name the composite image:

**Enter a new raster file name**
**Enter 'list' for a list of existing raster files**
**Enter 'list -f' for a list with titles**
**Hit RETURN to cancel request**
>

Now, the program creates the composite image, and the percent complete is echoed to the screen. Support files are created automatically.

## 10. IMAGERY GROUPS

The purpose of grouping imagery raster files is to identify collections of raster files for classification and rectification. For example, the imagery programs *i.points* and *i.rectify* ask the user for the imagery group to be analyzed, which contains the raster files to be rectified. All of the raster files in this group, regardless of whether they have the same row and column coordinates, that is, regardless of whether they are different subsets of the same image, have been extracted from the same image (i.e., the same TM quad, the same SPOT tape, the same aerial photo tape, or the same MSS tape set). Giving the name of this group to the *i.points* and *i.rectify* programs tells the programs which tape data is to be rectified, and tells the programs where to store the rectification points marked in *i.points* and the transformation equations created by the programs. During *i.rectify* the program asks the user to select the specific raster files in the imagery group to be rectified.

Similarly, the programs *i.cluster* and *i.maxlik* ask the user for the imagery group *and* the imagery subgroup to be analyzed. The imagery subgroup contains the band raster files (and/or band ratio raster files, elevation raster files, aspect raster files, slope raster files, etc.) to be classified. The imagery group, which contains this subgroup (and all of the raster files extracted from an image as described in the previous paragraph), also serves as a repository for the signature files generated by *i.cluster* and the classified map generated by *i.maxlik*.

When the imagery band raster files are extracted from tape and assigned to an imagery group (called *prefixname* in subsection 7.3.4., **Prefix/Group Name**), these raster files actually exist *ungrouped* as individual raster files in the current LOCATION and MAPSET, because only their file names are really grouped into the imagery group. By giving the programs the names of the files, the programs know what files to operate on. The raster files are therefore available at all times to be used as they exist in the LOCATION individually. The raster files are always available to be grouped into any number of other imagery groups or subgroups of imagery groups using the GRASS imagery command *i.group*.

*i.group* is an imagery program that allows the user to group any of the raster files that are present in the current LOCATION and currently selected MAPSETS into:

1 A subgroup of the first group created during tape extraction

2 A new imagery group, and subsequently, a subgroup of a new imagery group

3 An already existing group in the current LOCATION and MAPSET, and subsequently, a subgroup of an already existing group

*i.group* also allows the user to remove raster files from a group in his or her MAPSET. Cell files, however, cannot be removed from, or added to existing subgroups. If a new subgroup is needed, the procedure recommended is to create a new subgroup.

If raster files are removed from the hard disk using the GRASS commands *g.remove*, or the UNIX command **rm**, *and*, if *i.group* has not been directed to look for these files following selection of one of its options, the removed raster files will still appear as belonging to a group. They will, however, be tagged with a phrase stating: "raster files do not exist." The raster file names may be removed from within the group by using the remove option of *i.group*.

To remove an entire imagery group, the GRASS program *g.remove* can be used. If there are no imagery groups present in the user owned MAPSET, this option of *g.remove* will not be available. Removing an entire group will not remove raster files named in the group, but it will remove all signature files and rectification points stored in the group. At present, there is no option to remove subgroups.

*i.group* is able to offer its options to all of the raster files in the MAPSETS that were selected using the GRASS command *g.mapsets*. *i.group*, however, is restricted to the files and selected MAPSETS in the current LOCATION. The raster files available to *i.group*, then, will include all raster files that result from analyses using GRASS, i.e, classified images, color composites, contrast stretches, neighborhood filters, results of r.mapcalc, r.buffer, etc.

Since *i.group* can be run in any GRASS coordinate system LOCATION, it is possible to group any GRASS map layer into an imagery group. The only GRASS 4.0 programs that utilize the imagery group and imagery subgroup at present, however, are the imagery programs.

The options to select, edit, and create imagery groups are available in the *i.group* main menu.

To start *i.group*, type:

**i.group**

The first screen in *i.group* asks you to select an imagery group. If the group name you select does not yet exist, you will be asked if you would like to create a new group:

**This program edits imagery groups. You may add data layers to, or remove data layers from an imagery group. You may also create new groups.**
**Please enter the group to be created/modified**

**GROUP: _____ (enter 'list' for a list of groups)**

**AFTER COMPLETING ALL ANSWERS, HIT <ESC> TO CONTINUE (OR <Ctrl-C> TO EXIT)**

If the word **list** is entered above, groups that have already been created in the current LOCATION and MAPSET are displayed.

The second screen to appear is the main menu for *i.group*. It provides the following options:

**1. Select a different group**
**2. Edit group title**
**3. Include new raster files in the group or remove raster files from the group**
**4. Assign colors to the group**
**5. Create a new subgroup within the group**

**RETURN to exit**

If all of the raster files that were named in a group have been removed from the hard disk, and removed from within the group, only options 1 and 2 and the "include" portion of option 3 will be available.

Options 1 through 5 are described as follows:

*Select a different group*

If option number 1 is chosen, the following screen is displayed:

**Please enter the group to be created/modified**

**GROUP: _____ (enter 'list' for a list of groups)**

**AFTER COMPLETING ALL ANSWERS, HIT <ESC> TO CONTINUE**
**(OR <Ctrl-C> TO EXIT)**

This screen is like the first screen, it allows you to select a different group or create a new group. If the word **list** is entered, the already existing groups in the current LOCATION and MAPSET are displayed.

*Edit group title*

If option number 2 is selected, an entry space is provided to type in the group title. This title is useful in identifying each group:

**TITLE**_____

This option offers an opportunity to go back and change the entry if it is not correct by asking:

**Look ok? (y/n)**

*Include new raster files in the group or remove raster files from the group*

When choosing option number 3, the following screen is displayed:

**LOCATION:** *spear_xy* **GROUP:** *spot1* **MAPSET:** *mapsetname*

**If you wish to delete a file from group [***spot1***], remove the x**
**from in front of the file name.**

**x_ spot1.1 in mapsetname**
**x_ spot1.2 in mapsetname**
**x_ spot1.3 in mapsetname**

**AFTER COMPLETING ALL ANSWERS, HIT <ESC> TO CONTINUE**
**(OR Ctrl-C> TO CANCEL)**

To remove an **x** use the space bar.

Next, a screen listing all the other raster files present in the current MAPSET(S) will be displayed:

**LOCATION:** *spear_xy* **GROUP:** *spot1* **MAPSET:** *mapsetname*

**Please mark an x by the files to be added in group [***spot1***]**

**MAPSET:** *mapsetname*

**x_ composite1**
**x_ spotclass1**
**__ spotclass1_2**
**__ spotclass1_3**

**AFTER COMPLETING ALL ANSWERS, HIT <ESC> TO CONTINUE**
**(OR Ctrl-C> TO CANCEL)**

If more than one MAPSET is selected, screens for those mapsets will also be displayed. All files selected with an **x** will be included in the group being modified.

You will then have the opportunity to check the contents of the group that was just modified:

**Group [***spot1***] references the following files**

| | |
|---|---|
| **spot1.1** | **in mapsetname** |
| **spot1.2** | **in mapsetname** |
| **spot1.3** | **in mapsetname** |
| **composite1** | **in mapsetname** |
| **spotclass1** | **in mapsetname** |

**Look ok? (y/n)**

If you respond with the letter **y** then the following sentence is displayed on the screen:

**Group [***spot1***] updated!**

And the main menu for *i.group* returns.

If you respond with the letter **n**, the screen containing the group files, *after* it was modified, will be displayed and you will be asked to place an **x** in front of the files to be removed from the group. The screen(s) listing all of the other raster files in the current MAPSET(S) will be displayed again, and you will be asked to place an **x** in front of the files to be included in the group. This provides the opportunity to correct mistakes or make changes in the choice of files selected without exiting *i.group*.

*Assign colors to the group*

Option number 4 provides the following screen:

**Please indicate which files to use for red, green, and blue colors. You may leave any color out. You may specify more than one color per file. However, each color may only be specified once. For example, to get a full color image, specify r,g,b for 3 different files. To get a grey scale image, specify rgb for a single file.**

**b__ spot1.1**
**g__ spot1.2**
**r__ spot1.3**
**___ composite1**
**___ spotclass1**

**<<< r,g,b can only be specified once >>>**

**AFTER COMPLETING ALL ANSWERS, HIT <ESC> TO CONTINUE (OR <Ctrl-C> TO CANCEL)**

This screen allows you to assign to a band the shades of a primary color, red, green, and/or blue. Note, however, that composite images and classified images are already assigned colors during their creation. If they are assigned a new color here, the resulting color for the raster file will be a single color or a grey scale. (When the imagery program *i.colors* is completed, this option will disappear). An opportunity to change the choice of colors is offered after escaping the screen by:

**Look ok? (y/n)**

*Create a new subgroup within the group*

The following screen enables you to create a subgroup out of any combination of files in the group. Any number of subgroups may be created by repeating the option.

**LOCATION:** *spear_xy* **MAPSET:** *mapsetname*
**GROUP:** *spot1*

**SUBGROUP: _____ ('list' will show available subgroups)**

**AFTER COMPLETING ALL ANSWERS, HIT <ESC> TO CONTINUE (OR <Ctrl-C> TO CANCEL)**

After selecting or creating the above subgroup (*123*), this screen is displayed:

**Mark an x by the files to form subgroup [*123*]**

**x_ spot1.1**
**x_ spot1.2**
**x_ spot1.3**
**__ composite1**
**__ spotclass1**

**AFTER COMPLETING ALL ANSWERS, HIT <ESC> TO CONTINUE**
**(OR <Ctrl-C> TO CANCEL)**

Next you are given the opportunity to check the contents of the subgroup:

**Subgroup [*123*] references the following raster files**

**spot1.1 in mapsetname**
**spot1.2 in mapsetname**
**spot1.3 in mapsetname**

**Look ok? (y/n)**

If you respond with the letter **n**, the second screen will appear again enabling you to select files to form the subgroup.

## 11. TARGETING IMAGERY GROUPS

*i.target* is an imagery function that "targets" an imagery group to a GRASS database standard coordinate LOCATION. This is a LOCATION that has standard coordinates such as UTM or State Plane. You are probably wondering what it means to "target" a LOCATION. This can be explained as follows. During the imagery program *i.rectify* a specific LOCATION is required in which to transfer the rectified file or files just prior to completion of the program. *i.target* enables the user to specify this LOCATION for the imagery group that is about to undergo *i.rectify*. Because *i.points* also uses information about the target location, *i.target* must be run before running *i.points* and *i.rectify*.

To run *i.target* simply type:

**i.target**

The first prompt in the program asks for the name of the imagery group that needs a target. The imagery group must be present in the current LOCATION.

**Enter group that needs a target**
**Enter 'list' for a list of existing imagery groups**
**Enter 'list -f' for a verbose listing**
**Hit RETURN to cancel request**
>

Next, the following screen asks for the target LOCATION and MAPSET:

**Please select the target LOCATION and MAPSET for group** <*spot1*>

**CURRENT LOCATION:***spear_xy*_____
**CURRENT MAPSET:** *mapsetname*_____

**TARGET LOCATION:***spearfish*_____
**TARGET MAPSET:***mapsetname*_____

**(enter list for a list of locations or mapsets within a location)**

**AFTER COMPLETING ALL ANSWERS, HIT <ESC> TO CONTINUE**
**(OR <Ctrl-C> TO CANCEL**

The imagery group is now targeted to the selected LOCATION. An imagery group may be targeted to any GRASS LOCATION.

## 12. IMAGE CLASSIFICATION

One of the methods of deriving information from digital imagery is multispectral classification. Multispectral classification analyzes the spectral reflectance values of pixels in an image. Pixels, or picture elements, are the grid cells that make up an image. The word "pixel" is usually used in reference to a raster when it is displayed on a computer monitor, but it is also often used to refer to the grid cell in a stored digital image that we cannot see until it is displayed. In raw spectral digital imagery each grid cell has a digital number (DN) or spectral reflectance value ranging from 0 to 255. These are grouped into clusters or categories during image classification that can be interpreted to represent features on the surface of the earth.

There are many methods of multispectral classification, but, generally, they fall into three groups:

1 Supervised classification

2 Unsupervised classification

3 Combinations of supervised and unsupervised classification

GRASS 4.0 supports both supervised and unsupervised classification. Unsupervised classification is conventionally a two-pass process. The two GRASS programs that create an unsupervised classified image are:

*i.cluster*
*i.maxlik*

Both programs must be run to complete an unsupervised classification.

**Note:** The programs *i.cluster* and *i.maxlik* can be run either before or after *i.points* and *i.rectify*. Please see section 14, **A NOTE ABOUT PROGRAM SEQUENCE** and section 15, **PATCHING IMAGES**.

Before running an unsupervised classification, you must first run *i.group*, and create a subgroup that contains the imagery raster files (*prefixname.1, prefixname.2, prefixname.3*, or, band ratio raster files, elevation raster files, aspect raster files, slope raster files, etc.) that you wish to classify. Please see section 10, **IMAGERY GROUPS** for a discussion of groups and subgroups.

After creating a subgroup with the raster files in it, you can type

**i.cluster**

to begin the classification process. *i.cluster* must be run first, before *i.maxlik*.

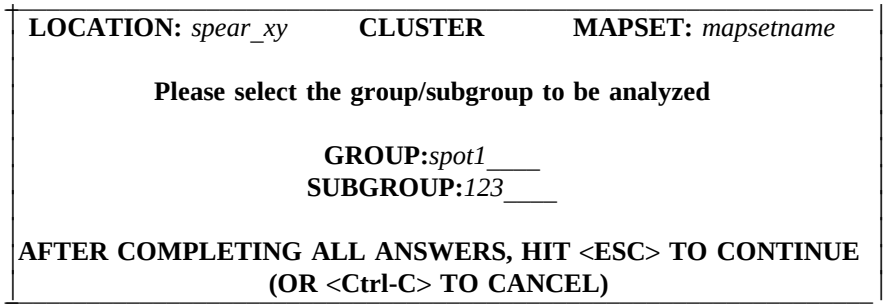
### 12.1. i.cluster

*i.cluster* is the program that generates the spectral signatures for the land cover types in the image using a clustering algorithm. It results in a signature file that is used as input for the second pass program *i.maxlik*. The clustering algorithm operates by reading through the imagery data and then building pixel clusters based on the spectral reflectances of the pixels. The spectral distributions of the clusters (which will be the landcover spectral signatures) are influenced by six parameters set by the user. The first parameter set by the user is the initial number of clusters to be discriminated. *i.cluster* starts by generating spectral signatures for this number of clusters and "trys" to end up with this number of clusters during the clustering process. The resulting number of clusters and their spectral distributions, however, are also influenced by the range of the spectral values in the image and the other parameters set by the user. These parameters are: the minimum cluster size, minimum cluster separation, the percent convergence, the number of iterations, and the row and column sampling interval.

The cluster spectral signatures that result are composed of cluster means and covariance matrices. These cluster means and covariance matrices are used in the second pass program *i.maxlik* to classify the image.

#### 12.1.1. i.cluster Inputs

The first screen in *i.cluster* asks for the imagery group and subgroup to be analyzed:




The purpose of the group and subgroup is to collect files for classification or analysis. The group should contain the image raster files that you wish to classify, and the subgroup, which is a subset of this group, should contain only the specific image raster files that you wish to classify.

**Note:** The subgroup must contain more than one raster file.

The first prompt in the program asks for the name of the resulting signature file. This signature file is the output file for *i.cluster* and the required input file for the GRASS program *i.maxlik*. It contains the cluster means and covariance matrices that are used to classify the image in *i.maxlik*.

**Enter a name for the resulting signature file**
**Enter 'list' for a list of existing subgroup [*123*] signature files**
**Enter 'list -f' for a list with titles**
**Hit RETURN to cancel request**
>

Note, *123* is the hypothetical name of the currently selected subgroup.

After entering the signature file name, you are asked to enter the name of a seed signature file. This is optional. Seed signatures are signatures that contain cluster means and covariance matrices that were calculated prior to the current run of *i.cluster*. They may be acquired from a previous run of *i.cluster* or a supervised classification signature training site selection. The purpose of seed signatures is to

optimize the cluster decision boundaries (means) for the number of clusters specified.

**SEED SIGNATURES**
**Select the signature file to use for the initial means**
**Enter 'list' for a list of existing subgroup [***123***] signature files**
**Hit RETURN to use DEFAULT means**
>

If you do not choose to enter a seed signature file, just hit RETURN and the means that are calculated during *i.cluster* will be used.

The final screen is:

| | |
|---|---|
| **Please set the following information** | |
| **Number of initial classes** | **15**_____ |
| **Minimum class size** | **17**_____ |
| **Minimum class separation** | **0.50**____ |
| **Percent convergence** | **98.00**____ |
| **Maximum number of iterations** | **30**_____ |
| **Your current region contains 100 rows and 100 cols (1000 cells)** | |
| **Please set the sampling intervals** | |
| **Row interval** | **2**_____ |
| **Col interval** | **2**_____ |
| **AFTER COMPLETING ALL ANSWERS, HIT <ESC> TO CONTINUE (OR <Ctrl-C> TO CANCEL)** | |

This screen asks for the clustering parameters. The default values in the screen are the suggested values based upon the total number of rows and columns in the image and common standards. This screen also informs you of the number of image rows and columns that are included in the current LOCATION and MAPSET region. Make sure that this region is large enough to include the portion of the image that you wish to classify. You can use *d.rast* followed by *d.where*, or *d.display*, or *g.region* to check the region.

A description of each field in the screen is listed below.

*Number of initial classes*

This is the number of clusters that will initially be identified in the clustering process before the iterations begin.

*Minimum class size*

This is the minimum number of pixels that will be used to define a cluster, and is therefore the minimum number of pixels for which means and covariance matrices will be calculated.

*Percent convergence*

The percent convergence is the point at which cluster means become stable during the iteration process. It is the percentage of pixels that no longer move from cluster to cluster during iteration. When clusters are being created, their means constantly change as pixels are assigned to them and the mean is recalculated to include the new pixels. After all clusters have been created, *i.cluster* begins iterations that change cluster means by maximizing the distances between them. As these means shift, a higher and higher percentage of pixels stay within their clusters. Because means will never become totally static, a percent convergence and a maximum number of iterations is supplied to stop the iterative process. The percent convergence should be reached before the

maximum number of iterations. If the maximum number of iterations is reached, it is probable that the desired percent convergence was not reached. In this event, you may want to increase the number of iterations and run *i.cluster* again (see *maximum number of iterations* below). The number of iterations is available in mail when *i.cluster* is finished.

*Maximum number of iterations*

This is a number greater than the number of iterations predicted to achieve the optimum percent convergence. If the number of iterations reaches the maximum designated, you may want to rerun *i.cluster* with a higher number of iterations, a lower percent convergence, or a larger minimum class separation.

*Minimum class separation*

This is the minimum separation below which clusters will be merged in the iteration process. If the separation is relatively large, clusters with similar spectral distributions will be merged. If the separation is small, clusters that are more similar spectrally than the resulting clusters of a larger separation will remain distinct. Smaller separations result in fewer cluster mergers, while larger separations result in more mergers.

The optimum minimum class separation is an image-specific number that depends on the image data being classified and the number of final clusters that are acceptable. Its determination requires experimentation. Commonly used minimum class separations range from 0.5 to 1.5. The separability matrix that is output via mail at the completion of the program presents the results of the minimum class separation.

Note that as the minimum class separation is increased, the maximum number of iterations should also be increased to achieve this separation with a high percent convergence. This is because of the larger number of mergers that occur when separations are larger. As more clusters are merged, the percent convergence decreases, requiring an increased number of stablizing iterations to achieve a high percent convergence.

*Row and column sampling intervals*

These numbers are the default row and column skip, which are based on the size of the data set. They are computed to sample 10,000 pixels throughout each image band at 100 pixels per sampled row, and 100 pixels per sampled column. If the user changes the default intervals, and the sample size is too large for the available computer space, or too small for accurate analysis, a warning will be delivered by mail. The reason for sampling the data set is to save computer time and memory during the sampling process, while obtaining an accurate representation of the statistical range of the data.

### 12.1.2. Program Execution

*i.cluster* will run in the background and notify you by mail when it is complete. The mail message will contain the results, i.e., the statistics for each cluster. Also included in the mail message are the resulting percent convergence for the clusters, the number of iterations that were required to achieve the convergence, and the separability matrix.

## 12.2. i.maxlik

*i.maxlik* is the program that classifies the image based on the cluster spectral signature information generated in *i.cluster*. Remember, clusters are the spectral information categories or classes that were determined for the spectral values of the pixels in the image.

*i.maxlik* is a maximum likelihood discriminant analysis classifier. It is the second pass in a two-pass procedure known as an unsupervised classification. It can also serve as the second pass in a supervised classification. The first pass in the unsupervised classification is executed by the GRASS program *i.cluster*. Both programs are required to complete an unsupervised classification.

The maximum likelihood classifier uses the cluster means and covariance matrices from the *i.cluster* signature file and determines to which class (or cluster) each pixel in the image has the highest

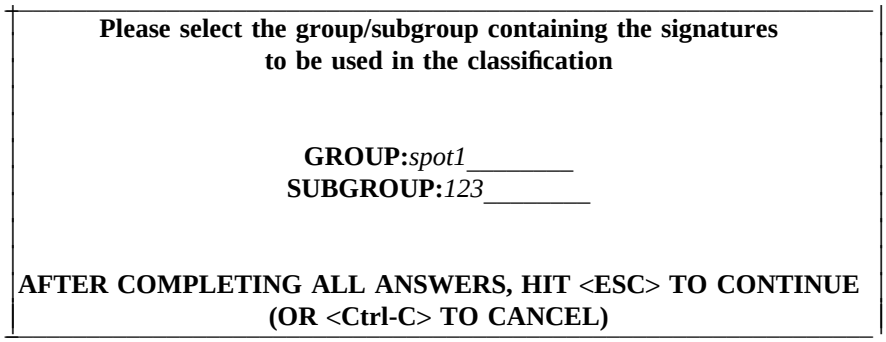
probability of belonging. The resulting file is a classified image in which each pixel has been assigned to a spectral class or category. The spectral classes can be related to specific land cover types on the ground.

To start *i.maxlik* simply type:

**i.maxlik**

### 12.2.1. i.maxlik Inputs

The first screen asks for the group and subgroup to be classified:




The group should contain the raster files and/or band raster files to be classified. The subgroup should contain the raster files and/or band raster files that were used to create the signature file in the program *i.cluster*.

The first prompt asks for the name of the signature file to be used for the classification. The signature file contains the cluster and covariance matrices that were calculated in the GRASS program *i.cluster*. These signatures are what determine the categories in the classification process.

**Enter signature file to be used for classification**
**Enter 'list' for a list of existing subgroup [*123*] signature files**
**Enter 'list -f' for a list with titles**
**Hit RETURN to cancel request**
>

The next prompt asks for the name of the classified map layer that will be generated in *i.maxlik*. This is a raster file that will be a map of categories that can be related to landcover categories on the ground.

**Please name the CLASSIFIED map layer to be generated**
**Enter 'list' for a list of existing raster files**
**Enter 'list -f' for a list with titles**
**Hit RETURN to cancel request**
>

Next, you are asked to enter the name of the reject threshold map layer that is also generated in *i.maxlik*.

**Please name the REJECT THRESHOLD map layer to be generated**
**Enter 'list' for a list of existing raster files**
**Enter 'list -f' for a list with titles**
**Hit RETURN if you don't want this layer**
>

While classifying an image, *i.maxlik* performs a chi square test on each discriminant result at various threshold levels of confidence to determine at what confidence level each pixel is likely to belong to the

assigned class. This is the reject threshold map layer. It contains one calculated confidence level for each classified pixel in the classified image. One of the possible uses for this map layer is as a mask, to identify pixels in the classified image that have the lowest probability of being assigned to the correct class. This can be accomplished by using the program *r.mask* and then displaying the classified map layer while this mask is set. See section 6, **REGIONS AND MASKS** for cautions about using a mask.

Next, the following screen is displayed listing the raster files in the subgroup you are classifying:

**COLOR CONFIGURATION FOR CLASSIFIED LAYER [***spotclass1***]**

**Please indicate which files to use for red, green, and blue colors. You may leave any color out. You may specify more than one color per color file. However, each color may only be used once.**

**For example, to get a full color image, specify r,g,b for 3 different files. To get a grey scale image, specify rgb for a single file.**

**b___ spot1.1 in mapsetname**
**g___ spot1.2 in mapsetname**
**r___ spot1.3 in mapsetname**

**AFTER COMPLETING ALL ANSWERS, HIT <ESC> TO CONTINUE (OR <Ctrl-C> TO CANCEL)**

The purpose of this option is to assign colors to the raster files making up the final classified map. This subgroup could have more than three raster files in it. This is because more than three raster files can be classified at one time, but, because there are only three additive primary colors, red, green, and blue, (which are the values used to create the many shades of colored light), only three raster files can be displayed on a color graphics monitor at a time. Therefore, regardless of how many raster files are being classified, only three of the raster files are assigned a color.

*i.maxlik* runs in the background and notifies the user by mail when it is complete. The classified image becomes a member of the selected imagery group.

**Note:** The maximum likelihood classifier assumes that the spectral signatures for each class in each band or raster file are normally distributed (i.e., Gaussian in nature). Clustering algorithms, however, can create signatures that are not normally distributed. If this occurs, *i.maxlik* will reject them and display a warning message. The program will proceed but the signatures will not be used.

### 13. IMAGE RECTIFICATION

Rectification is the mapping of an image from one coordinate system to another. The geometry of an image extracted from tape into a GRASS LOCATION having an x,y coordinate system is not planimetric. To make an image planimetric, that is, to convert the x,y coordinate system into a standard map coordinate system (for example, the UTM coordinate system or the State Plane coordinate system) points from a map having the standard coordinates must be associated with the same points on an image to be rectified. This is done by marking points on an image and then inputing standard coordinates from a map. Then a least squares regression can be calculated using the two types of coordinates for each marked point. A matrix containing transformation coefficients is the result. The transformation coefficients are used in an equation called a transformation equation to calculate a standard coordinate for each x,y pixel in the image. A rectified planimetric image having the standard coordinates is deposited in a target database.

The two programs that accomplish the rectification process in GRASS are:

1 *i.points*

2 *i.rectify*

**Note:** *i.points* and *i.rectify* can be run either before or after *i.cluster* and *i.maxlik*. Please see section 14, **A NOTE ABOUT PROGRAM SEQUENCE** and section 15, **PATCHING IMAGES**.

### 13.1. i.points

*i.points* enables the user to mark coordinate system points on an image to be rectified and then input the coordinates of each point for calculation of a transformation matrix. *i.points* must be followed by *i.rectify* which rectifies the image using the transformation matrix coefficients calculated in *i.points*. *i.target* must be run before *i.points* to enable the PLOT CELL option to be used and to identify the database LOCATION in which the rectified image will be deposited following *i.rectify*.

*i.points* requires the use of a graphics monitor. A summary of the program follows:

In the first step, the unrectified image is displayed, and points having known standard coordinates are marked on the image. To identify the precise location of a point to be marked, *i.points* has a zoom option. In addition to marking points on an image to be rectified and inputing the standard coordinates using the keyboard, *i.points* also has the option to simultaneously display (using PLOT CELL) another map layer available in the targeted database, and use the standard coordinates for that map layer as the coordinates for the marked point. When this option is chosen, these coordinates are input automatically. Any GRASS map layer in the current group, or in the targeted database LOCATION can be displayed using *i.points*.

During the process of marking points and entering map coordinates, the user can compute the RMS (root mean square) error for each point entered. *i.points* does this by calculating a transformation equation (the same one that is calculated in the GRASS program *i.rectify*), using a least squares regression. Coefficients are computed for the equation. The coefficients are then used in the equation along with the x,y coordinates of the marked points. The results are plugged into an equation for RMS error. The interpretation of RMS error is described in subsection 13.1.4., **ANALYZE**.

The procedure for marking points (registering points), entering coordinates, and calculating RMS error is described in the following sections.

The first prompt in the program asks for the imagery group to be registered.

**Enter imagery group to be registered**
**Enter 'list' for a list of existing imagery groups**
**Enter 'list -f' for a verbose listing**
**Hit RETURN to cancel request**
>

Note that if *i.target* is not run before *i.points*, the program will display the following error message:

**ERROR: Target information for group [***spot1***] missing**
**Please run** *i.target* **for group [***spot1***]**

The imagery group entered above should contain the files that you wish to rectfy. After entering the group to be registered, the terminal screen displays the message:

**use mouse now...**

And, the graphics monitor displays the following screen:

| **imagery** *filename* **(mag)** | **target** *filename* **(mag)** |
|---|---|
| | |
| | |
| QUIT ZOOM PLOTCELL ANALYZE | |

"mag" refers to the magnification of the image when it is displayed. A pop-down menu will be superimposed on the left half of the screen that looks like this:

| **Double click on raster file to be plotted**<br>**Double click here to cancel** |
|---|

| **Mapset** *mapsetname* | |
|---|---|
| **spotclass1** | **spot1.1** |
| **composite1** | **spot1.2** |
| **spot1.3** | |

Any single file in the imagery group may be used on which to mark points, and points can be marked on more than one file in the imagery group to accumulate the suggested minimum number of 12 points. Any file in the imagery group can be rectified (using *i.rectify*) based on the transformation matrix computed from these points.

The chosen imagery file is displayed in the upper left quadrant of the monitor at a default magnification based on the size of the file.

### 13.1.1. ZOOM

To enlarge the displayed file, all you have to do is place the mouse cross hairs on the word ZOOM, and the following menu will be displayed at the bottom of the screen:

| **Cancel** | **Box** | **Point** | **Select type of ZOOM** |
|---|---|---|---|

You have the option to identify the zoom region using either the mouse to make a box, or the mouse to mark two diagonal points of the desired region. The terminal screen will display a mouse menu to

guide you in identifying the corner points of the region.

### 13.1.2. Marking Points

While the image is displayed, the first thing you need to do is locate points on a standard map that correspond to points on the image that you can easily see. This is an important step because often points that look like good reference points on a map (i.e., road intersections stable landmarks, etc.) will not be visible on the image. Circle these points on the paper or mylar standard map. Next, use whatever means you have available to identify as precisely as possible the standard coordinates for these points. Digitizing software is recommended, especially the GRASS 4.0 program *v.digit*. Once you have determined the standard coordinates (for example, the UTMs) of each circled point, you are ready to mark the points on the displayed image.

To mark the points on the image that correspond to the points on the standard coordinate map, place the mouse cross hairs on the point of the image to be marked (you will probably have to ZOOM to find the exact spot) and press the left hand button on the mouse. A diamond shaped symbol will be marked on the image. The terminal will display the following screen:

| | |
|---|---|
| **Point 1 marked on the image at**<br>**East: 1023.77**<br>**North: -164.41** | |
| **Enter coordinates as east north:** | |

You can then enter the easting and northing for the point marked on the image. (The "East:" and "North:" displayed above are the x,y coordinates for the image.) When you enter an easting and a northing, the marked point is saved. If you decide not to enter a coordinate, you can simply hit RETURN and control goes back to the mouse and the marked point disappears.

### 13.1.3. PLOT CELL

In addition to acquiring reference points from a standard map, you also have the option to acquire the reference points from a map in the target database. The target database map is displayed by placing the mouse cross hairs on the words PLOT CELL. The following line is displayed at the bottom of the monitor:

| | |
|---|---|
| **Cancel** | **Indicate which side should be plotted** |

**Note:** If there are no raster files present in the target database, the PLOT CELL option will not be available.

To indicate which side of the monitor is to be plotted, place the mouse cross hairs on the half of the monitor screen that you would like to use and then press the left mouse button. The following menu

will appear on the half of the screen that was chosen:

| **Double click on raster file to be plotted**<br>**Double click here to cancel** |
|---|

| **Mapset** *mapsetname* | |
|---|---|
| **tm.rectified** | |
| **tm.classified** | |
| **Mapset PERMANENT** | |
| **elevation** | **geology** |
| **slope** | **soils** |
| **aspect** | |
| **roads** | |
| **streams** | |
| **airfields** | |

After the map is displayed the following message appears at the bottom of the monitor:

| **input method -->** | **keyboard** | **screen** |
|---|---|---|

When using the plotted target database map only as a comparative reference, the keyboard option can be chosen and the standard coordinates from a hardcopy map corresponding to the marked points on the image can be typed in. To choose the keyboard option, place the mouse cross hairs on the word KEYBOARD and press the left button on the mouse.

If you select the SCREEN option, the points marked on the image will automatically be associated with the coordinates from the corresponding points on the target database map. In this option, when the user marks a point on the imagery group raster file, the following screen is displayed on the terminal:

| **Point 5 marked on the image at**<br>**East: 1023.77**<br>**North: -164.41**<br><br>**Point located at**<br>**East: 679132.57**<br>**North: 4351080.67** | |
|---|---|
| **use mouse now...** | |

The "East:" and "North:" at the top of the monitor are the x,y coordinates for the marked point on the imagery group raster file. The "Point located at East: and North:" are the standard coordinates for the

marked point from the target database map (in this case UTMs). The coordinates for the target database map are automatically saved as the coordinates corresponding to the marked point on the image.

### 13.1.4. ANALYZE

After a number of points have been marked (four to seven), the RMS error of the points marked on the image can be checked. This is done by placing the mouse cross hairs on the word ANALYZE at the bottom of the monitor. The following error report is superimposed on the monitor:

| | error | | | image | | target | |
|---|---|---|---|---|---|---|---|
| # | row | col | target | east | north | east | north |
| **1** | **0.0** | **-0.9** | **1.0** | **1048.5** | **-144.8** | **679132.5** | **4351080.6** |
| **2** | **0.4** | **1.0** | **1.3** | **2153.1** | **-567.2** | **684314.7** | **4399001.4** |
| **3** | **-1.2** | **-0.5** | **.6** | **1452.8** | **-476.5** | **567841.4** | **3457682.8** |
| **4** | **1.1** | **0.5** | **1.3** | **1034.0** | **-109.2** | **677573.8** | **4352626.4** |
| **5** | *-2.7* | *14.0* | *14.2* | **1048.6** | **-144.9** | **679132.6** | **4351080.7** |
| | **overall** | **rms** | **error:** | **4.46** | | | |

The following menu appears at the bottom of the monitor:

| | | |
|---|---|---|
| **DONE** | **PRINT FILE** | **Double click on point to be included/excluded** |

The RMS error for the image being rectified is recorded under the column titled "error" and subtitled "row" and "col." In the above report, the marked point number 1 is 0.0 rows and -0.9 columns from the predicted location calculated by the transformation equation.

The RMS error for the target database map is recorded under the heading "error" and the subheading "target." This is the RMS error for the east and the north coordinates of the target map, but it is presented in the table using one general value.

The overall RMS error for the image is displayed at the bottom of the screen in meters. Points that generate a high RMS error are displayed in red on the monitor (represented here in italics).

The x,y coordinate of the point marked on the image being rectified is recorded under the heading "image" and the subheadings "east" and "north."

The standard coordinate of the point in the target database is recorded under the heading "target" and the subheadings "east" and "north".

If the user would like to exclude or include a point, this can be accomplished by placing the mouse cross hairs on the point number to be included (if the point is absent) or excluded (if the point is displayed) and then pressing the left button on the mouse twice. When a point is excluded, it is not included in the calculation of the RMS error, or included in the final transformation matrix. However, it can be retrieved within *i.points* at any time by double clicking with the mouse as described above. To exit the ANALYZE option, place the mouse cross hairs on the word DONE and the RMS error screen will disappear. You can then continue marking points on the image.

### 13.1.5. QUIT

To exit the *i.points* program, place the mouse cross hairs on the word QUIT at the bottom of the

monitor and all of the marked points (including coordinates) will be saved.

### 13.1.6. Note

A good rule of thumb is to mark at least twelve to fifteen points evenly distributed over the entire image (although they may be marked on different files) in order to obtain an accurate transformation equation for the rectification process. The RMS error may increase with more points added but the transformation equation will be more accurate.

An RMS error of less than or equal the resolution of the image being rectified is generally considered acceptable.

## 13.2. i.rectify

*i.rectify* is the imagery function that rectifies an image by computing a coordinate transformation for each pixel in the image using the transformation coefficient matrix created by the GRASS program *i.points*. *i.rectify* plugs the coefficients computed in *i.points* into an equation and then converts x,y coordinates to standard map coordinates for each pixel in the image. The result is a planimetric image with a transformed coordinate system (i.e., a different coordinate system than before it was rectified).

*i.points* must be run before *i.rectify*, and both programs are required to rectify an image. An image must be rectified before it can reside in a standard coordinate LOCATION, and therefore be analyzed with the other map layers in the standard coordinate LOCATION. Upon completion of *i.rectify*, the rectified image is deposited in the target standard coordinate LOCATION. This LOCATION is selected using *i.target*.

### 13.2.1. i.rectify Inputs

The first prompt in the program asks for the name of the group containing the files to be rectified.

**Enter the group containing files to be rectified**
**Enter 'list' for a list of existing imagery groups**
**Enter 'list -f' for a verbose listing**
**Hit RETURN to cancel request**
>

This is the same imagery group that was selected in *i.points* and the group that contains the raster files with the marked points and their associated coordinates. You are then asked to select the raster file(s) within the group to be rectified:

**Please select the file(s) to rectify by naming an output file**

| | |
|---|---|
| **spot1.1 in mapsetname** | **.....** |
| **spot1.2 in mapsetname** | **.....** |
| **spot1.3 in mapsetname** | **.....** |
| **spotclass1 in mapsetname** | *spotrectify1***..** |
| **spotreject1 in mapsetname** | **.....** |

**(enter list by any name to get a list of existing raster files)**

**AFTER COMPLETING ALL ANSWERS, HIT <ESC> TO CONTINUE**
**(OR <Ctrl-C> TO CANCEL**

More than one raster file may be rectified at a time. Each raster file should be given a unique output file name.

Next, you are asked to select one of two regions:

**Please select one of the following options**
**1. Use the current region in the target location**
**2. Determine the smallest region which covers the image**
>

*i.rectify* will only rectify that portion of the image or raster file that occurs within the chosen region, and only that portion of the raster file will be relocated in the target database. It is important therefore, to check the current mapset region in the target LOCATION if choice number one is selected.

If you are rectifying a file with plans to patch it to another file using the GRASS program *r.patch*, choose option number 1, the current region in the target location. This region, however, must be the default region for the target LOCATION. When a file being rectified is smaller than the default region in which it is being rectified, zeros are added to the rectified file. Patching files of the same size that contain 0/non-zero data, eliminates the possibility of a no-data line in the patched result. This is because, when the images are patched, the zeros in the image are "covered" with non-zero pixel values. When rectifying files that are going to be patched, rectify all of the files using the same default region.

### 13.2.2. Program Execution

*i.rectify* will run in the background and notify you by mail when it is finished. The process may take an hour or more depending on the size of the image, the number of files, and the size of the region.

**Note:** The rectified image or rectified raster files will be located in the target LOCATION when the program is completed. The original unrectified files are not modified or removed.

## 14. A NOTE ABOUT PROGRAM SEQUENCE

*i.cluster* and *i.maxlik* can be run before or after *i.rectify* without altering classification accuracy. This is because *i.rectify* is a nearest neighbor algorithm and not a cubic convolution. Rectification using *i.rectify*, therefore, does not introduce changes in the raw spectral data. In addition, *i.maxlik* is a per-point classifier and not a neighborhood classifier, so classification accuracy is not affected if the data has been previously rectified.

If an image is rectified first and then classified in GRASS, a few pointers are helpful. As described in subsection 12.1.1., **i.cluster Inputs**, *i.cluster* operates only on a subgroup of collected yet separate raster files. If you choose to rectify an image first, and then classify it, you must rectify all of the raster files as single files (these must total more than one), so that they can be classified by *i.cluster* as single raster files later. It is very easy in GRASS 4.0 to rectify more than one raster file at a time. To see how this is done, refer to subsection 12.2.1. **i.rectify Inputs**.

When classifying an image after rectifying it, *i.cluster* and *i.maxlik* will have to be run in the target LOCATION in which the rectified raster files are deposited following the running of *i.rectify*. *i.group* will also have to be run in the target LOCATION after "deposition" and before running *i.cluster*, to create the subgroup required by *i.cluster*.

## 15. PATCHING IMAGES

### 15.1. Patching and the Image-Processing Sequence

Like all programs in GRASS, except for the *i.tape* programs, *r.patch* can be run in any GRASS LOCATION whether it is an x,y LOCATION or a standard map coordinate system LOCATION. This means that *r.patch* can be run either before or after image rectification. To run *r.patch*, image files to be patched must be in the same LOCATION. When considering patching, there are several sequence

options. Below, are three examples.

Scenario One

1 **Patch** band *a* of one image to band *a* of another image, and the result of that patch to band *a* of a third image, and so on. The result of the last patch = band *A*.

2 **Patch** band *b* of one image to band *b* of another image, and the result of that patch to band *b* of a third image, and so on. The result of the last patch = band *B*.

3 **Patch** band *c* of one image to band *c* of another image, and the result of that patch to band *c* of the third image, and so on. The result of the last patch = band *C*.

4 **Classify** the bands *A, B,* and *C* normally, using *i.cluster* and *i.maxlik*. This will create a classified image.

5 **Rectify** the classified image (using *i.points* and *i.rectify*).

Scenario Two

1 **Patch** the bands for *n* number of images as in steps 1-3 Scenario One.

2 **Rectify** the patched bands *A, B,* and *C* (using *i.points* and *i.rectify*). The rectified bands = *RA, RB,* and *RC*. *RA, RB,* and *RC* are all separate files.

3 **Classify** the rectified bands *RA, RB,* and *RC,* normally using *i.cluster* and *i.maxlik*.

If you are working on a computer with limited disk space, the following method is recommended.

Scenario Three

1 **Rectify** each image or quad separately by repeating the following process for each image or quad:

- Run *i.points* on the image.
- Run *i.rectify*, selecting as the region option the default region for the target LOCATION (see subsection 13.2.1 regarding selection of the rectifying region). Selection of this region will also eliminate a no-data line that can occur in the image after *r.patch* if the cell header for the patched files was not adjusted properly (see subsection 15.2, **Patching Unrectified Cell Files**), or if the last row in a TM quad was extracted (see subsection 7.2.2., **Row and Column Extraction**), or if an image header record was not skipped in 1600 bpi SPOT imagery (see subsection 7.3.6., **A Word About SPOT**). Rectify each band as a separate file. Use this same target LOCATION default region when rectifying the bands of the other images.
- Remove each raw unrectified band file after rectification. Rectified raw band files can be used for everything for which an unrectified raw band file can be used (*i.composite*, analysis of raw data, etc.). Therefore, unless you are making a comparison of unrectified raw imagery with rectified raw imagery, the unrectified raw data will just take up unnecessary disk space.

2 **Patch** the *like* rectified bands of each image as in numbers 1-3 in Scenario One. The result will be a rectified band *A*, a rectified band *B*, and a rectified band *C*. If the patch is successful, you may want to remove the unpatched rectified bands.

3 **Classify** the rectified bands *A, B,* and *C,* normally using the programs *i.cluster* and *i.maxlik*. The result will be a single classified image. You may want to keep the rectified bands *A, B,* and *C* around for further analysis.

It is not recommended to classify images or quads separately and then patch them, because the clusters (the spectral categories) generated in *i.cluster* may be different for each image or quad. It is also not recommended to patch digitized aerial images at all, either before or after classification, because the density of the film emulsion layers, and therefore the related spectral reflectances of imaged features, may vary from image to image. When using *r.patch* make sure that the region is large enough

to contain the patched image. For information about the actual process of patching see the manual entries for *r.patch*, and the next subsection entitled **Patching Unrectified Cell Files**.

### 15.2. Patching Unrectified Cell Files

Unrectified raster files are those residing in an x,y coordinate LOCATION and are referenced by the rows and columns in which they were stored on tape. Unrectified raster files (classified images, imagery band raster files, band ratio raster files, etc.) to be patched then, will all be referenced with row and column numbers. The cell header for each unrectified file will indicate these rows and columns, as will the mapset region in which they were extracted (although, unless this region was saved or unchanged, it may no longer exist). Whether this region still exists or not, the region an image was extracted in (and therefore its x,y cell header) can be determined by using the *g.region* command.

Unrectified files to be patched may have similar or overlapping reference coordinates. For example, a subset of Quad One band one of a TM image containing the area of interest may have these coordinates in its cell header:

| | |
|---|---|
| **proj:** | **0** |
| **zone:** | **0** |
| **north:** | **-0.5** |
| **south:** | **-3000.5** |
| **west:** | **1000.5** |
| **east:** | **2000.5** |
| **n-s resol:** | **1.0** |
| **e-w resol:** | **1.0** |

and, a subset of Quad Two band one, to be patched to Quad One band one, may have these coordinates in its cell header:

| | |
|---|---|
| **proj:** | **0** |
| **zone:** | **0** |
| **north:** | **-0.5** |
| **south:** | **-2500.5** |
| **west:** | **500.5** |
| **east:** | **2500.5** |
| **n-s resol:** | **1.0** |
| **e-w resol:** | **1.0** |

Notice the overlapping west and east coordinates. If these two bands were patched, they would be patched on top of each other instead of side by side. It is necessary, therefore, to change the coordinates of the bands of one of the quads above by changing the coordinates in the cell header. This is done by running *r.support*.

After typing the command **support**, the first prompt will ask for the raster file for which support files are to be created or edited. Type in the name of the raster file that needs to be changed. For this case, remembering how the TM quads are juxtaposed in the TM image, the Quad Two band one raster file is selected.

The following screen, showing the cell header for Quad Two band one, will be displayed:

**IDENTIFY CELL HEADER**

**NORTH EDGE:-0.5**

**WEST EDGE** **EAST EDGE**

| | | |
|---|---|---|
| **500.5** | **SOUTH EDGE:-2500.5** | **2500.5** |
| | **GRID RESOLUTION**<br>**East-West:1.0**<br>**North-South:1.0** | **COLS:2000**<br>**ROWS:2500** |
| **PROJECTION:0**<br>**ZONE:0** | | |

**AFTER COMPLETING ALL ANSWERS HIT <ESC> TO CONTINUE**
**(OR <Ctrl-C> TO CANCEL)**

To edit the cell header, change the *north*, *south*, *east*, and *west* coordinates that need to be changed so that the files to be patched will be connected at an edge and will not overlap. To patch Quad One band one to Quad Two band one, the Quad Two band one cell header would be changed to this:

**IDENTIFY CELL HEADER**

| | | |
|---|---|---|
| | **NORTH EDGE:-0.5** | |
| **WEST EDGE**<br>**2000.5** | | **EAST EDGE**<br>**4000.5** |
| | **SOUTH EDGE:-2500.5** | |
| | **GRID RESOLUTION**<br>**East-West:1.0**<br>**North-South:1.0** | **COLS:2000**<br>**ROWS:2500** |
| **PROJECTION:0**<br>**ZONE:0** | | |

**AFTER COMPLETING ALL ANSWERS HIT <ESC> TO CONTINUE**
**(OR <Ctrl-C> TO CANCEL)**

Since the two bands to be patched will be joined on the side, Quad One on the left and Quad Two on the right, it is unnecessary to change the north and south coordinates. Only the west and east coordinates need to be changed. Note that the east coordinate of Quad One is the same as the new west coordinate of Quad Two. This will create a seamless patch.

When the changes have been made, hit <ESC> to continue, answer **no** to the other options in *r.support*, and the support program will end. Change the cell headers for all raster files that require it prior to running *r.patch*.

## Table of Contents


1. INTRODUCTION ..... 2
2. THE IMAGERY COMMANDS ..... 2
3. DESCRIPTIONS OF THE IMAGERY COMMANDS ..... 3
4. WHAT IS DIGITAL IMAGERY? ..... 4
5. THE IMAGERY LOCATION ..... 4
   5.1. Creating an X,Y Coordinate LOCATION ..... 5
   5.2. Sample X,Y Default Region ..... 6
6. REGIONS AND MASKS ..... 7
7. TAPE EXTRACTION ..... 7
   7.1. Extracting MSS Imagery From Tape ..... 7
   7.2. Extracting TM Imagery From Tape ..... 12
   7.3. Extracting Other Types of Digital Imagery From Tape ..... 16
8. THE CONTRAST STRETCH ..... 23
9. THE COLOR COMPOSITE ..... 23
10. IMAGERY GROUPS ..... 25
11. TARGETING IMAGERY GROUPS ..... 30
12. IMAGE CLASSIFICATION ..... 31
   12.1. i.cluster ..... 32
   12.2. i.maxlik ..... 34
13. IMAGE RECTIFICATION ..... 36
   13.1. i.points ..... 37
   13.2. i.rectify ..... 42
14. A NOTE ABOUT PROGRAM SEQUENCE ..... 43
15. PATCHING IMAGES ..... 43
   15.1. Patching and the Image-Processing Sequence ..... 43
   15.2. Patching Unrectified Cell Files ..... 45