MANUAL FOR THE USE OF SOFTWARE FOR AGRICULTURAL STATISTICS USING REMOTELY SENSED DATA

MARS-PED

JOINT
RESEARCH
CENTRE

COMMISSION OF THE EUROPEAN COMMUNITIES
Institute for Remote Sensing Applications
Agriculture Project
Ispra (VA), Italy

MAY 1991

Edited by A. Annoni, F. Dicorato, J. Stakenborg

Revision: 20/05/91

INDEX

INTRODUCTION

This introduction describes the interactive software systems for making **agricultural crop acreage estimates** using Remotely Sensed data. These acreage estimation procedures use features previously developed in EDITOR [1].

EDITOR was an interactive file management and image processing system developed by the Center for Advanced Computation in collaboration with USGS/DI, NASA/AMES, and USDA/SRS. The crop acreage estimation software was implemented as part of the EDITOR system, now called **MARS-PED.**

Before describing the structure of **MARS-PED,** it's necessary to briefly describe the **estimation procedure.** A more detailed description can be found in Annex 1 The estimates are produced by **strata,** that is area of land with similar land-use characteristics. Within each stratum, areas of land known as **segment** are selected as sample units in some land use stratum for data collection so they are the basic unit of data.

These segments are scattered spatially throughout the entire strata. The total number of possible segments within a stratum, the population, is the **number of frame units** in the stratum.

The process of delineating strata and segments within the strata is known as the construction of an **area-sampling frame.** STRATIFICATION is most usally done using topographic and thematic maps and satellites images.The information about land use in the selected segments is collected each year by enumerators who visit the segments during the growing season, interview a subsample of farmers for yields and other informations, record the crop or ground cover and draw the boundaries of fields on an overhead with the help of an enlarged aerial photograph.

Using this information, the **field boundaries of the segments** are digitized and then mapped into a geographical coordinate system. With this mapping and the geographical registration of the LANDSAT/SPOT data to a map projection, a mask is generated which maps the fields of the segments into the LANDSAT/SPOT pixels.

Once the masking is completed, pixels may be selected automatically from groups of segments to isolate those meeting certain criteria (especially ground cover) for **training.** In addition, fields may be sampled using statistical techniques. Thus some fields may be used for training and the remainder for testing the results of that training. Finally, tabulation of the categorized data may be done either to evaluate the results or as an input to statistical estimation.

The actual procedures used for clustering and for statistical classification are previously developed functions of EDITOR following LARSYS algorithms and so they will not be discussed in this introduction.

Using the tabulation feature, the ground truth information collected by the enumerators, and the masks associated with the digitized segment data, sample estimation of crop acreages can be performed in the various land use strata using statistical methods. **The segments are the sample and are selected over the region desired.** Selection criteria are used to choose which ground covers will be estimated. The results of sample estimation are used both to aid in evaluating classification performance and as parameters for large scale estimation.

However, before large scale estimation may be performed, an additional digitizing and masking operation is needed. In the same way as for segments, but on a larger scale and using highway maps instead of photos or maps 1:10,000, the land use strata boundaries are digitized. The masks thus generated are applied to the classified data to produce aggregations by category and land use strata. Using these aggregations in connection with parameters derived from the sample estimation a final or large scale estimation is performed of areas in various crops by land use strata and administrative units.

The program uses the word "county" for the administrative units for which the estimates are computed separately. If you to process together the data of a particular region, give the name of the region when the program asks for "COUNTY".

Before proceding further, it's useful to give an overview of the MARS-PED structure and mention briefly the various files used in the estimation process.

MARS-PED consists of a large collection of individual programs. **Each program is self-contained** and may be run separately from any other program. <u>There is no direct passing of parameters</u> between programs. Tipically, MARS-PED program will read in one or more files, process the information, and create one or more output files which may be new files or updated versions of certain of the input files. A few MARS-PED programs only display the contents of files [2].

MARS-PED provides a main program allowing the user to call various programs using simple commands. Each program must specifies the names of input and output files. In many cases, particular types of files have standard names so that the user need not enter file names explicity. This is particurarly useful for programs which must process a large number of files of the same type.

The following files are required using MARS-PED for estimation (they are generally created and used during the procedure although this order may be changed):

1 **Segment catalog file**, containing various attributes for each segment such as <u>county, stratum and analysis district</u>.

2 **Frame unit file**, containing the <u>number of frame units</u> by county and strata.

3 **Frame calibration file**, containing <u>polynomial coefficients for transformation</u> between map and LANDSAT/SPOT coordinates.

4 **Segment network** containing the <u>digitized field boundaries</u>. The strata network file contains digitized strata boundaries in counties.

5 **Mask file** containing an overlay of the segment field or county strata boundaries into the LANDSAT/SPOT coordinate system.

6 **Ground truth file**, one for each segment, containing for each field, the field name, size, cover, etc.

7 **Totals file** containing a single ground data value for each segment and an option selection such as cover type.

8 **LANDSAT/SPOT data window file** containing all the LANDSAT/SPOT data for a segment.

9 **Packed LANDSAT/SPOT** containing only those pixels corresponding to <u>selected ground covers</u> for a group of segments.

10 **Statistics file**, containing the means and variance-covariance matrix for various categories of LANDSAT/SPOT data, usually with one or more categories representing a particular cover.

11 **Categorized LANDSAT/SPOT data files**, both window and packed, containing a category for each LANDSAT/SPOT pixel. This is typically a classified window.

12 **Table file**, containing a tabulation by segment of pixels by category and reported cover.

13 **Estimator parameter file**, containing parameters from sample estimation to be used in large-scale estimation.

14 **Aggregation file**, containing a tabulation, generally for a county, of pixels in each strata and category.

15 **Estimator results file**, containing the estimates and variances computed by large-scale estimation.

As will be seen, these files are created and read by MARS-PED programs.

For each segment there are five files:

.segment network file,
.mask file,
.local shift file (1 for group of segments),
.ground truth file,
.window file of LANDSAT/SPOT data (1 for group of segments).

The **segment network file** contains the field boundary coordinates and is the direct result of digitization.

The **local shift file** contains the parameters to horizontally and/or vertically shift the segment from the location predicted by the global geographic registration of the image to its actual location (determined by hand registration of each segment to displays of LANDSAT/SPOT data). All data of all shifts are in one file (shiftfile).

The **mask file** results from this process of segment to LANDSAT/SPOT data registration. It is a mask assigning the pixels to the various fields in the segment.

The **ground truth file** is a computer readable version of the ground truth information collected by the enumerators for a segment.

The **window file** consists of a LANDSAT/SPOT data for a rectangle containing a particular segment. All data are in one file (multiwindow).

The files are assigned default names based on the segment number for a convenience in processing. In addition, several window files may be combined into one large file, called a **multi-window file**, for convenience of handling.

SEGMENT DIGITIZATION

As we seen before, a **segment** is an area of land. It will usually be divided into several (perhaps noncontinguous) ownership **tracts** (a tract is the set of fields in a segment that belongs to a farm. If you are not using farm information, give a dummy answer to the question "Tract :", for instance "A"), and further subdivided into **fields**. Each field represents an area on the ground to be considered homogeneous with respect to ground cover, The set of these areas do not overlap. There is no restriction (other than computer file storage space limitations) on the complexity of the field boundaries.

Segment digitization is the process of converting segments from fields drawn on aerial photographs or topographic maps to a file of coordinates in a geographic coordinate system. Location of points on the images or maps are measured by hand using a data tablet digitizer, in conjunction with interactive software subsystems, and assembled into a convenient computer-readable data structure.This data structure contains all the topological, geographic, and naming information needed to completely reconstruct the segment.

We will describe the process of digitizing agricultural segments (digitization of land-cover boundaries within both county-wide land-use strata maps and LANDSAT/SPOT frame-wide classified images).

SOURCE DATA

Aerial photographs are expected to be sufficiently orthographic that a simple linear transformation will serve to map ground points in the photo to geographic coordinates.

Some large segments require several photographs to cover completely the ground area of the segment. To avoid the problem of matching up partial segments digitized from different photographs, we simply consider segments (and partial segments) on a one-photo-at-a-time basis, but this situation should be avoided choosing a suitable scale for the photograph..

Field boundaries and field numbers are drawn directly on each photograph or on a plastic overlay. These boundaries and charges to them were determined on location by enumerators. Boundaries of fields that changed between visits were appropriately added using a color-coded marking scheme and the ground-truth information on crop cover and field acreage updated. The field boundaries thus recorded have proved to be *accurate to at least 15 meters* on the ground, in the central part of the original photograph, if the land is not too hilly.

Field boundaries should be drawn as black lines on transparencies and filed numbers must be written in red. This speeds up the digitization of segments with the video camera (TTS).

SEGMENT DATA STRUCTURE

A simple representation of the tract and field boundaries associated with a segment has proved convenient to use. Curved edges of a field are approximated by straight-line pieces, and fields themselves are a collection of one or more polygons. **Fields with "holes"** (e.g., ponds within pastures) **are stored as two or more polygons**, one for each separate boundary.

For use with the video system these island polygons have to be connected with a simple line to the outer polygons. This linepiece is then removed during conversion to segment network file.

The outer boundary in such a case is denoted by a polygon whose vertices travel <u>clockwise</u> around the field. Inner or hole boundaries have polygons with vertices travelling <u>counterclockwise</u>.

For adjacent polygons, the edges in common are stored only once. Thus, the data structure for each segment is really a **network of edges delineating the polygon** boundaries of the fields.

THE SEGMENT NETWORK DIGITIZATION

Segment digitization is done using one of two methods: manual digitization or video digitization.

For **manual digitization**, a digitizing tablet is used. Each field is assumed to be a polygon and the vertices are marked on the image. The fields are digitized by placing the cursor at each vertex and recording the vertex location. The resultant segment network file is a collection of edges and vertices representing the polygon digitized.

For **video digitization**, the registration of the segment is done by means of a video camera. The field boundaries are traced onto clear acetate. The image is then captured into a raster frame buffer using a video camera. The boundaries are thinned in the sense of removing excess boundary pixels but never letting previously separate fields be joined. Using the graphics system, the image is displayed and fields are labelled interactively by the user. The result is a polygon file in the coordinates of the graphics system. These simple ploygon files are then converted to network files.

NETWORK CALIBRATION AND REGISTRATION

Segment registration is done during manual digitization. Network coordinates are digitized directly in map coordinates with an affine transformation.

Registration of the video digitized network is done during conversion.

The parameters of the transformation are computed by a least-squares fitting procedure. Corresponding points are denoted on both map and segment and coordinates pairs obtained are used to compute the transformation coefficients. Maximum and root-mean-square errors are reported to the user on the terminal. Then the user may accept the calibration or proceed to obtain a different set of control points, based on the magnitude of the errors. RMS errors will be relatively large if the corresponding points do not truly correspond geographically or were digitized inaccurately.

Each LANDSAT/SPOT image is also separately geographically registered. Together, the segment and LANDSAT/SPOT registration transformations yield a mathematical mapping from digitized segment coordinates to LANDSAT/SPOT pixels. The accuracy of this composite transform may be checked visually by overlaying LANDSAT/SPOT coordinate plots of the segment on X-terminal of the appropriate LANDSAT/SPOT window. Since individual fields are usually visible within the plot, it becomes possible to specify the small shift corrections that may be necessary for the LANDSAT/SPOT plot to overlay the fields exactly. The program **XSFT** is available for specifying these shifts and generating **local shift files** storing the corrected X and Y shift of each segment to LANDSAT/SPOT transformation. On the PC the program **PCSHIFT** is avaible.

MASK GENERATION

After the registration of each segment with respect to the LANDSAT/SPOT image has been verified and any necessary local re-registration performed, it is possible to determine the corresponding LANDSAT/SPOT pixels for any field of a segment. This mapping information is generated and stored in the **segment mask file**.

The mask file generator program can also flag boundary pixels, i.e., those pixels that lie on field boundaries and are thus mixed spectrally with respect to the signatures of the two adjacent fields. Mask files are generated for each segment and LANDSAT/SPOT frame pair using a scan-conversion procedure on the digitized field polygons. The resulting set of field-pixel correspondences are then compressed via run-length encoding.

SELECTION, SAMPLING, AND TABULATION OF FIELDS

The selection of ground truth samples for **classifier training** involves several processes. First, a file may be created containing only pixels which meet certain criteria, such as having a particular ground cover (or one of a group of covers), and belonging to a certain group of segments. Such a file is called a **packed file.**

An alternative way to create a packed file is to sample statistically from all fields which meet the criteria. This permits classification results to be evaluated with respect to the remainder of the fields. Also, once clustering or classification has been done, a feature is available for tabulation of pixels by the various classification (or clustering) categories and the ground covers actually recorded for the fields so that the results may easily be checked. This tabulation process will also produce a table file which can then be used for estimation as will be described below.

SELECTION CRITERIA

Two types of selection must be performed. First, a region or collection of segments to process must be chosen by the user. Second, criteria (called "options") must be selected. These two phases are named SELECT REGION and SELECT OPTIONS respectively.

A **region is a collection of segments** which will be processed in the course of a particular analysis. However, to avoid the tedium of entering many segment numbers, a short notation is available for describing groups of segments having common attributes, provided that a special **catalog file** has been created which lists the attributes of the segments of interest. The attributes include the LANDSAT/SPOT frame(s) containing the segment, the land-use stratum to which the segment belongs, and the county in which the segment is found. An implicit assumption is made that a particular catalog will contain segments for one state only. Listing the attribute and appropriate parameter will then generate a list of segments satisfying that condition.

Once the collection of segments has been specified using SELECT REGION, the options specified under SELECT OPTIONS are used to choose processed. Again, to avoid entering by hand numerous field numbers, a shorthand notation is available allowing specification of field attributes, e.g. ground cover. These attributes are then used in selection of fields by checking the various fields in the ground truth file against the specified criteria for each segment processed.

The **ground truth file** is a computer readable representation of the ground survey data collected for a segment by enumerators. In generating these ground truth files, careful checking is done to ensure consistency with the results of segments digitizations. Among the data collected for each field in the segment and stored acreage, field appearance, intended use of crop, and date of field survey observation.

The user inputs SELECT REGION and SELECT OPTIONS statements using a simple Boolean expression language having the operators AND, OR, and NOT and allowing parentheses to facilitate grouping. A selection statement is such a Boolean expression followed by the terminator symbol "#".

SELECT REGION

SELECT REGION has the following operands: SEGMENT, FRAME, COUNTY, and STRATA. Each of the operands is followed by an appropriate parameter:

> SEGMENT followed by a segment number;
> COUNTY followed by a county name;
> FRAME followed by a frame number or name;
> STRATA followed by a land-use stratum number.

To get the segments desired, these basic descriptor variables are combined using AND, OR, or NOT.

Thus the expression "FRAME 2194-16042 AND STRATA 11" indicates all segments which are in the frame 2194-16042 and also in land-use stratum 11. On the other hand, "COUNTY CHAMPAIGN OR COUNTY DOUGLAS OR COUNTY MOULTRIE" generates segments which are in (any) one of Champaign, Douglas, or Moultrie counties. In fact, we may abbreviate the expression as "COUNTY ONEOF (CHAMPAIGN, DOUGLAS, MOULTRIE)" or more simply "COUNTY (CHAMPAIGN, DOUGLAS, MOULTRIE)".

Combining the above to obtain a more complex expression, we obtain "FRAME 2194-16042 AND STRATA 11 AND COUNTY (CHAMPAIGN, MOULTRIE, DOUGLAS)" which generates all segments in frame 2194-16042 and strata 11 and one of the counties Champaign, Moultrie, or Douglas.

SELECT OPTIONS

SELECT OPTIONS has a similar Boolean expression language with the operands COVER, ALL, BACKGROUND, FIELD, TRACT, SIZE, APPEARANCE, and INTENDEDUSE. The operands ALL (the entire area covered by the mask for a segment) and BACKGROUND (areas outside the boundaries of all digitized fields) have no parameters. The remaining operands must be followed by a parameter as follows:

COVER	followed by a ground cover name;
FIELD	followed by a full field name including the segment number, the tract (one or two letters), and the field number within the segment and tract;
TRACT	followed by a segment number and tract (one or two letters);
SIZE	followed by a relational operator and size in acres;
APPEARANCE	followed by an appearance code name;
INTENDEDUSE	followed by an intended use code name.

Except for FIELD, these return all fields selected from the various segments generated by SELECT REGION. FIELD gives a specific field in one segment and is used chiefly for packing using the fields generated by field sampling.

In SELECT OPTIONS additional operators are available to handle **boundary pixels**. To improve classifier training, it is often useful to eliminate boundary pixels. To eliminate such default is to include boundaries. For completeness, the operator "+" is provided to specify explicitly the inclusion of boundary pixels.

As an example of SELECT OPTIONS usage, "COVER CORN" or equivalently "+COVER CORN" returns all fields with a ground cover of corn and will also return the boundary pixels when generating a packed file. To exclude boundary pixels, we would use "-COVER CORN".

As in SELECT REGION, the ONEOF construct may be used so that "-(COVER CORN OR COVER WHEAT OR COVER SOYBEANS)" may be entered simply as "-COVER(CORN, WHEAT, SOYBEANS)".

Of course, we may use the AND operator as in "-(COVER CORN AND APPEARANCE MATURE)" to indicate all fields with a cover of corn and an appearance code of mature, all without boundary pixels.

PACKING

A **packed file** is one which contains pixels of either raw or categorized data taken from a collection of segments obtained using SELECT REGION and from fields meeting the criteria specified within SELECT OPTIONS. These pixels may then be used for any kind of further analysis as representing only pixels of those categories of interest. Spatial relationships between pixels in a packed file are lost; however, sufficient information is retained in the packed file to restore the spatial relationship if needed; this is done by unpacking a packed file.

The packing process requires a mask and a ground truth file for each segment processed as well as a data (**window**) file from which the pixels to be packed will be extracted. The files are given agreed-upon default names used by this and other programs in the MARS-PED system to avoid user entering of several file names for each segment. In addition, the data or pixels to be used for several segments may be placed in one file, known as a **mullti-window file**, for convenience in handling and to save disk space.

The packing process then consists of using SELECT REGION to get a list of segments and then using SELECT OPTIONS code on each segment to generate a list of fields which meet specified the segment, is used to determine which pixels fall within the specified fields. Pixels thus masked are moved into the packed file.

TABULATION

Once categorization (clustering or classification) has been performed on packed files of LANDSAT/SPOT data, it is often useful to see a tabulation of the categories and the various ground covers. For this purpose, the TABULATE option is used on a packed categorized file. Tables may be generated individually for each segment or for all segments in the region combined. The rows of the table displayed represent verified ground covers and the columns spectral categories. This allows checking the accuracy of the categorization. The tabulation process uses the mask and ground truth files along with information representing the user-entered SELECT REGION and SELECT OPTIONS to assign the pixels to the various fields. thus, the correspondance between the category (of the pixel in the packed categorized file) and the ground cover (of the field as recorded in the ground truth file) is obtained.

In addition to displaying the table, it is also possible to create a table file containing the tabulated information. Such a **table file** may then be used to produce another tabular printout showing the percentage of correct categorization by cover type. Such table files are also saved for later use as one of the inputs to the sample estimation procedure to be described below.

ACREAGE ESTIMATION

This section describes the software available to do acreage estimation based on a stratified sampling design. The mathematical basis of the system will be described as well as its implementation details.

The acreage estimation command has two main functional subcommands which perform the following functions:

1. Sample estimation
2. Large scale estimation

We will describe these functions together since they depend on the same statistical theory.

Sample estimation computes regression, ratio, and reciprocal of probability of selection (commonly called direct expansion) estimates for any region (set of segments) selected by the user with SELECT REGION. The region selected is usually all segments in a set of contiguous counties. Acreages may be estimated for any cover type selected by the user with SELECT OPTIONS. SELECT OPTIONS works as described above except that any input relating to boundaries are ignored so that a particular field enters into the acreage computation if its interior, boundaries, or both are returned.

A table file is necessary to do the regression and ratio estimation. Any or all land use strata may be pooled and some, such as non-agricultural or urban, may be deleted.

When a regression or ratio estimator is specified, the sufficent statistics (means, variances by strata or pooled strata) may be output to a file called an **estimator parameter file** for use in large scale estimation.

Large scale estimation is used to estimates acreages for large geographic regions using all LANDSAT/SPOT imagery for that area and not just a sample. Printed output from sample estimation is as follows:

1 coefficient of determination (r-square),

2 an estimate of the variance of the estimated acreage for the cover type in the segments selected using SELECT REGION, and

3 the relative statistical efficiency of the regression estimator vs. the direct expansion estimator for the dependent variable.

IMPLEMENTATIONS DETAILS

The direct expansion option of sample estimation is capable of using three different variables to estimate the acreages for a region selected by the user:

1 **Digitized acreage** - This is the acreage computed by digitizing each field from an aerial photograph and having the crop type specified from the ground-truth file. This acreage is computed and stored in the mask file.

2 **Reported acreage** - This is the acreage reported by enumerators and stored in the ground-truth file.

3 **Pixel acreage** - If the user so desires, all pixels classified into a single category or several categories from a table file may be used to estimate the acreages for the region selected. A user supplied constant is needed to convert pixels to hectares.

The computations performed for the direct expansion options as well as the ratio and regression options will be given below. For the segments generated by the SELECT REGION specified, let

L = total number of different strata to which the segments generated belong

N_h = total number of frame units in the SELECT REGION, for the **h**-th strata.

The set of all possible segments (called **frame units**) for a state is stored in a file called the frame-unit file by county and strata.

$$N = \sum_{h=1}^{L} N_h$$ = total number of frame units (over all strata),

n_h = total number of segments selected using SELECT REGION for the **h**-th strata,

$$n = \sum_{h=1}^{L} n_h$$ = total number of sample segments (over all strata) generated,

$$y_{hj} = \begin{cases} \text{Total digitized acreage for a particular cover types selected for the j-th segment in the h-th strata.} \\ \text{OR} \\ \text{Total reported acreage for a particular cover types selected for the j-th segment in the h-th strata} \end{cases}$$

x_{hj} = total number of pixels corresponding to the cover type selected for the **j**-th segment in the **h**-th strata.

The set of pixels corresponding to the cover type(s) selected using SELECT OPTIONS is stored in a table file for each segment (described above). The user must specify the table file name and the category number(s) associated with the cover types(s). The formulae here in after are applied separately for each land cover c, hence we omit the corresponding index.

The direct expansion estimate of the total acreage in the region for a particular cover type is:

$$\hat{Y} = \sum_{h=1}^{L} \frac{N_h}{n_h} \cdot \sum_{j=1}^{n_h} y_{hj}$$

and the estimated varaince of $\hat{Y}$ is:

$$Var(\hat{Y}) = \sum_{h=1}^{L} \cdot \frac{N_h^2}{n_h(n_h-1)} \cdot \frac{N_h-n_h}{N_h} \cdot \sum_{j=1}^{n_h} (y_{hj} - \bar{y})$$

A ratio estimate of the total acreage in the region for a particular cover type is:

$$\hat{Y}_{RATIO} = \sum_{h=1}^{L} (\bar{y}_h \;/\; \bar{x}_h) X_h = \sum_{h=1}^{L} r_h X_h, \qquad \text{where } r_h = \bar{y}_h \;/\; \bar{x}_h$$

This ratio estimate is computed in the large scale estimation process.

Note that X_h is the total pixels by stratum in an entire large region, typically a county or group of counties, classified into the categories corresponding to the cover type(s). The computation of totals by stratum with a strata mask is called **aggregation**. The aggregation process is similar to the tabulation process described earlier: however larger mask files must be handled for a large data sets and thus the distinction between these two process.

The results of aggregation are stored in an aggregation file.

The variance of the ratio estimate is:

$$Var(\hat{Y}_{RATIO}) = \sum_{h=1}^{L} \frac{N_h(N_h-n_h)}{n_h} \cdot \left[S_{h,y}^2 + r_h^2 S_{h,x}^2 - 2\, r_h\, \rho_h\, S_{h,y}\, S_{h,x} \right]$$

where,

ρ_h = sample correlation coefficient between **x** and **y** for the **h**-th strata,

$S_{h,y}^2$ = sample variance for the **h**-th strata for the **y** variate,

$S_{h,x}^2$ = is similary defined.

The regression estimate for a SELECTED REGION for a particular cover is:

$$\hat{Y}_R = \sum_{h=1}^{L} N_h \, \bar{y}_{h(reg)}$$

$$\bar{y}_{h(reg)} = \bar{y}_h + \hat{b}_h \, (\bar{X}_h - \bar{x}_h)$$

$$\hat{b}_h = \frac{\sum_{j=1}^{n_h} (x_{hj} - \bar{x}_h)(y_{hj} - \bar{y}_h)}{\sum_{j=1}^{n_h} (x_{hj} - \bar{x}_h)^2}$$

and the estimated variance is:

$$Var(\hat{Y}_R) = \sum_{h=1}^{L} \frac{N_h^2}{n_h} \frac{N_h - n_h}{N_h} \sum_{j=1}^{n_h} (y_{hj} - \bar{y})^2 \frac{1 - \rho_h^2}{n_h - 2}$$

Note that $Var(\hat{Y}_R)$ is computed from only the sample and from the frame unites in the region selected (wich are known apriori); thus <u>classification strategies that minimize this variance (rather than maximize the percent correct) can be determined</u>.

Of course the computation of X_h, the mean number of pixels classified into the categories for the cover of interest, is for the population and not just the sample. Thus large subsets corresponding to political boundaries such as counties must be classified and aggregated by strata to compute these estimates. The estimation of acreages using aggregation files is done using large scale estimation.

In large scale estimation an option exists for regression estimation with subsets of the area originally selected. With this option one can estimate the acreages for an individual county and estimate the variance of that estimate. Since the regression estimator is that the individual counties add to the total for the SELECT REGION.

The regression estimate for the acreage of an individual county or a proper subset of counties is:

Let $S_{h,y}^2$, $S_{h,x}^2$, $\hat{b}_h$, $\sum_{i=1}^{n_h} x_{hi}^2$, $\sum_{i=1}^{n_h} x_{hi}$, n_h be computed from the SELECT REGION sample describes earlier, and let :

$N_{h,c}$ = total number of frame units in the h-th strata for the subset of c counties in the SELECT REGION, and let

$\bar{X}_{h,c}$ = total number of pixels classified as a cover for the subset of c counties divided by $N_{h,c}$.

$X_{h,c}$ is computed by aggregating the individual aggregation files for each county. Then, the estimate of the total for the set of c counties is:

$$\hat{Y}_{REG,c} = \sum_{k=1}^{L} N_{k,c} \left(\bar{y}_k + \hat{b}_k \left(\bar{\bar{X}}_{k,c} - \bar{\bar{x}}_k \right) \right)$$

And the variance of this estimate is:

$$Var(\hat{Y}_{REG,c}) = \sum_{h=1}^{L} N_{h,c}^{2} \cdot \frac{N_h - n_h}{n_h} S_{h,y}^{2} \frac{n_h - 1}{n_h - 2} (1 - \rho_h^{2}) \left(I(c) + \frac{1}{n_h} + \frac{(\bar{\bar{X}}_{h,c} - \bar{\bar{x}}_h)^2}{\sum_{h=1}^{n_h} (x_{h,i} - \bar{x}_h)^2} \right)$$

where:

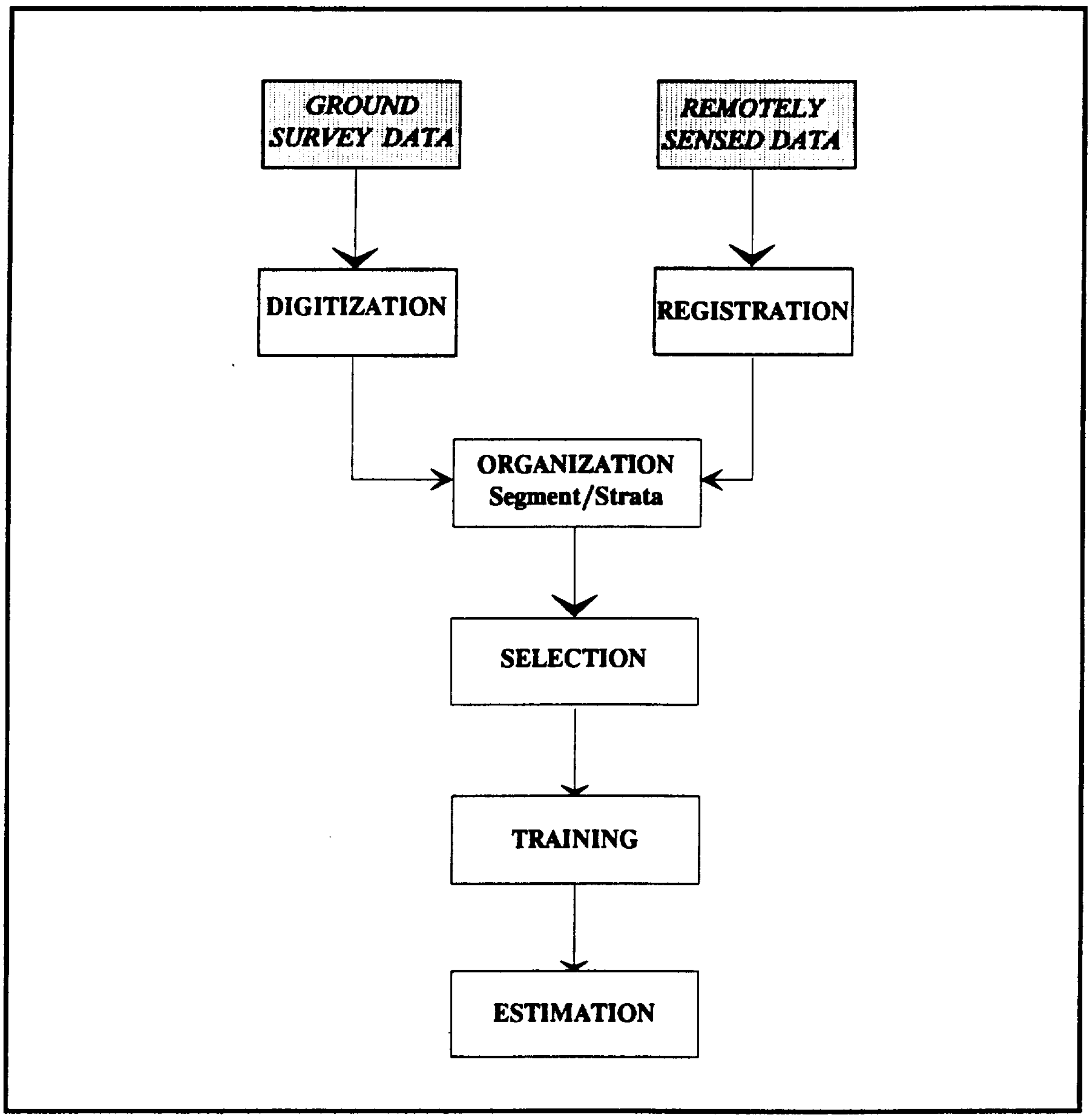
I(c) = 1 if 0(c) <total number of counties wholly contained in the SELECT REGION
= 0 otherwise

0(c) = is the cardinality of the set c.

PROGRAMS OVERVIEW

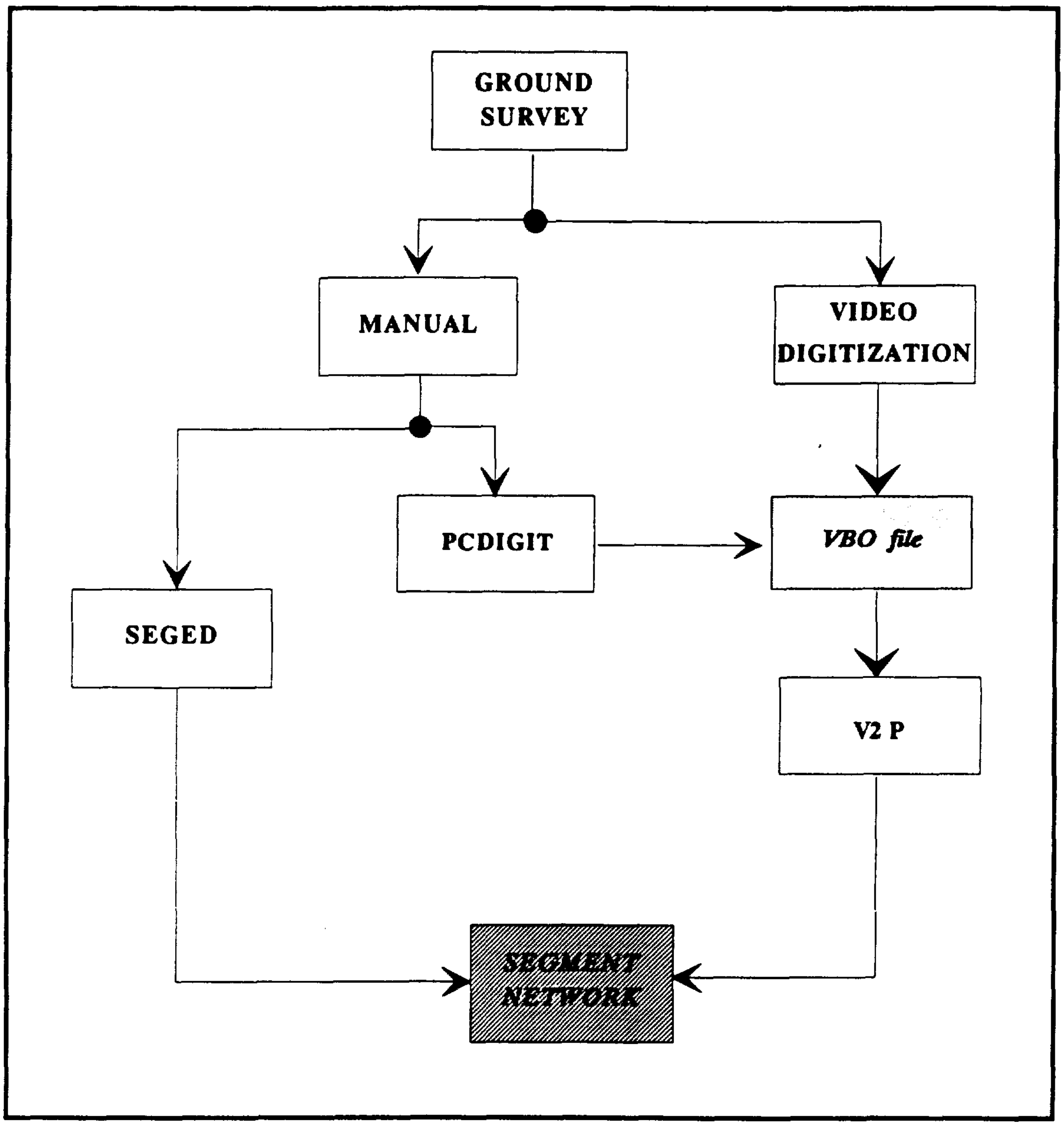
A short description of the different steps in using MARS-PED is shown in the following flow-charts.

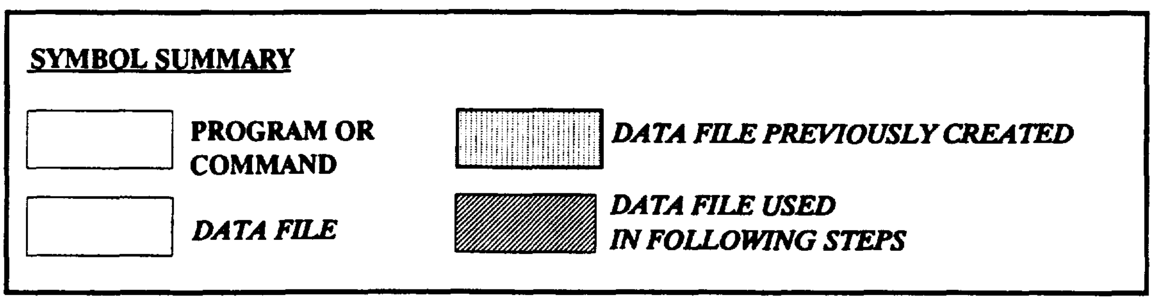
MARS OVERVIEW SCHEME



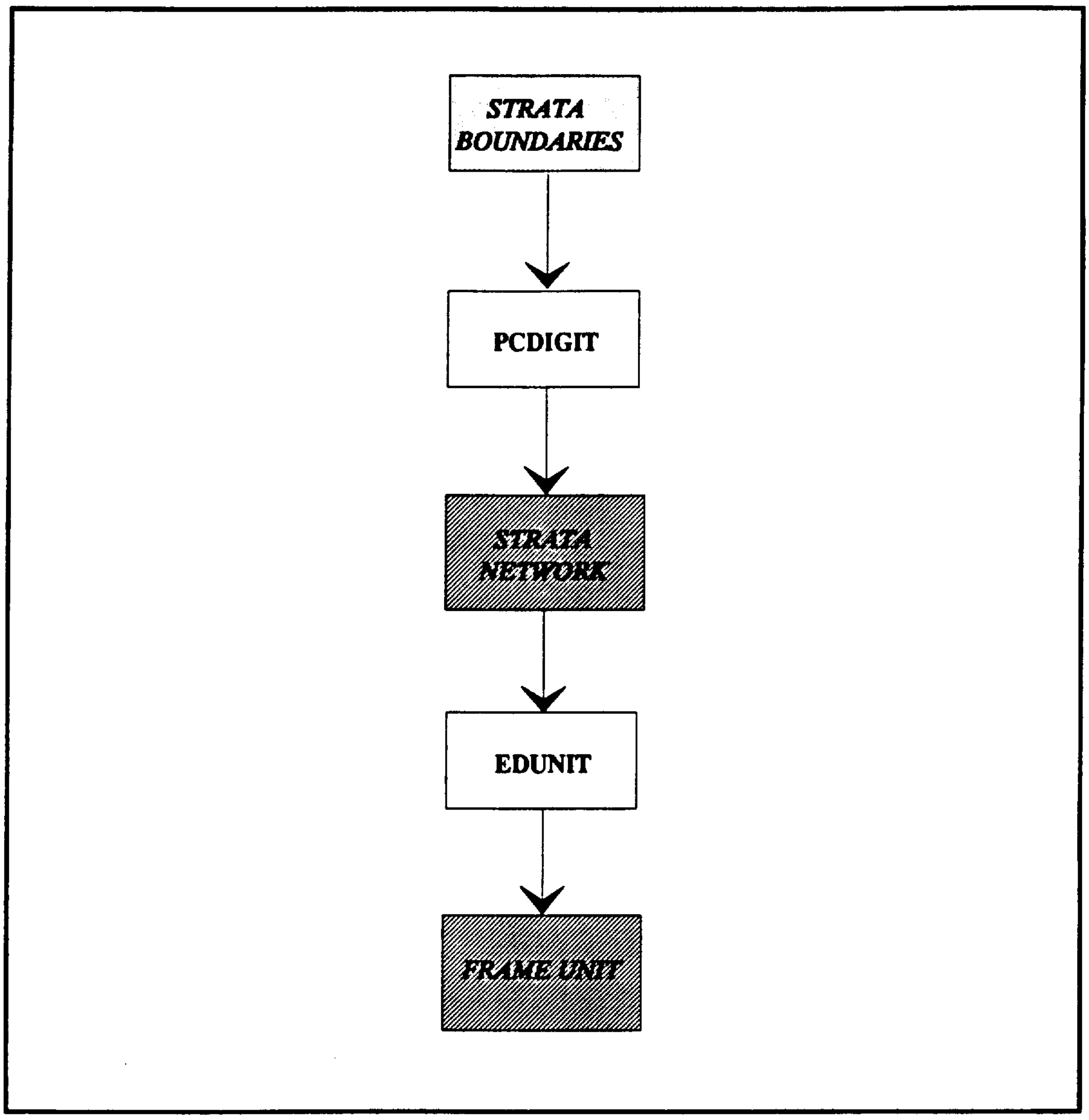
DIGITIZATION

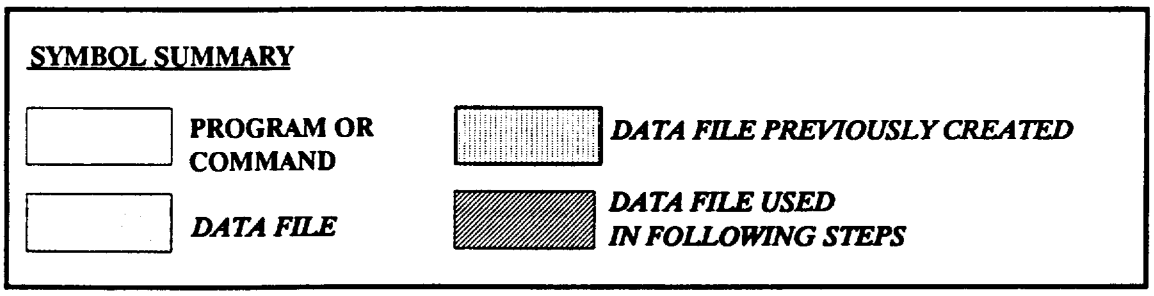
SEGMENT DIGITIZATION SCHEME





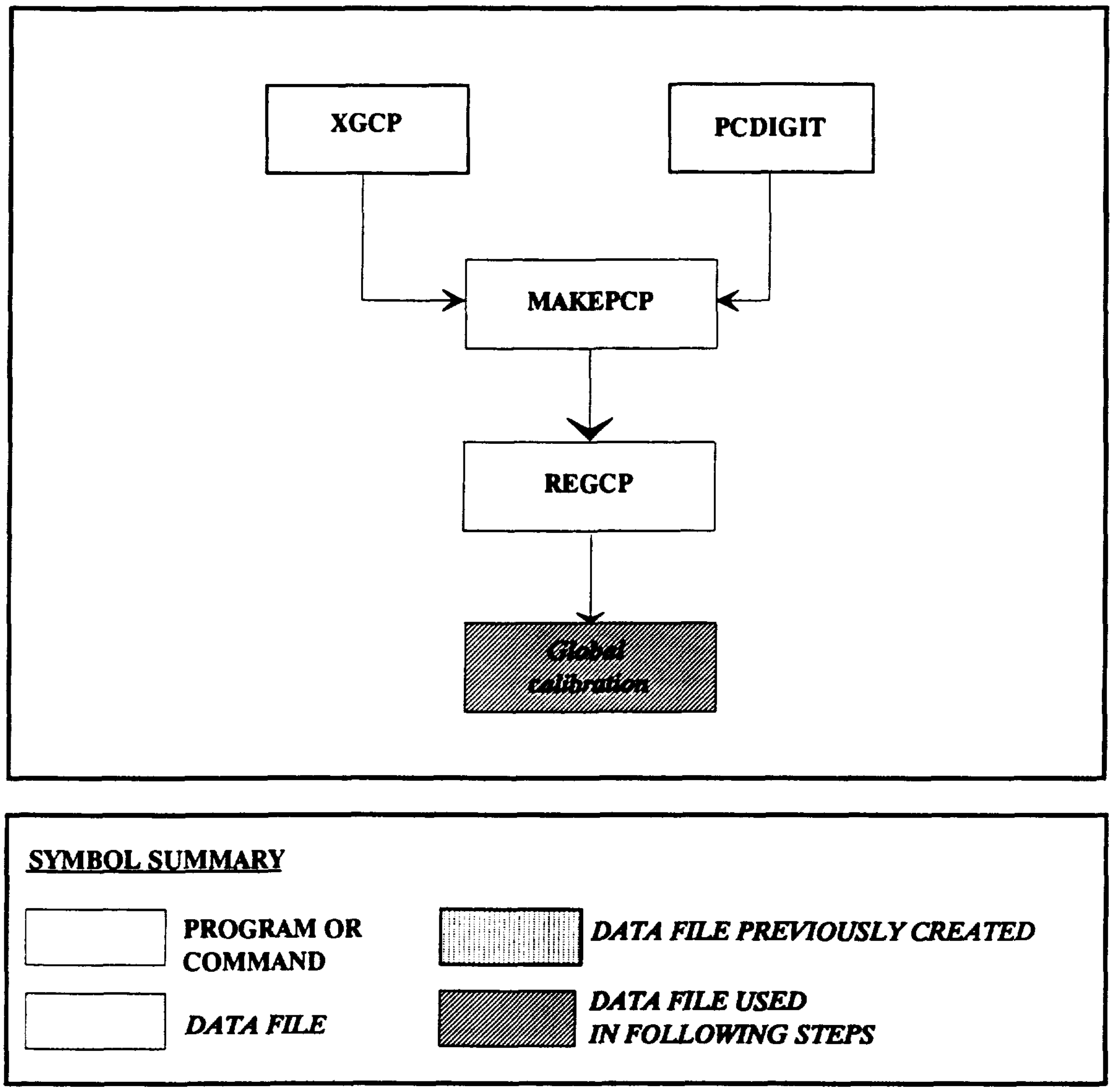
STRATA BOUNDARIES DIGITIZATION SCHEME





After the creation of a STRATA NETWORK file the FRAME UNIT file must be created.
It contains some basic informations about the strata. The EDUNIT program is used to create the file. This program can also update the Frame Unit File in the Strata Organization phase in order to add some other information useful in Estimation step.

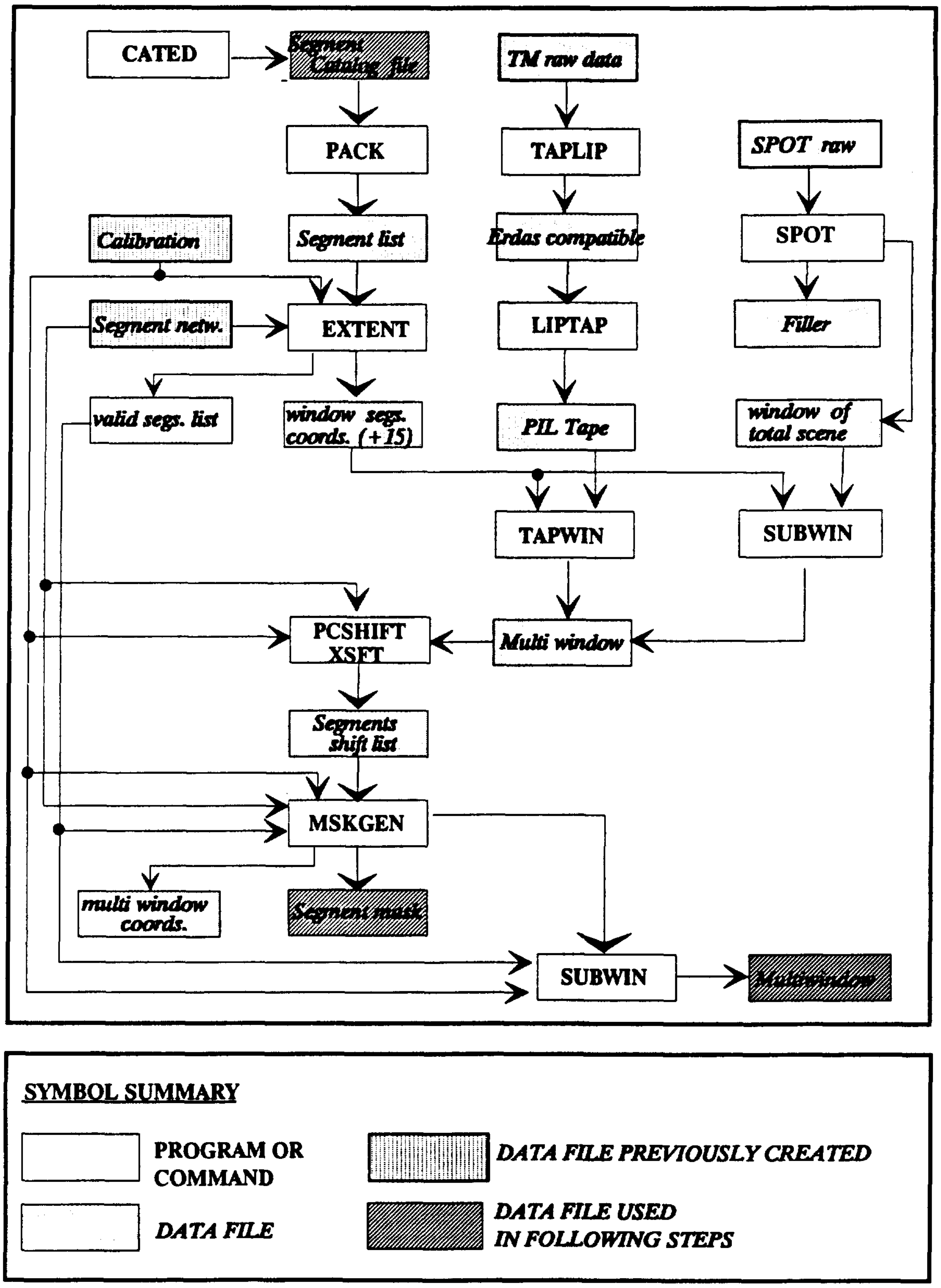
IMAGE REGISTRATION SCHEME (Map to Image)



ORGANIZATION

After the digitization ground truth data file, segment network and calibration files must be present for data organization that is splitted in two levels: **segment and strata.**

SEGMENT ORGANIZATION SCHEME



SEGMENTS DATA BASE

To create or edit a **segment catalog** containing information and characteristics of user-defined segments the program **CATED** must be used; in this way, groups of segments with common characteristics may be easily accessed by other programs.

Output data are in the Segment Catalog File (MARS-PED file type 49).

The following step in segment data organization is the creation of a segment list from the catalog edited before.The commands LIST OF SEGMENT CREATION and SELECT REGION (both from the program **PACK**) are used here.

LIST OF SEGMENTS FILE CREATION creates a list of segments in the specified output ASCII file. This list of segments is suitable for use in a SELECT REGION statement or in most places where a list of segments is requested.

The SELECT REGION statement is used to select segments by either explicitly specifying those segments or using attributes of the segments to find them in a segment catalog file. The SELECT OPTIONS statements is used to select fields, and hence segments in those fields, based on attributes of the fields in the ground truth files.

FINDING THE LOCATION IN THE IMAGE

After the list creation **EXTENT** program must be used to determine image file window extents of a group of digitized segments, and optionally produce a segment region file and a window text file of all segments that are wholly contained in the image file.

Input data for EXTENT program are a global calibration file name, and list of Segment numbers. Output data are summary listing on the console (and in a file) of each segment, with those segments not contained in the image file flagged as outside, a segment region text file (optional) and a segment window text file (optional).

The EXTENT routine requests the name of a global calibration file and information regarding the associated image file, and a list of segment numbers. There are several options that can be specified:

a. Output a region file containing the segment numbers of all segments that are wholly contained in the image file.

b. Output a window text file containing the row/column extents and the segment number for each contained segment.

c. If a window text file is requested, a border of specified size can be placed around the segment extents (1 up 15 pixels) wich is useful to be able to shift later the network over the window.

Once all options have been selected, the routine reads the global calibration file, then for each segment it will read the associated segment network file (MARS-PED file type 47), convert all vertices in the segment network file to row/column values, and determine the extent of the rectangular window that surrounds the segment. The information about the segment, including its window, will be displayed on the console and, optionally, all contained segments will be entered in the region file and in the window text file. The windows written to the window text file will be expanded by the border, if any.

For copying the selected portions of image data from tape to disk window file the programs **TAPWIN** or **SUBWIN** must be run. Input for TAPWIN are magnetic tapes containing image data in a recognizable form, in fact TAPWIN works on a PIL TAPE obtained from TM raw data handled by TDPLIP and LIPTIP programs. Output file is a Window file (MARS-PED file type 51); the name of the file must be supplied by the user.

CREATING THE SUB-IMAGES (MULTIWINDOWS)

The user is asked to enter the type of input tape to be used. TAPWIN obtains various tape parameters (e.g.,tape format, tape record length,image size) either from the tape header, or in the case of headerless tapes, directly from the user. Some tape parameters, such as the number of channels to be used in the windows or the row and column sampling increments, can be altered via user commands.

When more images are on one tape one has to use the appropriate system command to move to the requested file.

TAPWIN orders the user-specified window list by (supplied by the window segments coordinates +15 file) the northwest corner so it can make just one pass over the tape. When the windows are to be written to the disk window file, TAPWIN uses a separate procedure for each tape data format. For writing band-interleaved by line windows to disk, TAPWIN reads each row on the input tape, and if it is contained in one or more of the user-specified windows, it is written to the appropriate places in the window file. TAPWIN finishes by rewinding the input tape.

If the remotely sensed image is of SPOT type, SUBWIN must be run instead of TAPWIN. SUBWIN works on a one window file created by the SPOT program that process SPOT raw data. SPOT program creates also a filler data file of the image.

LOCAL ADJUSTMENT AND FINAL REGISTRATION

When a scene is registered on disk, a relationship is formed between map projection on the earth and row-column in the LANDSAT/SPOT scene. This relationship is used to predict the pixels in the scene that match the ground truth data. However, this prediction is not perfect. The purpose of segment shifting (sometimes called local calibration) is to adjust the predicted location of the segment within the LANDSAT/SPOT scene to match the ground truth data as closely as possible. Segment shifting is an important step in the analysis because the precision of the final results can hinge on how well the segments are located.

Shifting can be done on a PC with a VGA (or super-VGA) monitor using the program PCSHIFT or an X-terminal (DECWindow terminal) using greyscale or pseudo colors with the program XSFT.

Whatever method is used, it will produce a list of segment shifts. This file of shifts is then used to create a final segment mask and to clip the 15 pixel border window file to a 1 pixel border file for the remaining analysis process.

TAPE CONVERSION PROGRAM

Program diagram for reading raw data.

For SPOT we espect Band Interleaved by Line (BIL) format.

For TM we espect Band Sequential (BSQ) format.

Input	**SPOT**	**TM-Full**	**TM-quarter**
Program or Command	run SPOT	run TAPLIP @ TMF	run TAPLIP "@ TMQ
Output	WINDOW FILE	' ERDAS 7.2 IMAGE FILE	' ERDAS 7.2 IMAGE FILE

Program SPOT reads a SPOT tape and dumps an Multi-Window file to disk. One should use the XGCP, REGCP, EXTENT, and SUBWIN program to extract the Multi-Windows of the segments before backing up the big image.

The program TAPLIP with the subcommand @TMF reads three Landsat full scene tapes and dumps the channels 1,2,3,4,5 and 7 to disk in an ERDAS 7.2 file format.

The program TAPLIP with the subcommand @TMQ does the same but dumps all channels from a quarter scene TM to an ERDAS 7.2 image file.

One must use XGCP, and REGCP before executing LIPTAP with subcommand @TOTAPE that stores the ERDAS image to a Pixel Interleaved Tape (MARS-PED format to be read with TAPWIN). With TAPWIN you can create a Multi-Window file of the segments, using this Pixel Interleaved Tape.

To move to the specified file on tape one can use the DCL command
"SET MAGTAPE/SKIP=FILES:n device:"
after rewinding it with "SET MAGTAPE/REWIND device: ".
This is normally not necessary.

MASK GENERATION OF SEGMENTS

Now a mask file from a segment network file hasto be generated The program that performs this work is MSKGEN and as input it requires:

1) Global calibration file (No MARS-PED file type)
2) Segment network file (MARS-PED file type 47)
3) File of segment shifts (if available)
4) Segment region file (if available) (list of segments)
5) Possible user entered data as state name and year, window border for output mask window file, LANDSAT/SPOT frame date, boundary width for boundary generation.

The output is:

1) Segment Mask File (MARS-PED file type 17)
2) File containing mask window coordinates

The pogram works in this way: with a global tion file as input,the program converts all vertices of a segment network file to row and column coordinates. These coordinates for all of the segment edges are used to create a rasterized version of the segment network file - the mask file. The mask file contains several entries describing the number of pixels of each field that exist in varying locations on each raster line. If desired, pixels that are within a specified distance from any segment edges can be flagged as boundary pixels.

The mask generation program produces a file with subwindow coordinates, usefull with the program SUBWIN, used to strip off unused data after shifting and to patch window file.

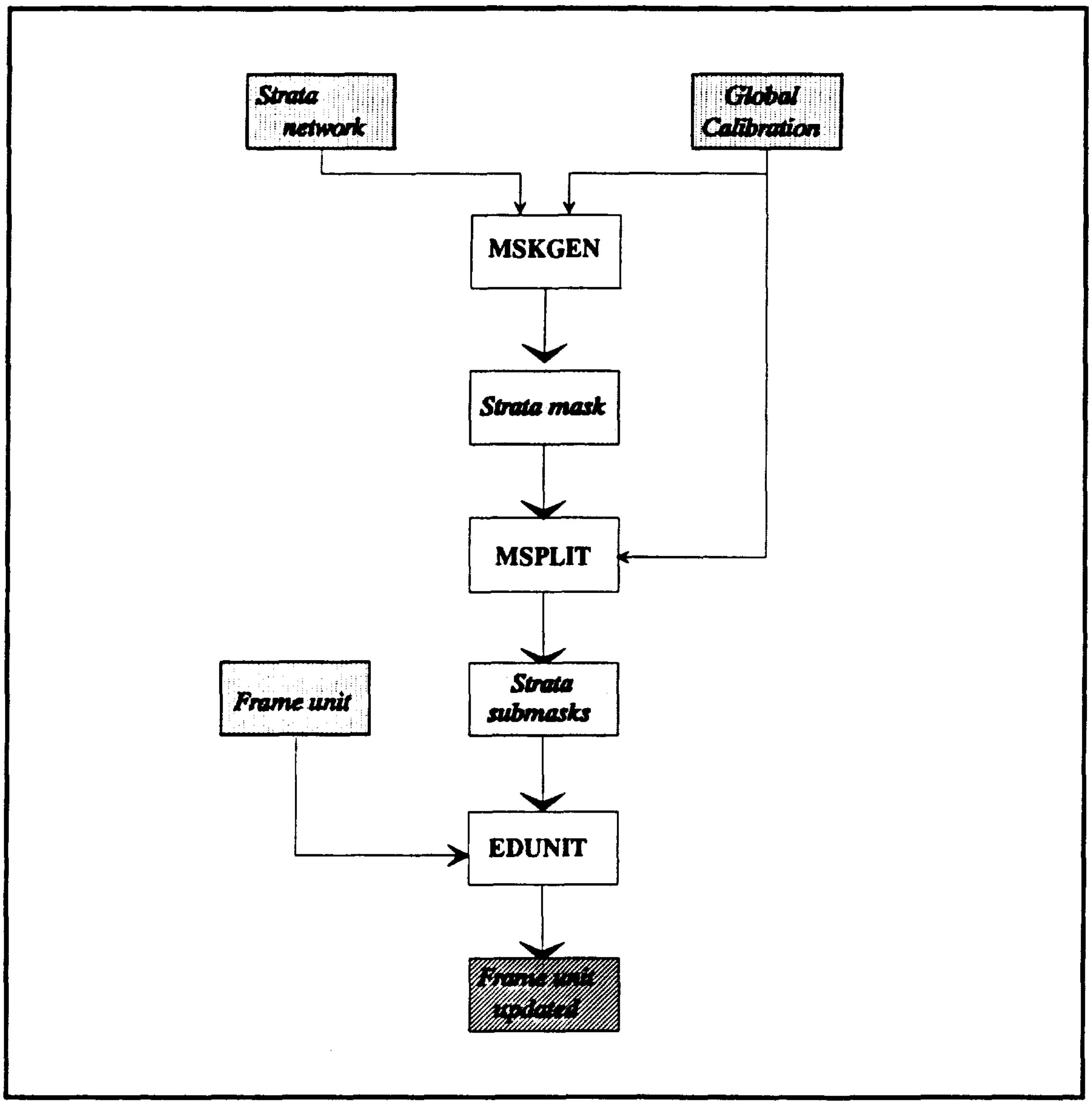
To transfer subwindows or complete copies of individual windows from one or more MARS-PED window files to a new window file the program **SUBWIN** must be used. Input data are: one or more Window files (MARS-PED file type 51), one or more text files containing segment numbers and their corresponding subwindow coordinates, and the user supplied commands. The output is a newly created Window file (MARS-PED file type 51), and a listing of input file headers, list of windows containing subwindow coordinates, and output window tables.

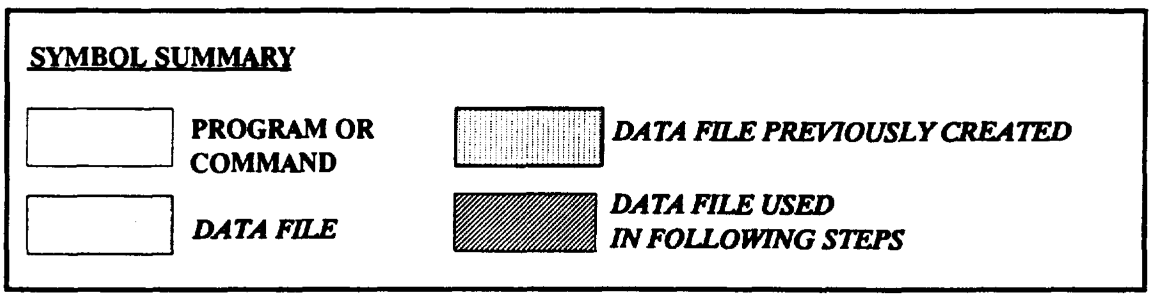
An output **window table file** is generated by the user with appropriate editing commands in SUBWIN. This window table describes the input file, window number, and subwindow coordinates for each new window in the output window file. When the output table is complete, the command WRITE is used to generate the new output window file according to the details spelled out in the output window table.

Additional commands are available to clear out the window table, and to list out information regarding the input file and the output window table.

At this point the segment level data organization is complete, now the frame level data organization must be edited.

STRATA ORGANIZATION SCHEME





MASK GENERATION OF STRATA

The **Frame unit file** is the basic data structure for large scale estimation. It is created using EDUNIT.

MSKGEN is used in the same way as in the segment level but in this case the input data file is a Strata Network File in place of a Segment Network File and a Global Calibration File.

MSPLIT is used because several mask files are often created for one file of segment or strata data due to the overlap of LANDSAT/SPOT frames. Each of these mask files may contain data common to other mask files as well as filler (blank) data. The MSPLIT program allows the extraction of selected parts of each mask (submasks) so as to eliminate frame overlapping and filler data. The Input data are user entered data as: county name, LANDSAT/SPOT frame name for each input mask, strata mask type, state name and year, directories, left and right frame margin to avoid filler, limits of usable frame area, column filler in the four corners of the usable frame area. A Calibration File (no MARS-PED file type) for each input LANDSAT/SPOT frame and a Segment or Strata Mask File (MARS-PED file type 17) for each input LANDSAT/SPOT frame are are needed also. In this case the Output is a Strata Submask File (MARS-PED file type 17).

How MSPLIT works: after all calibration file and mask file data for all frames are received for a county, each frame is processed in order of user specified priority. All of the area in the first frame (mask) that is within the county mask is retained for output to a submask file. Each subsequent frame is processed in terms of the previous frame's calibration file data and only those areas not already output for a previous frame will be written to other submask files created for each county and frame combination.

EDUNIT is used to enter and/or edit (sub)county, analysis district, strata, strata factor and frame unit information in a frame unit file or to create subcounties using segment network or mask file data or to check frame unit and digitizer area statistics. Here the frame unit level is considered. Input data are: a Frame Unit File (MARS-PED file type 50) to edit an existing file, Segment Network File(s) (MARS-PED file type 47) for subcounty creation and frame unit area checking and Possible user entered data (depending on command chosen): (sub)county names, analysis district names, strata and strata factor numbers and the number of frame units, LANDSAT/SPOT frame names, tract names, state name and year,fips code and crop reporting district (CRD). Output is the Frame Unit File (MARS-PED file type 50) containing input or updated information named by state unit.

The program takes user-entered parameters and places them in two data structures, one for county entries and the other for strata factor entries. The user may read data into these structures from a frame unit file disk file and then write it out after editing. The user may also optionally select the ASSIGN command with

1) the CREATE command to create subcounties,

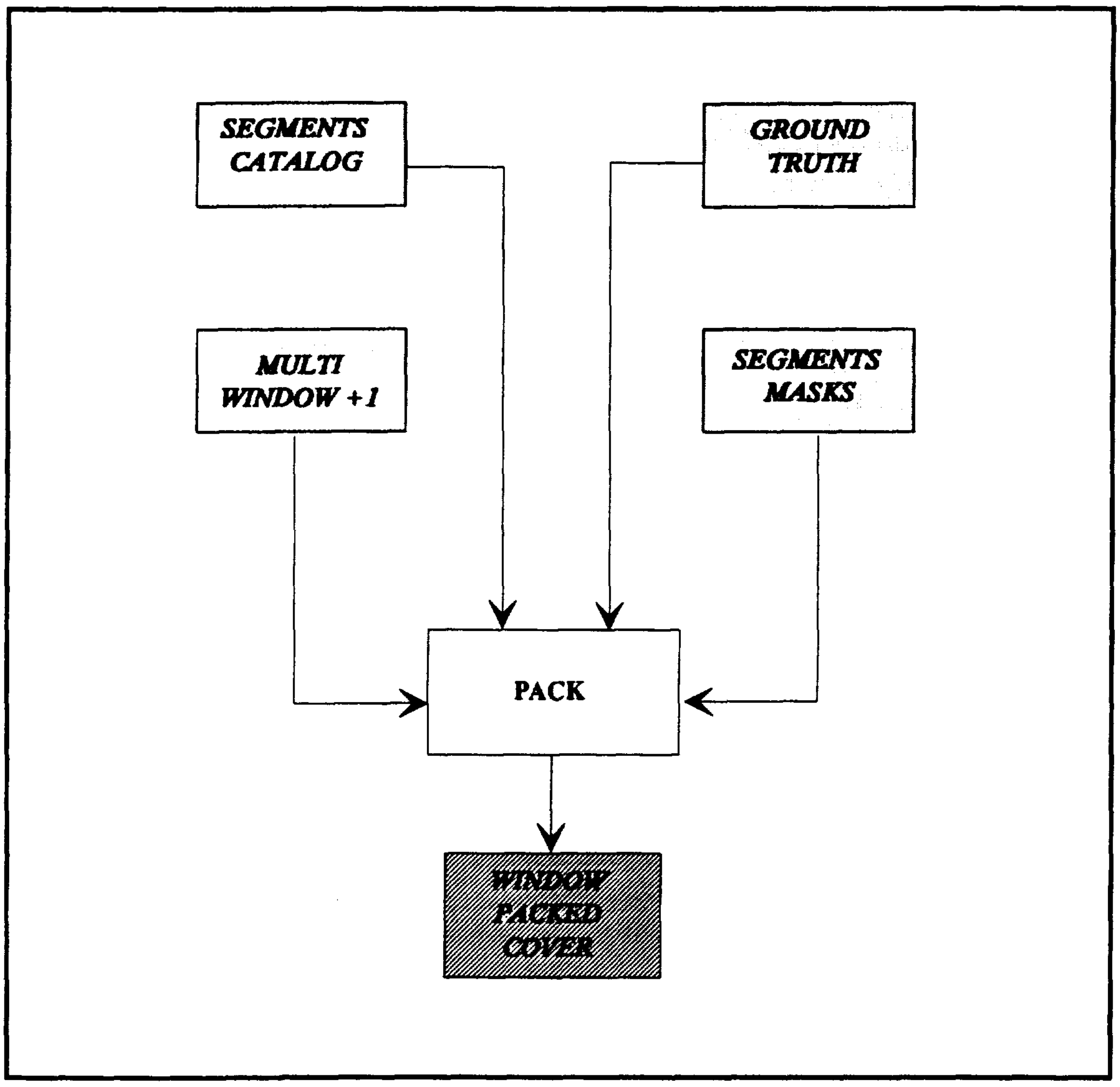
or

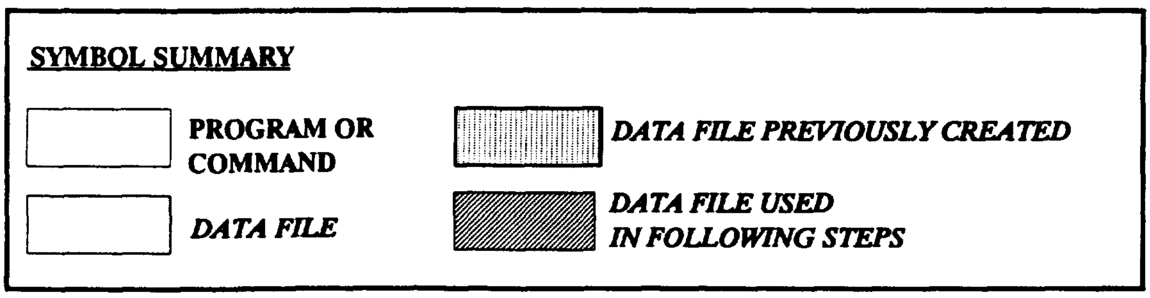
2) the FRAME command to compare frame unit and digitizer area statistics.

SELECTION

After the organization phase is completed the selection can be made using the **PACK** program. The PACK program is used to create packed files, descriptively packed files, and table files.

DATA SELECTION SCHEME





PUTTING PIXEL TOGETHER AND TABULATE INFORMATION

A **packed file** contains only pixels that meet criteria specified by a SELECT OPTIONS statement and for segments specified by a SELECT REGION statement. **As classification of the image is done by Analysis District, this is the area covered by images of the same date and sensor, also packing of pixels must be done by analysis district.**

However, one can adapt this methodology according to specific needs (i.e. classification stratum by stratum). Any other strategy however will increase significantly the cost of the analysis, putting the total effort probably above the cost/benefit ratio.

A **descriptively packed file** is similar to a packed file, but it contains additional information for each pixel. This additional information consists of the segment number, tract, field number, cover, and LANDSAT/SPOT row and column. The descriptively packed file is intended for external use.

Input Files are:

{ssyy}.CATLG	Segment catalog file
{segment number}.GTRUTH_ssyy	Ground truth files
{anything}.MWN	One pixel border multi-window file
{segment number}.MASK_{first five of scene-id}.$\{second five of scene-id}_{ssyy}	Segment mask files

Output Files that PACK may create:

NB.PACK	Packed file including border pixels (MARS-PED file type 57)
$NB.PACK	Packed file excluding border pixels (MARS-PED file type 57)
NB.TABLE	Table file including border pixels (MARS-PED file type 56)
$NB.TABLE	Table file excluding border pixels (MARS-PED file type 56)
COVER.PACK	Individual crop packed file (MARS-PED file type 57)
{pack identifier}.PACK$NAMES	Textfile with all the names of the files that are packed.This is useful to enter in SCAT program.

Packing is the grouping together of pixels, from all segments within an analysis district, by crop or cover using the ground truth information. A separate PACK file is created for each cover or crop. The file does not include field boundary pixels or pixels from fields which were coded as bad fields in the field level edit.

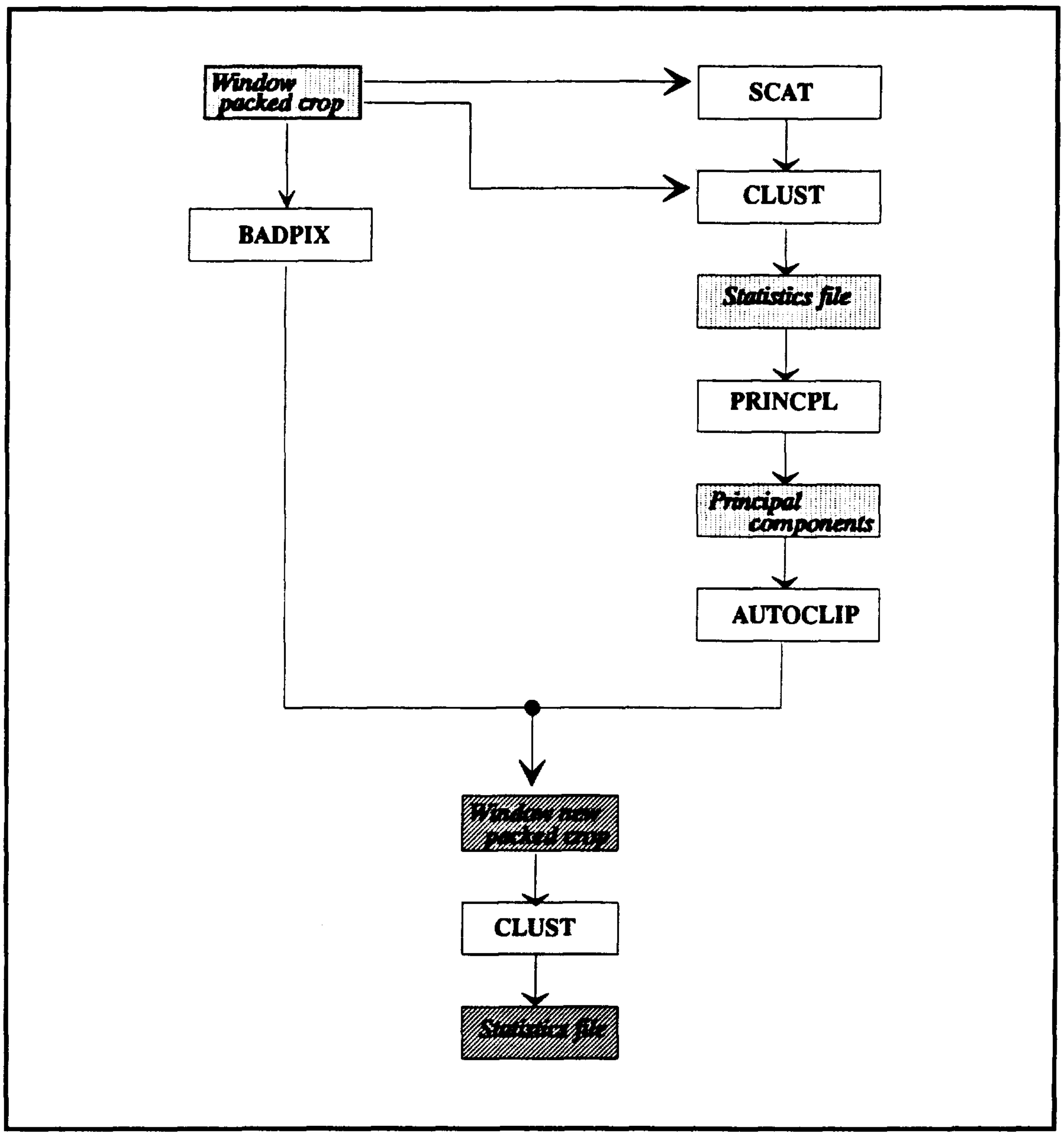
The NB.PACK file includes all pixels within all segments. The file $NB.PACK contains pixels for all good fields but no boundary pixels. The file contents are controlled by parameters which are called for in the pack program. Each PACK file contains reflectance values and location information for pixels identified by parameters. TAB files are created for each of the PACK files named above and are nothing more than a tabulation of the pixels in the PACK files.

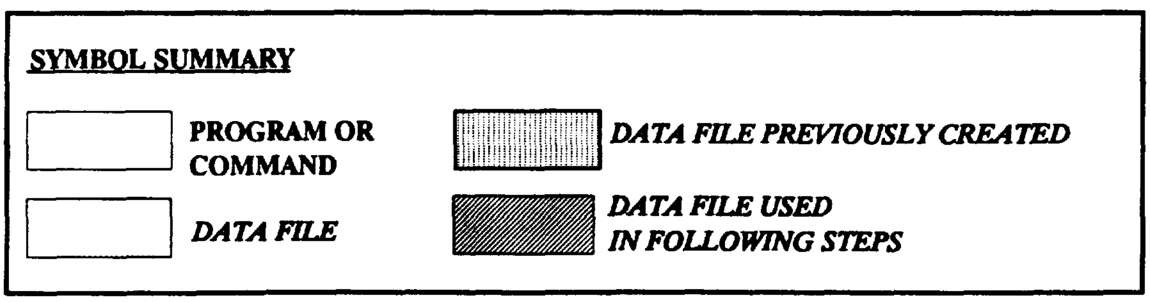
The PACK program can packing in two ways: automatic pack and pack a file. They are essentially the same program where 'automatic pack' names the files, selects the parameters, etc. and 'pack a file' must have all parameters and file names input by the analyst. Both programs are shown in the INTERACTIVE COMMAND SEQUENCES in the PACK program part.

The Individual crop packed file (MARS-PED file type 57), segments catalog and ground truth file are used in the training phase.

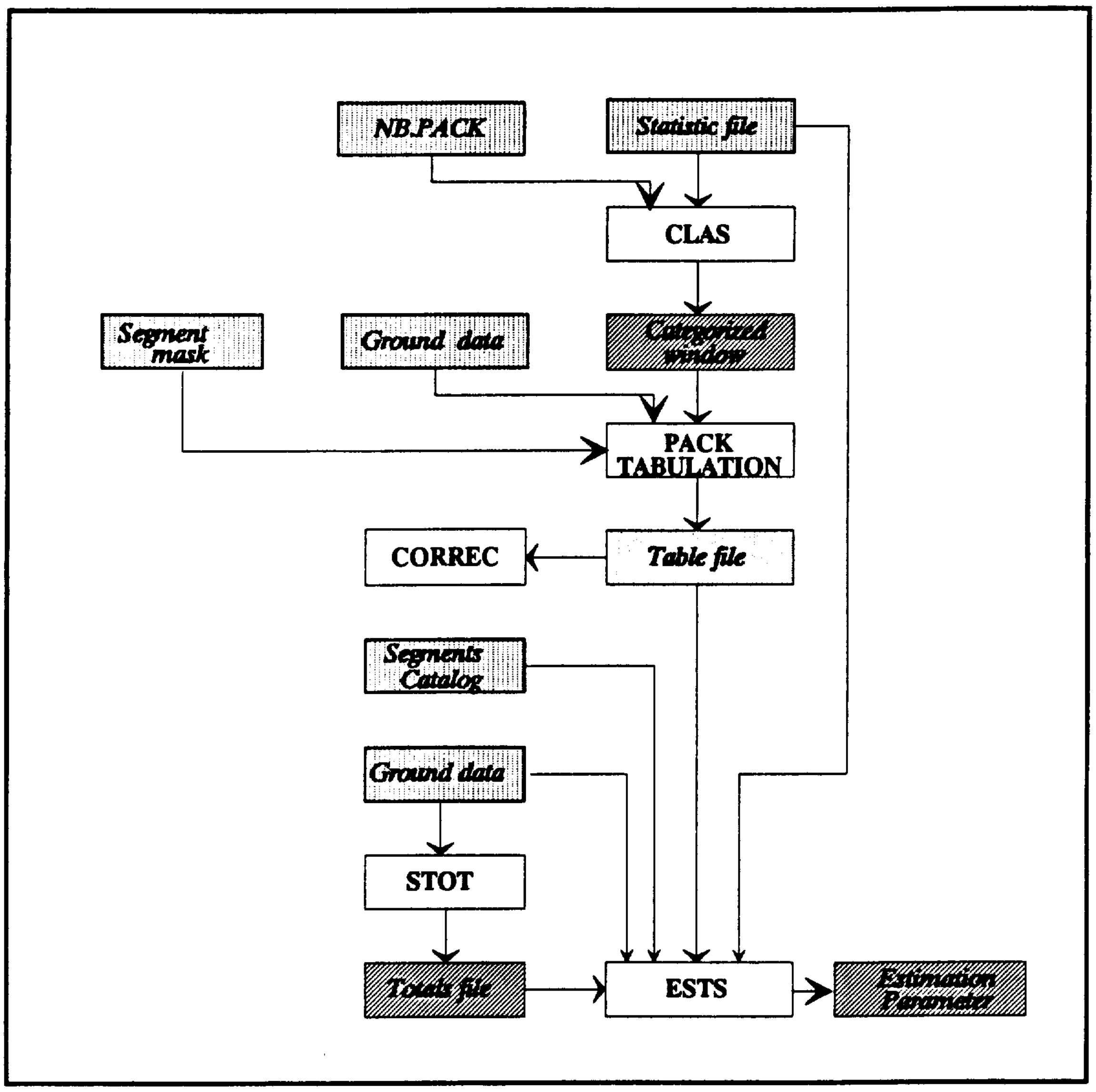
TRAINING

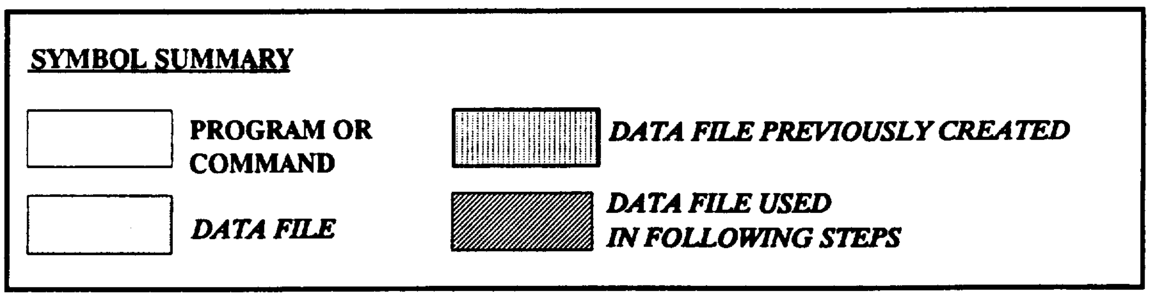
CLIP & CLUST SCHEME





TRAINING SCHEME





Training needs as input data files a packed window file, a catalog, and a Ground truth file as created in the previous steps (digitization and selection).

The first action to do is scattergraming in order to get a visual picture of the pixels in a packed cover to find stray pixels. Stray pixels may be caused by weed patches or waste (too small to break out). Stray pixels may be clipped (removed) by using the BADPIX program. The program used is SCAT and it requires as input file a packed covers raw data file. The output is on video or a textfile that can be printed.

To remove the bad pixels in a packed window file input the program **BADPIX** is used. Range values for these pixels are obtained from the user. When the out of range pixels are found, the mean of the good pixels will be substituted or a zero (0) will be inserted for deletion of the bad pixels. Input file: Window files (MARS-PED file type 51) with name of file supplied by the user. Output file: a new revised Window file is created if any bad pixels are found.

BADPIX is designed to find pixels which are out of the range specified by the user. The user types in an input file name and the program will print pertinent information on the screen. The user enters range values that the pixel must be within for each channel. The program then asks if the bad points should be deleted or substituted for. An output file is only created when badpixels are found, and this file name is supplied by the user.

This procedure is called clipping. A more automatic way of doing this is provided by the **AUTOCLIP** program. AUTOCLIP compares the principal component value generated from the raw pixel value to the principal component mean +- X times the standard deviation of the principal component. Out of range pixels and their principal component values are output in a text file named the same as the output clipped file with the last 3 characters replaced with 'TXT'.

CLUSTERING

After the analysis district packed files have been clipped to eliminate outlier pixels, the remaining pixels in each individual packed file need to be clustered into groups that are more similar than the overall population. CLUST is the program for multi-channel clustering. As input file it needs a window file (MARS-PED file type 51), as output it gives a statistics file (MARS-PED file type 54).

The set of pixels specified by the user is read into core memory and clustered using the LARS algorithm. The data is first accessed to determine the mean for the pixels in each channel. The data is then accessed again to determine the variance from the mean for each channel. The starting mean values (seedpoints) are then computed and distributed along the diagonal in multi-dimensional space according to the total variance found.

From this point on, the clustering algorithm is iterative and is terminated when the convergence criterion specified by the user is satisfied or the maximum number of iterations is exceeded. During each iteration, each pixel is accessed and compared with each mean determined by the previous iteration (or starting value for the first iteration). Each pixel is assigned to the class number wich has the smallest Euclidean distance to a mean. After all pixels have been assigned to the nearest class, new means are computed from the pixels assigned to each class. Convergence is defined as the percentage of pixels which have not changed class since the last iteration. Thus, 100% convergence occurs when each pixel is assigned to the same class as the previous iteration.

CLUST provides the facility to split clusters as well as merging them. In this way CLUST creates a statistics file, one for cover.

EDITING THE IMAGE STATISTICS

Once the individual cover statistics files have been generated a combined statistics file for all the covers in the segment or strata must be generated. This can be done with the statistical editor **STATED**. In general, to create an analysis district statistics file requires 4 steps :

1) open an individual cover statistics file,
2) name the categories from that statistics file,
3) write these categories to the output buffer,
4) after all input individual cover statistics files are written to the output buffer, generate the final analysis district statistics file (in general the more information that can be gathered the better).

Normally a listing of the names and number of pixels in each category and a measure of the distance between categories are needed. The means, variances, and generalized variance can also be useful together with the different separability criteria provided in STATED.

The creation of the combined statistics file is one of the most important functions in the analysis of LANDSAT/SPOT data. All of the pixels included in an analysis district are classified to covers according to the information contained in the analysis district statistics file. Minor changes in the statistics file can cause major changes in the number of pixels classified to each cover included in the statistics file. Even though this is a major function in the analysis, there are no hard and fast rules for editing a statistics file. However, there are certain steps that must be done to generate and properly edit a statistics file.

The program that creates the combined statistics file is STATED from the different individual cover statistics files.

Also a priori probability can be given and must be entered during creation of statistics files in STATED.

PRINCPL is used to update the principal component file (MARS-PED file type 66) that contains statistical data about the training; **DISPRNCP** displays this file. This two programs are optional, so they have not effect on the training.

CLASSIFICATION

After the creation of the file there is the classification of a multi-channel window file using maximum likelihood. A multi-channel data set is classified with a statistics file supplied by the user using the maximum likelihood algorithm by the **CLAS** program. Each point in the data set set is assigned the class number corresponding to the closest spectral class in the statistics file as determined by the maximum likelihood decision rule. If during the inversion of the spectral classes in the statistics file, a class is found to be ill-conditioned, then that class is given a very low probability so that no points will be assigned to it.

The output of this program is a single-band (one channel) categorized image, and (optionally) a single-band threshold image.The threshold image contains values from 1 to 16 which index a table of probabilities, i.e., the threshold values indicate the probability of correct classification for each point. This file can be printed and if is never used anymore it can be deleted directly.

The input files for CLAS are: Raw or filtered data window file (MARS-PED file type 51), Statistics file (MARS-PED file type 54). The output file is a Categorized window file (MARS-PED file type 51) or a Threshold window file (MARS-PED file type 51).

TABULATION OF CLASSIFICATION OUTPUT

After the classification a table file must be generated for the sample step. A **table file** is a tabulation of a packed file by cover and category of the number of pixels packed. A table file created from a raw data packed file is assumed to have one category. A table file may be by segment with a separate tabulation for each segment, or for all segments with a single tabulation containing the totals for all segments. TABULATE is the subcommand of PACK that generates a table file from an input packed file.

SUMMING ALL DATA TOGETHER

Before doing sample estimation a totals file must be create with the program STOT. The **totals file** contains sizes by segment for various SELECT OPTIONS statements and will be used to input segment size values to the estimator program.

The totals file editor, STOT is used to create, update, and list the totals file. The totals file contains one list of segments. The actual data is placed into separate data blocks which will be referred to as option blocks. Each option block is specified by a SELECT OPTIONS statement (for a description of SELECT OPTIONS see the PACK program), a default use, and a size type. The SELECT OPTIONS statement is any statement acceptable to the SELECT OPTIONS parser and is used to select the fields which will be included in the size computation for each segment. The default use is applied to any SELECT OPTIONS parameters requiring a use but for which no use is explicitly supplied. The default use is also used to build the list of covers for the option block by looking up the cover for each field selected.

This list of covers is of limited capacity, currently ten covers, and is associated with the entire option block and not with individual segments. The size type is field, planted, harvested, or abandoned. Abandoned is the difference between harvested and planted. Thus, it is possible to have several option blocks with the same SELECT OPTIONS statement, but with different default use or different size type.

ESTIMATION

The sample estimation program **ESTS** uses segment data to generate parameters for large-scale estimation (either single-variable estimation and multi-variable regression) and to display some results. The input files required by sample estimation are a segment catalog file, a frame unit file, and a totals file. If pixels is used for any of the independent variables, the usual case, a table file is required. A statistics file is required if the list of categories to be used to get pixel totals is to be generated rather than input by the user. All output files from sample estimation are optional. The output files which may be created are the estimator parameter file and the segment totals file.

For **single-variable estimation**, the R-SQUARE, B[0], B[1], standard deviation, and C.V. (coefficient of variation) are displayed for each group and unassigned strata. Values are displayed in both acres and hectares, with -A indicating acres and -H indicating hectares. R-SQUARE and C.V. are the same for both acres and hectares, of course. If there are more than one group or unassigned strata, the total standard deviation and C.V. are displayed.

For **multi-variable regression**, the R-SQUARE, standard error, and C.V. are displayed, the latter two for Y-MEAN and Y-MEAN[MR]. The regression coefficients, B, are displayed for the dependent and independent variables. The sample correlation coefficients are shown between the independent variables. If the absolute value of a correlation coefficient is greater than 0.25, a warning message about possible bias between the independent variables is displayed. Finally, an optional residual display is provided. If this display is selected, the actual value, computed value, and residual are displayed for each segment. The residual is the actual value minus the computed value. For values where units are pertinent, the units used is either acres or hectares, as selected by the user when the program is first invoked. The input files required by sample estimation are a segment catalog file, a frame unit file, and a totals file. If pixels is used for any of the independent variables, the usual case, a table file is required. A statistics file is required if the list of categories to be used to get pixel totals is to be generated rather than input by the user. All output files from sample estimation are optional. The output files which may be created are the estimator parameter file and the segment totals file.

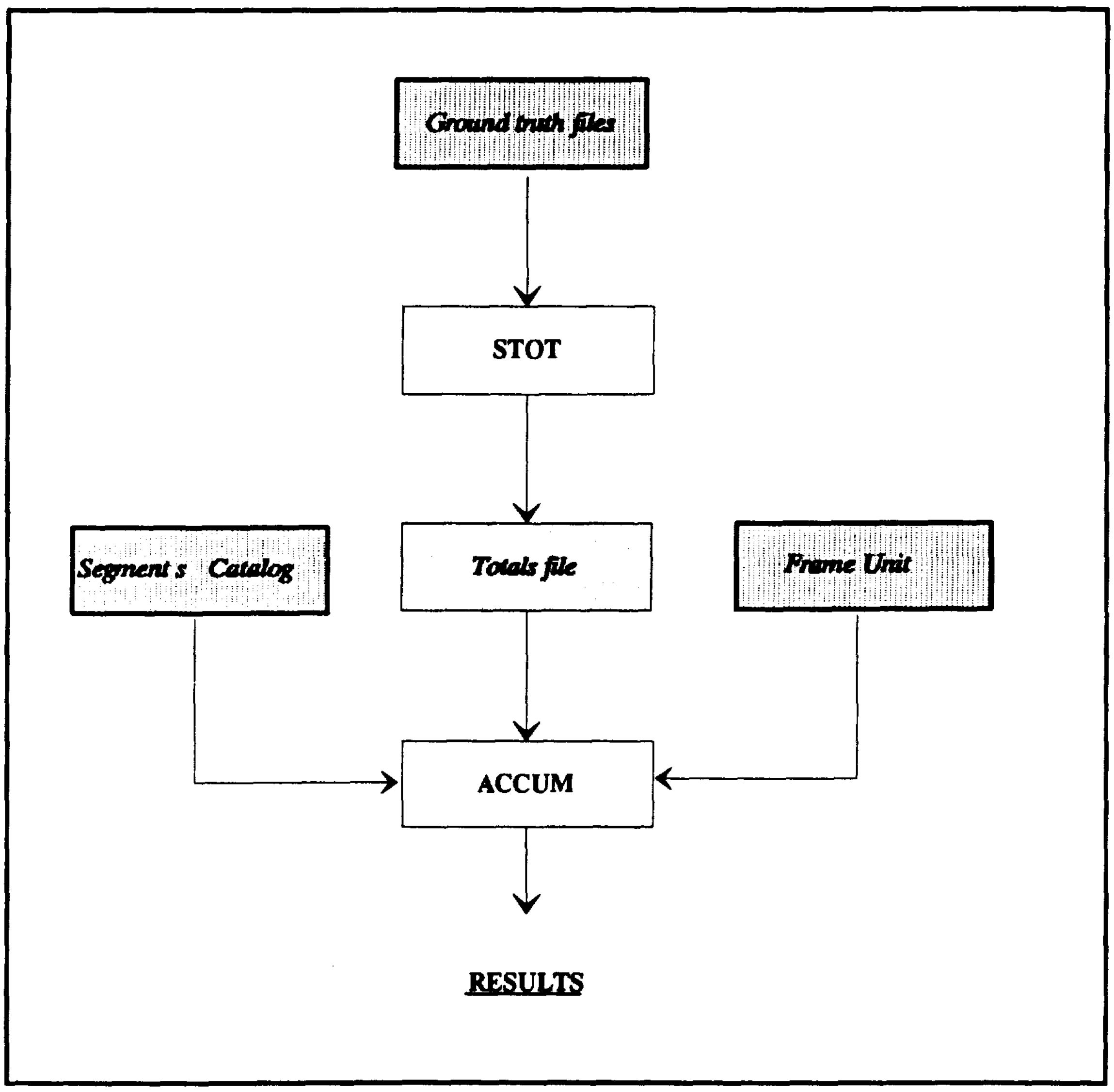
If the quality of the data in the Estimator Parameter File it isn't satysfing the training process can be repeated restarting with CLUST or STATED programs.

Now all the data for estimation are ready.

ESTIMATION

Classification, aggregation, large scale estimation and accumulate are the steps in estimation .

CLASSIFICATION WITHOUT RS DATA SCHEME



SYMBOL SUMMARY

PROGRAM OR COMMAND

DATA FILE PREVIOUSLY CREATED

DATA FILE

DATA FILE USED IN FOLLOWING STEPS

The accumulate estimates program ACCUM is used to bring together the various estimates of a specific crop for a large region. A region, usually an entire state, might consist of one or more anlaysis districts and is defined by a corresponding frame units file.
There are three possible outputs: tables displayed directly on the terminal, tables sent to text files for printing, and a machine readable file for input into the county estimates modules.

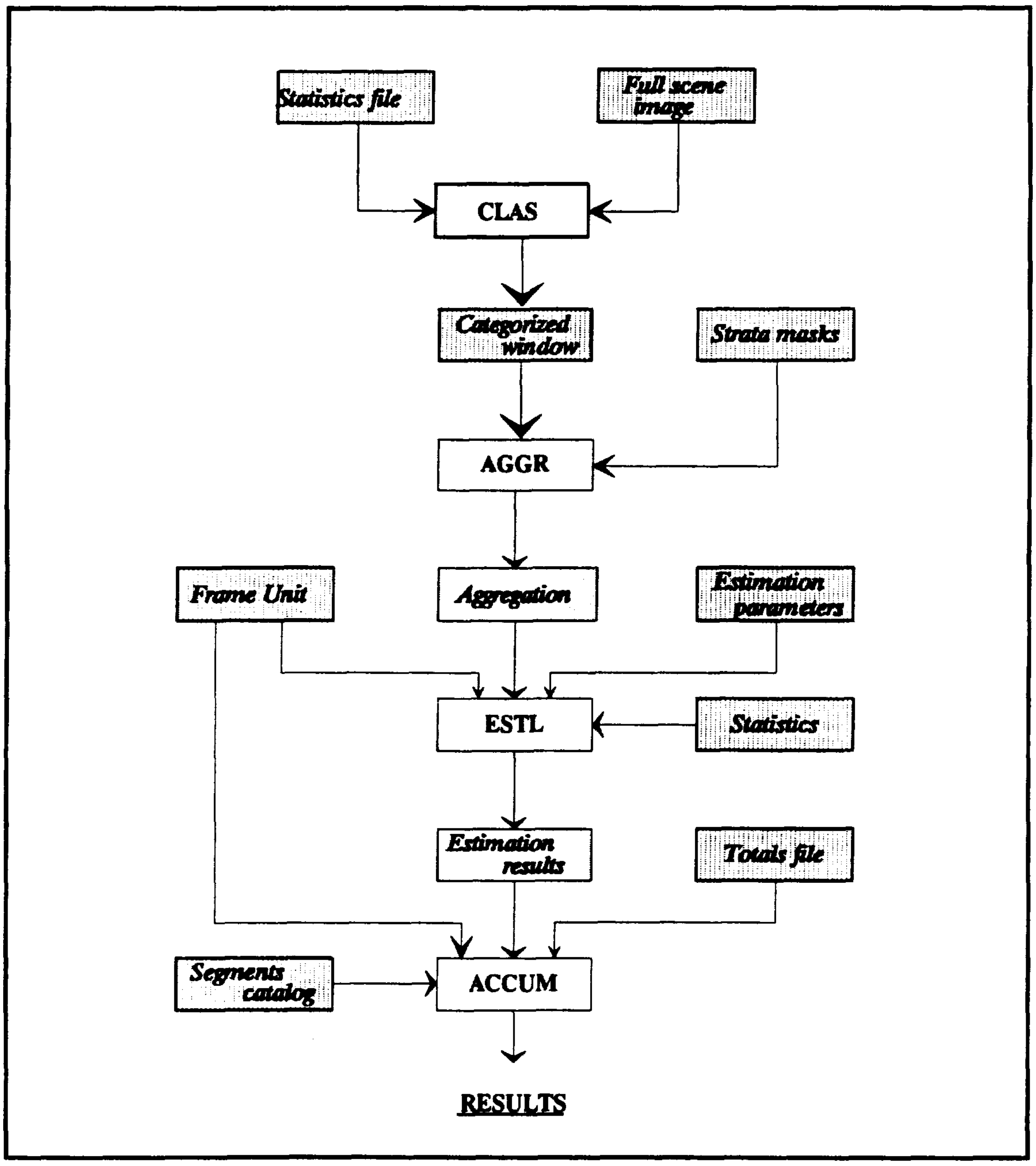
Estimates and corresponding input information are "accumulated" into display tables. Estimates for analysis disctrict/strata combinations may come from single-variable regression, multiple-variable regression, direct expansion, and/or proration estimators. Two varieties of tables are available, a collection of tables showing analysis district by strata data and a JES table by strata only showing both the JES estimate and the Landsat regression and non-regression estimates, the latter for areas where Landsat data is not available.

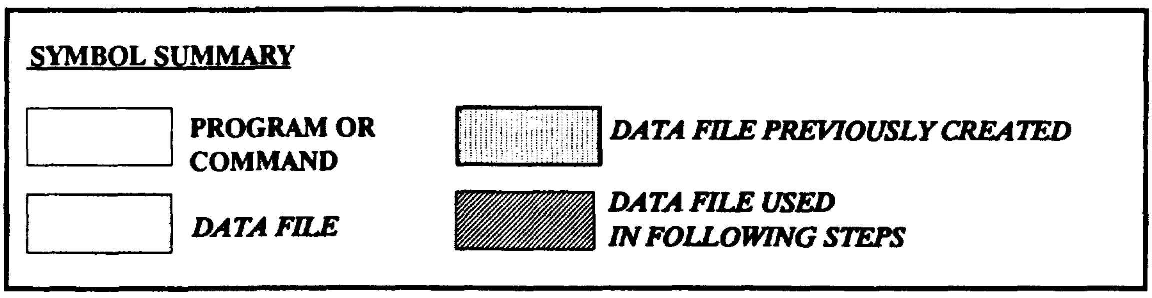
The accumulate estimates program operates at three levels, depending on which input files have been read. The first level is proration only. At the first level, the frame unit, segment catalog, and (segment) totals files have been read. These files are required for any use of the accumulate estimates program. At the first level, a limited collection of proration estimator values and inputs may be displayed by analysis district and strata; or estimator results files may be read to get to the second level.

The second level of the accumulate estimates program is reached when one or more estimator results files have been read. The estimator results files are a product of large-scale estimation.
At the second level, all tables by analysis district and strata may be displayed. Estimator type may be changed between direct expansion and regression for specified analysis districts and strata. Additional estimator results files may be read at any time. The machine readable file for county estimates is also created at this level. Finally, the JES input file may be read to rech the third level.

The third level of the accumulate program is reached when the JES input file has been read. The JES input file is an ASCII file created externally to EDITOR listing JES estimates by strata and crop. At the third level, all tables including the table of Landsat and JES values may be displayed and all operations allowed at the second level may be performed including reading additional estimator results files.

FULL SCENE CLASSIFICATION WITH RS DATA SCHEME





FULL SCENE CLASSIFICATION

The user supplies a statistics file (created in the training process) and the full scene image file to the CLAS program in order to get a categorized window file about the general scene. The CLAS program works like in the training process, the difference here is in the width of the scene.

Input files are:

1) Statistics file (MARS-PED file type 54). From training.
2) Multiwindow file (MARS-PED file type 51). Full scene image.

Output files is:

1) Categorized file (MARS-PED file type 51);

AGGREGATING THE RESULTS

With the categorized image file so obtained and the mask file of the same scene, **AGGR**(egate) tabulates the two. For each row in the area specified by the user, AGGR reads one row of data from the categorized file. It then reads the mask data for the corresponding values from the STRATA mask file. AGGR uses these two layers to produce a table of the area coverage by stratum and category.

Input files are:

1) Categorized window file (MARS-PED file type 51).
   This file contains only 1 channel of data, the category.
2) Mask file (MARS-PED file type 64).
   The names of the window and mask file nust be supplied by the user.

Output file is:

1) Aggregation file (MARS-PED file type 24).
   The name of this file must be supplied by the user.

AGGREGATE can be used in two ways. To produce a new aggregation file, the user enters "AGGREGATE" in response to the main prompt. The program will then ask for the names of the two input files. To write the aggregation file, the user enters "WRITE"; again, AGGREGATE will prompt for the name of the output aggregation file. The other commands in this section are optional. AGGREGATE may also be used to display or print an existing aggregation file. To display/print an aggregation file, the user should enter "DISPLAY" or "PRINT" in response to the main prompt. AGGREGATE will then ask for the name of the input and, for print, and output file.

ESTIMATION FOR EACH STRATUM

Large-scale estimation **ESTL** is used to compute the estimate, standard deviation, C.V. (coefficient of variation), and possibly other information by strata for an analysis district. Large-scale estimation incorporates the classification and aggregation by strata of complete LANDSAT/SPOT scenes with the parameters generated by sample estimation to compute estimates. Also, large-scale estimation will optionally create an estimator results file which may be used with the accumulate estimates program.

Required input files for large-scale estimation are the estimator parameter file, the frame unit file, and one or more aggregation files. A statistics file is an optional input file, used to get the list of categories for computing pixel totals from the aggregation file for the independent variables.

The only optional output file from large-scale estimation is the estimator results file used as an input to the accumulate estimates program that accumulate estimates from the various analysis districts.

Input files are:

Statistics file for an analysis district, state and year.

County masks for all counties completely contained in the scene.

County submasks for all counties partially contained in the scene.

Prior probability file for an analysis district, state and year (optional).

If the new MARS-PED form of the statistics file is ready to be used, the priors file will become obsolete since the prior probabilities will be maintained within the new statistics file.

Output file is

County aggregation files.

COMMANDS

COMMAND LIST

Programs actually avalaible in MARS-PED are listed below.

ACCUM	The accumulate estimates program is used to bring together the various estimates of a specific crop for a large region.
ADDAGG	Adds or subtracts the contents of user-specified aggregation files. The addition or subtraction can be done on a by-field or a by-tract basis.
AGGR	Provides aggregation statistics from classified image and mask files. These statistics consits of the number of pixels in each category in each stratum of the mask file.
AUTOCLIP	AUTOCLIP compares the principal component value generated from the raw pixel value to the principal component mean +- X times the standard deviation of the principal component. Out of range pixels and their principal component values are output in a text file named the same as the output clipped file with the last 3 characters replaced with 'TXT'.
BADPIX	To find the bad pixels in a packed window file input. Range values for these pixels are obtained from the user. When the out of range pixels are found,the mean of the good pixels will be substituted or a zero (0) will be inserted for deletion of the bad pixels.
CALC	To take as input 1) An image calibration file, and 2) Coordinate points in Latitude-Longitude, UTM, or Image coordinate systems And to output to the terminal 1) The corresponding coordinate points in any of the other coordinate systems, or 2) DWR quad number and block.
CATED	To create or edit a segment catalog containing information and characteristics of user-defined segments; in this way, groups of segments with common characteristics may be easily accessed by other programs.
CLAS	Classification of a multi-channel window file using maximum likelihood.
CLUST	Cluster a multi-channel window file.
CMASKP	By field, this command lists the non-background pixels for both the boundary and non-boundary pixels separately and gives the total number of pixels. The program also returns the total number of pixels masked, the total number of non-background pixels for this file, and the first and last row in which a non-background pixel was found.

CORREC — Reads a Table file (type 56) and prints a matrix showing ground-data cover versus categorized cover,in a 'categorized from vs. categorized to' format. It also prints a percent correct and percent commission for each cover type.

CPVERT — Generate an ascii file of segment calibration marks to be used with video-digitized segments.

DISPRNCP — This a program that reads and displays a Principal Components file.

DSPWIN — DSPWIN allows the user to print out selected subwindows from a window file (type 51) out on the user's terminal, a text file (for later printing), or to a special graphics metafile (see gmf.ld for more information). Windows printed out to a graphics metafile can additionally be scaled by specifying the size of a pixel in meters. The MARS-PED module GMFRAS can then be used to convert the graphics metafile into a raster format file, which can then be printed out on raster format devices using the appropriate driver, such as RASPRT for the Printronix line printer.

EDUNIT — 1) To enter and/or edit (sub)county, analysis district, strata, strata factor and frame unit information in a frame unit file
2) To create subcounties using segment network or mask file data
3) To check frame unit and digitizer area statistics.

ESTL — Large-scale estimation is used to compute the estimate, standard deviation, C.V. (coefficient of variation), and possibly other information by strata for an analysis district.

ESTS — The sample estimation program uses segment data to generate parameters for large-scale estimation and to display some results.

EXTENT — Determine image file window extents of a group of digitized segments, and optionally produce a segment region file and a window text file of all segments that are wholly contained in the image file.

GROUP — To allow the re-grouping and/or re-ordering of categories specified by the user into new categories and write them to a different file.

GTRUTH — The ground truth editor is used to create, update, check, and list ground truth files.

IDENT — Used to determine the type of a file if the user is unsure of it. It prints out the file type code, name, and some (perhaps) other useful information.

LIPTAP — Reads an ERDAS 7.2 image and dumps it to a Pixel Interleaved (PIL) tape, that can be read again with TAPWIN.

MAKEPCP — To create control point pairs file (.PCP) of the image control points and map control points.

MCTYNM — To compare county names between a frame unit file and a segment catalog file.

MEDIT	To create a submask file from an input mask file given the part number and submask coordinates that are within the mask file boundaries.Medit may also be used to change the field numbers in a file.
MSKGEN	To generate a mask file from a segment network file.
MSPLIT	Several mask files are often created for one file of segment or strata data due to the overlap of Landsat frames. Each of these mask files may contain data common to other mask files as well as filler (blank) data. Module MSPLIT allows the extraction of selected parts of each mask (submasks) so as to eliminate frame overlapping and filler data.
PACK	This program is used to create packed files, descriptively packed files, and table files. It is also used to unpack files, display table files, and create lists of segments.
PARTS	To manipulate multi-part segment network files, strata network files or segment mask files.May be used to combine several one part files into a multi-part file or to break out a part of a multi-part file into a single part file.
PCDIGIT	To digitize boundaries or strata with a digitizer on the PC.
PCSHIFT	This program is used to display grey-scales of raw data, display categorized data using colors, and do segment shifting using the raw data grey-scale display or the categorized data display.
PRINCIPL	This program calculates principal components, eigan values, and the rotation matrix from a statistics file and outputs a new Principal Components file.
PRINCPWN	This module creates a "window" file with principal component values replacing raw data values. SUBWINDOW module then can be used to reduce the dimensionality of the data.
PRXSEG	Generate plot files in Landsat or Digitized image from input segment network files. These plot files contain the necessary code to be able to be printed, in plot mode, on a Printronix line printer.
REGCP	Generates and stores either image-to-map or image-to-image calibration information useful in the subsequent location of data on selected remotely sensed images.
SEGED	The digitization program, allows commands to be entered in two ways, from the keyboard and from the digitizing cursor
SPOT	Reads a SPOT tape and dumps an Multi-Window file to disk.
STATED	Display, editing, and creation of statistics files.
STOT	The totals file editor is used to create, update, and list the totals file.
SUBWIN	To transfer subwindows or complete copies of individual windows from one or more window files to a new window file.

SVCAL	Save and restore the calibration portion of Segment Network Files.
TAPLIP	Reads a Landsat scene on tape and dumps channels 1,2,3,4,5 and 7 to an ERDAS 7.2 image file. The subcommand indicates if one wants to read a full scene or a quarter scene.
TAPWIN	Allows copying of selected portions of image data from tape to disk window file
TTS	To digitize boundaries with a video camera on the PC.
XGCP	To define Ground Control Points into a multiwindow or ERDAS image format.
XSFT	To perform the segment shifting on a image in multiwindow format displayed on the screen, using the mouse.

ACCUM	ACCUMUlate estimates

PURPOSE

The accumulate estimates program is used to bring together the various estimates of a specific crop for a large region. A region, usually an entire state, might consist of one or more analysis districs and is defined by a corresponding frame units file.
There are three possible outputs: tables displayed directly on the terminal, tables sent to text files for printing, and a machine readable file for input into the county estimates modules.

Estimates and corresponding input information are "accumulated" into display tables. Estimates for analysis disctrict/strata combinations may come from single-variable regression, multiple-variable regression, direct expansion, and/or proration estimators. Two varieties of tables are available, a collection of tables showing analysis district by strata data and a Ground Survey table by strata only showing both the Ground Survey estimate and the Landsat regression and non-regression estimates, the latter for areas where Landsat data is not available.

The accumulate estimates program operates at three levels, depending on which input files have been read. The first level is proration only. At the first level, the frame unit, segment catalog, and (segment) totals files have been read. These files are required for any use of the accumulate estimates program. At the first level, a limited collection of proration estimator values and inputs may be displayed by analysis district and strata; or estimator results files may be read to get to the second level.

The second level of the accumulate estimates program is reached when one or more estimator results files have been read. The estimator results files are a product of large-scale estimation.
At the second level, all tables by analysis district and strata may be displayed. Estimator type may be changed between direct expansion and regression for specified analysis districts and strata. Additional estimator results files may be read at any time. The machine readable file for county estimates is also created at this level. Finally, the Ground Survey input file may be read to rech the third level.

The third level of the accumulate program is reached when the Ground Survey input file has been read. The Ground Survey input file is an ASCII file created externally to EDITOR listing Ground Survey estimates by strata and crop. At the third level, all tables including the table of Landsat and Ground Survey values may be displayed and all operations allowed at the second level may be performed including reading additional estimator results files.

INPUT

The accumulate estimates program requires as input files the segment totals (file type 58), segment catalog (file type 49), and frame unit (file type 50) files. Estimator results (file type 60) files are optional input files but must be read if something other than proration results are to be displayed. Several estimator results files may be read, one for each different analysis district. The Ground Survey input file (ASCII) is optional but must be read if the Ground Survey and landsat table is to be displayed. The Ground Survey text (totals) file is described in the table below:

Ground Survey Text File:

- Each line contains data for one Ground Survey district (stratum)
- Items are separated by blanks
- Each line has the following format:

FF DDDD CCC TTTTT SSSSS PPPP

where

FFF	=	(2 digit) State Fips Code - Numeric
DDDD	=	(4 digit) Ground Survey District code, first two digits are stratum
CCC	=	(3 digits) Numeric crop code (usually corresponds to that of questionnaire)
TTTTT	=	District total direct expansion estimate
SSSSS	=	District standard deviation of total
PPPP	=	Coeffic. of Variation of district total (%).

- More than one crop may be included in the Ground Survey text file
- A total line for each crop is required, with DDDD=TOTAL.

OUTPUT

One optional output file is the listing of tables sent to a disk file (via the "Change Output Display File" command). These files have no specific format.

The only other file cretaed is the "**Machine Readable File**". This ASCII file consists of the following:

- The first line of this file will contain the cover type name supplied by the user
- Subsequent lines, one for each analsis district, stratum pair have the following format:

AAAAA SS YYYY TTTTTTT DDDDDD

where

AAAAA	=	Analysis district name
SS	=	Stratum number (2 digits)
YYY	=	Type of estimate (PR, DE, S-RE, M-RE)
TTTTTTT	=	Total estimate (to two decimals)
DDDDDD	=	Standard Deviation of total (to two dec.)

Not other output files are created.

HOW TO USE IT

The user is asked for the list of directories. Any directories containing files required for the accumulate estimates program should be entered, one per line. Normally, the user's default or working directory need not be entered.

The user is asked for the state and year. This is a four-character indentifier with a two-letter postal state (FIPS) abbreviation followed by the low-order two digits od the year. The state and year identifier is used to form the standard namesof the segment catalog and frame unit files and also tu supply some information for the heading of the Ground Survey and Landsat table.

The user is asked for the units to use, acres or hectare. All tables are printed in the units selected, except the Ground Survey and Landsat table which is always printed in acres.

The user is asked to enter the name of the totals file and to select the option from the totals file. The totals file contains the ground truth size for a collection of segments and various options. The list of segments which is used for sample size and values is obtained from the totals file as are the variable values associated with each segment. It is important to note that should the totals file contain fewer segments than the segment catalog file, only the segment in the totals file will be used in computing values to be displayed in the various tables. Any other segments will be ignored since the totals file is considered to be the primary input file for the first level of the accumulate estimates program.

The segment catalog file is searched for under its standard name and in the directories specified. If it cannot be found, the user must supply the name of the file since the segment catalog file is a required input file. The analysis district and strata are obtained from the segment catalog file for each segment in the totals file. If the segment does not appear in the segment catalog file, or if the entry for the segment does not contain the analysis district or strata for the segment, the segment is ignored and a message is printed.

The frame unit file is searched for under its standard name and in the directories specified. If it cannot be found, the user must supply the file name since the frame unit file is a required input file to the accumulate estimates program. The frame unit file is used to determine; the number of frame units for each analysis district, strata pair.

At this point, all files required for proration have been read. However, to delete unwanted data, the user is shown a list od strata with the number of frame units and segments in each and allowed to delete strata. Similarly, the user is shown a list od analysis districts and allowed to delete analysis districts. In each case, the items to be deleted are entered one per line and the deletion is terminated with a line with carriage return only.

Finally, the user is asked for the listing file. If a listing file is desired, a disk file name is entered; a carriage return only here sends output to the terminal. The listing file may be changed later when in display mode.

MAIN LEVEL COMMANDS

The following commands are available at the main level of the accumulate estimates program, that is when prompted by ACCUM. All commands are avilable at level two or higher, but some commands are not available at level one. The avilability will be specified for each command in brackets ( ).

Display (All levels)
Enters display mode to display various tables. The commands available in display mode will be described in section 4.

Jets text file read (Level two or higher only)
Reads the input ASCII file of Ground Survey data and enters level three of the accumulate estimates program- The user must supply an integer crop code to select which lines will be read from the Ground Survey input file. No attempt is made to ensure that the cover for which data is read from the Ground Survey input corresponds to the options selected from the totals file, such checking is entirely the user's responsability. The user is asked if the Ground Survey input file contains early data. The reponse only affects one of the footnotes to the Ground Survey and Landsat table, it has no affect on the values in the table.

Proration district change (Level two or higher only)
Allows the user a limited facility to change proration districts for anlysis district, strata pairs. The only proration district of importance is proration district one, which normally contains the

analysis district, strata pairs for which the type of estimation is proration. This command allows the user to move analysis district, strata pairs into or out of proration district one.

Results, read estimator results file (All levels)
Reads estimator results files, which are created in large-scale estimation, usually one per analysis district (and crop). The user enters the names of the estimator results files one per line and terminates with carriage return only. If the accumulate estimates program is at level one and some estimator results files are read, the program moves to level two.

Type of estimate change (Level two or higher only)
Allows the user to change the type of estimates for an analysis district, strata pair with restriction. A regression (sigle or multi-variable) estimate may be changed to direct expansion. A direct expansion estimate may be changed back to the original type or regression. Other changes are prohibited. Type of estimator changes done with this main-level command will be reflected in both display tables and in the machine readable file. Changes done using the "Set Display Estimate Type" described in Section 4 affect only the display tables.

Quit (All levels)
Exits the accumulate estimates program.

DISPLAY COMMANDS

The purpose of the accumulate estimates program is to display tables of values relating to estimation. This is done using display mode commands, when prompted with DISP. A few of the commands also allow the user to set parameters for the tables to be displayed. Not all commands are available at alla levels. The availability will be listed for each command in brackets ( ).

All tables except the Ground Survey and Landsat table are by analysis district and strata. The Ground Survey and Landsat table is by strata only and cobers all analysis districts. The machine readable file contains one line for each analysis district, strata pair.

All tables are written to the listing file which may be the user's terminal or a disk file to be printed later or transmitted electronically to some other site.

Change display output file (All levels)
Changes the display output listing file. The user enters a file name for a disk file or carriage return for the terminal. The previous listing file, if any, is closed. Thus, if the user goes from a disk listing file to the terminal and then to a disk listing file, the second disk listing file will be a different file from the first so that the user will have two disk listing files, each with a portion of the total listing. This command is also used to direct output to and name the "machine readable file".

Coefficient of variation (Level two or higher only)
List the coefficient of variation as a percentage.

Districts (Level two or higher only)
Shows which analysis district, strata pairs are in proration district one. Other entries are left blank.

Frame units (All levels)
Lists the number of frame unites.

JES and Landsat table (Level three only)
Displays the GROUND SURVEY and Landsat table. The table is displayed by strata for all analysis districts in four columns. The first column is the JES values as read from the Ground Survey input file. The second column is Landsat regression, being the portion of the computed values attributable to regression estimation using Landsat data. The third column is Landsat non-regression, the portion of the computed values generated using proration. The fourth column is Landsat total, the entire computed estimate. in addition, relative efficiency, the ratio of Ground Survey variance to Landsat variance, is shown. The estimates and standard deviations are shown to the nearest acre.

If the estimate is zero for a column in some strata, the entire column is left blank. If the estimate is zero in all four columns for some strata, that strata is not shown even though it has not been deleted.

Large scale variable means (Level two or higher only)
The average number of pixels (X) classified to the specific crop is displayed for any analysis district, strata pairs that have regression estimators based on the results files read. All other entries are left blank.

Machine readable table (Level two or higher only)
An ASCII file is created for use with the county estimates modules. Do not forget to set the display output to a file name before execution, otherwise the file will be listed on the terminal. This command asks for a cover type name as input.

Means (All levels)
The mean values are shown. These means are calculated as the current totals estimate for that analysis district, stratum pair divided by the number of frame units. For those pairs where a regression estimator exists, this entry is the adjusted mean. For other entries it would be the ground data men (direct expansion or proration).

Regression coefficients (Levels two of higher only)
The user must specify which coefficient (B0 or B1) from the (currently read) estimator results files is desidered. Analysis district, stratum pair entries are shown where regression exist, other entries are blank.

Relative efficients (Level two or higher only)
The relative efficiency, the ratio of direct expansion variance to regression varaince, is shown. Where regression estimates are not available, the value is 1.0 except where the variance is zero in which case the value is set to 0.0.

Sample sizes (All levels)
The sample size, the number of segments sampled, for each analysis district, strata pair, is shown. These are calculated from the segment totals file, nor from the catalog.

Set display estimate type (Level two or higher only)
Changes the types of estimate to be displayed, RESULTS or PRORATION. Under RESULTS, either DIRECT EXPANSION ONLY or REGRESSION WHERE AVAILABLE may be called for. Only the latter will show regression results and it is the default after some estimator results files have been read. This command does not affect the Ground Survey and Landsat table or the machine readable file.

Small scale variable means (Level two or higher only)
The user is asked to select either pixel (x) or ground data (y) means for display . These means are based on segment data only; the y's are not adjusted by regression. Only those analysis district, stratum pairs with regression estimators are shown, other entries are blank.

Standard deviations (All levels)
Displays the standard deviations. Note that in some cases the sum of the squares of the standard deviations by strata may not be the square of the strata total standard deviation due to the way proration standard deviations are computed. Also, note that the total standard deviation is the square root of the sum os the squares of the strata total standard deviations which is not necessarily the same as the square root of the sum of the squares of the analysis district total standard deviations.

Totals (All levels)
Shows the totals (estimates) for the current level. PRORATION ONLY and DIRECT EXPANSION ONLY tables are designated as such, otherwise the table contains a mixture of estimators.

Type of estimation (Level two or higher only)
Shows the type of estimation in effect for each strata, analysis district pair. The abbreviations used are PR for proration, DE for direct expansion (if changed from regression), S-RE for single-variable regression, and M-RE for multi-variable regression.

Quit (All levels)
Exits display mode.

ADDAGG	ADD or subtract AGGregation files

PURPOSE

Adds or subtracts the contents of user-specified aggregation files. The addition or subtraction can be done on a by-field or a by-tract basis.

ACCESS

ADDAGG

INPUT

1) Aggregation files (type 24); the names of the files must be supplied by the user.
2) User entered data in response to prompts.

OUTPUT

1) A new aggregation file (type 24) is created. The name of this file must be supplied by the user.

HOW THE PROGRAM WORKS

The first file name typed in by the user is designated as the base file; subsequent files are either added or subtracted from the base.
For each file entered by the user,ADDAGG compares the relevant aggregation file parameters (ie, file coordinates, number of categories and fields) with those of the base file. If they agree, or if the user instructs ADDAGG to ignore discrepancies, the aggregation data from the input file are read in row by row (ie, field by field) and added to or subtracted from the aggregation data already accumulated. At any time the user may look at the current totals, write the aggregation to a new file, or list the files comprising the current aggregation.

COMMAND SUMMARY

ADDAGG recognizes the following commands:

(1) Add <filename> to aggregation
Adds the aggregation data from the specified file to the current cumulative aggregation.If this is the first file specified, it becomes the base file.

(2) Subtract <filename> from aggregation
Subtracts the aggregation data from the specified file from the current cumulative aggregation. If this is the first file specified, it becomes the base file.

(3) Print current aggregation
Displays on user's terminal (or produces a file for later printing) the aggregation totals thus far. This is the same command as in AGGR.

(4) File list (list files used in aggregation)
Displays (or creates a file for later printing) the files used to make up the current accumulation.

(5) Change aggregation mode
Allows user to specify whether data are to be added/subtracted by tract or by field. The user is asked this question at the start of the session; this command lets him change the mode. If the mode is changed without reinitializing the count, the resultant aggregation may be meaningless.The user may also use this command to change the default program action when negative totals are obtained, or duplicate fields are found.

(6) Group categories
Allows the user to combine or reassign specified classifications as in AGGREGATE.

(7) Begin new aggregation
Restarts accumulation; previous total are all erased. Settings, such as tract vs field accumulation, are retained.

(8) Current category grouping
As in AGGREGATE, displays the category mappings.

(9) Write accumulated aggregation file
Writes the output file

(10) Tract list (list tracts in aggregation)
Displays (or creates a file for later printing) the tracts found in the base file. If the program is operating in FIELD mode, all the tract names are displayed.

(11) Rename tracts
Allows the user to rename the tracts in the base file (TRACT mode only).

(12) Quit
Ends ADDAGG.

HOW TO USE IT

ADDAGG may be used to add or subtract files in one of two ways; when the program is started, the user is first asked "Add/Subtract by tract or field?".
If field mode is selected, corresponding fields of the files are added or subtracted. (ie, the first field occurring in the specified file is added or subtracted to the first field in the base file, and so forth). No further checking is done; it is left to the user to make sure that the same fields in different files will produce meaningful data when combined.
If tract mode is selected, all fields in a file are first accumulated to tract level before adding to or subtracting from the base file (ie, all fields in tract AA are first added together). The user may set the action taken if files contain duplicate fields (ie, two fields with tract AA, field 1.1).
To add or subtract files, the user enters "add filename" or "subtract filename". The first filename entered is designated the base file; all other input files are compared with it in terms of coordinates, number of fields and classifications, and tract assignments.
To create the output aggregation file, the user enters "Write";the program will then ask for the name of the output file.
The other commands are optional, and may be used to obtain a record of the session. "Begin" can be used to restart the addition or subtraction with a new base file.

NOTES/LIMITATIONS

(1) On Midas systems, the size of the aggregation table causes the program to run slowly. If it is certain that all categories and fields are less than a certain threshold, the sizes of MAXCATS and MFIELDS in the program constant definition area can be decreased. Depending on the limits chosen, ADDAGG may run faster.
(2) At any time, the user may switch from Field to Tract mode and back. However, the user is advised to reinitialize the output file first, (with command Begin); the results are not guaranteed to have any meaning otherwise.

AGGR	AGGRegation functions

PURPOSE

Provides aggregation statistics from classified image and mask files.
These statistics consists of the number of pixels in each category in each stratum of the mask file.

INPUT

1) Categorized window file (type 1). This file must contain only 1 channel of data.
2) Mask file (type 17). The names of the window and mask file nust be supplied by the user.
3) User entered data in response to prompts.

OUTPUT

1) Aggregation file (type 24); the name of this file must be supplied by the user.

HOW THE PROGRAM WORKS

When the user supplies a categorized image file and a mask file, AGGR determines whether the two cover the same coordinates.If so, then for each row in the area specified by the user, AGGR reads one row of data from the window file. It then reads the mask data for the corresponding values from the mask file; the mask data consist of byte pairs, with the first byte containing the number of pixels in the mask and the second byte containing the field assignment. AGGR uses these data to produce a table of the area coverage by field and category.

COMMAND SUMMARY

The main menu of AGGR recognizes the following commands:

1) Aggregate a categorized file using a mask
This command takes as input a type 51 categorized window file and a type 17 mask file and produces a type 24 aggregation file. The following submenu selections are optional:

(a) Select coordinates: allows the user to perform the aggregation over part of the image.
(b) Window file select: allows the user to specify a different window file at any time prior to aggregation.
(c) Mask file select: allows the user to select a different mask file.
(d) List coordinates: displays boundaries of window and mask files, as well as coordinates selected for aggregation.
(e) Scale factor: allows user to specify the area represented by 1 pixel.
(f) Tract and field list: allows user to display tract and field data from mask file as they are being read.
(g) Group categories: allows user to reassign category groupings
(h) Display current groupings: allows user to examine current category groupings.
(i) Write file: aggregates image and mask data and writes the result to an aggregation file.
(j) Quit: returns user to the main menu.

2) Display an aggregation file on terminal
This command produces the same information as (3) below, but displays the results on the user's terminal.

3) Print an aggregation file
This command produces a hard-copy listing of an aggregation file.

4) Quit
This command terminates the aggregation program.

HOW TO USE IT

AGGR can be used in two ways. To produce a new aggregation file, the user enters "AGGREGATE" in response to the main prompt. The program will then ask for the names of the two input files. To write the aggregation file, the user enters "WRITE"; again, AGGR will prompt for the name of the output aggregation file. The other commands in this section are optional. AGGR may also be used to display or print an existing aggregation file. To display/print an aggregation file, the user should enter "DISPLAY" or "PRINT" in response to the main prompt. AGGR will then ask for the name of the input and, for print, and output file.

NOTES/LIMITATIONS

On Midas systems, the size of the aggregation table causes the program to run slowly. If it is certain that all categories and fields are less than a certain threshold, the sizes of maxcats and mfields in the program constant definition area can be decreased. Depending on the limits chosen, AGGR may run faster.

AUTOCLIP	AUTO CLIPping

PURPOSE

A New Program has been added to MARS-PED. The program sequence SCAT followed by BADPIX may be replaced with WINSTAT, PRINCPL then AUTOCLIP. AUTOCLIP compares the principal component value generated from the raw pixel value to the principal component mean +- X times the standard deviation of the principal component. Out of range pixels and their principal component values are output in a text file named the same as the output clipped file with the last 3 characters replaced with 'TXT'. Two options are available for packed files. The Delete option clips out of range pixels from the output file. The Save option truncates the out of range principal component value(s) at the editing limit(s) and generates a new pixel value. If the input file is image as opposed to packed, then the Save option is used.

BUGS FIXES ETC;

October 24, 1989 sbw
FURTHER STUDY SHOWS 3.5 STD DEV BETTER CLIP LIMIT. CHANGED DEFAULT TO REFLECT.
USE DELETE OPTION ON PACKED WINDOWS. SAVE OPTION ON MULTI-WINDOW FILES. RESULTS COMPARE WITH OLD METHOD. SAVES WORK.

INPUT

Principal component file type 66
Window file type 51 image or packed

OUTPUT

Clipped window file type 51 with pixels altered or deleted
Text file

USER INPUT FROM TERMINAL OR BATCH FILE

OPTION { Deleted, Save } Caps indicate required
EDIT LIMIT If desired Default 3.5 Standard Deviations
{3.0, 4.0 etc}
File Names

EXAMPLE

```
User input enclosed  with{}

{WINSTAT<CR>}
COMPUTE WINDOW FILE STATISTICS, VERSION 1.0, JUNE 27, 1989
Directories to use, one per line; CR only when done.
:{[WININGS.PACK]<CR>}
:{<CR>}

Input Window File (CR to quit):{OTHER.PCK<CR>}

Number of channels=3 number of pixels=37227

MEANS:
    66.24    63.53    80.31

VARIANCE/COVARIANCE MATRIX:
    84.32   100.53    31.69
            125.82    32.68
                     110.94

Output Statistics File (CR for none):{OTHER.STAT<CR>}

Input Window File (CR to quit):{<CR>}

CPU USED=7.
$ {PRINCIPL<CR>}
Principl Ver 1.1 September 28, 1989
Directories to use, one per line; CR only when done.
:{[WININGS.PACK]<CR>}
:{<CR>}

 Enter Name of Output Principal Components File to Save, <CR> no
save
:{OTHER.PRINCP<CR>}

 Input StatFile Name
:{OTHER.STAT<CR>}

 Enter Range of Catagorys For Principal Component Analysis
:{1<CR>}


     From Raw Data

Chan Means  Std. Dev.       1        2        3
   1    66.24   84.3174  0.7802  0.5923 -0.2011
   2    63.53  125.8218 -0.6243  0.7168 -0.3106
   3    80.31  110.9395 -0.0398  0.3678  0.9290

Principal Components
   1 Mean=      8.82363 Sigma=     1.50462
   2 Mean=    114.32244 Sigma=    15.02191
   3 Mean=     41.56138 Sigma=     9.65178

 get another Category ?:{N<CR>}

OTHER.PRItmp Deleted
CPU USED=0.
$ {AUTOCLIP<CR>}
 Auto clipping of a packed File
Version 1.1 October 24, 1989 by Sherman B. Winings
Directories to use, one per line; CR only when done.
:{[WININGS.PACK]<CR>}
:{<CR>}
Input Principal Components File:{OTHER.PRINCP<CR>}
 Enter Edit limit in terms of S.D.-- Default 3.5:{<CR>}
```

```
Enter Packed File Name:{OTHER.PCK<CR>}

Save or Delete out of range Pixels:{DEL<CR>}

Delete Pixels
 Enter Output Clipped Pack File Name:{OTHER.CLIP<CR>}
A temporary file called TEMPWIN.WIN has been created
Deleted Pixels Will Be in OTHER.CTXT

 Pixels exceeding Upper Principal component Limit 1503
 Pixels Less than Lower Principal component Limit  623
 Total Pixels Deleted                             1769
TEMPWIN.WIN Deleted
CPU USED=20.
$ {TYPER  OTHER.CTXT<CR>}

Principal Component Means
     8.824   114.322    41.561

Eigan values
     2.264   225.658    93.157


Rotation Matrix
      0.78020      0.59233     -0.20106
     -0.62426      0.71683     -0.31057
     -0.03983      0.36783      0.92904

    edit limits
 Channel           Max       Min
   1               12.59      5.06
   2              151.88     76.77
   3               65.69     17.43

LIMIT EXCEEDED
  PRINCIPAL COMPONENT EXCEDED(1=3RD, 2=1ST, 3=2ND)
    3RD
         1ST
              2ND
                    1     2     3 ORIGINAL PIX
L 1  2.6110.0 55.6   56    60    92
L 1  4.0133.3 45.3   73    79    91
U 1 32.2183.3 33.2  127   101    97
U 2  9.6168.2 40.4   99   102    99
U 1 13.8117.8 47.6   71    61    87
U 1 19.7144.8 40.4   93    79    90
U 1 17.8120.7 46.6   76    61    87
U 1 13.2112.5 49.7   67    57    87    67    57    87   MORE THAN ONE
U 3  9.5 92.6 66.0   49    40    95
.
.
.
L 1  4.7102.9 47.7   55    56    82    55    56    82
L 1  4.5101.9 49.2   54    55    83    54    55    83
L 1  3.9102.7 48.9   54    56    83
U 1 13.0 96.1 25.0   62    53    58    62    53    58

OUTPUT OF TEXT FILE UNDER SAVE OPTION

$ TYPE OTHER2.CTXT

Principal Component Means
     8.824   114.322    41.561

Eigen values
     2.264   225.658    93.157
```

```
Rotation Matrix
      0.78020         0.59233       -0.20106
     -0.62426         0.71683       -0.31057
     -0.03983         0.36783        0.92904

    edit limits
Channel                Max       Min
  1                   14.84       2.81
  2                  174.41      54.23
  3                   80.17       2.95


LIMIT EXCEEDED
  PRINCIPAL COMPONENT EXCEDED(1=3RD, 2=1ST, 3=2ND)
    3RD
         1ST
              2ND     ORIGINAL PIX  ALTERED PIX
                       1    2    3    1    2    3
L 1  2.6110.0 55.6    56   60   92   56   60   92
U 1 32.2183.3 33.2   127  101   97  108  105   94
U 1 19.7144.8 40.4    93   79   90   90   82   90
U 1 17.8120.7 46.6    76   61   87   74   63   88
L 1  2.4 84.4 29.1    46   50   58   46   49   58
L 1  2.0 95.3 54.9    47   50   86   48   49   86
L 1 -2.0117.6 50.2    58   70   90   62   67   90
L 1  0.6125.2 43.0    66   76   86   68   74   86
U 1 15.4124.6 43.9    77   66   86   77   67   86
U 1 15.6111.5 45.8    69   56   83   69   57   83
U 1 19.4158.5 59.7    97   83  113   94   86  114
.
.
.
.
U 1 15.3141.1 37.4    88   80   86   88   80   86
U 1 17.9133.6 50.1    83   69   95   81   71   95
```

BADPIX	BAD PIXels to be deleted or substituted

PURPOSE

To find the bad pixels in a packed window file input. Range values for these pixels are obtained from the user. When the out of range pixels are found,the mean of the good pixels will be substituted or a zero (0) will be inserted for deletion of the bad pixels.

INPUT

1) Window files (type 51). The file must be a packed file and the name must be supplied by the user.
2) User entered data in response to the prompts.

OUTPUT

1) A new revised window file is created if any bad pixels are found. If no bad pixels are found, then no file is created.

HOW THE PROGRAM WORKS

This program is designed to find pixels which are out of the range specified by the user. The user types in an input file name and the program will print pertinent information on the screen. The user enters range values that the pixel must be within for each channel. The program then asks if the bad points should be deleted or substituted for. An output file is only created when badpixels are found, and thisfile name is supplied by the user.

COMMAND SUMMARY

None, this program is simply prompt oriented.

HOW TO USE IT

BADPIX is user oriented. Prompts are self explanatory and are helpful. If mistakes are made, an error message will print on the screen with prompts to help the user correct the error.
When the program begins execution, the user will be asked,

"Input Window File Name:".

Upon entering the file name, if it is correct and exists in the user's current directory, file header information will be printed on the screen along with the next prompt which is

"Input Channel number desired:".

Any channel may be entered as long as it is within the range of channels for that file. A <cr> to this prompt means the user does not wish to change the default values.If a valid channel was entered, a prompt asking for the

"Minimum value for channel [0..255]: "

will be printed.

A <CR> will default to zero (0), or a minimum value which the user entered will become the minimum. The following prompt will then ask for the maximum value at which time the user will enter a <CR> for a default value of 255 or enter a maximum value. The prompt for the channel value will appear again. If the user has completed changing the range values for the channels desired, then enter a <CR>. Otherwise, enter a channel and pixel range values until all changes are completed.

The program then prints the user entered default values and prompts whether the bad pixels should be deleted or substituted for by prompting,

"Would you like to delete the points (Y or N)?:"

Yes means deletion will take place and no, the mean value will be substituted for the bad pixels.If no bad pixels are found, the program completes execution without creating a new file. When a bad pixel is found, the number of bad pixels will be printed along with a message stating either that thepixels will be deleted or substitution of pixels with the mean value will take place. The user will be prompted for an output file name and a file with the revised information will be created. With the output file specified the user is asked the following:

"Do you want to perform corner clipping (Y/N) ? "

If "YES", then the user will be prompted as to which channels to use. Selection is done by either answering Yes or No. After all channels have been queried, their limit values are displayed. The program then processes the specified channels, showing either the number delete or replaced sets upon completion.

If "NO", then process in an individual channel basis. At this time, the user may choose to continue or quit the running of this program by answering Y or N to the final prompt.

NOTES/LIMITATIONS

This program was built with a channel constant of up to eight (8) channels and a maximum of 1000 bad pixels. Also the window file entered must be in PIL format and must be a packed file.

REVISION NOTES

Made exceeding bad pixel limit a warning message.Added a procedure called CNRCLIP which allows the user corner clipping of scattergrams. This procedure permits the user to compare specified channels and those values not within the comparision levels are replaced with their respective mean values or deleted.

CALCOR	CALculate COoRdinates

PURPOSE

To take as input
1) An image calibration file, and
2) Coordinate points in Latitude-Longitude, UTM, or Image coordinate systems

And to output to the terminal
1) The corresponding coordinate points in any of the other coordinate systems, or
2) DWR quad number and block

INPUT

1) An image calibration file
2) Coordinate point in Latitude-Longitude, UTM or Image coordinate system
3) If UTM is selected, a nearby Latitude-Longitude coordinate point

OUTPUT

1) Coordinate point in one of the remaining coordinate systems, or
2) DWR quad number and block

HOW THE PROGRAM WORKS

The program uses the transformation functions utilized in the segment digitizer, mask generator, and others to convert coordinates in one coordinate system to another. A central latitude and longitude value is necessary (obtained from the segment network file in other modules) to calculate from or to UTM coordinates. The routine to calculate DWR quad number and block originate with the State Department of Water Resources in Sacramento, California.

COMMAND SUMMARY

Calcor is not a menu driven program.

HOW TO USE IT

Calcor begins by requesting the name of an image calibration file. Then the format of the input and output coordinates is requested in two separate questions. Note that all possible formats are displayedin parentheses in the prompt.
A response of single characters of either upper or lower case is sufficient.Then the coordinates themselves for the input coordinate system must besupplied in the proper format as displayed in parentheses.
A prompt for an appropriate Latitude-Longitude coordinate point will initially appear if input UTM coordinates were requested. This allows general placing of the UTM coordinates.

Latitude-Longitude and UTM coordinates may be typed as floating point, but Image coordinates must be integers. The corresponding points in the output coordinate system are then written to the terminal in the format displayed in parentheses. Immediately, another coordinate point in the input coordinate system is requested. If the point is supplied, the same transformation will be applied. However, if the user enters a carriage return only, the user is sent back to the point of specifying an input coordinate system. The same image calibration file will be used unless a <return> only is entered in which case the user will be prompted for a new image calibration file. Another <return> only will cause the program to exit.

NOTES/LIMITATIONS

None.

CATED	segment CATalog EDiting

PURPOSE

To create or edit a segment catalog containing information and characteristics of user-defined segments; in this way, groups of segments with common characteristics may be easily accessed by other programs.

INPUT

Segment Catalog File (type 49) (optional)
User disk or tape file of updates (optional)
User-entered commands and values

OUTPUT

Segment Catalog File (type 49)

HOW IT WORKS

A segment entry contains the segment number and values for eight possible parameters. These eight parameters and their data types are :

1) County (character string)
2) Frame Ids (character string)
3) Strata (integer value)
4) Quad (character string)
5) Map type (character string)
6) Adjustment Factor (real value)
7) Expansion Factor (real value)
8) Analysis District (character string)

Only the segment number is required, all other eight parameters may have values or be undefined (missing). With the exception of the frame ids string each parameter is a single value, e.g. one county name or a single Adjustment factor. The Frame Ids string may contain the ids for several frames, seperated by commas.
The program allows the user to create a new segment catalog or to examine/update an existing catalog. Any existing catalog is read into memory, and all updates, deletions, and insertions are done to the catalog in memory. The catalog is written out to a file either by an explicit command or when exiting the routine.

COMMAND SUMMARY

Level 1

OPEN - Open an existing catalog file for editing.
CREATE - Create a new catalog file from user input.
QUIT - Exit the program

Level 2

UPDATE - Change an existing segment entry
MASS UPDATE - Change many existing segment entries
DELETE - Delete one segment entry
INSERT - Add one segment entry
LIST - List certain segments in the catalog
SORT & LIST - Same as list but in order by segment no.
SHOW MISSING ENTRIES - List segment with undefined entries.
WRITE CATALOG TO DISK - Output catalog to a disk file.
AUTOMATIC UPDATE FROM DISK/TAPE - Update/insert with data from a file.

Level 3

ALL - Perform operation (e.g. list) on all segments.
SEGMENT - Perform operation on a particular segment.
COUNTY - Select segments with a certain county name.
STRATA - Select segments with a certain strata.
FRAME - Select segments with a certain frame id.
QUAD - Select segments with a certain quad name.
MAP TYPE - Segment selection based on map type.
EXPANSION FACTOR - etc.
ADJUSTMENT FACTOR - etc.
ANALYSIS DISTRICT - etc.
PLACE ENTRY INTO CATALOG - Indicates update complete.

HOW TO USE IT

At the top level of the program, indicated by the '1)' prompt,there are only three commands. QUIT to exit the routine, CREATE to create a Segment Catalog File, or OPEN to examine and/or edit an existing Segment Catalog File.
The second level of the routine, reached by either the OPEN or CREATE command, has nine subcommands. These commands and their functions are:

LIST - List the segment entries for either all segments, or for those segments having a specific parameter value.The segments are listed in order they appear in the catalog. For example, 'list all segments with in quad 15'.

DELETE - Delete an entry for one segment, indicated by number.

INSERT - Add an entry for a segment. The routine prompts for the values for the segment number and all parameters.

UPDATE - Change or add a parameter value for a specified segment entry. All parameters may be updated for a single segment. The update is ended by the 'Place entry in catalog' command.

SORT AND LIST - The same as LIST above, but the segment entries are listed in order by segment number.

SHOW MISSING PARAMETERS - List all segments that have a missing, or undefined, value for a specific parameter. For example, 'list all segments with no value for the Map Type parameter'.

WRITE CATALOG TO DISK - Output the current catalog to a disk file. The routine prompts for the file name.

MASS UPDATE - Allows changing or adding a parameter value to many segments. All segments can be updated, or only those segments meeting a specified criteria will be updated. For example, 'change the frame parameter value for all segments to 1234-76543', or 'change the analysis district value to D31 for all segments with strata value 34'.

AUTOMATIC UPDATE FROM DISK/TAPE FILE - Allows updating of existing segments and insertion of new segments from a disk or tape file. The user is prompted for the name of the update file, the format of the file (in a FORTRAN-type format statement), and the names of the elements to be updated. Also allows the segment catalog to be "cleaned-up" by automatic deletion of segments not updated or inserted in the current run.

QUIT - Exit to level one. If there have been any changes to the catalog since the initial OPEN command, or since the last WRITE TO DISK command, the routine will prompt for an output file name to save the altered catalog.

NOTES/LIMITATIONS

The current version of the routine does not allow segment number input from a region file. This option will be added someday.
There is some confusion as to what an Expansion Factor is. The routine defines it as a real value, but perhaps it should be an integer. The routine has only been tested with rather small catalogs. There may be some problems with large catalogs that have not appeared during testing, e.g. How long does a sort & list take with a big catalog?

CLAS	maximum likelihood CLASsification

PURPOSE

Classification of a multi-channel window file using maximum likelihood

METHOD

A multi-channel data set is classified with a statistics filesupplied by the user using the maximum likelihood algorithm. Each point in the data set set is assigned the class number corresponding to the closest spectral class in the statistics file as determined by the maximum likelihood decision rule. If during the inversion of the spectral classes in the statistics file, a class is found to be ill-conditioned, then that class is given a very low probability so that no points will be assigned to it. The output of this program is a single-band (one channel) categorized image, and (optionally) a single-band threshold image.The threshold image contains values from 1 to 16 which index a table of probabilities, i.e., the threshold values indicate the probabilityof correct classification for each point.

INPUT

Raw data window file (type 51)
Statistics file (type 54)
User entered values

OUTPUT

Categorized window file (type 51)
Threshold window file (type 51)

HOW TO USE IT

The user is prompted for the necessary information to run the program as follows:

"Input statistics file:" The name of a valid statistics file of type 54 should be entered. A carriage return only terminates the program.

"Want to use apriori probabilities?" A YES response indicates that the apriori probabilities are to be taken from the statistics file and used in the program. A NO response indicates that apriori probabilities are not wanted.

"Input window file:" The name of a valid window file of type 51 should be entered. A carriage return only terminates the program.If the number of channels in the window file does not agree with the number of channels in the statistics file previously entered, then the program is terminated.

"Classify which window:" This prompt appears only if the input window file contains more than one window. Response indicates which window is selected for classification. A carriage return only indicates that all windows are to be classified.

"Categorized window file:" The file name of the output categorized window should be entered. If the file already exists, the user will be prompted whether to overwrite it. A carriage return only indicates that the output categorized window is not to be saved.

"Threshold window file:" The file name of the output thresholdwindow shoud be entered. If the file already exists, the user will be prompted whether to overwrite it. A carriage return only indicates that the output threshold window is not to be saved.

LIMITATIONS

1) The maximun number of channels is 8.
2) The maximum number of classes is 64

CLUST	multi-channel CLUSTering

PURPOSE

Cluster a multi-channel window file

METHOD

The set of pixels specified by the user is read into core memory and clustered using the LARS algorithm. The data is first accessed to determine the mean for the pixels in each channel. The data is then accessed again to determine the variance from the mean for each channel. The starting mean values (seedpoints) are then computed and distributed along the diagonal in multi-dimentional space according to the total variance found.
From this point on, the clustering algorithm is iterative and is terminated when the convergence criterion specified by the user is satisfied or the maximum number of iterations is exceeded. During each iteration, each pixel is accessed and compared with each mean determined by the previous iteration (or starting value for the first iteration). Each pixel is assigned the class number of the mean closest in Euclidean space. After all pixels have been assigned to the nearest class, new means are computed from the pixels assigned to each class. Convergence is defined as the percentage of pixels which have not changed class since the last iteration. Thus, 100% convergence occurs when each pixel is assigned to the same class as the previous iteration.

INPUT

Window file (type 51)
User entered values

OUTPUT

Statistics file (type 54)

USAGE

The user is prompted for the necessary information to run the program as follows:

Input window file: The name of a valid multi-channel window file of type 51 should be entered. A carriage return only terminates the program. The message "Can't cluster more than nn channels" also terminates the program.

Cluster which window: This prompt appears only if the input window file contains more than one window. Response indicates which window is selected for clustering. A carriage return only means all windows are clustered together. The message "Window can not be selected" terminates the program.

Want to sample the data?: If this prompt is preceeded by the message "Can't cluster more than nnnn pixels" then the data must be sampled; a NO response terminates the program. Otherwise sampling is optional. A YES response is followed by the prompt "Enter sampling increment".

Number of categories: Enter the number of output classes desired. A carriage return only will enter a default of one.

Percent convergence: Enter the desired convergence value. A carriage return only will enter a default value of 98.5%.

Max number of iterations: Enter the maximum number of iterations allowed. A carriage return only will enter the default value of 20. A value of 0 indicates no limit to the number of iterations.

Delete final classes with ill-conditioned matrices?: A YES response will result in a check of the variance-covariance matrix of the clusters resulting after the last iteration. Classes with ill-conditioned matrices will be deleted. A NO response will result in all final classes being retained.

Statistics file write interval: Enter a number to indicate the number of iterations between writes of the output statistics file. A carriage return only indicates that only the final statistics will be saved.

Output statistics file: Enter the name of output statistics file to be created. A carriage return only indicates that the output statistics file is not to be saved.

NOTES/LIMITATIONS

1) The maximum number of channels is 8.
2) The maximum number of classes is 25.
3) The maximum number of pixels is 8000.

CMASKP	Pixel Count of MASKed fields

PURPOSE

By field, this command lists the non-background pixels for both the boundary and non-boundary pixels separately and gives the total number of pixels. The program also returns the total number of pixels masked, the total number of non-background pixels for this file, and the first and last row in which a non-background pixel was found.

INPUT

Mask file (type 17); the name of the file is supplied by the user.

OUTPUT

None.

HOW THE PROGRAM WORKS

This program is prompt oriented. To the prompt, a file name is entered. The data is then displayed on the screen. When all the data has been shown, the user can then enter another file or abort the program.

COMMAND SUMMARY

None.

HOW TO USE IT

Upon entering the program, the user is asked to supply mask file (type 17) to be displayed. If a non existent file is entered, the user will be prompted for the file name again.
After the command has completed displaying the data, the user will be asked if there are more files to be entered. The prompt for a new file will appear if "y" is entered and will abort if "n" is entered.

NOTES / LIMITATIONS

None.

CORREC	calculate and print percent CORRECt

PURPOSE

Reads a Table file (type 56) and prints a matrix showing ground-data cover versus categorized cover, in a 'categorized from vs. categorized to' format. It also prints a percent correct and percent commission for each cover type.

INPUT

Table File (type 56)
Statistics File (type 54) (optional)
User Supplied Input

OUTPUT

Printed on the CRT Screen
A text file with percent correct information (optional)

HOW IT WORKS

The table file contains information by segments. For each segment there is a table (hence the name) telling how many pixels of each ground cover type has been categorized into which category. (The cover types are from the ground data, the categories from the clustering.)
This program reads this table, gets from the user a cover type assignment for each category, then prints out a 'categorized from - categorized to' matrix of pixel counts and a percent correct / percent comission figure for each cover.
There are options for saving the percent correc/percent comission list in a text file, and for printing the variability stats for the segments (see below).

COMMAND SUMMARY

There are no 'commands'. The user interface is a straight-line query-response dialog.

HOW TO USE IT

When the routine is entered it prints the version number and immediately begins its queries

1. **Enter other directories to be searched**

2. **Enter the table file name.** The routine reads the file and prints some info about it.

3. **Use a Statistics file for crop assignment to categories.**
   a. If yes, the routine asks for a statistic file name. It will read the file, use the info to make the category-cover type assignments, then report on those covers that have had no categories associated and the categories that were not assigned to a cover.

b. If no, the routine asks the user to specify, for each cover type, the categories to be associated with it.
A CR only means no categories will be associated with that cover type.

4. Show unassigned categories and use in percent correct?
a. If yes, those categories that were not assigned to a cover will be printer in the matrix grouped into a cover called 'other', and their pixel counts will be used in the percent correct calculation.
b. If no, the pixel counts for the unassigned categories will be ignored.

5. Use unassigned covers in computations?
a. If yes, the ground covers that have no associated categories will be printed in the matrix and the pixel counts used in the percent correct calculation.
b. If no, the pixel counts for those ground covers will be ignored.

6. At this point the 'from-to' matrix will be calculated and printed on the screen. The ground-data cover types are printed by name on the vertical (left) axis and the categorized covers are printed by number along the horizontal (top) axis. The covers are ordered such that the diagonal entries in the matrix show the pixel counts for the number of pixels correctly categorized. Following the pixel count matrix comes the percent correct / percent comission list.

7. Save percent correct in a file?
If yes, the routine asks for the name of a file, then writes out the percent correct list as text to this file.

8. If the table file contained information for only one segment the routine is done and it will exit at this point. If it was a multi-segment table file the routine asks if you wand to see the segment variability statistics.
If no, skip to step 9.
If yes, the routine will print the following information for each ground cover:
The segment number and percent correct for the segment with the lowest percent correct.
The segment number and percent correct for the segment with the highest percent correct.

9. The routine will ask if you want to see percent correct by segment.
If no, the routine exits.
If yes :
a. The routine asks for a list of segment numbers to show. You can specify all segments by a CR only.
b. The routine will print a pixel count matrix and percent correct for each segment. This table is similar to the full pixel count matrix except only the covers that actually occur in the segment are shown.
c. If all segments were requested in step a. above,the routine is done and will exit.
If not all, The routine asks if more segments are desired.
If yes, steps a and b are repeated.
If no, the routine exits.

NOTES/LIMITATIONS

None.

CPVERT	Calibration Point conVERsion (Not used in MARS-PED)

PURPOSE

Generate an ASCII file of segment calibration marks to be used with video-digitized segments.

INPUT

1) List of segments.
2) State and Year identification
3) Segment network files

OUTPUT

ASCII file of calibration marks.

HOW IT WORKS

User is asked for a list of segments to use and the file name for the ascii output. The processing steps for each segment file is: open segment file and check; read file; convert x and y digitized coordinates into UTMs; convert x and y utms into latitude and longitude; convert real values into a string; write ascii output file.

DSPWIN	DiSPlay WINdow file data

PURPOSE

DSPWIN allows the user to print out selected subwindows from a window file (type 51) out on the user's terminal, a text file (for later printing), or to a special graphics metafile (see gmf.ld for more information). Windows printed out to a graphics metafile can additionally be scaled by specifying the size of a pixel in meters. The MARS-PED module GMFRAS can then be used to convert the graphics metafile into a raster format file, which can then be printed out on raster format devices using the appropriate driver, such as RASPRT for the Printronix line printer.

INPUT

1) User supplied commands.
2) Window file (type 51).
3) Plot Character text file.See pltchr.fd for details.
4) Map Table text file.See maptab.fd for details.

OUTPUT

Depending on user commands:
1) Map Table listings, Histograms, and Printer Plots to either the user's terminal or a text file.
2) Graphics Metafiles of plotted subwindows. See gmf.fd for details.
3) Map Table text files. See maptab.fd for details.

HOW IT WORKS

DSPWIN uses a multi-level command menu structure to allow the user to set options, manipulate a mapping table, select desired input files, window and channel number, and subwindow coordinates to select data for subsequent histogramming and subwindow display. These same options are available for multiple window(s) and/or channel(s).

Four types of subwindow plots can be generated:

- Value printer plots, where the actual numeric value of the pixel is printed out;
- Single-Character printer plots, where a single character is used to represent a pixel value;
- Multi-Character printer plots, where 1 or more over-strike characters can be used to represent grey scale for a pixel value;
- Graphic Metafile plots, which create a special graphic metafile that can be subsequently plotted out on printers, plotters, bit-mapped CRT's, and other graphic devices using Graphic Metafile plotting programs.

Graphic Metafile plots can be scaled according to pixel size, allowing the results to be usable with maps. Single- and Multi-Character print plots use a Plot Character text file to determine which character or sequence of characters are to be used for each pixel value. Also, special characters may be specified which allow the user to insert prescribed characters in the generated output plot. See PLTCHR.FD for details on the format of these Plot Character text files.
DSPWIN can also generate histograms from the selected subwindow. Histograms can be printed out with either pixel totals, percentages, and/or a bar graph, all selectable by the user. Both subwindow plots and histograms are mapped through a lookup table called the map table.

Commands allow the user to initialize the map to a 1-to-1 linear map (the default), to load or save a map to/from a text file, or to generate a linear or equiprobable map from the current histogram.

COMMAND SUMMARY

Upon program startup, DSPWIN prints out a welcome message and then prompts for the name of a window file. If the entered window file is multi-channel or multi-window, DSPWIN will prompt for the desired channel(s) or window(s) to use. If the entered file name cannot be found, is not a valid window file, or does not use BYTE values, an error message is sent and the user is prompted to re-enter another file name. If all is successful, DSPWIN then enters the top level command loop described below.

TOP LEVEL COMMAND MENU:

1) INput File Selection:Select another window file for input.

2) CHannel Number Selection:Select channel(s) from current file for processing.

3) WIndow Number SelectionSelect window(s) from current file for processing.

4) SUbwindow Selection:Select subwindow from current file and window for processing.
NOTE:This subwindow will be the same for all current channel(s).

5) RAw Data Defaults:Select Default values for Raw Data. This option sets up parameters within DSPWIN to generate value printer plots.These parameters can be overridden using the "Set Options command" described later.

6) CAtegorized Data Defaults: Select default values for Categorized Data. This option sets up parameters within DSPWIN to generate single character printer plots. If a plot character file has not yet been specified, the user is asked to supply it. Like the "RAw Data Default" command above, the user can override these values using Set Options.

7) Greyscale Data Defaults:Select default values for Greyscale Output. This option sets up parameters within DSPWIN to generate single character printer plots. If the file by the name of PLTCHR.DEF (upper case letters) is not present in the PEDITOR system directory in which PEDITOR.CROPS resides, the user is asked to supply it. The output is also routed to a file, for which a name is requested from the user.

8) SHow Current DISPLAY Status:Presents current window file header, current channel(s) and window(s) set, and DSPWIN parameter settings.

9) MApping Table Control:Transfers the user to the Mapping Table Control menu. See below for details.

10) SEt Plotting/Histogram Options:Transfers the user to the Set Options menu. See below for details.

11) Histogram:Transfers the user to the Histogram Options menu. See below for details.

12) PLot Generation:Generates subwindow plots based upon parameters set above.

13) Quit:Exits the DSPWIN program.

MAPPING TABLE MENU:

<CR> will return the user to the TOP LEVEL MENU.

1) INitialize Mapping Table:Initializes mapping table to a 1-to-1 linear mapping. This is the default setting when DSPWIN is started.

2) LISt Current Mapping Table:Lists the current mapping table(s) to output for all channel(s) and window(s).

3) LOad Mapping Table From File:Loads the mapping table from the user specified map table text file. See maptab.fd for details on file format.

4) SAve Mapping Table to File: Saves the mapping table to a user specified map table text file. See maptab.fd for details on file format.

5) GEnerate Mapping Table: Allows user to manually construct a mapping table. This option is not currently supported. If you wishes to build their own mapping table, see maptab.fd for details on map table file format, and use a standard text editor to construct one. Use the Load Mapping Table command described above to load in the generated file into DSPWIN.

6) LINear Mapping From Histogram: Generates a linear mapping table, based upon the current histogram information. User is requested to supply the number of grey levels desired.

7) EQuiprobable Mapping From Histogram: Generates a equiprobable mapping table, based upon the current histogram information. User is requiested to supply the number of grey levels desired.

SET OPTIONS MENU:

<CR> will return the user to the TOP LEVEL MENU.

1) Histogram Options: allows user to specify a number of parameters relating to the generation and printing of histogram information. See below for more details.

2) Output Routing: allows user to select destination of map table listing, histograms, and printer plot output. Available options are "Terminal" or "File". If "File" is requested, the user is then prompted to supply the name of a text file for all subsequent map table, histogram, and printer plot output.

3) Display Width: output for histograms and printer plots is set up for a standard 80 character terminal. This options allows the user to specify a shorter or longer column width for histogram and printer plot output.

4) Plot Type Selection: allows user to select between value printer plots,single character printer plots, multi-character printer plots, or graphic metafile plots. For single and multi-character printer plots, if a plot character file has not yet been supplied, the user is prompted for it. For graphic metafile plots, if a pixel scaling size has not yet been supplied, the user is prompted for it.

5) Character File for Plots: allows user to select a Plot Character File for subsequent single- and multi-character printer plots.

6) Graphics File Parameters: allows user to define parameters for graphics plots. The user is asked to supply the name of a graphics file to use for output, the maximum mapping scale to be used when the graphics file is created, and the size of a pixel in meters for the current input scene, both in the horizontal and vertical directions.If the pixel size and mapping scale would result in a plot where the resulting pixels would be too small to plot legibly, a warning message is printed. The user is then asked if s/he would like to have a title and/or top and side annotations on the final plot. If a title is selected, the user is then prompted for the scale of the title in "pixels", where pixels refers to the size of a single pixel in the finished plot. If the selected size is too small to be plotted legibly given the specified pixel size and mapping scale, a warning message is printed.If top and side annotations are requested, the user is then prompted to supply the scale of the annotation in pixels, plus an increment and top/side offsets.
The increment defines the frequency of annotations, with a increment of one meaning that every pixel row and column will be annotated. Note that if an annotation size is 2, the lowest increment that can be specified is also 2. Top and side offsets determine when the annotation will begin, starting from the top/left corner. For example, an offset of 0 for both the top and side annotations will start annotation on the upper left corner.

7) File Identifier: allows user to select a title for plot output other than the original header data stored with the window file.

HISTOGRAM MENU:

<CR> will return you to the TOP LEVEL MENU.

1) Generate Histogram: generate histogram based upon current parameter settings.

2) Print Histogram: print histogram to output using current parameter settings.

3) Set Histogram Options: allows user to set parameters controlling the generation and printing of the histogram. Histogram printing options allow user to choose which of the following three items are to be printed (based upon a yes or no answer): pixel totals, percentages, and a bar graph of the results.
The user is then asked to supply the range over which he wishes the histogram to be calculated:
using just the selected subwindow (SUBRANGE),
the entire selected window (WINRANGE)
or the entire file with all its windows (FILERANGE).
This last option is only available for PIL format window files.

HOW TO USE DISPLAY

To generate a simple value printer plot of a raw window file:

DSPWIN
< supply window file name when asked >
< supply desired channel number(s) if multi-channel data >
< supply desired window number(s) if multi-window data >

SUbwindow Selection
supply desired subwindow coordinates when asked >

PLot Generation

QUit

To generate a single character printer plot from a categorized window file:

DSPWIN
< supply window file name when asked >
< supply desired channel number(s) if multi-channel data >
< supply desired window number(s) if multi-window data >

SUbwindow Selection
< supply desired subwindow coordinates when asked >

CAtegorized Data Defaults
< supply name of Plot Character File when asked >

PLot Generation

QUit

To generate a histogram from a selected subwindow:

DSPWIN
< supply window file name when asked >
< supply desired channel number(s) if multi-channel data >
< supply desired window number(s) if multi-channel data >

SUbwindow Selection
< supply desired subwindow coordinates when asked >

HIstogram

Generate Histogram

Print Histogram

<CR>

QUit

Other more advanced tasks can be accomplished by setting options and selecting menu commands as appropriate. See previous section on Command Summary for details on using each command.

NOTE/LIMITATIONS

The user should remember that histograms and subwindow plots are ALWAYS mapped through the map table. If the map table has been changed and the user wants to see an un-mapped histogram or subwindow plot, s/he must remember to initialize the map table to its default linear mapping.

The current version does not allow manual construction of the map table using the Generate Map Table command. Multi-Character printer plots are currently implemented using only a CHAR-BACKSPACE-CHAR type sequence; there is currently no support for a TEXT-<CR>-TEXT-<CR><LF> type overstrike or overstrike using fortran carriage control.

EDUNIT	frame UNIT file EDiting

PURPOSE

1) To enter and/or edit (sub)county, analysis district, strata, strata factor and frame unit information in a frame unit file
2) To create subcounties using segment network or mask file data
3) To check frame unit and digitizer area statistics.

INPUT

1) Possible user entered data (depending on command chosen):
   a) (sub)county names, analysis district names, strata and strata factor numbers and the number of frame units
   b) Landsat frame names
   c) tract names
   d) state name and year
   e) fips code
   f) crop reporting district (CRD)
2) Type 50 Frame Unit File (to edit an existing file)
3) Type 47 Segment Network File(s) (for subcounty creation and frame unit area checking)
4) Type 17 Segment Mask File(s) (for subcounty creation)

OUTPUT

1) Type 50 Frame Unit File containing input or updated information named by state unit (e.g. KS83.UNIT)
2) Terminal display of output from listing commands

HOW THE PROGRAM WORKS

The program takes user-entered parameters and places them in two data structures, one for county entries and the other for strata factor entries. The user may read data into these structures from a frame unit file disk file and then write it out after editing. The user may also optionally select the ASSIGN command with
  1) the CREATE command to create subcounties,
or
  2) the FRAME command to compare frame unit and digitizer area statistics.

COMMAND SUMMARY

(The main command menu is prompted with a '1):'. Other subcommand prompts are suffixed with a ':'. Options within a command are suffixed with a '-'.)

<u>OPEN INPUT FILE</u>
Open and read frame unit file data into the program data structures.

UPDATE

Add or change a parameter in either data structure

UPDATE OF:

COUNTIES

specify a county, strata and assign/change the number of frame units;assign county fips code and crop reporting district.

FACTORS

specify a strata and assign/change the strata factor.

ANALYSIS DISTRICTS

Update Analysis Districts By:

COUNTY

specify a county and assign its corresponding analysis district.

ANALYSIS DISTRICT

specify an analysis district and then assign the counties that correspond to it.

RENAME

Change the name of a county or analysis district.

RENAME OF:

COUNTIES

enter the name of the county to be changed followed by the new name of the county.

ANALYSIS DISTRICTS

enter the name of the analysis district to be changed followed by the new name of the analysis district. All counties with the old analysis district name will be changed.

DELETE

Remove a county or analysis district from the county data structure.

DELETE OF:

COUNTIES

enter the name of the county to be removed.

ANALYSIS DISTRICTS

enter the name of the analysis district to be removed from all county entries.

CREATE SUBCOUNTIES USING SEGMENT NETWORK OR MASK FILES

Make new subcounty entries in the data structure by dividing county frame unit data among areas delineated by 2 or more Landsat scenes.

ASSIGN TRACTS TO STRATA

Set up a data structure showing the tracts which correspond to certain strata. Enter strata assignments specify a tract and its corresponding strata Display current assignments- list on the terminal the current setup of tracts and strata. Clear assignments and start over- remove all assignments that have been entered.Use standard assignment- automatically enter a standard set of tracts and strata.Quit- exit from the tract to strata assignment command.

FRAME UNIT AND DIGITIZER STRATA CHECK

Compare areal statistics by strata existing in the frame unit data structure with digitized area numbers. Assign county fips code and crop reporting district.

CHECK CONSISTENCY OF SUBCOUNTIES

Check to see if 1) there are strata in the subcounty entries that do not exist in the main county entry and 2) the number of frame units of any subcounty entry exceeds the number of frame units in the main county entry.

LIST

List selected portions of the county and/or strata factor data structure.

LIST BY:

ALL

list by all of the following options.

COUNTY

specify a county and its strata and corresponding number of frame units will be listed.

STRATA FACTOR

all strata in the strata factor data structure and their corresponding strata factors will be listed.

FRAMEUNITS

the number of frame units for each strata in each analysis district along with the total frame units for each strata in the entire file are listed.

ANALYSIS DISTRICT

specify an analysis district and all of the counties corresponding to it are listed.

UNASSIGNED (TO ANALYSIS DISTRICTS) COUNTIES

a list of those counties which have not been assigned to an analysis district is created.

INITIALIZE

Remove all entries from the program's data structures.

WRITE OUTPUT FILE

Output the current data structures to a disk file.

USE BACKUP FILE FOR LISTING

Output the listings from the LIST command to a text file.

QUIT

Exit from the entire program.

HOW TO USE IT

1) The major use of EDUNIT is to enter and/or edit frame unit file information,(sub)county, analysis district, strata, strata factor, and number of frame units. Of the commands available to the user upon entering EDUNIT, UPDATE,RENAME, and DELETE facilitate frame unit file editing. The UPDATE command may be used to start a new body of frame unit data "from scratch". Or, if frame unit data is already existent on disk, the OPEN INPUT FILE command will ask the user for the name of a frame unit file which is made available for editing. The status of the frame unit data during the editing process may be checked by using the CHECK CONSISTENCY OF SUBCOUNTIES command (see command list) and also by invoking one or several subcommands of the command LIST. LIST allows the contents of the frame unit data to be displayed by county, analysis district, frame unit, unassigned to analysis district counties, strata factor, and all of the above. Once editing of the program's data structures is complete, the WRITE OUTPUT FILE command transfers the results to a user- specified file.

2) Another use of EDUNIT is the creation of subcounty frame unit data. After the information on the number of frame units by strata for the main county is entered using the entering/editing commands, the ASSIGN command (see below) must be invoked to assign all of the tracts of the entire county (as listed in the segment network or mask file) to the strata (as listed in the frame unit data).

The CREATE command begins by asking if there is a master mask file for all frames or if each frame is to be entered separately. The file name of the master mask or the frames and file names for each frame's segment network or mask file are then requested. The proportion of the whole county's acreage by strata per frame is calculated and multiplied by county entry's frame unit total to result in a frame unit count for each subcounty. This data is inserted into the program's county data structure and may be viewed using LIST.

3) A third use of EDUNIT is the checking of acreage statistics by strata between the digitized data and the calculated frame unit information. Acreage data by tract is required from a county segment network file whose name is generated or requested. After invoking the tract-assignment command, ASSIGN (see below), this data is transformed into frame units per strata and then compared with the frame unit data for the subcounty in the county data structure. A tolerance for deviation (default is 5%)is selected at the program beginning along with the amount ofcomparison information to be listed.

4) The ASSIGN command (used both for subcounty creation and for frame unit and digitizer area statistics checking) provides a means for converting data that is listed by tract and expressing it in terms of strata. A set of subcommands mainly allows the user to "enter tract and strata assignments" by specifying a tract name and then its corresponding strata number. Other subcommands permit
1) the allocation of a default tract-strata assignment,
2) the display of current assignments,
3) the initialization (removal) of all assignments.

NOTES/LIMITATIONS

The EDITOR module, FRDCK, Frame and Digitizer Strata Check, has been incorporated as a command into EDUNIT. The library functions to allow the automatic generation of segment network file names and to search other directories are machine dependent routines that have not been implemented as yet.

INTRODUCTION

Large-scale estimation is used to compute the estimate, standard deviation, C.V. (coefficient of variation), and possibly other information by strata for an analysis district. Large-scale estimation incorporates the classification and aggregation by strata of complete Landsat scenes with the parameters generated by sample estimation to compute estimates. Also, large-scale estimation will optionally create an estimator results file which may be used with the accumulate estimates program.
There is basically one path through large-scale estimation, but with some variation requiring choices by the user. That path will be described.

Directories
The user is asked for the list of directories. Any directories entered will be used to search for all input files for which the names are generated by the program and also for those for which the user supplies the name unless the user also explicitly supplies a directory name. All output files have names supplied by the user and are written to the connected directory unless the user also explicitly supplies a directory name.

Estimator Parameter File
The user is required to enter the name of the estimator parameter file. The estimator parameter file is a required input to large-scale estimation. Various information from the estimator parameter file will be displayed, including the state and year, the type of estimation (single-variable regression or multi-variable regression), the units (acres or hectares), the dependent and independent variables and the covers associated with each, and the list of counties.
For single-variable regression, the user may choose to use only a subset of the counties from sample estimation. In that case, the names of the counties to be used must be entered one per line and terminated with carriage return only. Only county names from among those displayed may be entered. The list of counties entered is displayed and the user asked if it is correct. For multi-variable regression, it is not possible to use a subset of counties.
The frame unit file is read. The list of strata is displayed. For each strata, the number of frame units is displayed. If combined estimation is being used, as specified by the estimator parameter file, the group to which each strata belongs is displayed.
If the group number is negative, the strata actually belongs to a group, otherwise the group number will be the same as the strata number. If only a subset of counties is being used, the total frame units will also be displayed, the total being the number of frame units for all counties from the estimator parameter file.

Aggregation File
The aggregation file is a table of the number of pixels by category and strata. Large-scale estimation requires at least one aggregation file, but will accept more than one and add the values.
The strata are actually stored in the aggregation as tracts since the same programs are used for digitizing strata boundaries as for digitizing field boundaries of segments. Therefore, it is necessary to assign tracts to strata. The program provides a standard assignment which will be displayed for all strata read from the estimator parameter file. This standard assignment for strata(tract) is:
11(A), 12(B),20(C), 31(D), 32(E), 33(F), 40(G), 50(H), 61(I), 61(J).
If the standard assignment is not suitable, the user may enter an assignment. For each strata, the program asks for a list of tracts. Each tract is assumed to be a single letter. The tracts for a particular strata are entered on a single line, separated by commas.

A particular tract may be assigned to at most one strata. Not all tracts need be assigned since some may represent strata not being estimated. Once the assignment is entered, it is displayed and the user is asked if it is correct.
The categories to be used for each independent variable must be selected. This may be done either by using a statistics file or by the user actually entering the categories. Note that for multi-variable regression the same method of category selection must be used for all independent variables.
If a statistics file is used, the categories assigned to each independent variable will be those for which a cover matches one of the covers associated with the variable. The list of categories assigned to each independent variable will be displayed.
If the user enters the list of categories, the list for each independent variable is a list of category elements separated by commas. Each category element is a single integer category number or a range of categories. A range of categories is two category numbers separated by a dash ("-") with the first less than or equal to the second. A range of categories indicates that all categories from the first to the second inclusive wil be used. Normally, the list should fit on a single line. If not, the line must end with a comma to indicate that the list will continue on the next line.

Displaying Results
If a subset of counties is being used, the regression terms (TERM 1, TERM 2, TERM 3) are displayed for each strata. In the case of combined estimation, TERM 1 and TERM 2 are displayed for each strata assigned to a group and TERM 3 is displayed for the group. All three terms are displayed for strata not asigned to any group.
In all cases, the means are displayed for each strata. The regression and sample means are displayed for the dependent variable and the sample and population means are displayed for each independent variable. For combined estimation, the means are displayed for each group and unassigned strata. In addition, if all counties from sample estimation are being used, the sample means are displayed for the dependent and independent variables for all strata assigned to a group.
Finally, the estimate, standard deviation, and C.V. are displayed for each strata or group. For combined estimation, the estimate, standard deviation, and C.V. are also displayed for each strata assigned to a group, but only if all counties from sample estimation are used. Otherwise, only the values for the group are displayed. The values for strata not assigned to a group are, of course, always displayed. For single-variable regression, the estimate and standard deviation are displayed in both acres and hectares, with -A indicating acres and -H indicating hectares.
If there is more than one strata or group, total estimate, standard deviation, and C.V. are also displayed.

Estimator Results File
The estimator results file is an optional output file from large-scale estimation. It is an input file to the accumulate estimates program which is used to accumulate estimates from the various analysis districts.

Files
Required input files for large-scale estimation are the estimator parameter file, the frame unit file, and one or more aggregation files.
A statistics file is an optional input file, used to get the list of categories for computing pixel totals from the aggregation file for the independent variables.
The only output file from large-scale estimation is the optional estimator results file used as an input to the accumulate estimates program.

INTRODUCTION

The sample estimation program uses segment data to generate parameters for large-scale estimation and to display some results. There is basically one path through the program, but with some variation depending on the type of estimation selected. The remainder of this paper will show that path.

Preliminaries
The state and year identifier is requested. This is the two-letter state postal abbreviation followed by the two low-order digits of the year. While the state and year is optional, it should generally be entered since the state and year identifier is used to find the segment catalog and frame unit files, both of which are required for sample estimation.
The list of directories is requested. The directories are entered one per line, terminated by carriage return only. Any directories entered will be used to search for all input files for which names are generated by the program and also for those with names entered by the user, unless the user explicitly supplies a directory name. All output files have names supplied by the user and are written to the connected directory unless the user supplies a directory name.
The list of frames is requested. The frames are entered one per line and terminated by a line with carriage return only. While the list of frames is optional, it will generally be required to handle subcounties. A subcounty is a portion of a county lying in a particular frame and recorded in the frame unit file with the number of frame units reflecting the portion of the county in the particular frame. When building the county list, the only subcounties used are those corresponding to a frame entered in the frame list.

Select Region
SELECT REGION is requested. This is the standard SELECT REGION statement used to create a list of segments and is described in more detail in the paper on the pack program. The list of segments generated is used in two ways, to read variables from the totals and table files, and to get the first list of counties.

Building the List of Counties
The list of counties is initially built by using the segment catalog file to get the county for each segment selected by the REGION statement. The user is given an opportunity to add more counties. Then, for all counties so far selected, the frame unit file is used to build a list of analysis districts containing those counties. The list of analysis districts is displayed and is used to find additional counties in the frame unit file. Finally, for all counties, if a subcounty exists for one or more of the frames selected and if the main county has been selected, the main county entry is replaced by the subcounty entries for all applicable frames. The final list of counties and subcounties is displayed.

Building the List of Strata
The list of strata is built by using the segment catalog file to get the strata for each segment selected by the SELECT REGION statement. The list of strata, number of frame units, and number of segments in each strata is displayed. The number of frame units is obtained from the frame unit file using the list of counties previously constructed.

Selecting the Estimator and Variables
The type of estimator is selected. It is either single-variable regression or multi-variable regression. The dependent variable is assumed to be from the totals file. The totals file name is requested and the file is opened. The options in the totals file are displayed, including the size

type, the units and the SELECT OPTIONS statement used. Each option is assigned an integer value and the user selects the value corresponding to the desired option. The list of covers for the dependent variable is taken from the totals file. The independent variable(s) are selected. In the case of multi-variable regression, the user must enter the number of independent variables. If only one independent variable is specified, single-variable regression is assumed. The independent variables will generally be pixels, but may be taken from the totals file. Unless the independent variables are all pixels, it is not possible to create an estimator parameter file for use with large-scale estimation. If the totals file is used for an independent variable, the selection is as for the dependent variable. If pixels is selected, the first independent variable is assumed to have the same covers as the dependent variable.

For multi-variable regression, the user must enter the cover list for each succeeding independent variable using pixels. Each cover is entered on a separate line and the list is terminated with carriage return only.

The number of pixels to assign to each segment for an independent variable using pixels is obtained from a table file strictly on the basis of a category list and ignoring the cover information in the table file. The list of categories may be obtained from a statistics file or may be entered by the user. If the list of categories is obtained from a statistics file, the same statistics file will be used for all independent variables using pixels and the categories used will be those with a cover matching a cover associated with the particular independent variable.

If the category list is entered by the user, the user enters a list of category elements separated by commas. Each category element may be a single integer category number or a range of categories. A range of categories is two category numbers separated by a dash ("-") with the first number less than or equal to the second. A range of categories indicates that all categories from the first to the second inclusive will be used. Terminating a line with a comma indicates that more categories will be entered on another line. Otherwise, the program assumes that all required categories have been entered.

The table file is read once and the number of pixels is computed for each segment for each independent variable using pixels. If a segment does not appear in the table file, a warning message is given and the number of pixels is set to zero for that segment for all independent variables using pixels.

Displaying and Deleting Strata

The list of strata is displayed. For each strata, the strata number, number of segments, number of frame units, and the total and mean value for each variable are displayed. The mean value is the total divided by the number of strata in the segment. The user is asked if all strata with fewer than five segments should be automatically deleted. Regardless of the reply, the user is also asked if any other strata are to be deleted. If so, the user must enter the list of strata numbers to be deleted separated by commas.

For single-variable regression, the sum of segments in all remaining strata must be at least three. If this is not the case, all strata except those with fewer than five segments which were automatically deleted will be undeleted and the user will be asked to delete strata again.

For multi-variable regression, each remaining strata must have at least three segments. If this is not the case, all strata except those with fewer than five segments which were automatically deleted will be undeleted and the user will be asked to delete strata again.

The reason for the difference in handling single-variable regression and multi-variable regression is that grouping (combined estimation) is allowed for single-variable regression but not for multi-variable regression.

Grouping, Combined Estimation

For single-variable regression only, use of combined estimation is allowed to group strata. If any strata have fewer than three segments, combined estimation is required.

The groups are assigned negative numbers by the program, starting at -1. For each group, the user enters a list of strata numbers separated by commas. Any particular strata may be assigned to at most one group. Not all strata need be assigned to a group. However, for the grouping to be acceptable, each group or unassigned strata must have at least three segments. Before the grouping is finally accepted, it is displayed and the user is asked if it is correct.

Segment Totals File

For single-variable regression only, a segment totals file may be created. The file is an ASCII file listing, for each segment, the segment number, strata to which the segment belongs, the reported size (from the totals file) and the number of pixels. The segment totals file is needed for the EDITOR program used to process segment totals files. When this program is rewriten for MARS-PED, the segment totals file will be eliminated. Therefore, the segment totals file is not pertinent to any system except BBN.

Displaying Results

For single-variable estimation, the R-SQUARE, B[0], B[1], standard deviation, and C.V. (coefficient of variation) are displayed for each group and unassigned strata. Values are displayed in both acres and hectares, with -A indicating acres and -H indicating hectares. R-SQUARE and C.V. are the same for both acres and hectares, of course. If there are more than one group or unassigned strata, the total standard deviation and C.V. are displayed.

For multi-variable regression, the R-SQUARE, standard error, and C.V. are displayed, the latter two for Y-MEAN and Y-MEAN[MR]. The regression coefficients, B, are displayed for the dependent and independent variables. The sample correlation coefficients are shown between the independent variables. If the absolute value of a correlation coefficient is greater than 0.25, a warning message about possible bias between the independent variables is displayed. Finally, an optional residual display is provided. If this display is selected, the actual value, computed value, and residual are displayed for each segment. The residual is the actual value minus the computed value. For values where units are pertinent, the units used is either acres or hectares, as selected by the user when the program is first invoked.

Estimator parameter file

After the displays, the estimator parameter file may optionally be created. The estimator parameter file may only be created if all independent variables are pixels. If some are not, the program will not even ask about the estimator parameter file. The estimator parameter file is a required input for large-scale estimation.

Continuing Estimation

Finally, the user may do other estimates or quit. If other estimates are to be done, the program returns to step 6., Selecting the Estimator and Variables, but with the same type of estimator as previously selected.

Input files

The input files required by sample estimation are a segment catalog file, a frame unit file, and a totals file. If pixels is used for any of the independent variables, the usual case, a table file is required. A statistics file is required if the list of categories to be used to get pixel totals is to be generated rather than input by the user.

Output files

All output files from sample estimation are optional. The output files which may be created are the estimator parameter file and the segment totals file.

EXTENT	Determine segment window EXTENT

PURPOSE

Determine image file window extents of a group of digitized segments, and optionally produce a segment region file and a window text file of all segments that are wholly contained in the image file.

INPUT

A global calibration file name.
A list of Segment numbers.
User entered information and values.

OUTPUT

Summary listing on the console of each segment, with those segments not contained in the image file flagged as outside.
A Segment Region Text File (optional).
A Segment Window Text File (optional).

HOW IT WORKS

The routine requests the name of a Global Calibration File and information regarding the associated image file, and a list of segment numbers. There are several options that can be specified.

a. Output a region file containing the segment numbers of all segments that are wholly contained in the image file.
b. Output a window text file containing the row/column extents and the segment number for each contained segment.
c. If a window text file is requested, a border of specified size can be placed around the segment extents.

Once all options have been selected, the routine reads the Global Calibration File, then for each segment it will read the associated Segment Network File (type 47), convert all vertices in the Seg. Net. File to row/column values, and determine the extent of the rectangular window that surrounds the segment. The information about the segment, including its window, will be displayed on the console and, optionally, all contained segments will be entered in the region file and in the window text file. The windows written to the window text file will be expanded by the border, if any.

COMMAND SUMMARY

There are no commands as such. The routine asks for all the needed information from the user in a 'straight-line' mode.

HOW TO USE IT

When the routine is entered, it immediately begins asking for information. The sequence of questions goes:

1. Determine the scene format, edips or not. If not edips the user will be asked which Landsat mission produced the image.
2. Input the global calibration file name.
3. For edips images, input the fill pixel counts.
4. The routine asks for a list of segment numbers to be checked, then asks if there will be an output region file. If yes it will also get the output filename.
5. The routine asks if a output window file is desired. If yes it will ask for the filename, then ask if a border is wanted.

Once all this information is gathered the processing begins. Infor- mation about each segment is printed on the terminal, and output to the optional files if they have been specified. Segments that are not wholly contained within the image file are flagged with the message '<--<<OUTSIDE||' on the terminal, and not put into either of the two possible output files.

NOTES/LIMITATIONS

The routine does not accept a segment list from a segment region file. This option will be added at a lated time.

There are limits to the size of the calibration file and the segment network file. Currently the maximum number of coefficients allowed in a calibration file is 16. The limits for a segment network file are 200 fields, 600 edges, and 600 vertices. The routine can accept up to 300 segment numbers.

Due to a shortage of calibration files and segment network files, the routine has been tested in only two cases, both examples with only a single seg. net. file for a segment that was not contained in the associated image file. Testing on a data set that includes multiple segments for a single calibration file, including at least some segments that are valid, must still be done.

GROUP	reGROUP categories in a file

PURPOSE

To allow the re-grouping and/or re-ordering of categories specified by the user into new categories and write them to a different file

INPUT

1) Categorized window file (type 2, 3, or 51)
2) The category numbers from the input categorized window file that are to be combined.

OUTPUT

Categorized window file (of the same file type as the input) with re-grouped and/or re-ordered categories

HOW IT WORKS

The program reads a categorized window file and prints out the number of categories, number of windows, and number of pixels. It then asks for the number of categories that are to be in the output categorized window file and proceeds to ask for the input categories that are to be combined for each of the output categories. The program then remaps the categories to their new configuration, asks for an output categorized window file name and writes out the new file.

COMMAND SUMMARY

GROUP is not a menu driven program, but it keeps prompting the user for more inputs. Since the program is small and the prompts are understandable, no sample program script will be provided.

HOW TO USE IT

GROUP is a very straightforward program with unambiguous prompts. No explanations are necessary.

NOTES/LIMITATIONS

The maximum number of categories is 256 (Category numbers 0 to 255).

GTRUTH	Ground TRUTH editor

INTRODUCTION

The ground truth editor is used to create, update, check, and list ground truth files. Briefly, a ground truth file represents a portion of the information collected by USDA enumerators about a segment. There is one ground truth file for each segment. Each ground truth file is made up of a number of fields. Each field is designated by a tract (a one or two-letter identifier) followed by a field number. The fields are composed of at most four uses, where each use represents the status of the field at some particular date. Within each use, information is stored for cover, field size, planted size, harvested size, other size, intended use, note code, and irrigation. The sizes are in tenths of a unit, usually acres. The remainig items are referred to by names, e.g. "corn" as an example of cover.

The program has a fairly complex structure with several levels of commands and some commands such as checking and ground truth file creation requiring a number of parameters. The main level commands will be described briefly followed by a more thorough explanation of the various functions and parameters involved.

Several of the commands request SELECT REGION. Such a region generates a list of segments. It may contain the operands SEGMENT, FRAME, COUNTY, ANALYSIS_DISTRICT, STRATA, and ALL. All operands except SEGMENT require that the segment catalog file be present. All operands except ALL take parameters. More than one parameter may be used in which case they are enclosed in parentheses and separated by commas. The effect is an OR of the parameters, to include segments satisfying one or more of the parameters. Boolean operators AND, OR, and NOT may be applied to the operands. Parentheses may be used to any level for grouping. Abbreviations may be used for operands and for nonumeric parameters. The SELECT REGION statement may extend over several lines and is terminated with "#". Some examples of SELECT REGION statements are:
SEG(1023,5111,6184)#

COUNTY CLAY AND STRATA(11,12)#

When the name of some element such as cover is entered, it should not contain spaces, but may contain "_" (underline) characters. Thus, winter wheat may be entered as WINTERWHEAT or WINTER_WHEAT but not WINTER WHEAT.

Main Commads
<u>OPEN INPUT FILE</u> Opens an input ground truth file and reads the contents. The file may then be updated, checked, or listed as specified by subcommands.

<u>LIST MANY SEGMENTS</u> The ground truth files for all segments in the region selected are listed.

<u>DOUBLE COVER DISPLAY</u> A listing is made for all segments in the region selected of all the instances in which different covers appear in two selected uses. The listing is by use pair and segment.

<u>PARAMETERS FOR CHECKING</u> The user is requested to enter various options and parameters for checking. The types of checking are internal checking within a ground truth file

and external checking against mask or segment network files. The options and parameters will be described in the section on checking.

CHECK MANY SEGMENTS Performs a check of segments in the region selected. If checking options and parameters have not been specified, the user is asked for them.

USE DELETION FOR MANY SEGMENTS Deletes a specified use for all segments in the region selected. The updated ground truth files are written out.

CREATE GROUND TRUTH FILES Ground truth files are created by reading in data from a card-image tape or disk file (the "card images" may be longer than the standard 80-character card). The input file would typically have been prepared externally, perhaps using SAS or similar software. Checking is done on each file as it is created. This command is described more fully in the section on creation of ground truth files.

INITIALIZE TO AN EMPTY GROUND TRUTH FILE Creates a shell of a ground truth file which has no fields. The fields may then be filled in interactively using update commands.

SUMMARIZE FIELD INFORMATION The user enters a size type and a use number. The number of fields and total size are shown for each cover for the type of size and use number selected. The user may optionally enter a note code. If a note code is entered,an additional summary is shown only for fields with that note code.

QUIT Exit the ground truth editor.

Creating Ground Truth Files

The ground truth files are created from a card-image tape or disk file. The tape may be in ASCII or EBCDIC, but the disk file may only be in ASCII. Each record must contain all required data for one field for all uses. An input tape may be blocked. The record length may be longer than 80 characters. The user is queried for the number of uses which will apply to all data read.

The input record is described by a FORTRAN-like format statement and a list of elements. The format statement allows the I, F, A, and X editing phrases only. For the F editing phrase, the form is Fw.d where the ".d" is optional. If present and there is no "." in the data, an implied decimal point is assumed. On the F and I editing phrases, if the input is all spaces except for the occurence of a single ".", the field is assumed to contain missing data and a value of zero is returned for sizes or UNKNOWN for other elements. As in FORTRAN, an integer constant may precede an editing phrase, or a group of editing phrases enclosed in parentheses, to indicate repetition.

The list of elements consists of some of STATE, SEGMENT, TRACT, FNUM (field number), COVER, FIELD SIZE, PLANTED SIZE, HARVESTED SIZE,OTHER SIZE, INTENDED USE, NOTE, and IRRIGATION. The list of elements is entered with the elements separated by commas and is terminated by "#". The elements SEGMENT, TRACT, FNUM, COVER, and FIELD SIZE must always be present. STATE is only used if a portion of the input is to be read based on the state number. PLANTED SIZE is set to FIELD SIZE if not present. HARVESTED SIZE is set to PLANTED SIZE if not present. OTHER SIZE is set to FIELD SIZE minus PLANTED SIZE if not present. INTENDED USE, NOTE, and IRRIGATION are set to UNKNOWN if not present.

There are two ways to use only a part of the input data to create ground truth files. The first is to select a certain state number. Of course, this assumes STATE is in the element list.The second is to select a segment number such that only segments with numbers greater than that selected will be used. Both methods may be used on a single run. If neither is selected, the entire input data set will be used.

Since COVER, INTENDED USE, NOTE, and IRRIGATION are entered as integers, a list of the names corresponding to the integer values must be entered. Each list consists of one entry per line with the entry being the name followed by one or more spaces followed by the integer value. An example is

CORN 23

The list is terminated by an empty line, carriage return only. The same name may be assigned to several numbers, but each number may be assigned to only one name. The user will be asked for the codes for all appropriate elements entered in the list of elements.

Checking Ground Truth Files

Ground truth files may be checked internally and externally. The internal checks look for consistency within the ground truth file and the external checks compare the ground truth file to the mask or segment network file for the same segment. When checking is called for, certain internal checks are always performed while others are optional. The external checks are always optional. The optional checks and associated parameters are specified as checking options and parameters by the user.

The mandatory checks are that FIELD SIZE is non-zero, PLANTED SIZE+OTHER SIZE=FIELD SIZE, and HARVESTED SIZE< =PLANTED SIZE.

Optional checks are the external checks, the difference check between planted and field size, the difference check between harvested and field size, the difference check between planted and harvested size, and the consistency check among uses.

The external check first finds any fields which appear in the mask or segment network file but not the ground truth file and vice-versa.

For the fields in both files, a size comparison is made. There will be an error only if the absolute value of the differnece is greater than or equal to a minimum difference and the percentage is greated than or equal to some minimum percentage. The percentage is the difference divided by the size from the ground truth file (reported size) and multipled by 100. The minimum size and percentage are parameters suppllied by the user. In addition, special percentages may be input for certain notes or covers. The special percentage, if applicable, is used in place of the general percentage. Special percentage by note takes precedence over special percentage by cover. Also, the note code may be changed in case of an error. If such a note is inserted, the file will have of course been changed and will be written out when checking is completed. Finally, a particular use number may be specified for the external check. The default is to do the check for all uses.

The planted and field size check, the harvested and field size check, and the planted and harvested size check compare sizes in each field and use. If the difference is greater than a specified value and the percent difference is greater than a specified percentage, an error is reported. The percentage is the difference divided by the field size multiplied by 100. The minimum difference and percentage are input by he user. Special percentages may be input for certain notes and covers. The same special percentages are used for all of the size checks that have been selected. The special percentage, if applicable, is used in place of the general percentage for each size check. Special percentage by note takes precedence over special percentage by cover. If requested by the user, the note code will be set to some value on an error.
The use consistency check will report any fields for which the field size is different among uses.

Subcommands to OPEN INPUT FILE
Once the OPEN INPUT FILE command is used, there are several commands which may be used on the file.

LIST Lists the file on the user terminal. The listing may be for the entire file or for a specified field.

UPDATE Allows the user to make a variety of changes to the file as will be described in the section on update mode.

PARAMETERS FOR CHECKING GROUND TRUTH EDITOR Allows entry of checking options and parameters.

CHECK Checks the file as specified by the checking options and parameters.

QUIT Closes the file, first writing it out if there have been any changes.

Update Mode
Update mode allows the user to update the file in various ways according to the subcommand entered.

FIELD UPDATE The user selects a field and one or more uses (ALL is used to ndicate all uses, otherwise a list of use numbers separated by commas is entered). If the field is not found in the ground truth file, it is added and all sizes are set to zero and other elements to UNKNOWN. The user selects the element to be accessed. The current contents of that element within the field and for the uses selected is displayed. At that point, the user may enter a new value or a carriage return to indicate no change. The elements which may be accessed are COVER, FIELD SIZE, PLANTED SIZE, HARVESTED SIZE, OTHER SIZE, NOTE, and IRRIGATION. FIELD UPDATE is exited by entering a carriage return only when asked for the field name.

DELETE FIELDS Deletes fields specified. Fields to be deleted are entered one per line terminated by carriage return only.

RENAME FIELD Changes the name of a field. The new name may not be the same as that for some other field in the file.

USE DELETION Deletes a specified use, Not allowed if there is only one use in the file.

QUIT Exits update mode.

AUTOMATIC PACKING A standardized method of creating several packed files for various covers using SELECT REGION with program generated SELECT OPTION STATEMENTS based on criteria supplied by the user. This command is described more fully in the section of automatic packing.

TABULATE Generates a table file from an input packed file.

PRINT A TABLE FROM A FILE Prints a table file onto the user terminal or onto a file for later printing. If the table file was created by segment, it is possible to print only certain segments.

UNPACK Generates an unpacked file from an input packed file.

LIST OF SEGMENTS FILE CREATION Creates a list of segments into the output ASCII file. The list of segments is the list generated by the input SELECT REGION statement. The list of segments is suitable to be input as a SELECT REGION statement or to most programs requiring a list of segments.

SELECT REGION and SELECT OPTIONS

The SELECT REGION statement is used to select segments by either explicitly specifying the segment numbers or using attributes of the segments from the segment catalog file. The SELECT OPTIONS statements is used to select fields, and hence the pixels contained in those fields, by using attributes of the field from the ground truth file.

Both SELECT statements are Boolean expressions using the operators AND, OR.and NOT. Some operands have parameters and others do not. If an operand has parameters, it may have a sinlge parameter spearated from the operand by a space or a group of one or more parameters enclosed in parentheses and separated by commas. The effect of more than one parameter is OR, that is select segments or fields satisfying at least one of the parameters. All operands and non-numeric parameters may be abbreviated. Also, operands and non-numeric parameters may contain underscore ("_") anywhere except the first character to improve readability, but no embedded spaces.

A SELECT statement may extend over more than one line and is terminated by a "#". No identifiers or numbers may be split at line boundaries. Parentheses may be used at any level for grouping.

When entering a SELECT statemnt, a "?" may be used at any time to get help. In particular,if a parameter is expedted, one may obtain, where pertinent, a list of all allowed parmeters.

IDENT	IDENTify a MARS-PED file

PURPOSE

Used to determine the type of a file if the user is unsure of it. It prints out the file type code, name, and some (perhaps) other useful information.

INPUT

User must supply the name of the file to be identified and, optionally, a list of directories to be searched.

OUTPUT

Outputs text to the terminal.

HOW IT WORKS

The routine attempts to open the file named by the user and reads the first two words. If the first word is zero and the second is a legal MARS-PED file type code, a specific handling routine is called for that file type. If the first word is zero and the second is not a legal MARS-PED type, it checks to see if the file is an older EDITOR file. If the first word is not zero it checks to see if the file may be a text file. If it does look like a text file the 1st line is printed on the terminal.

COMMAND SUMMARY

There are no commands as such. IDENT has a 'straight-line' user interface.

HOW TO USE THE PROGRAM

The user is queried for info in this order:
1) Enter other directory names (if any). If files cannot be found in the current directory these will be searched.
2) Enter the file name to be identified.
3) Some file types will print some preliminary info, then give the user an opportunity to get more detailed info.
4) Repeat steps 2 & 3 until a CR only is hit in step 2, causing an exit.

NOTES/LIMITATIONS

The file types 48 (scan mask) and 52 (ground truth) have not been tested due to lack of files of those types.

LIPTAP	LIP file to TAPe (DCL COMMAND FILE)

PURPOSE

Reads an ERDAS 7.2 image file and produces a Pixel InterLeaved (PIL) tape. This tape can be read by TAPWIN to create image windows.

INPUT

Erdas 7.2 image file of a Landsat Thematic Mapper full scene, or quarter scene or any other ERDAS image file (.LAN).

OUTPUT

PIL tape of image data.

HOW IT WORKS

The image on disk will be dumped to tape.

COMMAND SUMMARY

This is a command file started with
@LIPTAP
This command file will start up the LIPTAP program.
The subcommando @TOTAPE must be entered always and will read the necessary parameters from a file
TOTAPE.DAT on the executable directory.

HOW TO USE IT

Like the TAPLIP program LIPTAP needs a new tape to be mounted with the /FOREIGN and /NOMOUNT_VERIFICATION switches.

ALL INFORMATION ON THE MOUNTED TAPE WILL BE ERASED.

The user has to enter the subcommando @TOTAPE after starting the command file.
The name of the input file and the channels to be dumped on tape will be asked.
Take care not to dump all channels of a full scene TM to tape as it does not fit on one tape. Normally the channels 2,3,4,5 of TM will be sufficient for classification.

MAKEPCP	MAKE Control point Pairs file

PURPOSE

Generates a control point pairs file of two control point files.

INPUT

1) Image control point file (.ICP) which contains the ground control points in image coordinates. (Output from XGCP).
2) Map control point file (.MCP) which contains the ground control points in map coordinates.(Output from PCDIGIT on PC).

OUTPUT

1) Control point pairs file (.PCP) (Input to REGCP).

HOW IT WORKS

The program asks for the two files and creates a pairs file where map and image control points are paired. Please take care that the sequence of points in both files are the same.

COMMAND SUMMARY

Omitted

HOW TO USE IT

One can add some text in the header of the output file. This will be displayed when it is loaded into REGCP.

STANDARD FILE NAMING:

Output is a PCP pairs file with extension .PCP.