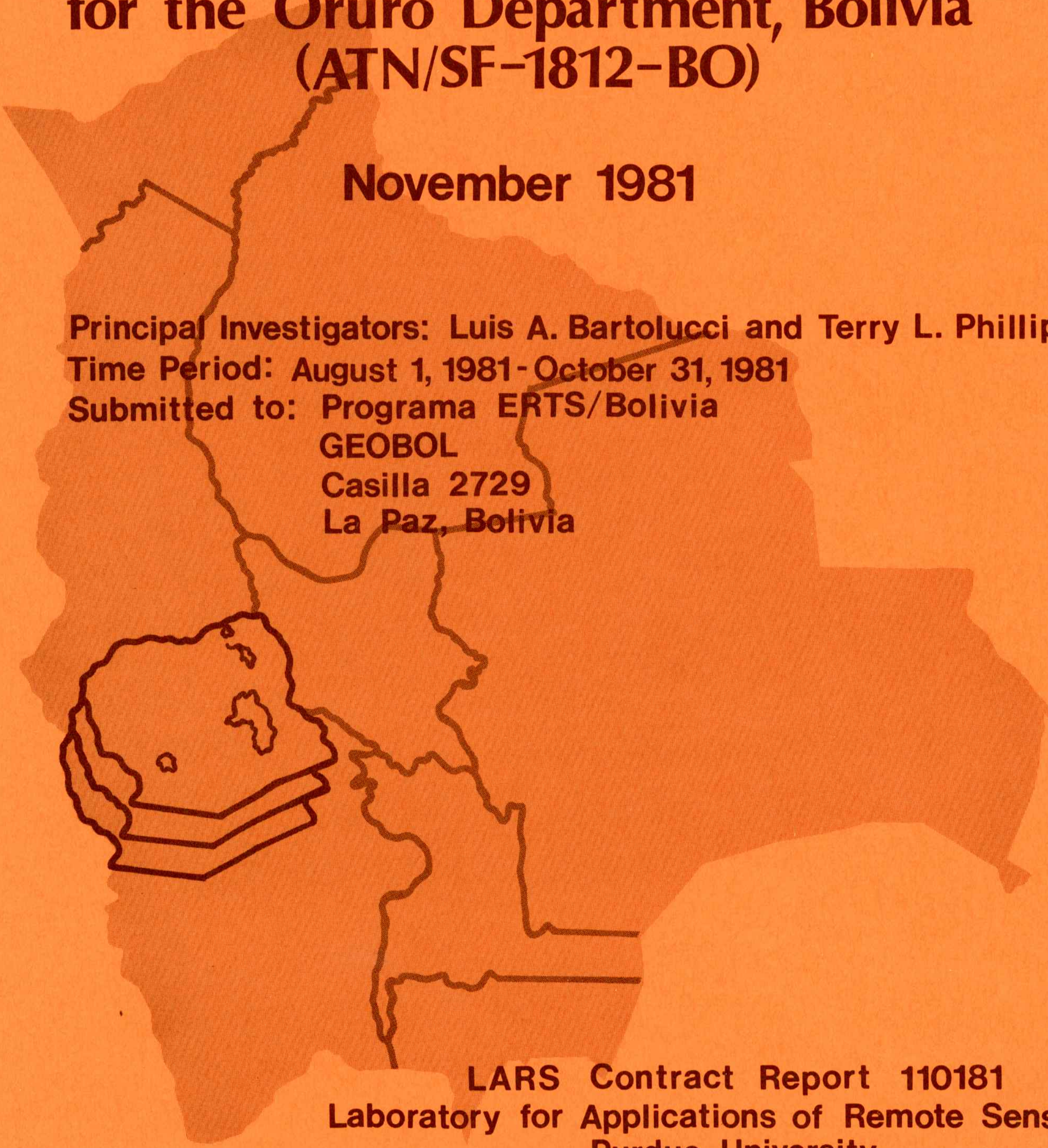
110181

Quarterly Progress Report

Digital Information System for the Oruro Department, Bolivia (ATN/SF-1812-BO)

November 1981



Principal Investigators: Luis A. Bartolucci and Terry L. Phillips

Time Period: August 1, 1981 - October 31, 1981

Submitted to: Programa ERTS/Bolivia
GEOBOL
Casilla 2729
La Paz, Bolivia

LARS Contract Report 110181
Laboratory for Applications of Remote Sensing
Purdue University
West Lafayette, Indiana 47906
USA

QUARTERLY PROGRESS REPORT

Period: August 1, 1981 - October 31, 1981

Project: Digital Information System for the
Oruro Department, Bolivia (ATN/SF-1812-BO)

Principal Investigators: Luis A. Bartolucci and
Terry L. Phillips
Purdue University/LARS
1220 Potter Drive
West Lafayette, Indiana 47906
USA

Submitted to: Programa ERTS/Bolivia
GEOBOL
Casilla 2729
La Paz, Bolivia

November 1981

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

TABLE OF CONTENTS

Digital Geographic Information System for the Oruro Department, Bolivia

INTRODUCTION

During this reporting period, i.e. August 1, 1981 - October 31, 1981, the following project tasks were accomplished:

1. Development of software for spatial resolution "aggregation" of Landsat pixels.
2. Research on the relationship between digitization cell size and two types of errors: locational and areal extent errors.
3. Quantitative assessment of the planimetric accuracy of the Landsat digital mosaic of Oruro.
4. Preparation, coding, and input of the hydrologic maps of Oruro into the digital data base.
5. Preparation and coding of the Oruro Department geologic maps for subsequent digitization and inclusion into the digital data base.

In addition to the above listed tasks, the Jet Propulsion Laboratory (JPL) is proceeding with the creation of the Landsat digital mosaic of the Oruro Department, as indicated in the enclosed monthly reports (see Appendix A) corresponding to the August-September-October time period.

SPATIAL RESOLUTION AGGREGATION ALGORITHMS

As stated in the System Design Draft section of the November 1980 quarterly progress report (LARS Contract Report 110180), the digital Geographic Information System designed for Bolivia is composed basically of a four-level-n-element grid cell data storage structure. The four levels of the system correspond to four different levels of information detail which in turn are related to four mapping scales. Table 1 shows the relationship between the information levels and a) cell size (minimum mapping unit), b) level purpose, and c) mapping scale.

Table 1. Levels of Information Detail of the Bolivian Digital Geographic Information System.

Information Level Number	Level Purpose	Cell Size (in meters)	Mapping Scale
1	Input	50x50	1:25,000
2	Local	100x100	1:50,000
3	Departmental	500x500	1:200,000-1:250,000
4	National	1000x1000	1:500,000-1:1,000,000

If data are input into the Bolivian GIS at the 50m x 50m resolution level (high level of detail), it should be possible to generalize or "aggregate" these data into the other three coarser resolution levels (lower levels of detail) through an appropriate aggregation criterion. It should be noted that this is a unidirectional process, i.e. it is possible to aggregate high resolution data into lower resolution data, but it is not possible to follow the inverse path. In other words, if the data are input at the 500m x 500m resolution level, it is not possible to derive 50m x 50m information from the 500m x 500m data set, but it is possible to aggregate the 500m x 500m data into a 1000m x 1000m data set.

Since the resource information available for the Oruro Department has been mapped at a scale of 1:250,000 and digitized at a 500m x 500m resolution level, but the Landsat digital mosaic data have been re-sampled to a 50m x 50m spatial resolution, it is necessary to aggregate the higher resolution Landsat MSS data in order to make them compatible with the rest of the digitized resource data.

To perform the aggregation of the Landsat multispectral data, four algorithms were written in MACRO-11 Assembly Language*, implemented and tested. The four algorithms used in the test were:

1. Image shrink (IMSHK)
2. Image Average (IMAVG)
3. Image Square (IMSQR)
4. Image Deviation (IMDEV)

The Image Shrink (IMSHK) algorithm selects the center pixel of a specified 50m x 50m data window (10x10 pixel block) and produces an image in which the selected center pixel represents a 500m x 500m aggregated

---

* MACRO-11 Assembly Language for a DEC PDP-11/34 processor. Documentation for MACRO-11 is available through Order No. DEC-11-OIMRA-B-D manual, Digital Equipment Corporation, Maynard, Massachusetts 01754.

pixel. A listing of the IMSHK program is included in Appendix B.

The Image Average (IMAVG) algorithm calculates the average spectral response value of a 10x10 high resolution pixel block and creates a new image in which the resulting averaged pixel represents a 500m x 500m cell. A listing of the IMAVG program is included in Appendix C.

The Image Square (IMSQR) algorithm computes a new pixel value using the following formula:

$$\text{New Pixel Value} = \sqrt{\frac{\sum_{i=1}^{n} x_i^2}{n}}$$

where,

x_i = 50m x 50m pixel value within the specified window (10x10 pixel block).

n = 100 (total number of pixels within the window).

The IMSQR subroutine is included in Appendix D.

The Image Deviation (IMDEV) algorithm calculates the standard deviation of the 10x10 pixel block and then creates a new image in which the new pixel values represent 500m x 500m cells. The listing of the IMDEV subroutine is included in Appendix E.

Visual inspection of the four resulting aggregated images suggested that the average-pixel algorithm (IMAVG) provided an image which appeared to have the least information loss. Therefore, in order to aggregate the 50m x 50m resolution Landsat MSS data into coarser (100m x 100m, 500m x

500m, and 1000m x 1000m cell sizes) resolutions, the averaged-pixels algorithm will be used.

The averaged-pixels criterion used for degrading the spatial resolution of the digital Landsat mosaic is only applicable to the spectral elements of the Bolivian GIS image-plane data base. For the other elements of the Bolivian GIS image-plane data base, different aggregation criteria should be developed and tested. For example for the soils and the land use elements neither of the four aggregation procedures described above would be applicable, instead a "voting" or "majority rule" criterion would be preferable.

Appendix F contains a series of multispectral scanner images (Thematic Mapper simulated data) of four different spatial resolutions.* The 30m, 45m, and 80m resolution images were obtained by averaging pixels of the highest resolution original data (15m data set). These examples should illustrate the effect of averaging pixel values to generate coarser spatial resolution MSS data sets.

---

* These images were obtained from "Computer-Based Forest Cover Classification Using Multispectral Scanner Data of Different Spatial Resolutions" by R.S. Latty, LARS Technical Report 052081.

RELATIONSHIP BETWEEN DIGITIZATION CELL SIZE AND MAPPING ERRORS

The inpact of digitization cell size on inventory and mapping errors in a cellular geographic information system has been investigated by Michael E. Wehde and the results have been documented by the Remote Sensing Institute, South Dakota State University.* A summary of this investigation and some of the implications related to the Bolivian GIS has been prepared by Mr. Bernard L. Spence

Summary of Wehde's Research

Wehde had two goals in mind which he hoped to accomplish from his research. He wanted to analyze the relationship between cell size and two types of error, locational error and error in areal extent. From this research these two errors could then be predicted prior to the creation of a map. Secondly, it was desired to be able to select a specific cell size for rasterization based on a cost-accuracy trade-off.

Research was performed using only one source of data which was a USDA, 1:20,000 scale soils map of Minnehaha County, South Dakota. By using a soils map, Wehde has practically a worst case mapping error. There are few other types of maps which can be imagined to be more irregular in shape or

----

* "Impact of Cell Size on Inventory and Mapping Errors in a Cellular Geographic Information System," Interim Technical Report No. SDSU-RSE-79-03, April 1979, RSI, South Dakota State University, Brookings, S.D. 57007.

having more classes displayed per unit area. Also by using a large scale map his map would display more erratic detail than a smaller scale map whose boundaries will have been smoothed. The actual test site was a region covering four square miles in area. This part of the map was selected based on the fact that it was of moderate boundary density and that there was a good mixture of sizes and shapes for the soil classes rather than just a uniform pattern. Specifically the map was broken down into 18 soil classes and 172 different soil regions over a 40.14 in area.

To create a measurable standard for his research, Wehde repeatedly rasterized his test map by decreasing the cell size until the rasterized map visually appeared to be the same as the original map. At this point the cell size was measured to be .4191 mm (.0165 in) along a side and it represented .0070 hectares (.0174 acres) on the map. This map he called his "true" map and assigned it zero percent (no) error. Having found a "true" cell size, the original map was rasterized thirteen additional times. Each iteration used a cell size which was a multiple of the "true" cell's width (i.e., a multiple of two equaled four times the "true" cell's area or .0281 ha). The specific multiples used were: 2, 3, 4, 6, 8, 12, 16, 24, 32, 48 & 64. Upon completion of each new map it was compared to the "true" map for errors.

As mentioned before, two errors were calculated. Locational error was calculated by overlaying a map of larger cell size over the "true" map and counting the number of "true" cells which were incorrectly assigned. The other error, that of areal extent is a root sum squared

$$\varepsilon = \sqrt{\sum_{i=1}^{18} e_i^{\,2}}$$

Where e_i=error of i^{th} soil type

and

$$e_i = \frac{|a_{Ti} - a_{Ai}|}{a_{Ti}}$$

where a_{Ti}=area of one soil type from the true map

a_{Ai}=area of one soil type from the larger cell size map

A plot of both of these errors can be found in Figures 1 and 2, where Figure 2 is an enlarged image of Figure 1 from the origin to the multiple of 16x.

Wehde drew the following conclusion from his research:

> "Although in the case of a single class, the behavior of locational error and error in areal extent with increasing cell size may behave very erratically, an averaging effect of many classes in a map segment is apparently a well behaved non-decreasing function."

Other conclusions that can be drawn from Wehde's research are numerous. Some of these other conclusions Wehde alludes to or comments upon, but does not elaborate upon.

1. Mapping error (locational error) is initially restricted to the border areas between two classes, but spreads and increases in size toward each region's interior as the cell size increases. This becomes obvious, for as the cell size increases the cells start to cover both the border region and more and more of the interior of any one separate region.

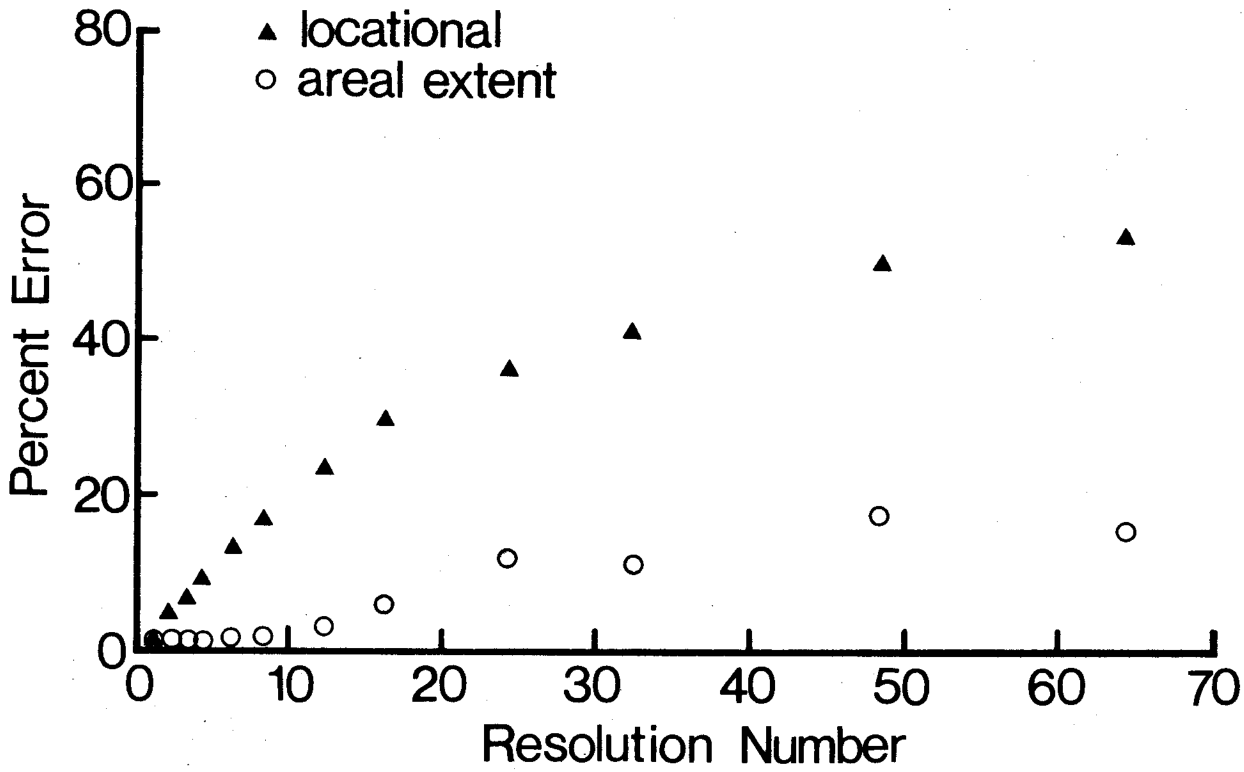


Figure 1. Mapping and inventory errors versus resolution number, the unit resolution according to Wehde, 1979 in 0.4191 mm.

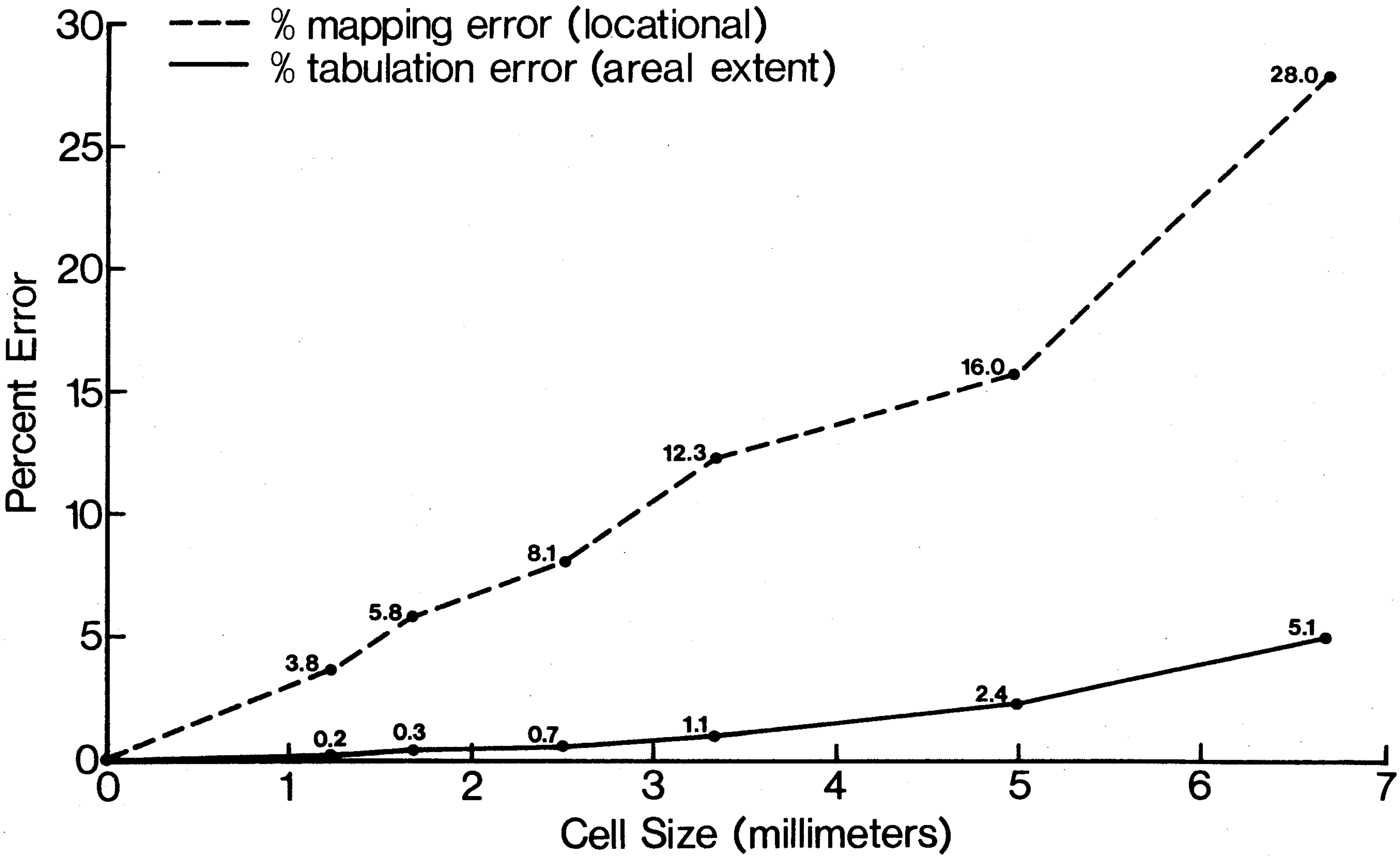


Figure 2. Relationship between mapping errors (in %) and digitization cell size.

2. As the summation of all perimeter lengths increase, error will increase. This occurs because with a longer total perimeter more cells will undergo an assignment decision based on two or more classes which that border cell encompasses. As one can see, those areas malassigned within a border cell will be increased proportionately to the number of cells undergoing a classification decision, hence the total malassignment will increase.

3. At an unknown, yet recognizable point the digitization process becomes the major source of error, if one is already using reliable equipment and trained analysts. If untrained analysts are employed they will induce the majority of error because they are unable to accurately trace the printed boundaries. Even with trained analysts 1mm digitizing errors can easily occur while tracing boundaries of the same width. For example by using a .5 mm cell size and a 1:250,000 scale map with a 1 mm digitization error will cause a 3.125 hectare error for every 1 mm along the boundary which has been digitized.

4. Classes represented by small regions will have a larger error in areal extent than classes with large regions. This is based upon the fact that the border cells of the smaller regional class make up a larger proportion of the entire class. If the border cells are assigned to the class (i.e., digitized the class to be area right) this will dramatically increase each small region's areal extent. If the border cells of a small regional class are assigned to outside of that class (i.g., area left) then there is a dramatic decrease in areal extent.

5. Lastly, if the borders are parallel to the sides of the cell grid pattern there will be an averaging effect upon the error, decreasing it somewhat as compared to a set of irregularly shaped borders. If the borders are irregular and diagonal, there is a greater probability of larger malassignment at each border grid cell.

LARS Input Subsystem

Working with the cell sizes for the different scales which are presently being used on the LARS system and the graph found in Figure 2, errors for each major map scale were determined and are listed in Table 2. This table shows that the LARS Input Subsystem locational error is 1.5% and the areal extent error is .1% for all major map scales except for the 1:25,000 and the 1:250,000 scale maps, whose errors are 6.8% and .4% respectively.

Table 2. LARS Input Subsystem Errors.

	Cell Size		Error(%)	
Scale	Ground(m)	Map(mm)	Locational	Areal Extent
1:25,000	50	2	6.8	0.4
1:50,000	50	1	1.5	0.1
1:100,000	100	1	1.5	0.1
1:250,000	500	2	6.8	0.4
1:500,000	500	1	1.5	0.1
1:1,000,000	1000	1	1.5	0.1

The Bolivian geographic information system developed at LARS will be primarily used for information concerning areal extent. For the purposes of the Bolivian geographic information system the selected cell sizes are all within 0.5% predicted error in area extent. Variations in the predicted error will be influenced by the five factors discussed earlier.

QUANTITATIVE PLANIMETRIC ACCURACY ASSESSMENT OF THE DIGITAL LANDSAT MOSAIC OF ORURO

Background

As indicated in JPL's Monthly Technical Status Report for the month of August, 1981, Ref. 384/RGM:gs-345 (see Appendix A), the Landsat green (0.5-0.6μm) spectral band of the digital mosaic for the Oruro Department was completed and sent to LARS on August 20, 1981 for LARS personnel to conduct a quantitative evaluation of the planimetric accuracy of the mosaic. JPL sent the digital mosaic in three computer tapes in VICAR format and also a set of photographic prints corresponding to each one of the 16 quadrants. The VICAR formatted tapes were reformatted to the LARSYS MIST (Multispectral Image Storage Tape) format.

A detailed description of the procedures followed for the selection of the control points from the seven Landsat frames and from the seven 1:250,000 topographic maps covering the Oruro Department for the creation of the digital mosaic was included in the February 1981 Quarterly Progress Report (LARS Contract Report 020181). The same procedures were followed to obtain the check points from the Landsat digital mosaic and from the 1:50,000 topographic maps that were used for the quantitative evaluation of the planimetric accuracy of the mosaic. Note that the control points used for the creation of the mosaic were obtained from 1:250,000 scale topographic maps because at that time they were the only available cartographic maps covering the Oruro Department. The 1:50,000 topographic maps were received at LARS much later, on August 17, 1981. The ERTS/GEOBOL Program sent to LARS 76 topographic maps at a scale of 1:50,000, which cover approximately 58% of the Oruro Department, that is, the Oruro Department is covered by approximately 130 topographic maps at a scale of 1:50,000. Figure 3 illustrates the distribution of the 1:250,000 and 1:50,000 scale topographic maps that cover the Oruro Department. The shaded 1:50,000 scale maps represent the 76 available maps.

Objective

The main objective of this task was to statistically measure the planimetric accuracy of the Landsat digital mosaic of the Oruro Department. In other words, the purpose was to determine the positional accuracy of selected points (features) on the mosaic with respect to corresponding points (features) on the 1:50,000 scale topographic maps.

Methodology

In order to accomplish this objective, the first step involved the estimation of the appropriate number of 1:50,000 topographic maps and the number of points (samples) within each map that are needed to obtain a statistically valid measure of the mosaic planimetric accuracy. The measured variable being the deviation (in meters) between the position of a specific point on the 1:50,000 maps and its corresponding position on the mosaic.

Since the most accurate measure of position (in this particular situation) is 50m, the variability of the accuracy estimate was chosen to

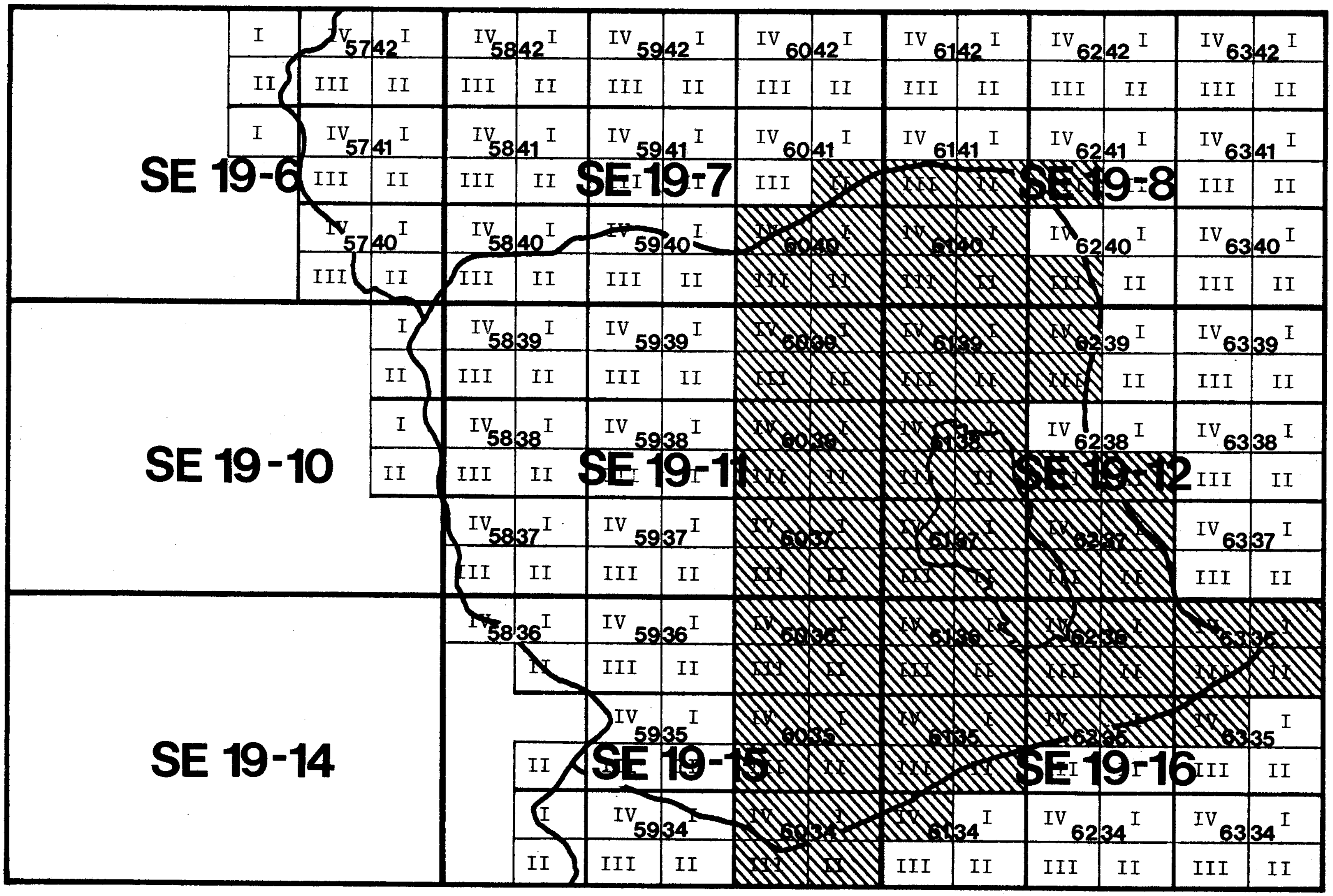


Figure 3. Distribution of 1:250,000 and 1:50,000 scale topographic maps that cover the Oruro Department. The shaded 1:50,000 scale maps correspond to those received at LARS.

be within 50m, and it was anticipated that the worst case observed would be a 400m deviation. Assuming that the 400m worst case corresponds to approximately 3 standard deviations away from the mean, then this provides an estimate of the population standard deviation s = 150m (inflated estimate). Also if it is assumed that the observations of a simple random sample is normally (gaussianly) distributed, then the number of samples "n" required would be estimated as follows:

$$\eta = \left(\frac{ts}{d}\right)^2$$

where t = is the appropriate percentage point of the student's "t" distribution
s^2 = is the population variance
d = is the allowable error of estimation

Therefore, at a 5% level of significance and assuming a large sample size,

$$\eta = \left(\frac{t\infty,\ \alpha/2\ s}{d}\right)^2 = \left(\frac{1.96 \times 150}{50}\right)^2 = 35$$

This sample size applies to the entire mosaic, and one would be 95% certain the 35 is a sufficient number of samples if the following assumptions are satisfied:

1. The population variance does not exceed s = 150m.
2. The accuracy of the checkpoints can be considered to be representative of the accuracy of all points on the mosaic.
3. The accuracy of the mosaic is the same for the areas having maps as those areas not having maps.
4. The accuracy of the mosaic is expected to be the same for each of the maps.
5. The population variance is relatively constant throughout the region.

6. The analyst does not bias the accuracy in his selection of samples.

- Assumptions 2 and 3 must be made in order to enable any discussion of the accuracy to be carried out.

- Assumptions 1 and 5 could be checked out by a pilot survey (which is probably too expensive) or could be inflated to insure that sufficient samples are selected.

- If assumption 4 is believed to be strictly true, then all sample points could be taken on one map. However, this is believed not to be true, therefore multiple maps were selected. The selection of maps was carried out following a stratified random sampling procedure.

The entire Landsat mosaic was stratified on the basis of the 16 mosaic quadrant, which provide a set of strata that satisfy the geographic dispersion requirement. Then, randomly, 22 topographic maps at a scale of 1:50,000 were selected from 9 out of the 16 quadrants as illustrated in Figure 4. No maps were selected from quadrants 1, 5, 9, and 13 because they were not available, and no maps were selected from quadrants 4, 8, and 16 because it was impossible to reliably identify checkpoints on that part of the mosaic (too much topographic relief). The 22 selected maps correspond to approximately 17% of the total number of maps covering the Oruro Department and 10% of the total area covered by the 16 quadrants of the mosaic.

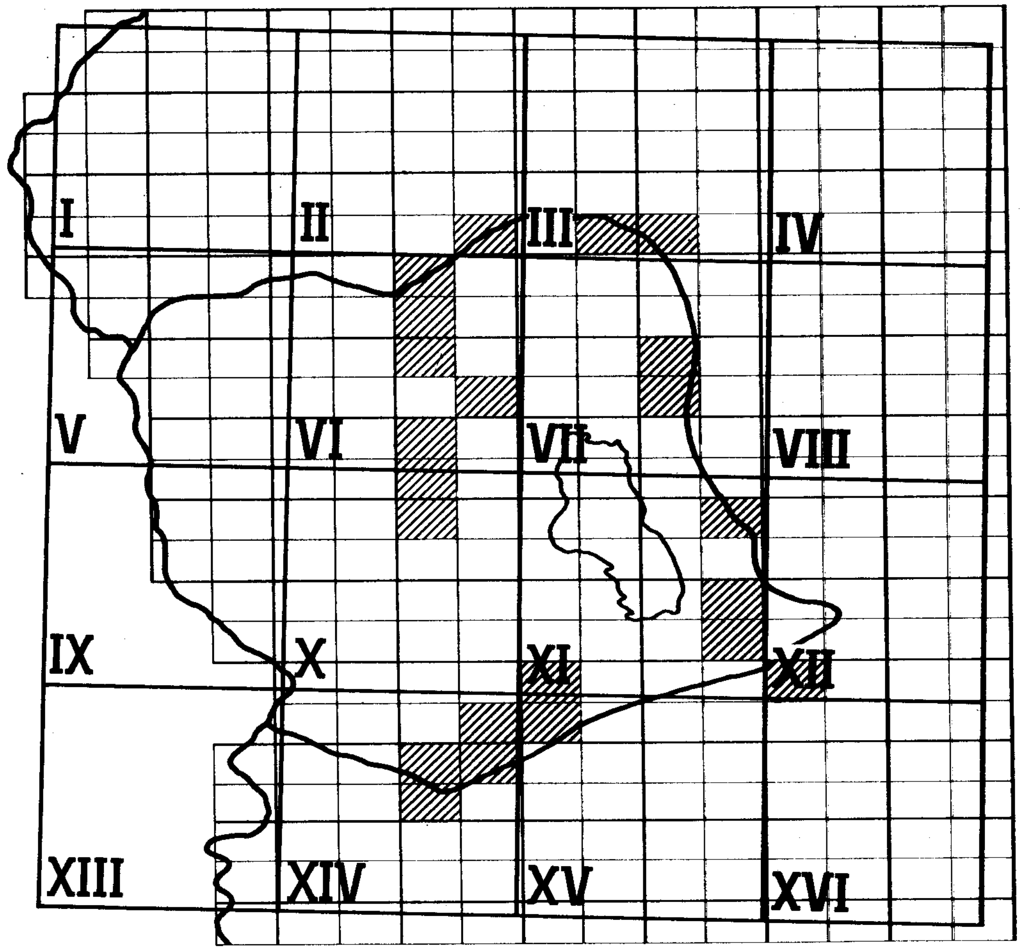


Figure 4. Geographic location of the 22 randomly selected 1:50,000 scale maps used for the planimetric evaluation of the Oruro digital mosaic.

Sixty-eight (68) checkpoints were selected randomly from the 22 topographic maps and its corresponding locations (in Albers addresses) were also determined on the mosaic. Subsequently, the difference between the Albers address of the checkpoints on the mosaic and the Albers addresses on the 1:50,000 scale topographic maps (Δx_i and Δy_i) were computed for both the x and y directions. The Δx_i and Δy_i were averaged for each one of the 22 topographic maps. The results of this planimetric evaluation are included in the following section of this report.

Results

To describe the planimetric accuracy of the Oruro Department digital mosaic, the Δx_i and Δy_i deviations, their standard deviations, and their corresponding RMS (root mean square) errors have been computed and are presented in Tables 3 through 9. The RMS errors were calculated using the following formulae:

$$RMS_x = \sqrt{\frac{\Sigma \Delta x_i^{\,2}}{\eta}}$$

$$RMS_y = \sqrt{\frac{\Sigma \Delta y_i^{\,2}}{\eta}}$$

$$RMS_D = \sqrt{\frac{\Sigma (\Delta x_i^{\,2} + \Delta y_i^{\,2})}{\eta}}$$

where the subscripts x, y, and D correspond to the x direction (longitude), y direction (latitude), and D is the direction of the Eucledian distance $D = \sqrt{x^2+y^2}$.

Table 3. Positional deviations in x (latitude) and y (longitude), and the Euclidian distance (D) for the 22 check points (in meters).

Checkpoint Number	Δx	Δy	D
1	183.5	-150.0	237.0
2	125.0	-300.0	325.0
3	-150.0	137.5	203.5
4	706.5	-994.0	1219.5
5	650.0	-950.0	1151.0
6	416.5	-733.5	843.5
7	-150.0	-50.0	158.0
8	100.0	0.0	100.0
9	25.0	255.0	226.5
10	266.5	200.0	333.0
11	0.0	150.0	150.0
12	25.0	-277.5	278.5
13	33.5	-116.5	121.0
14	20.0	-300.0	300.5
15	100.0	400.0	412.5
16	-150.0	-225.0	270.5
17	-50.0	-250.0	255.0
18	-150.0	-100.0	180.5
19	-133.5	-50.0	142.5
20	0.0	-150.0	150.0
21	25.0	-200.0	201.5
22	16.5	-200.0	202.0

Table 4. Positional deviation means, standard deviations and RMS errors for the digital mosaic including the Oruro frame.

	mean (in meters)	standard deviation (in meters)	RMS error (in meters)
Δx	85.5	238.0	247.5
Δy	-177.5	346.0	382.0
D	338.0	312.0	455.0

Table 5. Positional deviation means, standard deviations and RMS errors of the digital mosaic excluding the Oruro frame.

	mean (in meters)	standard deviation (in meters)	RMS error (in meters)
Δx	5.5	118.5	115.5
Δy	-66.0	201.5	207.0
D	223.0	82.0	237.0

Table 6. Positional deviation means, standard deviations and RMS errors calculated for each Landsat frame.

Landsat Frame	mean			standard deviation			RMS error		
	Δx	Δy	D	Δx	Δy	D	Δx	Δy	D
Desaguadero	61.0	94.5	280.5	53.5	336.5	131.0	75.0	290.5	300.5
Oruro	591.0	-892.5	1071.5	153.5	139.5	200.5	604.0	899.5	1084.0
Caipasa	-89.0	-208.5	247.0	69.5	112.5	76.5	109.0	232.5	256.5
Poopo	63.5	-200.0	223.0	86.5	61.0	65.0	100.0	207.5	230.5
Uyuni	18.5	102.5	204.0	177.0	122.0	87.0	159.0	105.0	218.5

Table. 7 Positional deviation means, standard deviations and RMS errors of some training control points calculated without the Oruro frame.

	mean (in meters)	standard deviation (in meters)	RMS error (in meters)
Δx	14.5	44.0	46.5
Δy	43.0	49.5	65.5
D	71.5	36.5	80.0

Table 8. Positional deviation means, standard deviations and RMS errors of some training control points calculated for the Oruro frame.

	mean (in meters)	standard deviation (in meters)	RMS error (in meters)
Δx	-475.0	285.0	575.5
Δy	300.0	500.0	583.0
D	691.0	441.0	819.5

Table 9. Positional deviation means, standard deviations and RMS errors calculated for the 1:250.000 topographic maps with respect to the 1:50,000 topographic maps.

	mean (in meters)	standard deviation (in meters)	RMS error (in meters)
Δx	66.0	63.5	90.0
Δy	-59.0	56.5	80.5
D	111.5	48.0	121.0

Discussions

The positional deviations in the x, y and D directions for the 22 checkpoints given in Table 3 are illustrated graphically in Figure 5. It is evident in this Figure the lack of check points in the Western half of the Oruro Department since no maps were available for this part of the Department. It is also obvious that the largest positional deviations were found in the Oruro Landsat frame (see Figures 5 and 6). This result was expected because of the poor data quality of this frame as described in the section entitled "Evaluation of the Landsat MSS Data Quality" of the February 1981 quarterly progress report (LARS Contract Report 020181, pp: 3-25).

It is also of interest to note that in all cases the RMS error is greater in the y direction than in the x direction. This is believed to be due to the higher spatial resolution of the Landsat MSS data in the x direction (56 meters) than in the y direction (79 meters).

Table 4 shows that the total RMS error for the Oruro mosaic, including the poor quality Oruro Landsat frame, is 455 meters. However, if the Oruro frame is excluded (Table 5), the RMS error for the mosaic is 237 meters.

Note however, that the RMS error for the control points used in the creation of the mosaic is only of 80 meters (Table 7), which approximates the nominal spatial resolution of the Landsat MSS data.

Finally, a comparison of the positional accuracy of the 1:250,000 scale topographic maps used for the creation of the mosaic with respect to

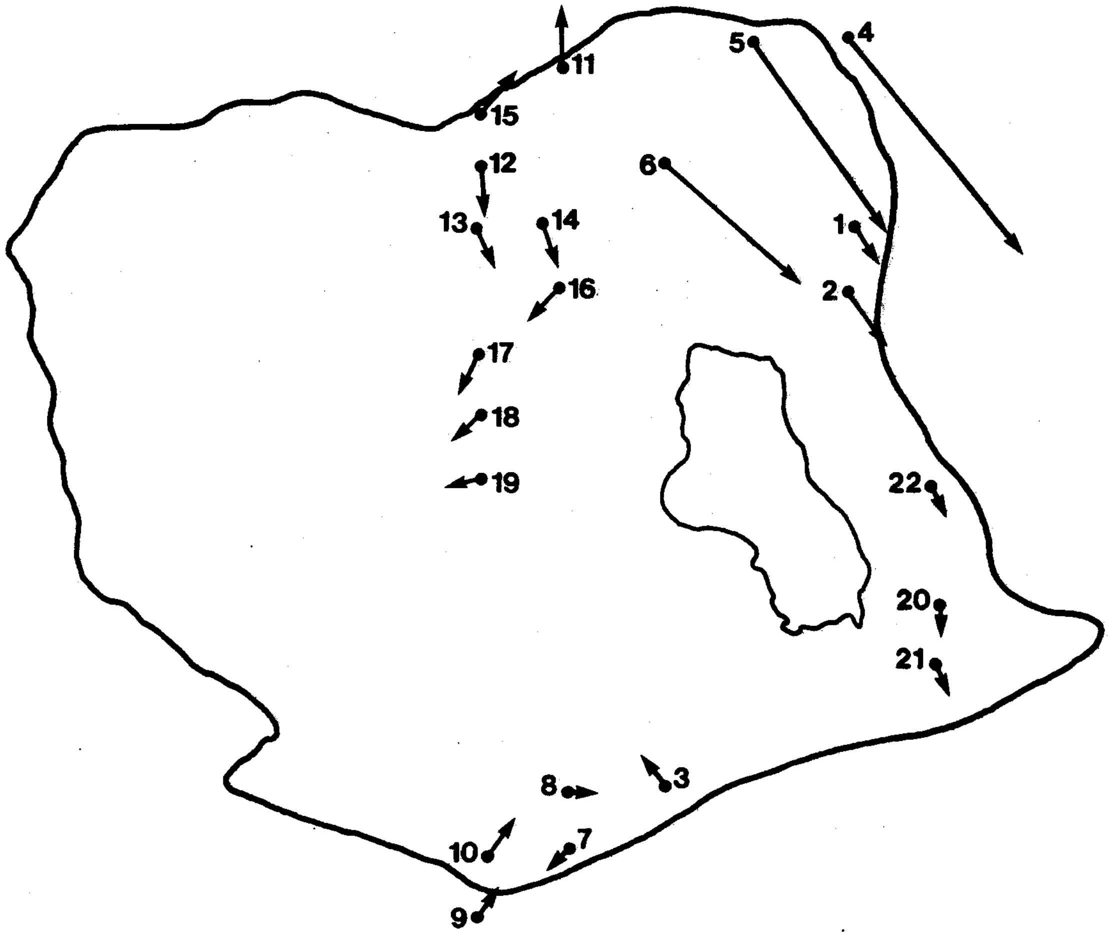
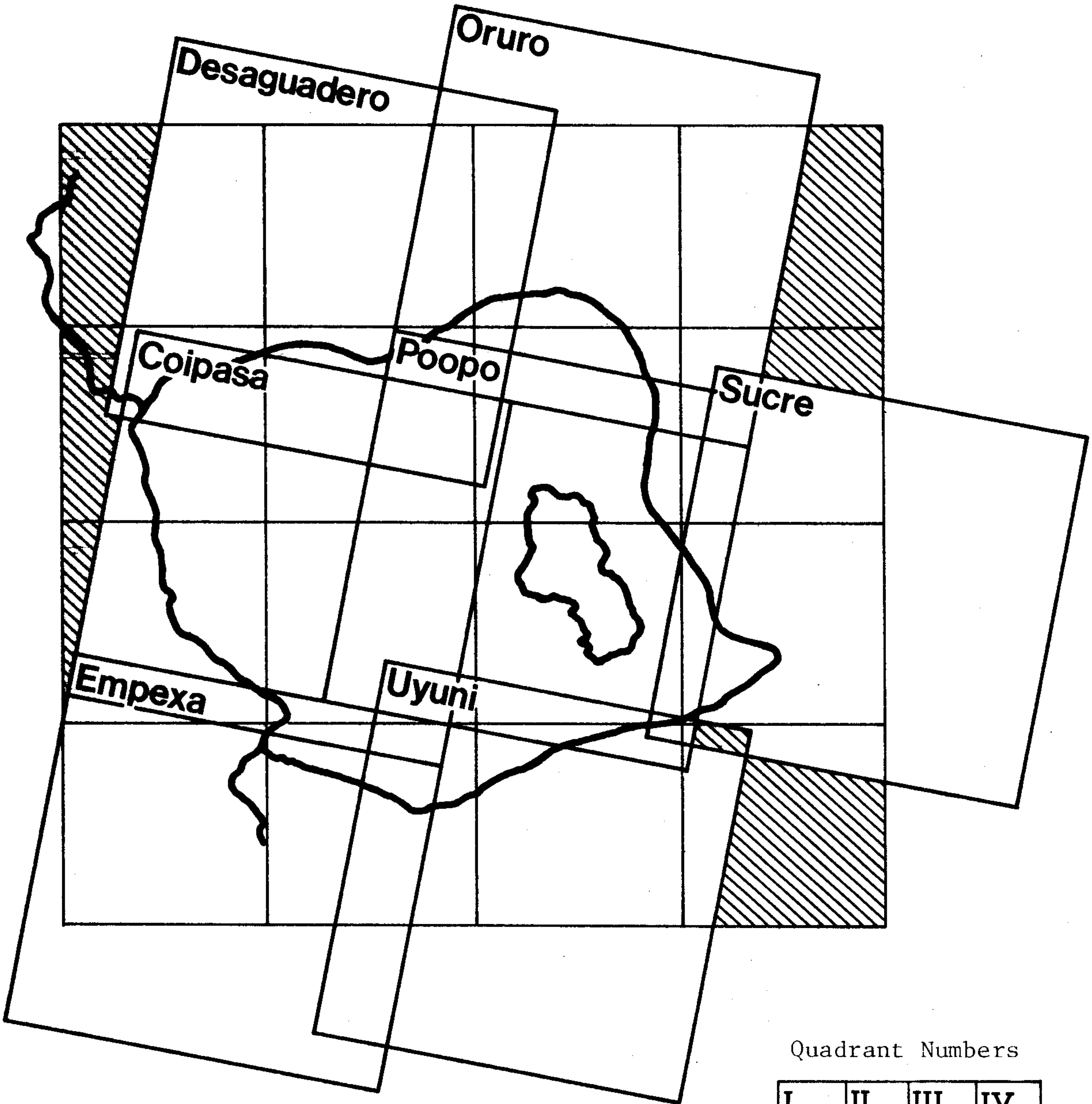


Figure 5. The arrows indicate the positional deviations of the 22 check points of the Oruro mosaic with respect to their position in the 1:50,000 scale topographic maps.



Quadrant Numbers

I	II	III	IV
V	VI	VII	VIII
IX	X	XI	XII
XIII	XIV	XV	XVI

Figure 6. Positional relationahip between the seven (7) Landsat frames and the sixteen (16) digital mosaic quadrants. Shaded area on the mosaic quadrants indicate lack of data.

the 1:50,000 scale topographic maps used for the evaluation of the planimetric accuracy of the mosaic was also conducted. The results of this comparison are given in Table 9, and it shows that the RMS error of the 1:250,000 scale maps is 121 meters, which is within the mapping standards for this scale, i.e. 125 meters.

PREPARATION, CODING AND INPUT OF THE HYDROLOGIC MAPS OF ORURO

Seven 1:250,000 hydrologic maps of the Oruro Department were received at LARS. These maps showed watershed boundaries and relative permeabilities. The watersheds delineated in these maps belong to the Altiplano Basin only. However, since the Oruro Department is located within three Major Basins of Bolivia, and because the GIS was designed for the entire Bolivian territory, the watersheds belonging to the Rio de la Plata and Amazon Basins had to be added to the maps. Also, some of the already existing watersheds had to be re-delineated in order to maintain consistency with the hierarchical data storage structure of the attribute data base.

Since the relative permeability is a derived parameter as opposed to being a basic variable, it will not be included as one of the data base elements. The relative permeabilities could be derived from other elements already included in the data base.

After all the watersheds were re-drawn, the maps were prepared, coded and input as a new element of the data base. Table 10 shows the hierarchical attribute data base structure including the names of the three major basins, second and third order watersheds, their codes and corresponding fill characters.

Table 10. Watersheds of the Oruro Department.

Watershed Name	Fill Character
1. Amazonas	
1. Mamore	
1. Grande	(1)
2. Rio De La Plata	
1. Pilcomayo	
1. Pilcomayo	(2)
3. Altiplano	
1. Titicaca/Poopo	
1. Desaguadero	(3)
2. Lago Poopo	(4)
3. Laguna UruUru	(5)
4. Laguna Soledad	(6)
5. Rio Mauri	(7)
6. Tacagua	(8)
7. Sevaruyo	(9)
8. Rio Marquez	(10)
9. Huari	(11)
10. Penas	(12)
11. Azanaques	(13)
12. Antequera	(14)
13. Khara Khara	(15)
14. Panza	(16)

PREPARATION AND CODING OF THE GEOLOGIC MAPS OF ORURO

The seven geologic/lithologic maps available for the Oruro Department were coded and prepared for digitization. It is expected that both the geologic/lithologic and the geomorphologic maps will be input into the data base by the end of the next reporting period.

The geologic hierarchical coding was prepared for the entire country. Table 11 shows the names, codes, and fill characters for the geologic element corresponding to the Oruro Department. A list of proposed geologic data items that should be included in the geologic attribute data base is given in Table 12.

Table 11. Geologic Units of the Oruro Department

Era Era	System Sistema	Group Grupo	Fill Character
2. Paleozoic Paleozoico			
	2. Ordovician Ordovisico		
		1. Cochabamba	(9)
	3. Silurian Silurico		
		1. Bustillos	(7)
	4. Devonian Devonico		
		* 1. Chuquisaca	(6)
3. Mesozoic Mesozoico			
	3. Cretacic Cretacico		
		1. Potosi	(8)
4. Cenozoic Cenozoico			
	1. Tertiary Terciario		
		* 1. Sedimentario	(4)
		* 2. Volcanico	(3)
		* 3. Intrusivo	(5)
		* 4. Evaporitico	(14)
	2. Quaternary Cuaternario		
		* 1. Volcanicos	(1)
		* 2. Aluvios	(2)
		* 3. Organicos	(10)
		* 4. Intrusivos	(11)
		* 5. Salar	(12)
		* 6. Agua	(13)

* Tentative
  Tentativo

Table 12. Proposed Geologic Data Items.

A. Stratigraphic
  I. Geologic Units
    1. System
      a. Name of Unit
      b. Rock Type
      c. Thickness
      d. Other (Complex, Series, Etc.)
    2. Group
      a. Name of Unit
      b. Rock Type
      c. Thickness
      d. Others
    3. Formation
      a. Name of Unit
      b. Rock Type
      c. Thickness
      d. Cores
      e. Logs
      f. Others
    4. Member
      a. Name of Unit
      b. Rock Type
      c. Thickness
      d. Cores
      e. Petrographics Thin
      f. Logs
      g. Other
B. Structural
  I. Fractures
    1. Faults
      a. Normal
      b. Reverse
      c. Thrust
      d. Fault Pattern
      e. Other
    2. Local
      a. Normal
      b. Reverse
      c. Thrust
      d. Overthrust
      e. Upthrust
      f. Underthrust
      g. Gravity-glide Fault
      h. Other Based on:
        i. Net Slip
        ii. Relation of Fault to Adjacent Strata
        iii. Fault Pattern
        iv. Other
    3. Joints

- II. Folds
  - 1. Regional
    - a. Anticline
    - b. Syncline
    - c. Homocline
    - d. Structural Terrace
    - e. Other
  - 2. Local
    - a. Anticline
    - b. Syncline
    - c. Dimentions
    - d. Other Based On:
      - i. Symmetry
      - ii. Attitude of Axial Surface
      - iii. Divergence of Limbs
      - iv. Fold Shape
      - v. Flexures
      - vi. Relations Among Folds
      - vii. Genetic Classification
      - viii. Fold Systems and Features of Folded Systems
      - ix. Nappe Structures
      - x. Other
- III. Unconformities
  - 1. Regional
    - a. Angular
    - b. Disconformity
    - c. Nonconformity
    - d. Other
  - 2. Local
    - a. Local Unconformity
    - b. Other
- IV. Intrusions
  - 1. Sedimentary
    - a. Salt Domes
    - b. Diapir
    - c. Diapir Fold
  - 2. Igneous Rock
    - a. Batholits
    - b. Stock
    - c. Laccoliths
    - d. Lopoliths
    - e. Phacoliths
    - f. Volcanic Vents
    - g. Ring-Dikes
    - h. Dikes
    - i. Sills
- V. Extrusive Igneous
  - 1. Lava Flows
    - a. Pahoehoe Lava
    - b. AA Lava
    - c. Block Lava

d. Tumuli
e. Pressure Ridges
f. Squeeze-up
g. Spatter Cones
h. Lava Tunnels
i. Collapse Depression
j. Other

2. Pyrollastic Beds
a. Volcanic Dust
b. Volcanic Ash
c. Lapilli
d. Cinders
e. Block
f. Tuff
g. Breccia
h. Nuees Ardemtes
i. Other

3. Volcanoes
a. Volcanic Chain
b. Volcanic Cluster
c. Lava Cones
d. Strato Volcanoes
e. Volcanic Domes
f. Exogenous Domes
g. Endogenous Domes
h. Compund Valcanoes
i. Others

C. Economic
I. Genetic of Mineral Deposits
1. Syngenetic Deposits
2. Epigenetic Deposits
II. Metallic Earth Material
1. Iron
2. Ferro-Allow Meters
3. Copper
4. Lead, Zinc, Silver
5. Gold, Platinium
6. Tin, Wolfram
7. Uranium, Rare Earths
8. Minor Metals
9. Other
III. Non Metallic Earth Material
1. Coal
2. Hidrocarbons
3. Gem Mineral
4. Materials of Miscellaneous Uses

D. Paleontology
I. Paleozoology
1. Vertebrates
2. Invertebrates
II. Paleobotanic
III. Palinology

APPENDIX A

JET PROPULSION LABORATORY California Institute of Technology • 4800 Oak Grove Drive, Pasadena, California 91103

Refer to: 384/RGM:gs-345

September 29, 1981

Mr. Luis A. Bartolucci
Purdue University/LARS
1220 Potter Drive
West Lafayette, Indiana 47906

SUBJECT: Monthly Technical Status Report for August 1981
(141-40-00-01-00; Oruro, Bolivia Mosaic)

Dear Mr. Bartolucci:

The first band (green) of the Bolivia mosaic has been completed and we are now waiting word from you and your staff regarding the appropriateness of the product. As soon as we hear, we will proceed with the remaining bands.

As noted in our phone conversation of 9-10-81, the boundary file for the Oruro Department will be included as a fifth channel in the finished data set. This will present no problems. I've included with this report a hard copy photograph of the Oruro Quad #III with the boundary superimposed on the Landsat data. Please examine this product and advise me if it is correct. I would like to make sure that the offset has been properly selected. This boundary bears little resemblance to existing maps that I have, so I'm having a hard time verifying the fit.

Contact me at your earliest convenience so that we may proceed.

Sincerely,

Ronald G. McLeod

Earth Resources
Applications Group

cc: R. J. Blackwell
D. B. Downey

Enclosure

JET PROPULSION LABORATORY *California Institute of Technology • 4800 Oak Grove Drive, Pasadena, California 91103*

Refer to: 384/RGM:gs-375

October 28, 1981

Mr. Luis Bartolucci
Purdue University/LARS
1220 Potter Drive
West Lafayette, Indiana 47906

SUBJECT: Monthly Technical Status Report for September 1981.
(141-40-00-01-00; Oruro, Bolivia Mosaic)

Dear Mr. Bartolucci:

No progress has been made since the last status report dated September, 1981. Efforts at JPL are on hold while you and your staff review the first band mosaic. I've been advised by Terry Phillips in a phone conversation (October 14, 1981) that a decision will be forthcoming by the end of October.

Sincerely,

Ronald G. McLeod
Earth Resources Application
Group

cc: R. J. Blackwell
C. L. Matthews

Telephone (213) 354-4321 *Twx 910-588-3269* *Twx 910-588-3294*

JET PROPULSION LABORATORY *California Institute of Technology • 4800 Oak Grove Drive, Pasadena, California 91103*

Refer to: 384/RGM:gs-419

November 25, 1981

Mr. Luis Bartolucci
Purdue University / LARS
1220 Potter Drive
West Lafayette, Ind. 47906

SUBJECT: Monthly Technical Status Report for October 1981
(141-40-00-01-00; Oruro, Bolivia Mosaic)

Dear Mr. Bartolucci:

We have received your letter documenting the LARS evaluation of the mosaic and per your instructions, we are proceeding with the remaining bands. We have completed the geometric correction of all frames, all bands. The mosaicking and quad extraction has been done for the first three bands. Only the IR2 band is still under-going processing. Tasks that remain to be completed are the generation of the color composites for each quad and the final report. We are experiencing no major technical problems.

Sincerely,

Ronald G. McLeod
Earth Resources Applications
Group

RGM:gs

cc: R. J. Blackwell
C. L. Matthews

Telephone (213) 354-4321 *Twx 910-588-3269* *Twx 910-588-3294*

APPENDIX B

FILE: IMSHK MAC A LARS / PURDUE UNIVERSITY

```
        .LIST MEB
        .PSECT .REL.RO
MAIN::  START
;THIS IS THE MAIN PROGRAM--IT CALLS THE IMAGE SHRINK SUBROUTINE WITH
;    ARGUMENTS SPECIFYING INPUT IMAGE IS 1. OUTPUT IMAGE IS 2.
;    WINDOW-WIDTH IS 10.. AND WINDOW-HEIGHTH IS 10.
        CALL  IMSHK,#1,#2,#10.,#10.,#0,#0,#511.,#511.,#0,#0
        STOP
;
IMSHK:  ENTER 10.
;THIS IS THE IMAGE SHRINK SUBROUTINE--IT MOVES THE CENTER PIXEL OF
;    EACH WINDOW OF SPECIFIED SIZE IN THE SPECIFIED INPUT IMAGE TO
;    THE OUTPUT IMAGE (OTHER PIXELS FROM THE INPUT WINDOW ARE IGNORED
;    HENCE THE IMAGE SHRINKAGE).  ITS PARAMETERS ARE
;    ARG1(R5)   INPUT IMAGE NUMBER
;    ARG2(R5)   OUTPUT IMAGE NUMBER (MUST DIFFER FROM INPUT IMAGE NR)
;    ARG3(R5)   WIDTH OF WINDOW ON INPUT IMAGE TO BE AVERAGED (X-DIR)
;    ARG4(R5)   HEIGHTH OF WINDOW ON INPUT IMAGE TO BE AVERAGED (Y-DIR)
;    ARG5(R5)   UPPER LEFT X-COORD OF INPUT IMAGE
;    ARG6(R5)   UPPER LEFT Y-COORD OF INPUT IMAGE
;    ARG7(R5)   LOWER RIGHT X-COORD OF INPUT IMAGE
;    ARG8(R5)   LOWER RIGHT Y-COORD OF INPUT IMAGE
;    ARG9(R5)   UPPER LEFT X-COORD OF OUTPUT IMAGE
;    ARG10(R5)  UPPER LEFT Y-COORD OF OUTPUT IMAGE
;LOCAL VARIABLE OFFSETS (RELATIVE TO SP)
HALFWD=0.
OXBASE=2.
OXLAST=4.
OYBASE=6.
OYLAST=8.
IYBASE=10.
ISTATU=12.
OSTATU=14.
IACADR=16.
OACADR=18.
        MOV  ARG3(R5),R1    ;LOAD WINDOW-WIDTH INTO R1
        MOV  R1,R2     ;AND INTO R2
        ASR  R1     ;CALCULATE AND STORE HALF-WINDOW-WIDTH
        MOV  R1,HALFWD(SP)
        MOV  ARG7(R5),R1    ;CALCULATE X-SIZE (WIDTH) OF INPUT IMAGE
        SUB  ARG5(R5),R1
        INC  R1
        CLR  R0     ;DIVIDE BY WINDOW-WIDTH TO GET WINDOWS IN X-DIRECTION
        DIV  R2,R0
        DEC  R0     ;CALCULATE AND STORE LAST X-POSITION OF OUTPUT IMAGE
        ADD  ARG9(R5),R0
        MOV  R0,OXLAST(SP)
        MOV  ARG4(R5),R2    ;LOAD WINDOW-HEIGHTH INTO R2
        MOV  R2,R1     ;CALCULATE AND STORE INITIAL INPUT Y-COORD
        ASR  R1
        ADD  ARG6(R5),R1
        MOV  R1,IYBASE(SP)
        MOV  ARG8(R5),R1    ;CALCULATE Y-SIZE (HEIGHTH) OF INPUT IMAGE
        SUB  ARG6(R5),R1
        INC  R1
        CLR  R0     ;DIVIDE BY WINDOW-HT TO GET WINDOWS IN Y-DIRECTION
        DIV  R2,R0
        DEC  R0     ;CALCULATE AND STORE LAST Y-POSITION OF OUTPUT IMAGE
        ADD  ARG10(R5),R0
        MOV  R0,OYLAST(SP)
        MOV  ARG9(R5),OXBASE(SP)    ;STORE OUTPUT WINDOW LEFT END
        MOV  ARG10(R5),OYBASE(SP)    ;STCRE OUTPUT WINDOW TOP END
        CLR  ISTATU(SP)     ;STORE STATUS CODE FOR INPUT
        CLR  OSTATU(SP)     ;STORE STATUS CODE FOR OUTPUT
        MOV  SP,R0
        ADD  #IACADR,R0     ;CALCULATE ADDR OF IACADR(SP) IN R0
        CALL INITMM,ARG1(R5),R0     ;INIT INPUT MEM MAP & GET ACCESS
        MOV  SP,R0
        ADD  #OACADR,R0     ;CALCULATE ADDR OF OACADR(SP) IN R0
        CALL INITMM,ARG2(R5),R0     ;INIT OUTPUT MEM MAP & GET ACCESS
        CALL WINDOW,ARG2(R5),OXBASE(SP),OXLAST(SP),OYBASE(SP),OYLAST(SP),OSTATU(
YLOOP:      MOV ARG5(R5),R2     ;PUT INITIAL INPUT X-COORD INTO R2
            ADD HALFWD(SP),R2
XLOOP:          CALL WINDOW,ARG1(R5),R2,R2,IYBASE(SP),IYBASE(SP),ISTATU(SP)
                MOV @IACADR(SP),@OACADR(SP)     ;MOVE SELECTED POINT TO OUTPUT
                ADD ARG3(R5),R2     ;INCREMENT INPUT X-CCORD BY WINDOW-WIDTH
                CMP R2,ARG7(R5)     ;COMPARE X-COORD TO RIGHT END OF INPUT
                BLE XLOOP     ;IF X-COORD WITHIN INPUT AREA THEN GOTO XLOOP
            MOV IYBASE(SP),R2     ;INCREMENT INPUT Y-COORD BY WINDOW-HEIGHT
```

```
           ADD ARG4(R5),R2
           MOV R2,IYBASE(SP)
           CMP R2,ARG8(R5)    ;COMPARE Y-COORD TO BOTTOM END OF INPUT
           BLE YLOOP    ;IF Y-COORD WITHIN INPUT AREA THEN GOTC YLOOP
         RETURN
;
;THE FOLLOWING SYMBOLS ARE USED BY THE WINDOW AND INITMM SUBROUTINES:
;    NOTE THAT NUMBERS WHICH DO NOT INCLUDE A DECIMAL POINT ARE OCTAL.
;
;FOR THE FOLLOWING CHANNEL REGISTER ADDRESSES:
;    ADD 0 FOR CHANNEL 0 (USED FOR IMAGE 1)
;    ADD 2 FOR CHANNEL 1 (USED FOR IMAGE 2)
;    ADD 4 FOR CHANNEL 2 (USED FOR IMAGE 3)
RACCES=171000
RXBASE=171040
RXLAST=171100
RYBASE=171140
RYLAST=171200
RSTATU=171240
;
;FOR THE FOLLOWING MEMORY MAP REGISTER ADDRESS:
;    ADD 0 FOR THE FIRST BLOCK (USED FOR IMAGE 1)
;    ADD 2 FOR THE SECOND BLOCK (USED FOR IMAGE 2)
;    ADD 4 FOR THE THIRD BLOCK (USED FOR IMAGE 3)
RMMAP=170000
;
;FOR THE LARS COMTAL CONFIGURATION, THE VALUES IN XBASE AND XLAST
;    REGISTERS SHOULD BE OFFSET BY 512.*CHANNELNUMBER AND THE MEMORY
;    MAP REGISTERS SHOULD BE LOADED WITH  REAL+PORT  WHERE
;    000 DESIGNATES PORT 0 (USED FOR IMAGE 1)
;    100 DESIGNATES PORT 1 (USED FOR IMAGE 2)
;    200 DESIGNATES PORT 2 (USED FOR IMAGE 3)
REAL=100000
;
WINDOW: ENTER
;THE WINDOW SUBROUTINE INITIALIZES RANC CHANNEL REGISTERS TO ACCESS
;    SPECIFIED WINDOW ON THE GIVEN IMAGE NUMBER.  ITS PARAMETERS ARE
;    ARG1(R5)    IMAGE NUMBER (1, 2, OR 3) (NOTE THAT THIS IS NOT THE
;                   IMAGE PLANE NUMBER WHICH AT LARS IS ALWAYS 1)
;    ARG2(R5)    XBASE VALUE (0 TO 511.)
;    ARG3(R5)    XLAST VALUE (0 TO 511. AND >= XBASE)
;    ARG4(R5)    YBASE VALUE (0 TO 511.)
;    ARG5(R5)    YLAST VALUE (0 TO 511. AND >= YBASE)
;    ARG6(R5)    STATUS VALUE (SEE VISION ONE/20 USER'S MANUAL, P.20)
        MOV ARG1(R5),R2
        DEC R2
        MOV R2,R1
        MUL #512.,R1    ;R1 CONTAINS XBASE & XLAST OFFSET FOR IMAGE NR
        ADD R2,R2    ;R2 CONTAINS CHANNEL REG OFFSET FOR IMAGE NR
        MOV ARG2(R5),R0
        ADD R1,R0
        MOV R0,RXBASE(R2)    ;LOAD XBASE REGISTER
        MOV ARG3(R5),R0
        ADD R1,R0
        MOV R0,RXLAST(R2)    ;LOAD XLAST REGISTER
        MOV ARG4(R5),RYBASE(R2)    ;LOAD YBASE REGISTER
        MOV ARG5(R5),RYLAST(R2)    ;LOAD YLAST REGISTER
        MOV ARG6(R5),RSTATU(R2)    ;LOAD STATUS REGISTER
        RETURN
;
INITMM: ENTER
;THE INITMM SUBROUTINE INITIALIZES THE RANC VIRTUAL MEMORY MAP AND
;    RETURNS THE ACCESS REGISTER ADDRESS FOR THE GIVEN IMAGE NUMBER.
;    ITS PARAMETERS ARE
;    ARG1(R5)    IMAGE NUMBER (1, 2, OR 3) (NOT IMAGE PLANE NUMBER)
;    ARG2(R5)    ADDRESS TO WHICH ACCESS REGISTER ADDRESS IS RETURNED
        MOV ARG1(R5),R2
        DEC R2
        MOV R2,R1
        ADD R2,R2    ;R2 CONTAINS CHANNEL & MAP REG OFFSETS FOR IMAGE NR
        MUL #8.,R1    ;R1 CONTAINS RANC PORT NUMBER
        BIS #REAL,R1    ;SET REAL-FLAG BIT IN R1
        MOV R1,RMMAP(R2)    ;LOAD MEMORY MAP REGISTER
        ADD #RACCES,R2
        MOV R2,@ARG2(R5)    ;RETURN ACCESS REGISTER ADDRESS
        RETURN
;
        .END MAIN
```

APPENDIX C

FILE: IMAVG   MAC   A   LARS / PURDUE UNIVERSITY

```
        .LIST MEB
        .PSECT ,REL,RO
MAIN::  START
;THIS IS THE MAIN PROGRAM--IT CALLS THE IMAGE AVERAGE SUBROUTINE WITH
;    ARGUMENTS SPECIFYING INPUT IMAGE IS 1, OUTPUT IMAGE IS 2,
;    WINDOW-WIDTH IS 10., AND WINDOW-HEIGHTH IS 10.
        CALL IMAVG,#1,#2,#10.,#10.,#0,#0,#511.,#511.,#0,#0
        STOP
;
IMAVG:  ENTER 14.
;THIS IS THE IMAGE AVERAGE SUBROUTINE--IT CALCULATES THE AVERAGE PIXEL
;    VALUE OF EACH WINDOW OF A SPECIFIED SIZE IN THE SPECIFIED INPUT
;    IMAGE AND OUTPUT ONE PIXEL OF AVERAGE VALUE PER WINDOW TO THE
;    OUTPUT IMAGE.  ITS PARAMETERS ARE
;    ARG1(R5)    INPUT IMAGE NUMBER
;    ARG2(R5)    OUTPUT IMAGE NUMBER (MUST DIFFER FROM INPUT IMAGE NR)
;    ARG3(R5)    WIDTH OF WINDOW ON INPUT IMAGE TO BE AVERAGED (X-DIR)
;    ARG4(R5)    HEIGHTH OF WINDOW ON INPUT IMAGE TO BE AVERAGED (Y-DIR)
;    ARG5(R5)    UPPER LEFT X-COORD OF INPUT IMAGE
;    ARG6(R5)    UPPER LEFT Y-COORD OF INPUT IMAGE
;    ARG7(R5)    LOWER RIGHT X-COORD OF INPUT IMAGE
;    ARG8(R5)    LOWER RIGHT Y-COORD OF INPUT IMAGE
;    ARG9(R5)    UPPER LEFT X-COORD OF OUTPUT IMAGE
;    ARG10(R5)   UPPER LEFT Y-COORD OF OUTPUT IMAGE
;LOCAL VARIABLE OFFSETS (RELATIVE TO SP)
PIXCNT=0.
IXBASE=2.
OXBASE=4.
IXLAST=6.
OXLAST=8.
IYBASE=10.
OYBASE=12.
IYLAST=14.
OYLAST=16.
ISTATU=18.
OSTATU=20.
IACADR=22.
OACADR=24.
ROUND=26.
        MOV  ARG3(R5),R1    ;LOAD WINDOW-WIDTH INTO R1
        MOV  R1,R2    ;AND INTO R2
        MUL  ARG4(R5),R1    ;MULTIPLY R1 BY WINDOW-HEIGHTH
        MOV  R1,PIXCNT(SP)    ;STORE PIXEL CNT PER WINDOW
        ASR  R1    ;CALCULATE AND STORE AVERAGE-ROUNDING VALUE
        MOV  R1,ROUND(SP)
        MOV  ARG7(R5),R1    ;CALCULATE X-SIZE (WIDTH) OF INPUT IMAGE
        SUB  ARG5(R5),R1
        INC  R1
        CLR  R0    ;DIVIDE BY WINDOW-WIDTH TO GET WINDOWS IN X-DIRECTION
        DIV  R2,R0
        DEC  R0    ;CALCULATE AND STORE LAST X-POSITION OF OUTPUT IMAGE
        ADD  ARG9(R5),R0
        MOV  R0,OXLAST(SP)
        DEC  R2    ;CALCULATE AND STORE INITIAL INPUT WINDOW RIGHT END
        ADD  ARG5(R5),R2
        MOV  R2,IXLAST(SP)
        MOV  ARG4(R5),R2    ;LOAD WINDOW-HEIGHTH INTO R2
        MOV  ARG8(R5),R1    ;CALCULATE Y-SIZE (HEIGHTH) OF INPUT IMAGE
        SUB  ARG6(R5),R1
        INC  R1
        CLR  R0    ;DIVIDE BY WINDOW-HT TO GET WINDOWS IN Y-DIRECTION
        DIV  R2,R0
        DEC  R0    ;CALCULATE AND STORE LAST Y-POSITION OF OUTPUT IMAGE
        ADD  ARG10(R5),R0
        MOV  R0,OYLAST(SP)
        DEC  R2    ;CALCULATE AND STORE INITIAL INPUT WINDOW BOTTOM END
        ADD  ARG6(R5),R2
        MOV  R2,IYLAST(SP)
        MOV  ARG5(R5),IXBASE(SP)    ;STORE INITIAL INPUT WINDOW LEFT END
        MOV  ARG6(R5),IYBASE(SP)    ;STORE INITIAL INPUT WINDOW TOP END
        MOV  ARG9(R5),OXBASE(SP)    ;STORE OUTPUT WINDOW LEFT END
        MOV  ARG10(R5),OYBASE(SP)    ;STORE OUTPUT WINDOW TOP END
        CLR  ISTATU(SP)    ;STORE STATUS CODE FOR INPUT
        CLR  OSTATU(SP)    ;STORE STATUS CODE FOR OUTPUT
        MOV  SP,R0
        ADD  #IACADR,R0    ;CALCULATE ADDR OF IACADR(SP) IN R0
        CALL INITMM,ARG1(R5),R0    ;INIT INPUT MEM MAP & GET ACCESS
        MOV  SP,R0
        ADD  #OACADR,R0    ;CALCULATE ADDR OF OACADR(SP) IN R0
```

```
        CALL INITMM,ARG2(R5),RC   ;INIT OUTPUT MEM MAP & GET ACCESS
        CALL WINDOW,ARG2(R5),OXBASE(SP),OXLAST(SP),OYBASE(SP),OYLAST(SP),OSTATU(
YLOOP:
XLOOP:          CALL WINDOW,ARG1(R5),IXBASE(SP),IXLAST(SP),IYBASE(SP),IYLAST(SP),IST
                CLR R1    ;CLEAR SUM ACCUMULATOR
                MOV IACADR(SP),R0    ;LOAD ADDRESS OF INPUT ACCESS REGISTER
                MOV PIXCNT(SP),R2    ;LOAD INLOOP INDEX WITH PIXEL COUNT
INLOOP:           ADD (R0),R1    ;ADD PIXEL VALUE FROM ACCESS REG TO SUM
                  SOB R2,INLOOP    ;DECREMENT AND TEST CNT OF PIXELS LEFT
                ADD ROUND(SP),R1    ;ADD HALF-DIVISOR SO RESULT IS ROUNDED
                CLR R0    ;DIVIDE SUM BY NUMBER OF PIXELS IN WINDOW
                DIV PIXCNT(SP),R0
                MOV R0,@OACADR(SP)    ;MOVE AVERAGE TO OUTPUT ACCESS REG
                MOV ARG3(R5),R1    ;LOAD WINDOW-WIDTH INTO R1
                ADD R1,IXBASE(SP)    ;ADD WINDOW-WIDTH TO IXBASE
                MOV IXLAST(SP),R2
                ADD R1,R2    ;ADD WINDOW-WIDTH TO IXLAST
                MOV R2,IXLAST(SP)
                CMP R2,ARG7(R5)    ;COMPARE IXLAST TO RIGHT END OF INPUT
                BLE XLOOP    ;IF IXLAST WITHIN INPUT AREA THEN GOTO XLOOP
            MOV ARG5(R5),R1    ;LOAD INITIAL INPUT WINDOW LEFT END INTO R1
            SUB IXBASE(SP),R2    ;RESTORE IXLAST TO ITS INITIAL VALUE
            ADD R1,R2
            MOV R2,IXLAST(SP)
            MOV R1,IXBASE(SP)    ;RESTORE IXBASE TO ITS INITIAL VALUE
            MOV ARG4(R5),R1    ;LOAD WINDOW-HEIGHTH INTO R1
            ADD R1,IYBASE(SP)    ;ADD WINDOW-HEIGHTH TO IYBASE
            MOV IYLAST(SP),R2
            ADD R1,R2    ;ADD WINDOW-HEIGHTH TO IYLAST
            MOV R2,IYLAST(SP)
            CMP R2,ARG8(R5)    ;COMPARE IYLAST TO BOTTOM END OF INPUT
            BLE YLOOP    ;IF IYLAST WITHIN INPUT AREA THEN GOTC YLOOP
        RETURN
;
;THE FOLLOWING SYMBOLS ARE USED BY THE WINDOW AND INITMM SUBROUTINES:
;   NOTE THAT NUMBERS WHICH DO NOT INCLUDE A DECIMAL POINT ARE OCTAL.
;
;FOR THE FOLLOWING CHANNEL REGISTER ADDRESSES:
;   ADD 0 FOR CHANNEL 0 (USED FOR IMAGE 1)
;   ADD 2 FOR CHANNEL 1 (USED FOR IMAGE 2)
;   ADD 4 FOR CHANNEL 2 (USED FOR IMAGE 3)
RACCES=171000
RXBASE=171040
RXLAST=171100
RYBASE=171140
RYLAST=171200
RSTATU=171240
;
;FOR THE FOLLOWING MEMORY MAP REGISTER ADDRESS:
;   ADD 0 FOR THE FIRST BLOCK (USED FOR IMAGE 1)
;   ADD 2 FOR THE SECOND BLOCK (USED FOR IMAGE 2)
;   ADD 4 FOR THE THIRD BLOCK (USED FOR IMAGE 3)
RMMAP=170000
;
;FOR THE LARS COMTAL CONFIGURATION, THE VALUES IN XBASE AND XLAST
;   REGISTERS SHOULD BE OFFSET BY 512.*CHANNELNUMBER AND THE MEMORY
;   MAP REGISTERS SHOULD BE LOADED WITH  REAL+PORT  WHERE
;   000 DESIGNATES PORT 0 (USED FOR IMAGE 1)
;   100 DESIGNATES PORT 1 (USED FOR IMAGE 2)
;   200 DESIGNATES PORT 2 (USED FOR IMAGE 3)
REAL=100000
;
WINDOW: ENTER
;THE WINDOW SUBROUTINE INITIALIZES RANC CHANNEL REGISTERS TO ACCESS
;   SPECIFIED WINDOW ON THE GIVEN IMAGE NUMBER.  ITS PARAMETERS ARE
;   ARG1(R5)    IMAGE NUMBER (1, 2, OR 3) (NOTE THAT THIS IS NOT THE
;                  IMAGE PLANE NUMBER WHICH AT LARS IS ALWAYS 1)
;   ARG2(R5)    XBASE VALUE (0 TO 511.)
;   ARG3(R5)    XLAST VALUE (0 TO 511. AND >= XBASE)
;   ARG4(R5)    YBASE VALUE (0 TO 511.)
;   ARG5(R5)    YLAST VALUE (0 TO 511. AND >= YBASE)
;   ARG6(R5)    STATUS VALUE (SEE VISION ONE/20 USER'S MANUAL, P.20)
        MOV ARG1(R5),R2
        DEC R2
        MOV R2,R1
        MUL #512.,R1    ;R1 CONTAINS XBASE & XLAST OFFSET FOR IMAGE NR
        ADD R2,R2    ;R2 CONTAINS CHANNEL REG OFFSET FOR IMAGE NR
        MOV ARG2(R5),R0
        ADD R1,R0
```

```
        MOV  R0,RXBASE(R2)    ;LOAD XBASE REGISTER
        MOV  ARG3(R5),R0
        ADD  R1,R0
        MOV  R0,RXLAST(R2)    ;LOAD XLAST REGISTER
        MOV  ARG4(R5),RYBASE(R2)    ;LOAD YBASE REGISTER
        MOV  ARG5(R5),RYLAST(R2)    ;LOAD YLAST REGISTER
        MOV  ARG6(R5),RSTATU(R2)    ;LOAD STATUS REGISTER
        RETURN
;
INITMM: ENTER
;THE INITMM SUBROUTINE INITIALIZES THE RANC VIRTUAL MEMORY MAP AND
;    RETURNS THE ACCESS REGISTER ADDRESS FOR THE GIVEN IMAGE NUMBER.
;    ITS PARAMETERS ARE
;    ARG1(R5)    IMAGE NUMBER (1, 2, OR 3) (NOT IMAGE PLANE NUMBER)
;    ARG2(R5)    ADDRESS TO WHICH ACCESS REGISTER ADDRESS IS RETURNED
        MOV  ARG1(R5),R2
        DEC  R2
        MOV  R2,R1
        ADD  R2,R2    ;R2 CONTAINS CHANNEL & MAP REG OFFSETS FOR IMAGE NR
        MUL  #8.,R1   ;R1 CONTAINS RANC PORT NUMBER
        BIS  #REAL,R1    ;SET REAL-FLAG BIT IN R1
        MOV  R1,RMMAP(R2)    ;LOAD MEMORY MAP REGISTER
        ADD  #RACCES,R2
        MOV  R2,@ARG2(R5)    ;RETURN ACCESS REGISTER ADDRESS
        RETURN
;
        .END MAIN
```

APPENDIX D

```
         .LIST MEB
         .PSECT .REL,RO
MAIN::   START
;THIS IS THE MAIN PROGRAM--IT CALLS THE IMAGE SQUARE SUBROUTINE WITH
;    ARGUMENTS SPECIFYING INPUT IMAGE IS 1, OUTPUT IMAGE IS 2,
;    WINDOW-WIDTH IS 10., AND WINDOW-HEIGHTH IS 10.
         CALL  IMSQR,#1,#2,#10.,#10.,#0,#0,#511.,#511.,#0,#0
         STOP
;
IMSQR:   ENTER 15.
;THIS IS THE IMAGE SQUARE SUBROUTINE--FOR EACH WINDOW OF A SPECIFIED
;    SIZE IN THE SPECIFIED INPUT IMAGE IT CALCULATES THE FOLLOWING
;        ISQRT( SUM@I=1,N( SQR( X(I)))/N)
;    AND PLACES THE RESULTING VALUE IN THE APPORPRIATE POSITION OF
;    THE OUTPUT IMAGE SPECIFIED.  THE ACTUAL FORMULA USED INCLUDES
;    PROVISIONS TO PREVENT OVERFLOW WHILE AVOIDING THE USE OF EITHER
;    MULTI-PRECISION INTEGER ARITHMETIC OR REAL ARITHMETIC:
;        2 * ISQRT( SUM@I=1,N(((N/2) + SQR( X(I)/2)) / N))
;    WHERE N = NUMBER OF PIXELS IN A WINDOW AND
;    WHERE X(I) = THE VALUE OF THE I-TH PIXEL AND
;    WHERE SQR( Y) = Y * Y AND
;    WHERE ISQRT( Y) = THE INTEGER SQUARE ROOT FUNCTION OF Y
;        = IF Y .GE. 0 THEN Z SUCH THAT SQR(Z) .LE. Y .LT. SQR(Z+1).
;    THE PARAMETERS OF THIS SUBROUTINE ARE
;    ARG1(R5)    INPUT IMAGE NUMBER
;    ARG2(R5)    OUTPUT IMAGE NUMBER (MUST DIFFER FROM INPUT IMAGE NR)
;    ARG3(R5)    WIDTH OF WINDOW ON INPUT IMAGE TO BE PROCESSED (X-DIR)
;    ARG4(R5)    HEIGHTH OF WINDOW ON INPUT IMAGE TO BE PROCESSED (Y-DIR)
;    ARG5(R5)    UPPER LEFT X-COORD OF INPUT IMAGE
;    ARG6(R5)    UPPER LEFT Y-COORD OF INPUT IMAGE
;    ARG7(R5)    LOWER RIGHT X-COORD OF INPUT IMAGE
;    ARG8(R5)    LOWER RIGHT Y-COORD OF INPUT IMAGE
;    ARG9(R5)    UPPER LEFT X-COORD OF OUTPUT IMAGE
;    ARG10(R5)   UPPER LEFT Y-COORD OF OUTPUT IMAGE
;LOCAL VARIABLE OFFSETS (RELATIVE TO SP)
PIXCNT=0.
IXBASE=2.
OXBASE=4.
IXLAST=6.
OXLAST=8.
IYBASE=10.
OYBASE=12.
IYLAST=14.
OYLAST=16.
ISTATU=18.
OSTATU=20.
IACADR=22.
OACADR=24.
SUMSQR=26.
ROUND=28.
         MOV  ARG3(R5),R1   ;LOAD WINDOW-WIDTH INTO R1
         MOV  R1,R2   ;AND INTO R2
         MUL  ARG4(R5),R1   ;MULTIPLY R1 BY WINDOW-HEIGHTH
         MOV  R1,PIXCNT(SP)   ;STORE PIXEL CNT PER WINDOW
         ASR  R1   ;HALVE PIXEL COUNT TO USE AS ROUNDING-VALUE
         MOV  R1,ROUND(SP)
         MOV  ARG7(R5),R1   ;CALCULATE X-SIZE (WIDTH) OF INPUT IMAGE
         SUB  ARG5(R5),R1
         INC  R1
         CLR  R0   ;DIVIDE BY WINDOW-WIDTH TO GET WINDOWS IN X-DIRECTION
         DIV  R2,R0
         DEC  R0   ;CALCULATE AND STORE LAST X-POSITION OF OUTPUT IMAGE
         ADD  ARG9(R5),R0
         MOV  R0,OXLAST(SP)
         DEC  R2   ;CALCULATE AND STORE INITIAL INPUT WINDOW RIGHT END
         ADD  ARG5(R5),R2
         MOV  R2,IXLAST(SP)
         MOV  ARG4(R5),R2   ;LOAD WINDOW-HEIGHTH INTO R2
         MOV  ARG8(R5),R1   ;CALCULATE Y-SIZE (HEIGHTH) OF INPUT IMAGE
         SUB  ARG6(R5),R1
         INC  R1
         CLR  R0   ;DIVIDE BY WINDOW-HT TO GET WINDOWS IN Y-DIRECTION
         DIV  R2,R0
         DEC  R0   ;CALCULATE AND STORE LAST Y-POSITION OF OUTPUT IMAGE
         ADD  ARG10(R5),R0
         MOV  R0,OYLAST(SP)
         DEC  R2   ;CALCULATE AND STORE INITIAL INPUT WINDOW BOTTOM END
         ADD  ARG6(R5),R2
         MOV  R2,IYLAST(SP)
```

```
        MOV ARG5(R5),IXBASE(SP)     ;STORE INITIAL INPUT WINDOW LEFT END
        MOV ARG6(R5),IYBASE(SP)     ;STORE INITIAL INPUT WINDCW TOP END
        MOV ARG9(R5),OXBASE(SP)     ;STORE OUTPUT WINDOW LEFT END
        MOV ARG10(R5),OYBASE(SP)    ;STORE OUTPUT WINDOW TOP END
        CLR ISTATU(SP)    ;STORE STATUS CODE FOR INPUT
        CLR OSTATU(SP)    ;STORE STATUS CODE FOR OUTPUT
        MOV SP,R0
        ADD #IACADR,R0    ;CALCULATE ADDR OF IACADR(SP) IN R0
        CALL INITMM,ARG1(R5),RC    ;INIT INPUT MEM MAP & GET ACCESS
        MOV SP,R0
        ADD #OACADR,R0    ;CALCULATE ADDR OF OACADR(SP) IN R0
        CALL INITMM,ARG2(R5),RC    ;INIT OUTPUT MEM MAP & GET ACCESS
        CALL WINDOW,ARG2(R5),OXBASE(SP),OXLAST(SP),OYBASE(SP),OYLAST(SP),OSTATU(
YLOOP:
XLOOP:
            CALL WINDOW,ARG1(R5),IXBASE(SP),IXLAST(SP),IYBASE(SP),IYLAST(SP),IST
            CLR SUMSQR(SP)    ;INITIALIZE SUM OF SQUARES TO 0
            MOV PIXCNT(SP),R2    ;LOAD INNER LOOP INDEX
INLOOP:
              MOV @IACADR(SP),R1    ;LOAD PIXEL VALUE X(I)
              ASR R1    ;R1 NOW CONTAINS X(I)/2
              MUL R1,R1    ;R1 NOW CONTAINS SQR( X(I)/2)
              ADD ROUND(SP),R1
              CLR R0
              DIV PIXCNT(SP),RC    ;R0 NOW CCNTAINS (SQR(X(I)/2))/N
              ADD R0,SUMSQR(SP)    ;ACCUMULATE SUM OF SQUARES
              SOB R2,INLOOP    ;DECREMENT AND TEST COUNT OF PIXELS LEFT
            MOV SUMSQR(SP),R2    ;INITIALIZE TEST SQUARE ROOT WITH SUM
            MOV #1,R0    ;INITIALIZE TEST QUOTIENT
I3LOOP:     CMP R2,R0    ;IF TEST SQUARE ROOT IS LESS THAN OR EQUAL
            BLE I3END    ;    TO TEST QUOTIENT THEN THE ISQRT FOUND
              ADD R0,R2    ;AVERAGE OLD TEST SQUARE ROOT AND OLD TEST
              ASR R2       ;    QUOTIENT TO GET NEW TEST SQUARE ROOT IN R2
              CLR R0
              MOV SUMSQR(SP),R1
              DIV R2,R0    ;NEW TEST QUOTIENT IS IN R0
              BR I3LOOP
I3END:      ASL R2    ;MULTIPLY SQUARE ROOT BY 2 TO GET RESULT
            MOV R2,@OACADR(SP)    ;MOVE RESULT IN R2 TO OUTPUT
            MOV ARG3(R5),R1    ;LOAD WINDOW-WIDTH INTO R1
            ADD R1,IXBASE(SP)    ;ADD WINDOW-WIDTH TO IXBASE
            MOV IXLAST(SP),R2
            ADD R1,R2    ;ADD WINDOW-WIDTH TO IXLAST
            MOV R2,IXLAST(SP)
            CMP R2,ARG7(R5)    ;COMPARE IXLAST TO RIGHT END OF INPUT
            BGT XEND
            JMP <XLOOP-A0>(R3)
XEND:     MOV ARG5(R5),R1    ;LOAD INITIAL INPUT WINDOW LEFT END INTO R1
          SUB IXBASE(SP),R2    ;RESTORE IXLAST TO ITS INITIAL VALUE
          ADD R1,R2
          MOV R2,IXLAST(SP)
          MOV R1,IXBASE(SP)    ;RESTORE IXBASE TO ITS INITIAL VALUE
          MOV ARG4(R5),R1    ;LOAD WINDOW-HEIGHTH INTO R1
          ADD R1,IYBASE(SP)    ;ADD WINDOW-HEIGHTH TO IYBASE
          MOV IYLAST(SP),R2
          ADD R1,R2    ;ADD WINDOW-HEIGHTH TO IYLAST
          MOV R2,IYLAST(SP)
          CMP R2,ARG8(R5)    ;COMPARE IYLAST TO BOTTOM END OF INPUT
          BGT YEND
          JMP <YLOOP-A0>(R3)
YEND:
        RETURN
;
;THE FOLLOWING SYMBOLS ARE USED BY THE WINDOW AND INITMM SUBROUTINES:
;   NOTE THAT NUMBERS WHICH DO NOT INCLUDE A DECIMAL POINT ARE OCTAL.
;
;FOR THE FOLLOWING CHANNEL REGISTER ADDRESSES:
;   ADD 0 FOR CHANNEL 0 (USED FOR IMAGE 1)
;   ADD 2 FOR CHANNEL 1 (USED FOR IMAGE 2)
;   ADD 4 FOR CHANNEL 2 (USED FOR IMAGE 3)
RACCES=171000
RXBASE=171040
RXLAST=171100
RYBASE=171140
RYLAST=171200
RSTATU=171240
;
;FOR THE FOLLOWING MEMORY MAP REGISTER ADDRESS:
;   ADD 0 FOR THE FIRST BLOCK (USED FOR IMAGE 1)
;   ADD 2 FOR THE SECOND BLOCK (USED FOR IMAGE 2)
;   ADD 4 FOR THE THIRD BLOCK (USED FOR IMAGE 3)
```

```
RMMAP=170000
;
;FOR THE LARS COMTAL CONFIGURATION, THE VALUES IN XBASE AND XLAST
;    REGISTERS SHOULD BE OFFSET BY 512.*CHANNELNUMBER AND THE MEMORY
;    MAP REGISTERS SHOULD BE LOADED WITH  REAL+PORT  WHERE
;    000 DESIGNATES PORT 0 (USED FOR IMAGE 1)
;    100 DESIGNATES PORT 1 (USED FOR IMAGE 2)
;    200 DESIGNATES PORT 2 (USED FOR IMAGE 3)
REAL=100000
;
WINDOW: ENTER
;THE WINDOW SUBROUTINE INITIALIZES RANC CHANNEL REGISTERS TO ACCESS
;    SPECIFIED WINDOW ON THE GIVEN IMAGE NUMBER.  ITS PARAMETERS ARE
;    ARG1(R5)    IMAGE NUMBER (1, 2, OR 3) (NOTE THAT THIS IS NOT THE
;                   IMAGE PLANE NUMBER WHICH AT LARS IS ALWAYS 1)
;    ARG2(R5)    XBASE VALUE (0 TO 511.)
;    ARG3(R5)    XLAST VALUE (0 TO 511. AND >= XBASE)
;    ARG4(R5)    YBASE VALUE (0 TO 511.)
;    ARG5(R5)    YLAST VALUE (0 TO 511. AND >= YBASE)
;    ARG6(R5)    STATUS VALUE (SEE VISION ONE/20 USER'S MANUAL, P.20)
        MOV  ARG1(R5),R2
        DEC  R2
        MOV  R2,R1
        MUL  #512.,R1     ;R1 CONTAINS XBASE & XLAST OFFSET FOR IMAGE NR
        ADD  R2,R2    ;R2 CONTAINS CHANNEL REG OFFSET FOR IMAGE NR
        MOV  ARG2(R5),R0
        ADD  R1,R0
        MOV  R0,RXBASE(R2)    ;LOAD XBASE REGISTER
        MOV  ARG3(R5),R0
        ADD  R1,R0
        MOV  R0,RXLAST(R2)    ;LOAD XLAST REGISTER
        MOV  ARG4(R5),RYBASE(R2)    ;LOAD YBASE REGISTER
        MOV  ARG5(R5),RYLAST(R2)    ;LOAD YLAST REGISTER
        MOV  ARG6(R5),RSTATU(R2)    ;LOAD STATUS REGISTER
        RETURN
;
INITMM: ENTER
;THE INITMM SUBROUTINE INITIALIZES THE RANC VIRTUAL MEMORY MAP AND
;    RETURNS THE ACCESS REGISTER ADDRESS FOR THE GIVEN IMAGE NUMBER.
;    ITS PARAMETERS ARE
;    ARG1(R5)    IMAGE NUMBER (1, 2, OR 3) (NOT IMAGE PLANE NUMBER)
;    ARG2(R5)    ADDRESS TO WHICH ACCESS REGISTER ADDRESS IS RETURNED
        MOV  ARG1(R5),R2
        DEC  R2
        MOV  R2,R1
        ADD  R2,R2     ;R2 CONTAINS CHANNEL & MAP REG OFFSETS FOR IMAGE NR
        MUL  #8.,R1     ;R1 CONTAINS RANC PORT NUMBER
        BIS  #REAL,R1     ;SET REAL-FLAG BIT IN R1
        MOV  R1,RMMAP(R2)    ;LOAD MEMORY MAP REGISTER
        ADD  #RACCES,R2
        MOV  R2,@ARG2(R5)    ;RETURN ACCESS REGISTER ADDRESS
        RETURN
;
        .END MAIN
```

APPENDIX E

```
        .LIST MEB
        .PSECT .REL,RO
MAIN::  START
;THIS IS THE MAIN PROGRAM--IT CALLS THE IMAGE DEVIATION SUBROUTINE WITH
;    ARGUMENTS SPECIFYING INPUT IMAGE IS 1, OUTPUT IMAGE IS 2,
;    WINDOW-WIDTH IS 10., AND WINDOW-HEIGHTH IS 10.
        CALL  IMDEV,#1,#2,#10.,#10.,#0,#0,#511.,#511.,#0,#0
        STOP
;
IMDEV:  ENTER 17.
;THIS IS THE IMAGE DEVIATION SUBROUTINE--FOR EACH WINDOW OF SPECIFIED
;    SIZE IN THE SPECIFIED INPUT IMAGE IT CALCULATES THE STANDARD
;    DEVIATION AND PLACES THE RESULTING VALUE IN THE APPROPRIATE
;    POSITION OF THE SPECIFIED OUTPUT IMAGE.  TO PREVENT ARITHMETIC
;    OVERFLOW, THE NUMBER OF PIXELS IN A WINDOW MUST BE GREATER THAN 1
;    AND LESS THAN 129.  THE FORMULA USED FOR CALCULATING THE STANDARD
;    DEVIATION INCLUDES PROVISIONS TO PREVENT OVERFLOW:
;        ISQRT( SUM@I=1,N( SQR( X(I) - U)/(N - 1)))
;    WHERE U = AVERAGE PIXEL VALUE = ( SUM@I=1,N( X(I)))/N AND
;    WHERE N = NUMBER OF PIXELS IN A WINDOW AND
;    WHERE X(I) = THE VALUE OF THE I-TH PIXEL AND
;    WHERE SQR( Y) = Y * Y AND
;    WHERE ISQRT( Y) = THE INTEGER SQUARE ROOT OF Y
;        = IF Y .GE. 0 THEN Z SUCH THAT SQR(Z) .LE. Y .LT. SQR(Z+1).
;    THE PARAMETERS OF THIS SUBROUTINE ARE
;    ARG1(R5)    INPUT IMAGE NUMBER
;    ARG2(R5)    OUTPUT IMAGE NUMBER (MUST DIFFER FROM INPUT IMAGE NR)
;    ARG3(R5)    WIDTH OF WINDOW ON INPUT IMAGE TO BE PROCESSED (X-DIR)
;    ARG4(R5)    HEIGHTH OF WINDOW ON INPUT IMAGE TO BE PROCESSED (Y-DIR)
;    ARG5(R5)    UPPER LEFT X-COORD OF INPUT IMAGE
;    ARG6(R5)    UPPER LEFT Y-COORD OF INPUT IMAGE
;    ARG7(R5)    LOWER RIGHT X-COORD OF INPUT IMAGE
;    ARG8(R5)    LOWER RIGHT Y-COORD OF INPUT IMAGE
;    ARG9(R5)    UPPER LEFT X-COORD OF OUTPUT IMAGE
;    ARG10(R5)   UPPER LEFT Y-COORD OF OUTPUT IMAGE
;LOCAL VARIABLE OFFSETS (RELATIVE TO SP)
PIXCNT=0.
IXBASE=2.
OXBASE=4.
IXLAST=6.
OXLAST=8.
IYBASE=10.
OYBASE=12.
IYLAST=14.
OYLAST=16.
ISTATU=18.
OSTATU=20.
IACADR=22.
OACADR=24.
ROUND=26.
DIVSOR=28.
AVERAG=30.
SUMDEV=32.
        MOV  ARG3(R5),R1    ;LOAD WINDOW-WIDTH INTO R1
        MOV  R1,R2    ;AND INTO R2
        MUL  ARG4(R5),R1    ;MULTIPLY R1 BY WINDOW-HEIGHTH
        MOV  R1,PIXCNT(SP)    ;STORE PIXEL CNT PER WINDOW
        DEC  R1    ;STORE ONE LESS THAN PIXEL COUNT AS STD DEV DIVISOR
        MOV  R1,DIVSOR(SP)
        INC  R1
        ASR  R1    ;CALCULATE AND STORE AVERAGE-ROUNDING VALUE
        MOV  R1,ROUND(SP)
        MOV  ARG7(R5),R1    ;CALCULATE X-SIZE (WIDTH) OF INPUT IMAGE
        SUB  ARG5(R5),R1
        INC  R1
        CLR  R0    ;DIVIDE BY WINDOW-WIDTH TO GET WINDOWS IN X-DIRECTION
        DIV  R2,R0
        DEC  R0    ;CALCULATE AND STORE LAST X-POSITION OF OUTPUT IMAGE
        ADD  ARG9(R5),R0
        MOV  R0,OXLAST(SP)
        DEC  R2    ;CALCULATE AND STORE INITIAL INPUT WINDOW RIGHT END
        ADD  ARG5(R5),R2
        MOV  R2,IXLAST(SP)
        MOV  ARG4(R5),R2    ;LOAD WINDOW-HEIGHTH INTO R2
        MOV  ARG8(R5),R1    ;CALCULATE Y-SIZE (HEIGHTH) OF INPUT IMAGE
        SUB  ARG6(R5),R1
        INC  R1
        CLR  R0    ;DIVIDE BY WINDOW-HT TO GET WINDOWS IN Y-DIRECTION
        DIV  R2,R0
```

```
        DEC R0    ;CALCULATE AND STORE LAST Y-POSITION OF OUTPUT IMAGE
        ADD ARG10(R5),R0
        MOV R0,OYLAST(SP)
        DEC R2    ;CALCULATE AND STORE INITIAL INPUT WINDOW BOTTOM END
        ADD ARG6(R5),R2
        MOV R2,IYLAST(SP)
        MOV ARG5(R5),IXBASE(SP)    ;STORE INITIAL INPUT WINDOW LEFT END
        MOV ARG6(R5),IYBASE(SP)    ;STORE INITIAL INPUT WINDCW TOP END
        MOV ARG9(R5),OXBASE(SP)    ;STORE OUTPUT WINDOW LEFT END
        MOV ARG10(R5),OYBASE(SP)    ;STORE OUTPUT WINDOW TOP END
        CLR ISTATU(SP)    ;STORE STATUS CODE FOR INPUT
        CLR OSTATU(SP)    ;STORE STATUS CODE FOR OUTPUT
        MOV SP,R0
        ADD #IACADR,R0    ;CALCULATE ADDR OF IACADR(SP) IN R0
        CALL INITMM,ARG1(R5),RC    ;INIT INPUT MEM MAP & GET ACCESS
        MOV SP,R0
        ADD #OACADR,R0    ;CALCULATE ADDR OF OACADR(SP) IN R0
        CALL INITMM,ARG2(R5),RC    ;INIT OUTPUT MEM MAP & GET ACCESS
        CALL WINDOW,ARG2(R5),OXBASE(SP),OXLAST(SP),OYBASE(SP),OYLAST(SP),OSTATU(
YLOOP:
XLOOP:      CALL WINDOW,ARG1(R5),IXBASE(SP),IXLAST(SP),IYBASE(SP),IYLAST(SP),IST
            CLR R1    ;CLEAR SUM ACCUMULATOR
            MOV IACADR(SP),R0    ;LOAD ADDRESS OF INPUT ACCESS REGISTER
            MOV PIXCNT(SP),R2    ;LOAD INLOOP INDEX WITH PIXEL COUNT
INLOOP:       ADD (R0),R1    ;ADD PIXEL VALUE FROM ACCESS REG TO SUM
              SOB R2,INLOOP    ;DECREMENT AND TEST CNT OF PIXELS LEFT
            ADD ROUND(SP),R1    ;ADD HALF-DIVISOR SO RESULT IS ROUNDED
            CLR R0    ;DIVIDE SUM BY NUMBER OF PIXELS IN WINDOW
            DIV PIXCNT(SP),R0
            MOV R0,AVERAG(SP)    ;STORE AVERAGE PIX VALUE FOR WINDOW
            CLR SUMDEV(SP)    ;INITIALIZE SUM OF DEVIATIONS SQUARED
            CALL WINDOW,ARG1(R5),IXBASE(SP),IXLAST(SP),IYBASE(SP),IYLAST(SP),IST
            MOV PIXCNT(SP),R2    ;LOAD I2LOOP INDEX WITH PIXEL COUNT
I2LOOP:       MOV @IACADR(SP),RC    ;LOAD PIXEL VALUE
              SUB AVERAG(SP),R0    ;CALCULATE (DIVIDED) DEVIATION SQUARE
              MUL R0,R0
              DIV DIVSOR(SP),R0
              ADD R0,SUMDEV(SP)    ;ACCUMULATE SUM OF (DIV) DEV SQR
              SOB R2,I2LOOP    ;DECREMENT AND TEST COUNT OF PIXELS LEFT
            MOV SUMDEV(SP),R2    ;INITIALIZE TEST SQUARE ROOT
            MOV #1,R0    ;INITIALIZE TEST QUOTIENT
I3LOOP:     CMP R2,R0    ;IF TEST SQUARE ROOT IS LESS THAN OR EQUAL
            BLE I3END    ;   TO TEST QUOTIENT THEN THE ISQRT FOUND
              ADD R0,R2
              ASR R2    ;NEW TEST SQUARE ROOT IS IN R2
              CLR R0
              MOV SUMDEV(SP),R1
              DIV R2,R0    ;NEW TEST QUOTIENT IS IN R0
              BR I3LOOP
I3END:      MOV R2,@OACADR(SP)    ;MOVE STANDARD DEVIATION TO OUTPUT
            MOV ARG3(R5),R1    ;LOAD WINDOW-WIDTH INTO R1
            ADD R1,IXBASE(SP)    ;ADD WINDOW-WIDTH TO IXBASE
            MOV IXLAST(SP),R2
            ADD R1,R2    ;ADD WINDOW-WIDTH TO IXLAST
            MOV R2,IXLAST(SP)
            CMP R2,ARG7(R5)    ;COMPARE IXLAST TO RIGHT END OF INPUT
            BLE XLOOP    ;IF IXLAST WITHIN INPUT AREA THEN GOTO XLOOP
          MOV ARG5(R5),R1    ;LOAD INITIAL INPUT WINDOW LEFT END INTO R1
          SUB IXBASE(SP),R2    ;RESTORE IXLAST TO ITS INITIAL VALUE
          ADD R1,R2
          MOV R2,IXLAST(SP)
          MOV R1,IXBASE(SP)    ;RESTORE IXBASE TO ITS INITIAL VALUE
          MOV ARG4(R5),R1    ;LOAD WINDOW-HEIGHTH INTO R1
          ADD R1,IYBASE(SP)    ;ADD WINDOW-HEIGHTH TO IYBASE
          MOV IYLAST(SP),R2
          ADD R1,R2    ;ADD WINDOW-HEIGHTH TO IYLAST
          MOV R2,IYLAST(SP)
          CMP R2,ARG8(R5)    ;COMPARE IYLAST TO BOTTOM END OF INPUT
          BGT YEND
          JMP <YLOOP-A0>(R3)
YEND:
        RETURN

;
;THE FOLLOWING SYMBOLS ARE USED BY THE WINDOW AND INITMM SUBROUTINES:
;    NOTE THAT NUMBERS WHICH DO NOT INCLUDE A DECIMAL POINT ARE OCTAL.
;
;FOR THE FOLLOWING CHANNEL REGISTER ADDRESSES:
;    ADD 0 FOR CHANNEL 0 (USED FOR IMAGE 1)
;    ADD 2 FOR CHANNEL 1 (USED FOR IMAGE 2)
```

```
;    ADD 4 FOR CHANNEL 2 (USED FOR IMAGE 3)
RACCES=171000
RXBASE=171040
RXLAST=171100
RYBASE=171140
RYLAST=171200
RSTATU=171240
;
;FOR THE FOLLOWING MEMORY MAP REGISTER ADDRESS:
;    ADD 0 FOR THE FIRST BLOCK (USED FOR IMAGE 1)
;    ADD 2 FOR THE SECOND BLOCK (USED FOR IMAGE 2)
;    ADD 4 FOR THE THIRD BLOCK (USED FOR IMAGE 3)
RMMAP=170000
;
;FOR THE LARS COMTAL CONFIGURATION, THE VALUES IN XBASE AND XLAST
;    REGISTERS SHOULD BE OFFSET BY 512.*CHANNELNUMBER AND THE MEMORY
;    MAP REGISTERS SHOULD BE LOADED WITH  REAL+PORT  WHERE
;    000 DESIGNATES PORT 0 (USED FOR IMAGE 1)
;    100 DESIGNATES PORT 1 (USED FOR IMAGE 2)
;    200 DESIGNATES PORT 2 (USED FOR IMAGE 3)
REAL=100000
;
WINDOW: ENTER
;THE WINDOW SUBROUTINE INITIALIZES RANC CHANNEL REGISTERS TO ACCESS
;    SPECIFIED WINDOW ON THE GIVEN IMAGE NUMBER.  ITS PARAMETERS ARE
;    ARG1(R5)    IMAGE NUMBER (1, 2, OR 3) (NOTE THAT THIS IS NOT THE
;                   IMAGE PLANE NUMBER WHICH AT LARS IS ALWAYS 1)
;    ARG2(R5)    XBASE VALUE (0 TO 511.)
;    ARG3(R5)    XLAST VALUE (0 TO 511. AND >= XBASE)
;    ARG4(R5)    YBASE VALUE (0 TO 511.)
;    ARG5(R5)    YLAST VALUE (0 TO 511. AND >= YBASE)
;    ARG6(R5)    STATUS VALUE (SEE VISION ONE/20 USER'S MANUAL, P.20)
        MOV  ARG1(R5),R2
        DEC  R2
        MOV  R2,R1
        MUL  #512.,R1     ;R1 CONTAINS XBASE & XLAST OFFSET FOR IMAGE NR
        ADD  R2,R2    ;R2 CONTAINS CHANNEL REG OFFSET FOR IMAGE NR
        MOV  ARG2(R5),R0
        ADD  R1,R0
        MOV  R0,RXBASE(R2)      ;LOAD XBASE REGISTER
        MOV  ARG3(R5),R0
        ADD  R1,R0
        MOV  R0,RXLAST(R2)      ;LOAD XLAST REGISTER
        MOV  ARG4(R5),RYBASE(R2)     ;LOAD YBASE REGISTER
        MOV  ARG5(R5),RYLAST(R2)     ;LOAD YLAST REGISTER
        MOV  ARG6(R5),RSTATU(R2)     ;LOAD STATUS REGISTER
        RETURN
;
INITMM: ENTER
;THE INITMM SUBROUTINE INITIALIZES THE RANC VIRTUAL MEMORY MAP AND
;    RETURNS THE ACCESS REGISTER ADDRESS FOR THE GIVEN IMAGE NUMBER.
;    ITS PARAMETERS ARE
;    ARG1(R5)    IMAGE NUMBER (1, 2, OR 3) (NOT IMAGE PLANE NUMBER)
;    ARG2(R5)    ADDRESS TO WHICH ACCESS REGISTER ADDRESS IS RETURNED
        MOV  ARG1(R5),R2
        DEC  R2
        MOV  R2,R1
        ADD  R2,R2     ;R2 CONTAINS CHANNEL & MAP REG OFFSETS FOR IMAGE NR
        MUL  #8.,R1     ;R1 CONTAINS RANC PORT NUMBER
        BIS  #REAL,R1     ;SET REAL-FLAG BIT IN R1
        MOV  R1,RMMAP(R2)     ;LOAD MEMORY MAP REGISTER
        ADD  #RACCES,R2
        MOV  R2,@ARG2(R5)     ;RETURN ACCESS REGISTER ADDRESS
        RETURN
;
        .END MAIN
```

APPENDIX F

Figure F1. A Dot Matrix Greyscale Image of a Portion of the 15 Meter Spatial Resolution Data, Channel 5 (1.00-1.30 µm). After Latty (1981).

Figure F2. A Dot Matrix Greyscale Image of a Portion of the 30 Meter Spatial Resolution Data, Channel 5 (1.00–1.30 μm). After Latty (1981).

Figure F3. A Dot Matrix Greyscale Image of a Portion of the 45 Meter Spatial Resolution Data, Channel 5 (1.00-1.30 μm). After Latty (1981).

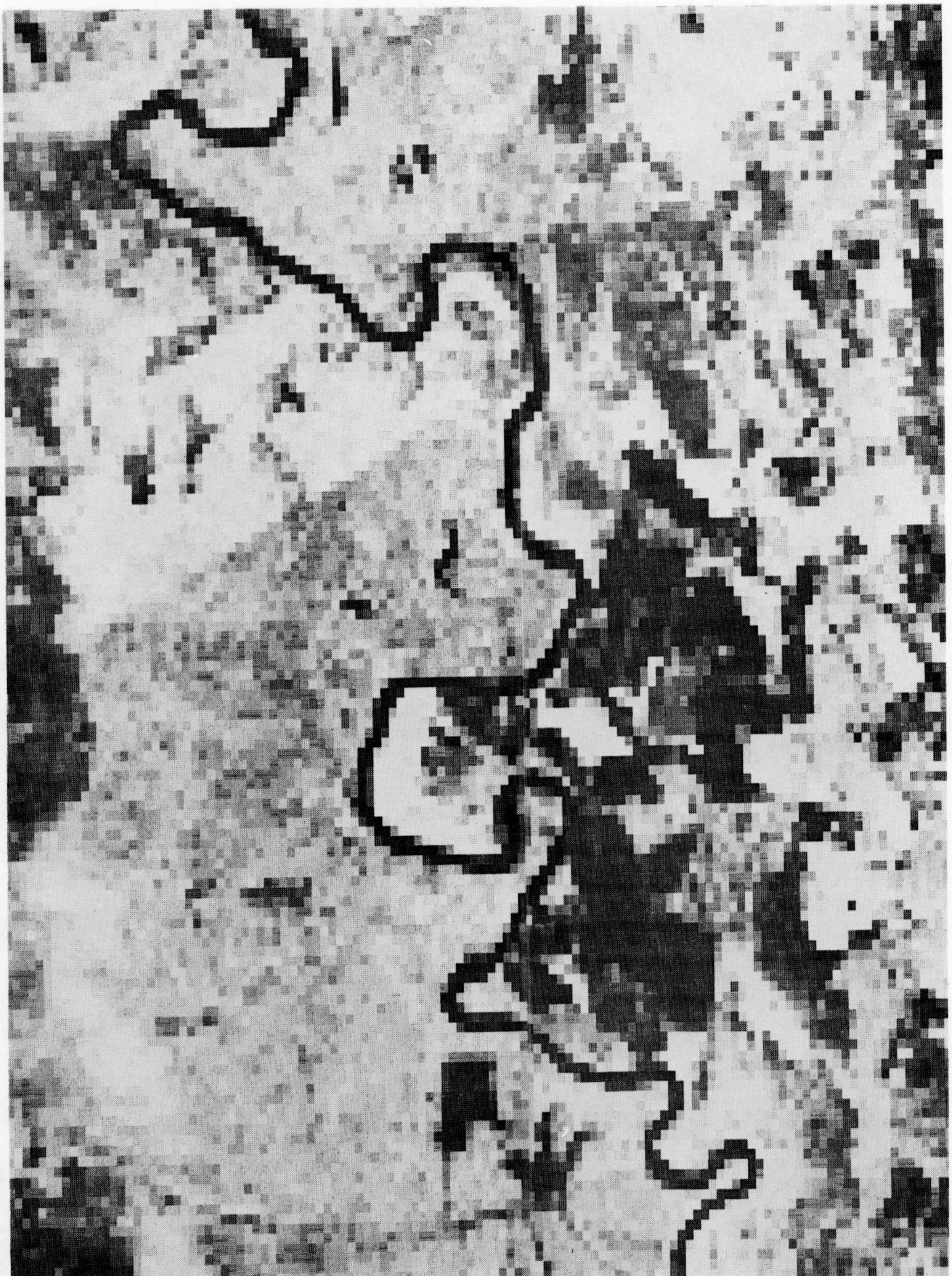


Figure F4. A Dot Matrix Greyscale Image of a Portion of the 80 Meter Spatial Resolution Data, Channel 5(1.00-1.30 μm). After Latty (1981).