USER'S GUIDE TO IMAGE PROCESSING APPLICATIONS OF THE NOAA SATELLITE HRPT/AVHRR DATA

NAS5-28096

PART I
INTRODUCTION TO THE SATELLITE SYSTEM AND ITS APPLICATIONS

PART II
PROCESSING AND ANALYSIS OF AVHRR IMAGERY

Remote Sensing and Image Processing Lab
150 Electrical Engineering Building
Louisiana State University
Baton Rouge, Louisiana 70803
(504) 388-6826

Prepared For

NASA Goddard Space Flight Center
Greenbelt, Maryland 20771
NASA Contract NAS5-28096

January 1986

RSIP TR 301.86

PART I

INTRODUCTION TO THE SATELLITE SYSTEM AND ITS APPLICATIONS

Oscar Karl Huh
Professor
Coastal Studies Institute and
Department of Geology

Scott G. Leibowitz
Research Associate
Remote Sensing and Image Processing Lab

Louisiana State University
Baton Rouge, Louisiana 70803

PART II

PROCESSING AND ANALYSIS OF AVHRR IMAGERY

Scott G. Leibowitz
Research Associate
Remote Sensing and Image Processing Lab

Donald DiRosa
Research Associate
Remote Sensing and Image Processing Lab

and

John M. Hill
Associate Professor
Department of Civil Engineering

Louisiana State University
Baton Rouge, Louisiana 70803

Contents

	Page
Contents	i
List of Figures	iv
List of Tables	vi
Acknowledgements	vii
Preface	ix
PART I - INTRODUCTION TO THE SATELLITE SYSTEM AND ITS APPLICATIONS	1
CHAPTER 1 - INTRODUCTION TO USER'S GUIDE	2
CHAPTER 2 - THE TIROS-N/NOAA A-G SERIES SATELLITES AND THE ADVANCED VERY HIGH RESOLUTION RADIOMETER	7
System and Instruments	7
Orbits	11
The NOAA AVHRR Sensors	16
NOAA AVHRR Calibration	19
NOAA AVHRR Data Types	21
CHAPTER 3 - IMAGE ANALYSIS AND APPLICATIONS	24
Thermal Infrared Data Analysis	24
Land/Water Boundary Mapping	41
Mapping Turbidity in Riverine, Estuarine, and Oceanic Waters	45
Snowfield Mapping	49
Vegetation Index	53
Image Classification	57
Thresholding	57
Unsupervised Classification	57
Supervised Classification	59
Specialized Data Bases	62
The World Data Bank II	62
The Synthetic Bathymetric Profiling System (SYNBAPS)	65
REFERENCES	67
APPENDIX - FACT SHEET ON TIROS-N/NOAA A-G SERIES SATELLITES	71

Page

PART II - PROCESSING AND ANALYSIS OF AVHRR IMAGERY... 73

CHAPTER 4 - THE I^2S IMAGE PROCESSING SYSTEM..... 74

Hardware Environment.......................... 74
Refresh Memory.......................... 74
Pipeline Processing Channels............. 75
Look-Up Tables.......................... 77
Hardware Zoom, Hardware Scroll, and Videometer............................ 78
Trackball and Footpedal................. 78
Command Syntax............................. 82
Input Files............................ 82
Application Functions................... 86
Output Files............................ 89
Miscellaneous Syntax.................... 89
System Functions........................... 91
CI...................................... 91
ACQUIRE................................. 91
LIST'CATALOG............................ 92
SESSION................................. 93
REDO.................................... 94
HELP.................................... 95
END..................................... 96

CHAPTER 5 - DATA ENTRY ON THE I^2S IMAGE PROCESSING SYSTEM...................................... 97

Data Transferal to Disk....................... 97
Tape to Disk............................ 97
Refresh to Disk......................... 99
Data Transferal to Refresh Memory............ 102
Disk to Refresh......................... 102
Pipeline to Refresh..................... 115
Data Transferal to Pipeline.................. 117

CHAPTER 6 - CALIBRATION AND GEOGRAPHIC RECTIFICATION OF NOAA AVHRR IMAGERY.............. 119

Level 1B Tapes............................. 120
The AVHRR Calibration Program................ 122
Tape Management............................. 129
The LCOPY Program........................... 132
The AWARP Program........................... 137

CHAPTER 7 - PREPROCESSING IMAGES ON THE I^2S...... 141

Image Enhancement........................... 141
Point Operators......................... 141
Neighborhood Operators.................. 154
Image Coloring.............................. 155

Page

Image Examination.............................. 160
Image Labeling and Graphics.................. 168

CHAPTER 8 - PRODUCT GENERATION ON THE I^2S........ 174

Sea Surface Temperature...................... 174
BINARY'IMAGE................................. 176
CLUSTER...................................... 177
Supervised Classification.................... 179
TRAIN.. 179
PREPARE...................................... 180
CLASSIFY..................................... 180
C'CLASSIFY................................... 182
MAP.. 183
The SYNBAPS Bathymetry Program............... 185

REFERENCES................................... 189

APPENDIX A - COMMAND SUMMARY FOR NOAA AVHRR IMAGE PROCESSING SOFTWARE ADDED TO THE I^2S PACKAGE..... 190
AVHRR.. 190
AWARP.. 193
LCOPY.. 194
MAP.. 196
POINTS....................................... 198
PROFILE...................................... 199
SYNBAPS...................................... 200

APPENDIX B - COMMAND LISTING FOR IMAGERY PRODUCED FOR THIS MANUAL................................ 202
Figure 14.................................... 202
Figure 15.................................... 203
Figure 16.................................... 204
Figure 17.a.................................. 206
Figure 17.b.................................. 207
Figure 20.................................... 208
Figure 21.................................... 209
Figure 22.................................... 210
Figure 23.................................... 211
Figure 24.................................... 212
Figure 25.................................... 213
Figure 26.................................... 214
Figure 27.................................... 215
Figure 28.................................... 216
Figures 29-30................................ 217
Figures 31-32................................ 218
Figure 33.................................... 219
Figures 34-36................................ 220
Figure 37.................................... 221
Figure 38.................................... 222

List of Figures

Figure		Page
1	Global Meteorological Satellite System........	4
2	Characteristics of the TIROS-N Spacecraft......	8
3	Afternoon (Near Solar Zenith) Orbit (TIROS-N, NOAA 7 and 9)........................	12
4	Morning (Near Termination) Orbit (NOAA 6 and 8)................................	13
5	Atmospheric Transmittance for Six Model Atmospheres................................	27
6	Multichannel Sea Surface Temperatures (MCSST)..	30
7	Thermal Inertia...............................	36
8	Transparency of Water with Respect to Ultra-violet, Visible, and Near Infrared Solar Radiation.....................................	42
9	Land/Water Edge Detection......................	44
10	Turbidity of Surface Waters....................	47
11	Albedo of Highly Turbid Surface Waters.........	48
12	Image Enhancement for Snowfield Patterns.......	52
13	Normalized Vegetation Index....................	55
14	Binary Image Illustrating Thresholding.........	58
15	Unsupervised Classification using CLUSTER......	60
16	Supervised Classification using CLASSIFY.......	61
17	Example of World Data Bank II Output...........	63
18	I^2S Trackball Unit and Footpedal..............	79
19	Example of On-Line Menu for I^2S Software Functions.....................................	81
20	Example of Subsectioning.......................	84

Figure		Page

Figure		Page
21	False Color Composite Image (Bands 1, 2, and 5 Assigned to Red, Green, and Blue Color Guns)...	106
22	False Color Composite Image (Bands 1, 2, and 5 Assigned to Green, Blue, and Red Color Guns)...	111
23	Sample Output from AVHRR Function.............	123
24	Sample Output from LCOPY Function.............	133
25	Sample Output from AWARP Function.............	138
26	Image Enhancement with the TLM Function........	144
27	Image Enhancement with the PIECEWISE'LINEAR Function....................................	146
28	Histogram of Unenhanced Image.................	148
29	Image Enhancement with the H'EQUALIZE Function.	150
30	Histogram of Image Produced with H'EQUALIZE....	151
31	Image Enhancement with the H'NORMALIZE Function......................................	152
32	Histogram of Image Produced with H'NORMALIZE...	153
33	Display of a Monochrome Image with the Blue Color Gun, Using Dummy Bands..................	159
34	Image Magnification (2X).......................	161
35	Image Magnification (4X).......................	162
36	Image Magnification (8X).......................	163
37	Illustration of POINTS Function...............	166
38	Subregions Defined with the BLOTCH Function....	172

List of Tables

Table		Page
1	Summary of the Operational History of the TIROS-N/NOAA Series Satellites.................	14
2	Spectral Band Width and Radiometer Aperture of TIROS-N/NOAA Satellites.....................	17
3	Algorithms for Calculating Sea Surface Temperatures..................................	34

Acknowledgements

The authors wish to express sincere appreciation to Drs. A. Ali and D. A. Quadir of the Bangladesh Space Research and Remote Sensing Organization (SPARRSO) for their time, patience, and scientific efforts during the processing of the AVHRR images in Part I of this manual. Particular thanks are due to Kevin Marshall and Chong Ng of the Remote Sensing and Image Processing Laboratory (RSIP), Louisiana State University (LSU), for the implementation and modification of the AVHRR and associated data base software on the I^2S computer system. David Evans (RSIP) and Taebo Shim (Coastal Studies Institute - CSI) are commended for their assistance during the training phase of this project, which eventually provided the capability to produce the results presented in this manual. We would especially like to thank the Naval Ocean Research and Development Activity (NORDA) at NSTL, Mississippi, for providing the AVHRR, LCOPY, and SYNBAPS programs.

Appreciation is also given to Charles Vermillion, Margaret Williams, and Dr. Harold Maurer for their guidance and trust from a contract management perspective. We are indebted to Gene Shaffer, Science Systems and Applications, Inc., for assistance provided during the implementation of the LSU developed software on the NASA/SPARRSO computer system, and for development of the MAP program.

This project was supported by the Agency for International Development (AID) through the National Aeronautics and Space Administration (NASA), Goddard Space Flight Center, under contract number NAS5-28096.

Preface

This AVHRR User's Guide was generated primarily from the results of an AVHRR software development and training contract. This contract was a part of the Agro-Climatic Environmental Monitoring Project for Bangladesh (ACEMP), and was specifically designed for scientists from the Bangladesh Space Research and Remote Sensing Organization (SPARRSO). Although all images in this manual are of Bangladesh and surrounding areas, its content relative to sensor/platform specifications and applications can be applied to the processing of AVHRR data for any geographic region of the world where there is sufficient coverage.

As this manual was being prepared, a devastating typhoon hit the Bangladesh coast, resulting in a documented loss of ten thousand lives. SPARRSO scientists who trained at LSU as a part of this NASA/AID project were able to use the hardware and techniques described in this manual to track the storm and issue evacuation warnings that prevented the loss of thousands of additional lives. This calamity serves as a reminder of the role that remote sensing technology can play in preventing human loss and improving our quality of life.

PART I

INTRODUCTION TO THE SATELLITE SYSTEM AND ITS APPLICATIONS

Oscar Karl Huh
Professor
Coastal Studies Institute and
Department of Geology

Scott G. Leibowitz
Research Associate
Remote Sensing and Image Processing Lab

Louisiana State University
Baton Rouge, Louisiana 70803

CHAPTER 1

INTRODUCTION TO USER'S GUIDE

In the last two and one half decades, sensors aboard earth orbiting satellites have advanced into highly sophisticated tools for multispectral imaging, atmospheric sounding, and environmental monitoring. The nearly universal and world-wide requirements for high quality, timely meteorological observations and forecasts have stimulated the progressive development of advanced meteorological remote sensing systems. As these systems have been progressively improved, their radiometric and spatial resolving powers have reached levels where whole new areas of applications have become possible. Terrestrial and oceanographic users have in recent years found such systems capable of detecting changes in crop health, ocean current activity, pollution, flooding, snow melt, and a host of other applications. Although this system still lacks the spatial resolution and spectral radiometric capabilities required by certain applications, these meteorological sensors have one primary advantage over all other systems: responsiveness. Image repetition rates range from 10-30 minutes for the geostationary satellites to 4-12 hours for the polar orbiting systems. These environmental satellites are designed with real time transmission systems to provide the user with access to the data within minutes of its acquisition. This responsiveness and high frequency of coverage helps solve some of the greatest problems of applied remote sensing: timely crisis response and the opportunity to acquire data under cloud-free conditions.

The Global Meteorological Satellite system consists of a series of geostationary (earth synchronous) satellites orbiting at fixed locations above the equator and several polar orbiting (sun synchronous) satellites (Figure 1). The former each acquire imagery of one full side of the earth and are best for resolving rapid changes along the equator and the surrounding tropical and temperate belts. These satellites have low spatial and spectral resolution and are suited primarily for atmospheric applications. The two polar orbiting NOAA satellites cover the entire globe with a daytime/nighttime overpass each 24 hours. Overlap of a 2800 km wide swath by each satellite provides for a maximum repetition rate at polar latitudes, while minimum overlap at the equator provides for twice daily coverage at lower latitudes. The relatively high spatial (1.1 km) and spectral (5 narrow banded channels) resolution of these satellites makes them practical for many earth surface applications.

In comparison with the meteorological satellite systems, the Landsat Multispectral Scanner (MSS) and Thematic Mapper (TM) provide extremely high quality, high resolution imagery for the full range of terrestrial and coastal applications. Imagery from these systems, when free

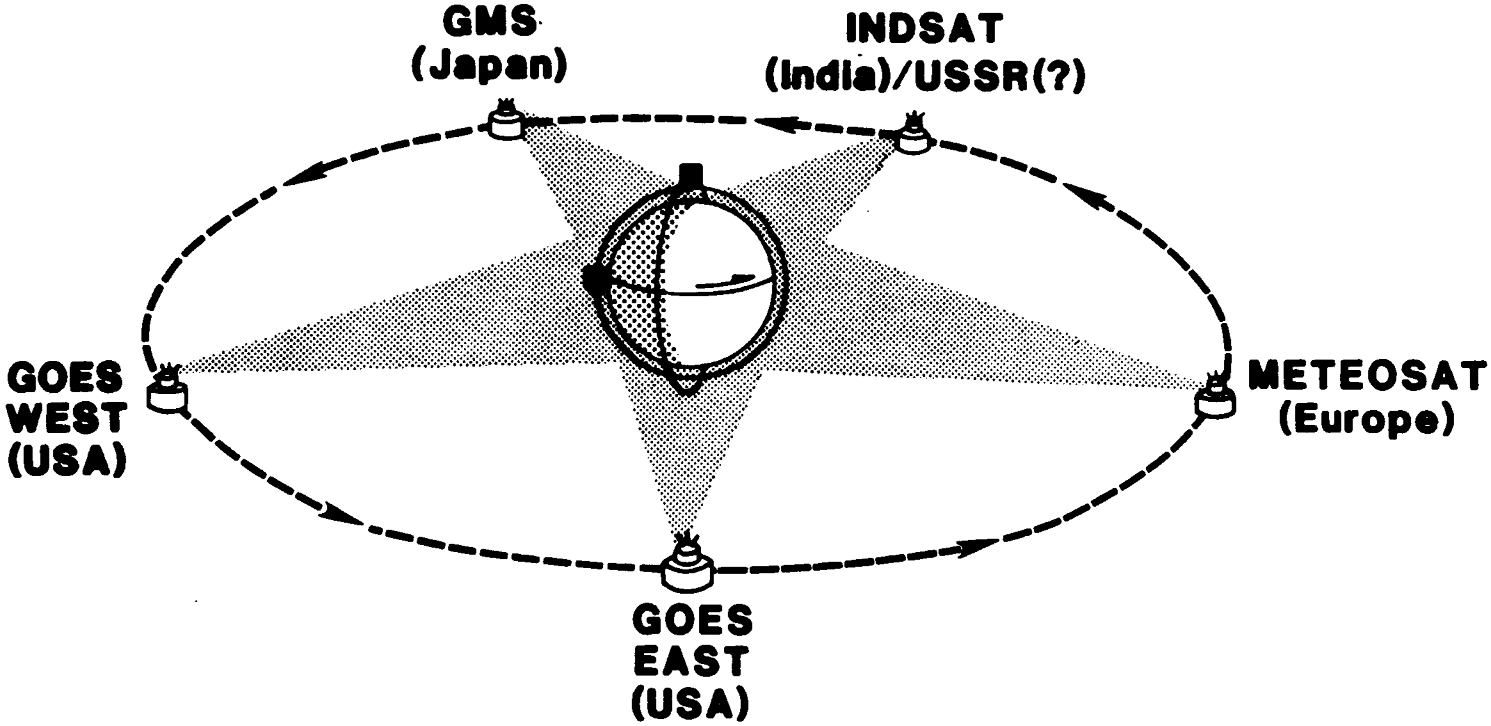


Figure 1. Global meteorological satellite system.

of cloud contamination and when available at the necessary times, provide the premium satellite imagery for most resource monitoring and damage or condition assessment requirements. The infrequent overpass rate (every 16-18 days per satellite) and the greater chance of being clouded out for long periods of time prompt us to urge that processing of Landsat and NOAA imagery be combined so that AVHRR images can track time-varying conditions between Landsat overpasses. The result will be an increased responsiveness and reliability of operational remote sensing for dealing with world environmental problems.

The aim of this manual is to illustrate the use of NOAA AVHRR/HRPT imagery for earth resource applications. It is primarily written for the applications scientist for use within the various earth science, resource, and agricultural disciplines. The purpose is to guide the analyst to processing NOAA AVHRR* and associated data using the hardware and software systems integrated for this NASA project. It will lead the reader through the processing steps from raw data on computer compatible tapes (1B data format) through usable qualitative and quantitative products for applications. The manual is divided into two parts. The first section describes the NOAA satellite system, its sensors, and the theoretical basis for using these data for

*To distinguish between the AVHRR image processing function and the NOAA satellite AVHRR sensor, the function will always be referred to as AVHRR and the sensor will be referred to as NOAA AVHRR.

environmental applications. Part 2 is a hands-on description of how to use a specific image processing system - the International Imaging Systems, Inc. (I^2S) Model 75 Array Processor and S575 software - to process these data. This guide does not replicate or supercede other available user's guides and standard references, but rather builds on their foundation. Other essential references are Barnes and Smallwood (1982), I^2S (1984), Kidwell (1984), Lauritson et al. (1979), and Schwalb (1982).

CHAPTER 2

THE TIROS-N/NOAA A-G SERIES SATELLITES AND THE ADVANCED VERY HIGH RESOLUTION RADIOMETER

System and Instruments

The TIROS-N/NOAA A-G series satellites (Figure 2) are part of a sophisticated system of polar orbiting satellites, data links, and associated ground stations. Its purpose is to provide daily, near real time access to high quality, digital data from a multispectral earth imaging system, an atmospheric sounding system, and a data collection and location system. These satellites carry the following three relevant instruments:

Advanced Very High Resolution Radiometer (AVHRR)

The AVHRR is a five channel scanning radiometer with a 1.1 km resolution. It is designed to discern clouds, land/water boundaries, snow and ice extent, cloud distribution (day and night), and large-scale sea surface and terrestrial features. Its precisely calibrated infrared sensors are designed to measure sea surface and cloud top temperatures.

TIROS Operational Vertical Sounder (TOVS)

The TOVS is a three sensor, 28 channel radiometer system for calculating temperature and humidity profiles of the atmosphere, total ozone

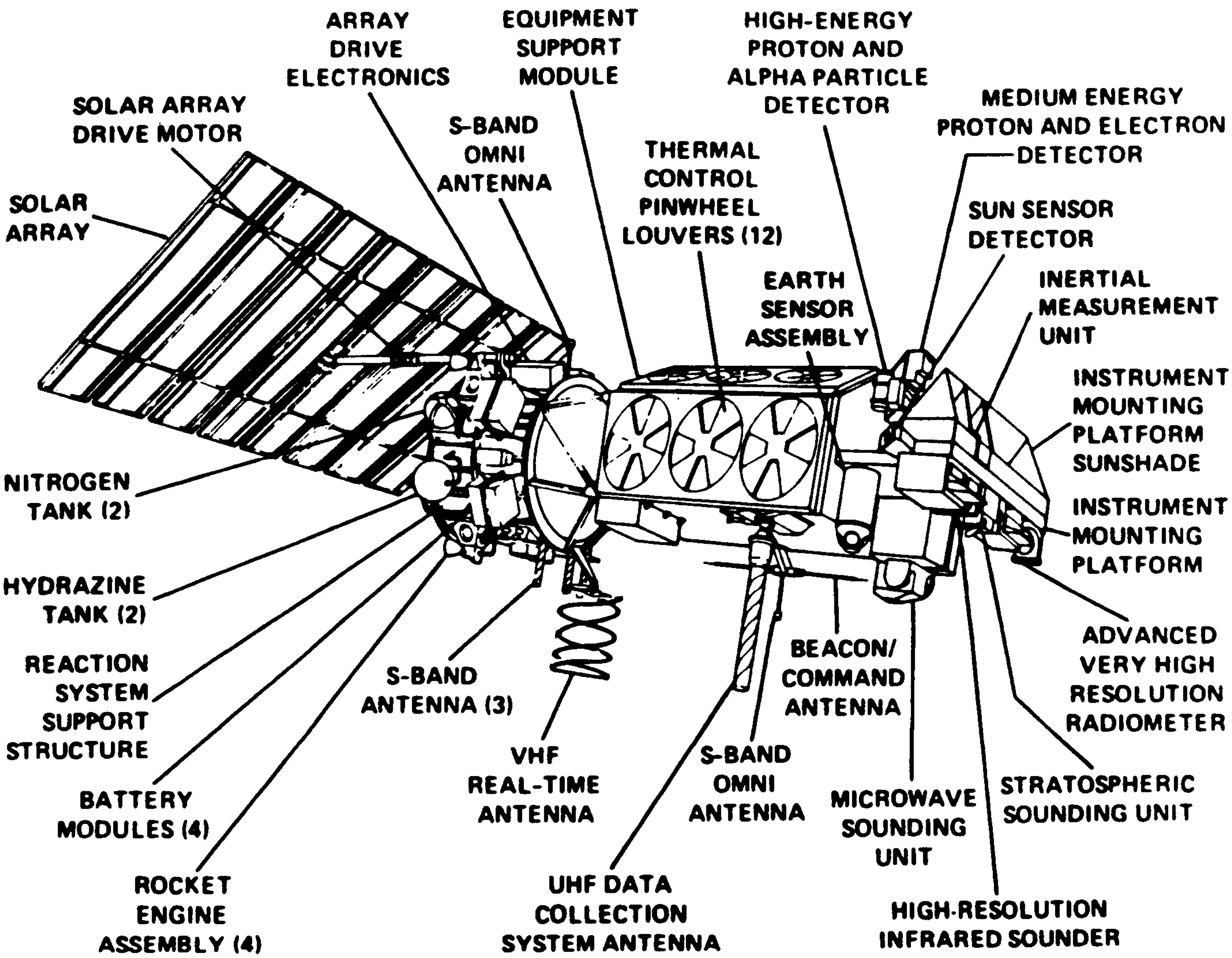


Figure 2. Characteristics of the TIROS-N spacecraft (Source: Schwalb 1978).

content, and derivation of regional pressure fields. Its components include the High Resolution Infrared Radiation Sounder (HIRS/2), the Stratospheric Sounding Unit (SSU), and the Microwave Sounding Unit (MSU). The HIRS/2 radiometer measures energy in 19 regions of the infrared spectrum and one region of the visible spectrum. The SSU, provided by the United Kingdom, is a three channel radiometer operating on a selective absorption technique by measuring radiance passed through a modulated pressure carbon dioxide cell. The MSU is a four channel Dicke radiometer that makes passive measurements in the 5.5 μm oxygen absorption band of the atmosphere (Kidwell 1984, Lauritson et al. 1979, Schwalb 1978, and Smith et al. 1983).

System Argos - The Data Collection and Location System (DCS)

System Argos was built and is operated by the Centre National D'Etudes Spatiales (CNES) of France, who refer to it as the ARGOS Data Collection and Location system. It provides a means for automatically obtaining a time series of measurements and precise earth location from any measurement device equipped with a special transmitter. The data are received and recorded on each satellite overpass and retransmitted to

CNES for processing and distribution (for further information refer to: Service Argos, Centre Spatial De Toulouse, 18 Avenue Edouard Belin, 31055 Toulouse Cedex, France).

These instruments make the NOAA series satellites an operationally efficient system for acquiring near real time information several times daily on terrestrial, oceanic, and atmospheric conditions from both remotely sensed (AVHRR and TOVS) and time series surface measurement (DCS) data.

Orbits

The orbit of the NOAA satellite controls the image repetition rate, the spatial resolution, and the image geometry - factors vital to the utility of NOAA AVHRR data. The near polar, sun synchronous orbits of the NOAA satellites are illustrated in Figures 3 and 4. The orbits of these satellites are nominally inclined some nine degrees from the pole, having an orbital period of 102 minutes and operating at an altitude of 850 km. Each satellite completes some 14.2 orbits per day, which results in daily shifting of satellite subtracks without exact repeats as with Landsat. The equatorial crossings (nodes) step westward around the equator some 25° with each successive orbit. The sun synchronous orbit keeps the satellite in a constant relationship to the sun so that radiometer data are acquired at approximately the same local solar time (LST) each day. Data are thus provided for each portion of the earth at the same solar illumination/nocturnal cooling conditions for each daytime and nighttime overpass per satellite. This two satellite system is designed to operate with the following approximate overpass times at each location on earth: (a) one satellite crossing the equator southbound (daytime pass) at 0730 LST and northbound (nighttime pass) at 1930 LST; (b) the other satellite crossing the equator southbound (nighttime) at 0300 LST and northbound (daytime) at 1500 LST. The satellite orbits and

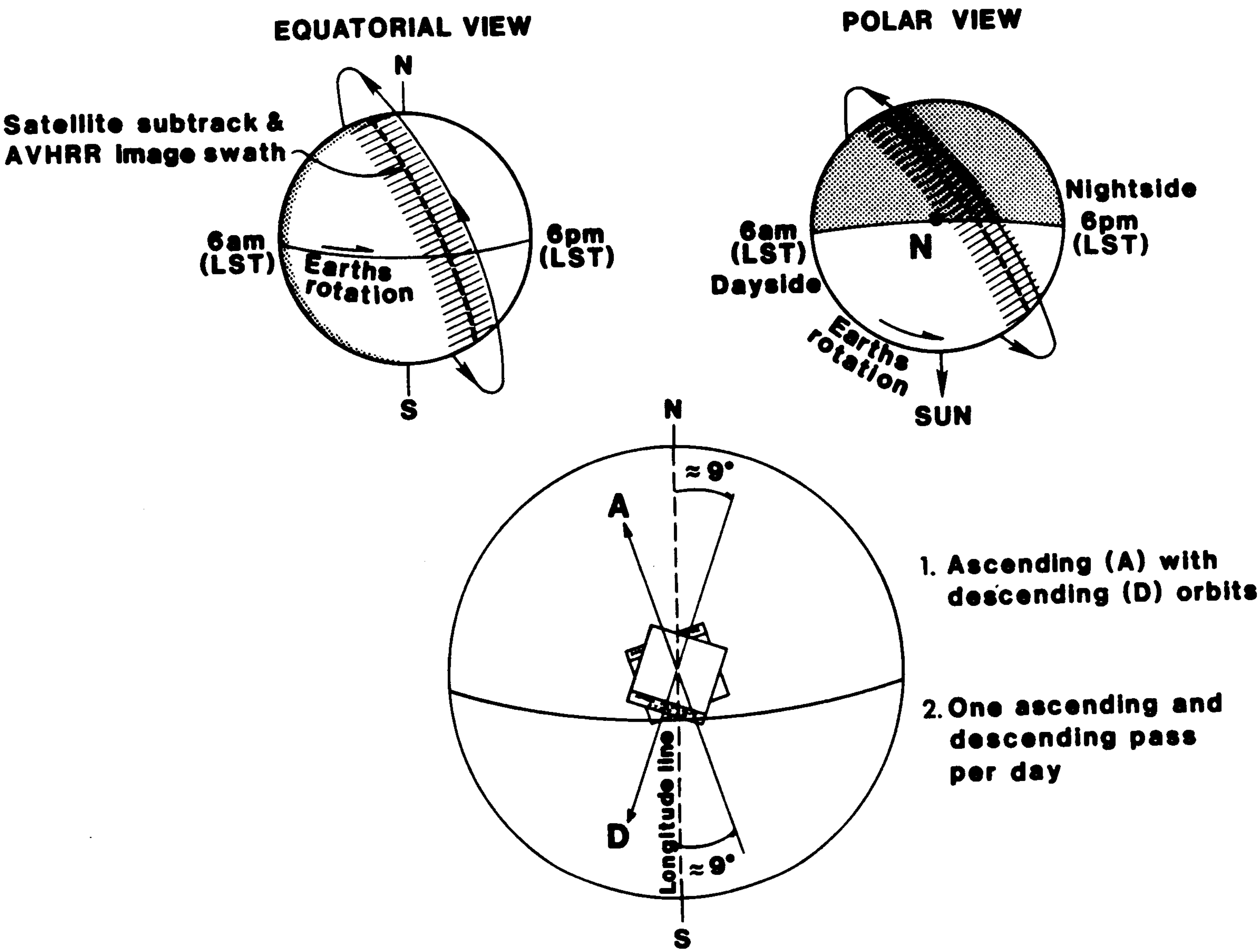


Figure 3. Afternoon (near solar zenith) orbit (TIROS-N, NOAA 7 and 9).

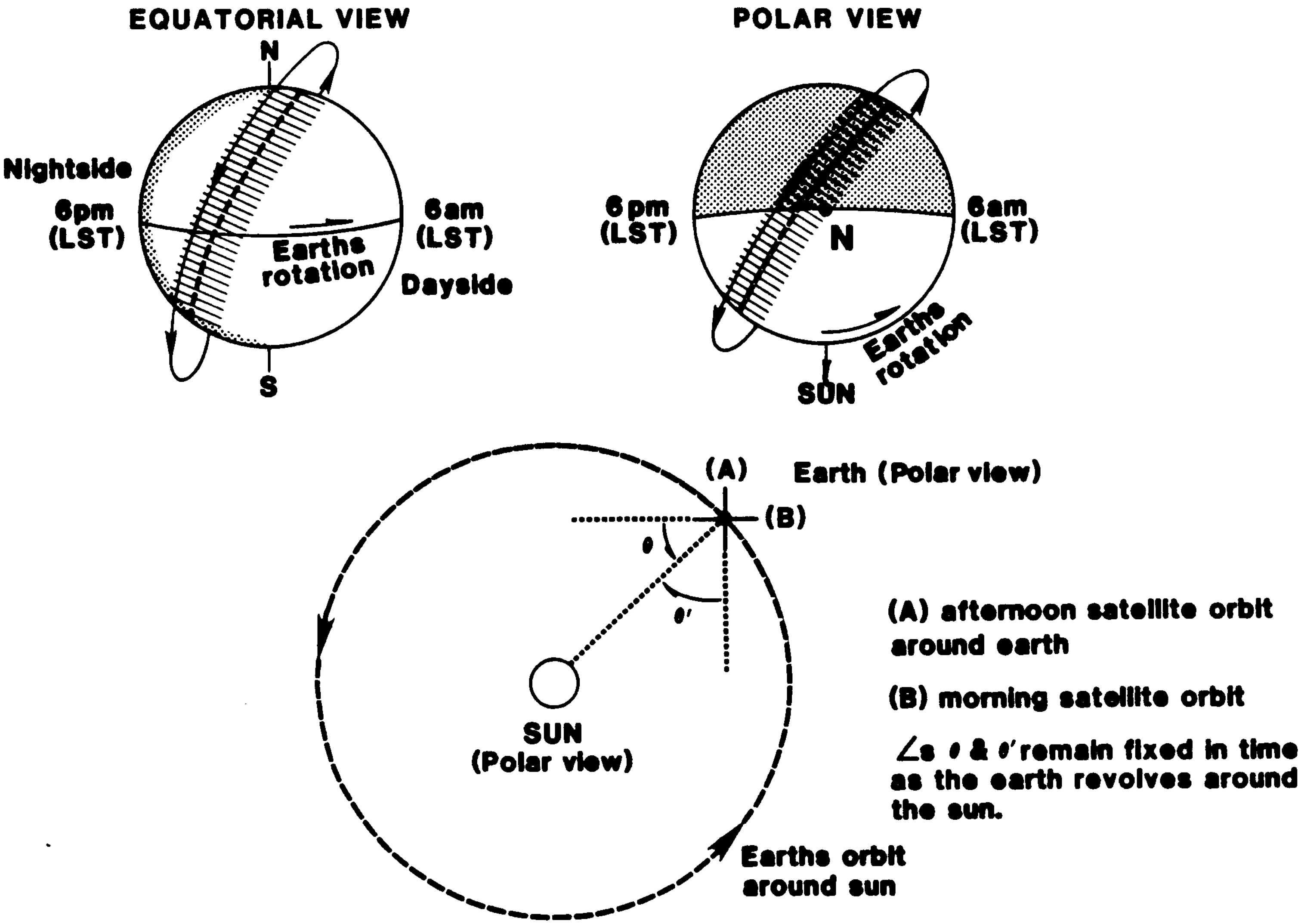


Figure 4. Morning (near termination) orbit (NOAA 6 and 8). Orbital planes of polar orbiting satellites are fixed with respect to sun (local sun time - LST). The earth rotates beneath the solar-stationary orbital plane to provide successive swaths around it.

Table 1. Summary of the operational history of the TIROS-N/NOAA series satellites.

Satellite (1)			Ascending Node (LST)	Descending Node (LST)	Operational Period (2)
TIROS-N	TIROS-1	------	1500	0300	10/19/78-1/30/80
NOAA 6	TIROS-2	NOAA A	1930	0730	6/27/79-3/05/83
-------	TIROS-3	NOAA B	----	----	(3)
NOAA 7	TIROS-4	NOAA C	1430	0230	8/24/81-2/18/85
-------	TIROS-5	NOAA D	----	----	(4)
NOAA 8	TIROS-6	NOAA E	1930	0730	5/03/83-6/12/85
NOAA 9	TIROS-7	NOAA F	1430	0230	12/28/84-present

(1) First column is satellite designation by NOAA once it is operational. Second column is satellite designation in ephemeris by NORAD for tracking. Third column is satellite designation by NASA before turnover to NOAA. The NOAA E through J spacecraft are the Advanced TIROS-N (ATN) satellites similar to NOAA A-D, but will include the Earth Radiation Budget Experiment (ERBE) sensor and the Solar Backscatter Ultraviolet Radiometer (SBUV/2) in its instrument payload (Schwalb 1982).
(2) NOAA 6 has been reactivated for direct readout, 1.1 km NOAA AVHRR imagery.
(3) Failed to reach orbit.
(4) Not launched.

---

the equatorial crossings (nodes) are illustrated in Figures 3 and 4. A summary of the operational history of TIROS-N/NOAA satellites is provided in Table 1.

Because of orbit and image geometry the unrectified images are not true charts; true north is nine degrees to the east or west of the satellite subtrack (Figure 3). Since the earth rotates beneath the satellite and the orbit is inclined to the pole, each successive scan line is offset westward from the previous one by a small increment. Both this offset as well as along-scan changes in pixel size must be rectified in order to have a geometrically correct image.

The NOAA AVHRR Sensors

The NOAA AVHRR is an extremely high quality, cross-track, line scanning multispectral radiometer. Scanning of the earth scene is done cross track from anti-sun to sun direction by a rotating mirror between the sensor aperture and the radiation detectors. This mirror rotates at a speed of 360 rpm which, when combined with satellite motion, altitude, and aperture, result in successive scan lines being contiguous at nadir. It views the earth scene through an angle of 110.8°, plus or minus 55.4° from nadir. The spatial resolution of the NOAA AVHRR is controlled by satellite altitude and radiometer aperture. The instantaneous field of view (IFOV) of the NOAA AVHRR is an aperture of 1.3 milliradians (Table 2), yielding a satellite subpoint resolution of 1.1 km and a 4.0 km resolution at the far edges of the image swath. The IFOV of the four or five channels (TIROS-N and NOAA 6, 8, and F had only 4 channels) is made coincident within plus or minus 0.1 milliradians (8%).

The NOAA AVHRR channels were chosen to permit multispectral analyses which have been shown to provide improved determination of terrestrial, hydrologic, oceanographic, and meteorologic parameters. The visible/near infrared (NIR) sensors (channels 1 and 2) are filtered silicon detectors that measure the incoming

Table 2. Spectral Band Width and Radiometer Aperture of TIROS-N/NOAA Satellites. (1)

Channel	Band Width			IFOV
1	0.55 - 0.90	0.58 - 0.68	0.58 - 0.68	1.39
2	0.725 - 1.10	0.725 - 1.10	0.725 - 1.10	1.41
3	3.55 - 3.93	3.55 - 3.93	3.55 - 3.93	1.51
4	10.5 - 11.5	10.5 - 11.5	10.3 - 11.3	1.41
5	(2)	(2)	11.5 - 12.5	1.3

(1) Band width in μm and instantaneous field of view (IFOV) in milliradians. First band width column is for TIROS-N, second column is for NOAA 6, 8, and G, and third column is for NOAA 7, 9, D, H, I, and J.
(2) These satellites only contained four channels. However channel 4 data were repeated to give a fifth channel.

---

radiation to a 3:1 signal to noise ratio at 0.5% albedo. The thermal channels use indium antimonide (channel 3) and mercury-cadmium-telluride (channels 4 and 5) detectors cooled to an operating temperature of 105 K. These latter two thermal channels have a noise equivalent differential temperature (NE Δ T) of 0.12 at 300 K. Channel 3 has consistently had serious noise problems requiring averaging or filtering for radiometric applications (Lynn and Svejkovsky 1984), a problem apparently not yet solved. The analog data signal output from the NOAA AVHRR detectors are digitized on board the satellite to 10-bit precision. Each sample step corresponds to a scanner rotation angle of 0.95 milliradians, giving 1.362 samples (oversampling) for each

IFOV. Bandwidths and IFOVs of the five channels are presented in Table 2.

NOAA AVHRR Calibration

Procedures for precise calibration of the HRPT/AVHRR channels have been established to rigorously relate sensor output data to radiance measurements. Calibration details are provided by Lauritson et al. (1979). In brief, both prelaunch and inflight calibration procedures are used. Prior to launch each NOAA AVHRR instrument is tested for stability, linearity of response, and sensitivity to radiance. The prelaunch calibration of the IR sensors is conducted in a vacuum to simulate space conditions and values are referenced to precision blackbody sources of the U.S. National Bureau of Standards (NBS). The typical prelaunch calibration will expose the sensor to radiation from a standard blackbody source whose temperature is varied in discrete steps over the appropriate range of temperatures. The temperature of the sensor itself is varied so that deviations in performance due to spacecraft temperature can be determined.

The visible and NIR channels of the NOAA AVHRR are calibrated precisely at ambient temperature in air against lamps whose output is precisely controlled. Calibration of these lamps is also based on NBS measurements.

Since the AVHRR sensor ages in orbit, prelaunch calibrations cannot be relied upon. Due to spontaneous instrument changes with time, these calibrations may become useless. For the thermal channels, a two-step inflight calibration is programmed. After scanning the earth scene below the satellite, the radiometer views deep space as a source of near 0 K radiance and then a temperature controlled 15 °C (288.14 K) on-board blackbody source. These radiance values, along with temperature data from platinum resistance thermometers imbedded in the on-board blackbody, are multiplexed into the NOAA AVHRR data stream for each scan line. There is no inflight calibration for the visible or NIR channels. The NOAA AVHRR data from the five channels, digitized on board to 10-bit precision, provide a maximum temperature resolving power of 0.125 °C (at the warm end of the dynamic range) for the thermal channels, and 0.25% albedo for the visible and NIR.

NOAA AVHRR Data Types

Use of the NOAA AVHRR data as described in this guide will begin with data processed to level 1B data base format. This level of processing involves quality checking and addition of calibration coefficients, earth location, and solar zenith information to every scan line. Each scan line has calibration coefficients appended after a series of 2048 samples (pixels) of the earth scene. The calibration coefficients are calculated from inflight measurements as described above. These coefficients are gain and intercept values (in units of milliwatts/[m^2 steradian cm^{-1}] per digital count) for the linear equation that relates sensor output to radiance. This calibrates the data through the full range of sensor response (Lauritson et al. 1979). The earth location (lat/long) and solar zenith angle (daytime passes) are inserted into the 2048 pixel record at every 40th pixel along the scan line, starting at pixel 25 (e.g., 25, 65, 105, ..., 2025), for a maximum of 51 possible earth location points per line.

To convert these level 1B data into geometrically corrected and radiometrically calibrated images on the I^2S system, the following programs are used (these programs are described in detail in Chapter 6, pgs. 122-128, 132-140). The AVHRR function (Figure 23, pg. 123) is used to convert engineering units (digital counts) to radiances (milliwatts/[m^2 steradian cm^{-1}]) through the use of the calibration coefficients; these are then converted to the geophysical parameters of temperature and albedo. AVHRR also produces a separate ground control point file. The LCOPY program is used to linearize the size of all pixels to 1.1 km by replicating the larger area pixels of the image margin, thus creating an equal area image (Figure 24, pg. 133). LCOPY also produces an amended control point file. The AWARP function takes this file of ground control points, arranges them in the correct geographic pattern, and "rubber sheets" the two dimensional image to fit the control points properly (Figure 25, pg. 138). At this point in processing of the 1B data, the image is a geographically corrected, geophysically calibrated map of the earth scene.

The NOAA AVHRR sensor provides calibratible digital image data in the following modes:

> High Resolution Picture Transmission (HRPT) - This is the five channel, 1.1 km resolution imagery broadcast directly in real time from the satellite and acquired by any appropriate tracking antenna.
>
> Automatic Picture Transmission (APT) - These are the two channel (selected by the Satellite Operations Control Center), 4.0 km NOAA AVHRR sensor data continuously broadcast in real time from the satellites.

High Resolution Recorded Imagery (Local Area Coverage) - This is the five channel, 1.1 km resolution imagery tape recorded on board the satellite for later transmission to U.S. ground stations.

Reduced Resolution Recorded Imagery (Global Area Continuously Recorded Coverage) - This five channel 4.0 km resolution data is transmitted to U.S. ground stations and archived permanently for the globe by National Environmental Data and Information Service.

CHAPTER 3

IMAGE ANALYSIS AND APPLICATIONS

Thermal Infrared Data Analysis

Surface temperature measurements by satellite infrared radiometry is now a common practice. Radiation temperatures of the earth scene are mapped using the thermal infrared channels for a number of important applications, including:

(1) Detection of earth surface and atmospheric features on the dark side of the earth.

(2) Measurement and observation of cloud tops to identify components of weather systems and to determine cloud heights through the use of radiation temperature and ground based atmospheric sounding data. Cloud pattern or nephanalysis is used for a host of meteorological and agro-climatological applications.

(3) Measurement and observation of water surface temperatures and temperature gradient features such as ocean currents, estuarine discharge, etc. Sea surface temperature measurements are used for a host of oceanographic applications including fisheries management, ocean current monitoring, pollution tracking, and climate modeling.

(4) Surface/soil temperatures, both single pass and a diurnal pair (day/night for thermal inertia), are used to detect regions of high and low levels of soil moisture which show up as temperature anomalies.

The earth's surface and atmosphere, whose mean temperature is approximately 250 K (-18 °C), emits infrared radiation with maximum intensity at about 10 μm. At this wavelength the earth's surface and thick clouds have an emissivity close to one and thus, for most purposes, may be assumed to be nearly a blackbody radiator. Thus, the infrared radiation temperatures detected by the NOAA AVHRR can be directly related to the thermodynamic temperatures of the emitting land, water, and cloud surfaces through the laws of blackbody radiation physics.

The infrared radiation temperatures of these planetary surfaces are calculated from the infrared channel digital data (channels 3, 4, and 5) in the following manner: as noted in Chapter 2 (pgs. 21-22), digital counts are converted to radiances using the calibration coefficients (slope and intercept values in units of milliwatts/[m^2 steradian cm^{-1}] per count) to create a count versus radiance look-up table. The radiances are then converted to radiation temperature values using the inverse of Planck's equation. Planck's equation was originally established to calculate the blackbody radiance in a specific wavelength for a given temperature. Its inverse form is:

$$T_E = \frac{C_2 \nu}{\ln (1 + C_1 \nu^3/E)}$$

where T_E = temperature (K) for energy value E
ν = central wave number of channel filter (cm^{-1})
C_1 = 1.1910659 * 10^{-5} milliwatts/(m^2 steradian cm^{-4})
C_2 = 1.438833 cm K
E = radiance in milliwatts/(m^2 steradian cm^{-1})

This equation yields a temperature value for each thermal channel from the satellite derived radiances. This estimate of a surface temperature does not take into consideration the effects of surface emissivity, if less than one, or atmospheric attenuation.

The atmosphere is not completely transparent, even in the so-called "window regions" - those bands of low atmospheric absorption within the infrared region of the spectrum. Figure 5 shows cloud free atmospheric transmittance versus wavelength for six standard atmospheres, ranging from the cold dry subarctic winter to the warm and humid tropical. The window regions appear as transmittance maxima, particularly in the 3.5-3.9 μm and the 8-13 μm bands. Note here how strongly the transmittances of channels 4 and 5 vary with different atmospheres. Channel 3 does not have quite as high a transmittance at cold and dry conditions as does channel 4; however channel 3 never degrades to less than ca. 65% transmittance. Outgoing "window region" radiation from land, water, and cloud

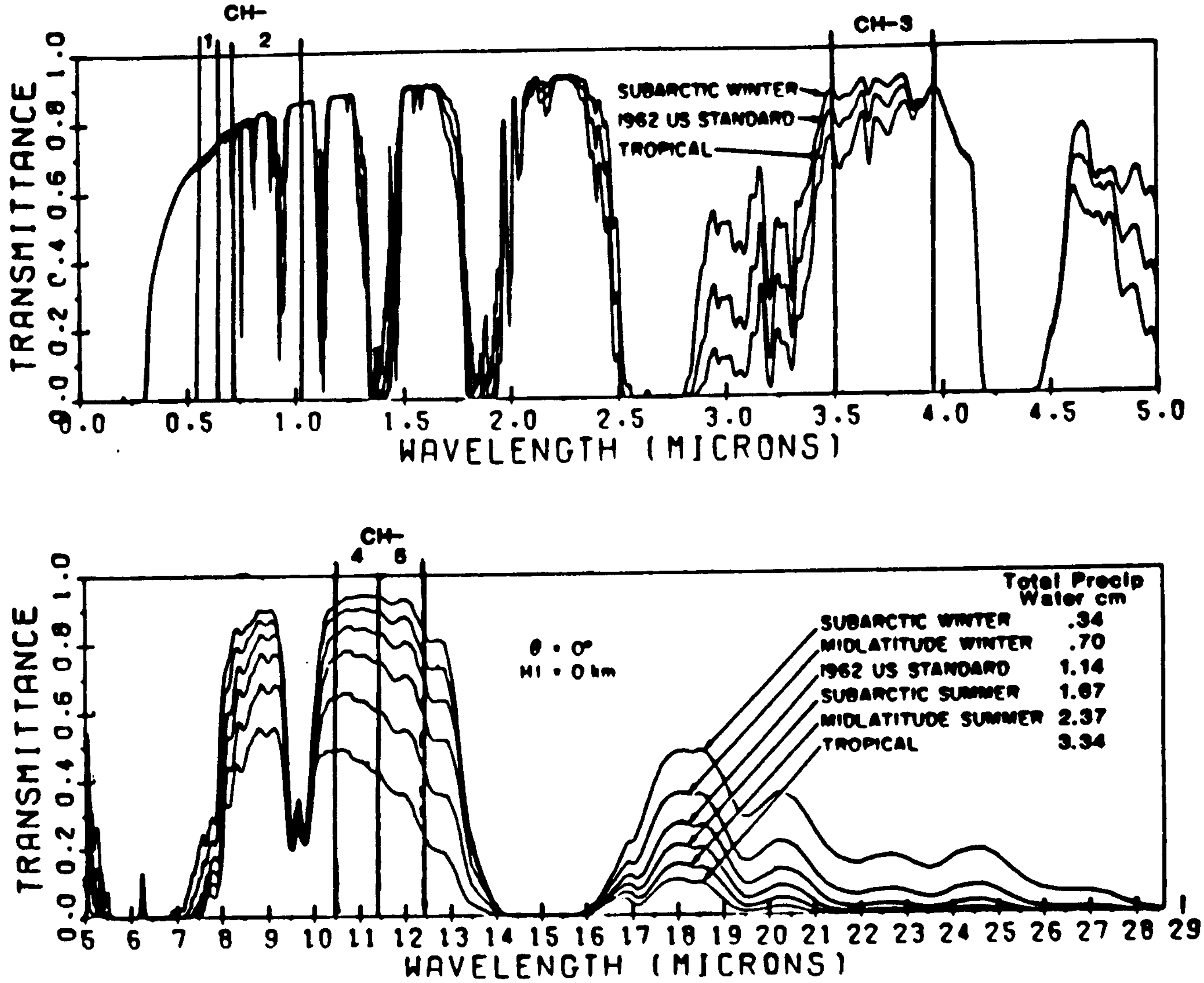


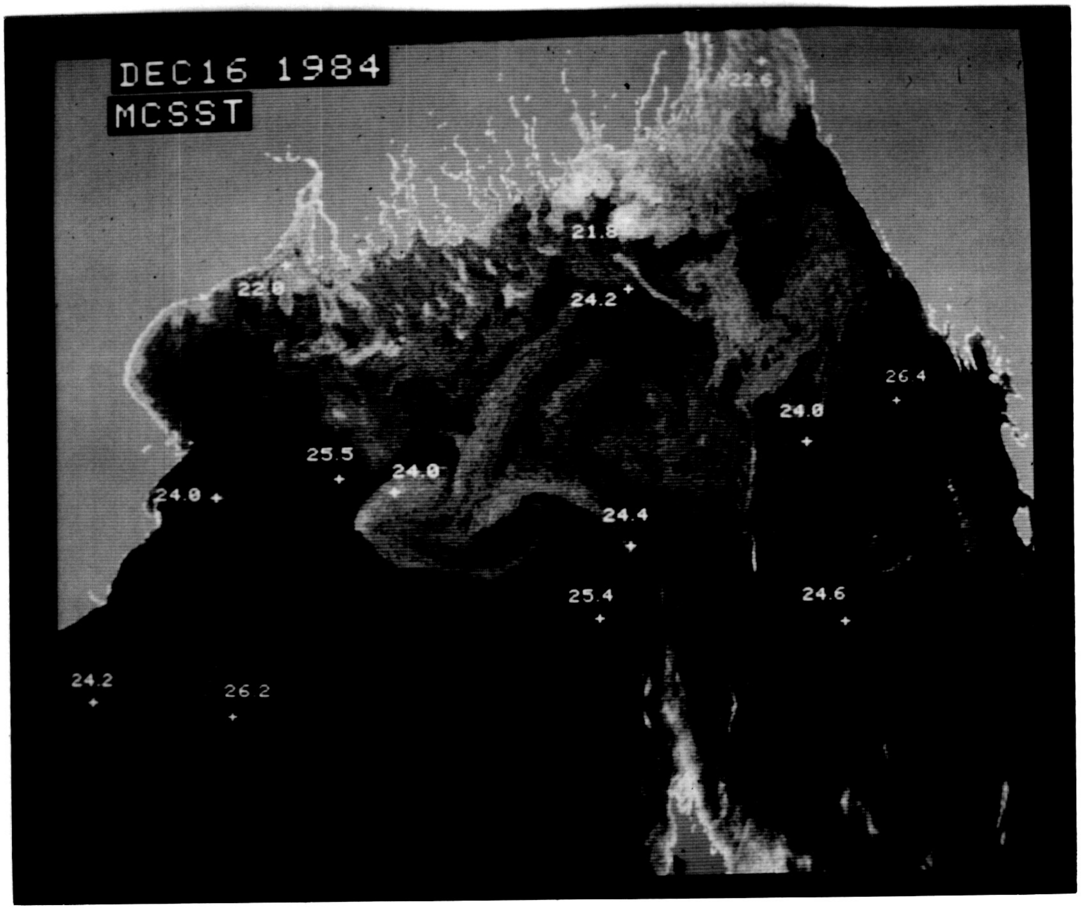
Figure 5. Atmospheric transmittance for six model atmospheres (modified from Selby and McClatchey 1975).

surfaces measured in space will be influenced by the temperature of the emitting surface, emissivity of that surface, and the composition and thermal structure of the atmosphere. Assuming an emissivity of one, the key variables are the atmosphere and thermodynamic temperature of the surface. Infrared radiation passing through the atmosphere undergoes complex absorption and reemission processes by a number of constituents - mainly the triatomic atmospheric gases (water vapor, ozone, carbon dioxide, etc.). Chief in importance among these is water vapor content, which is highly variable in time and space. It follows from these facts that slightly different surface temperature values will be provided by each of the three thermal channels. Both theoretical and experimental studies (Anding and Kauth 1970, McMillin and Crosby 1984, Prabhakara et al. 1974) have shown that the temperature differences measured in different portions of the IR spectrum (the NOAA AVHRR thermal channels) are themselves a function of atmospheric attenuation. This provides the basis for the algorithms available for determining absolute surface temperatures of the oceans. Since the NOAA AVHRR infrared channels are in window regions of the atmosphere, it follows that they differ largely due to the variably high concentration of water vapor in the lower levels of the atmosphere.

Functional uses of the thermal infrared channels and procedures for their use are provided as follows. Under conditions of cool, dry air (less than 1 cm total precipitable water in the atmosphere) channel 4 provides a superb and detailed rendition of the horizontal structure of the sea surface temperature field (Huh 1976, Huh and DiRosa 1981). The radiation temperature values in this channel are 1.5-2.0 °C below actual surface temperature and simple addition provides the best estimate of absolute sea surface temperature (Figure 6) under these conditions (Huh 1976, Maul 1983).

Use of the channel 4 data to detect surface temperature gradient features (sea or land) has certain pitfalls important to keep in mind from the outset. First of all, as atmospheric water vapor reaches levels of over 1 cm total precipitable water important things begin to occur. One insidious result is the progressive suppression of surface temperature differences (Huh et al. 1982). Radiation from a horizontally structured surface temperature field may propagate through a very warm, humid atmosphere, emerging from the top of the atmosphere as a featureless isothermal temperature field. The wetter the atmosphere the more probable is this effect.

The second complication is that atmospheric water vapor can create gradients in the radiation temperature field which do not exist on the surface. This can occur along a cloudless humidity front, a phenomenon which is rare (Huh et al. 1982). An infrared difference image of channel 4 minus



ORIGINAL PAGE
COLOR PHOTOGRAPH

Figure 6. Multichannel sea surface temperatures (MCSST). Split window algorithm calculated estimate of sea surface temperatures.

channel 5 will map areal distributions of atmospheric water vapor.

A third complication is cloud exclusion, both low altitude cloud masses at nearly the same temperature as the surface and clouds with areas less than the spatial resolution of the pixel. In the daytime, high albedo values (channels 1 and 2) from cloud and fog banks can be used to avoid incorrect interpretation; at night, however, without albedo data, the problem is greater. Temperature, structure, texture, and time stability characteristics of the infrared temperature field are helpful to correctly discern surface temperature features from clouds. Except for the lowest level clouds, cloud top temperatures are markedly lower than surface temperatures. A multimodal temperature distribution is an important clue to cloud contamination. In some circumstances, simple temperature thresholding can discriminate clouds from surface features. Low altitude cumulus clouds tend to be broken with a distinctive spotty texture, whereas ocean temperature fields typically show fluid flow patterns. The spatial structure is an important clue to the presence of low, warm stratus clouds covering the earth's surface. Although similar in temperature to surface features, they mask major features such as coastlines and rivers, and recur in locally typical patterns. The temporal stability of atmospheric features is generally much less than oceanic features, so repeated passes will show changing atmospheric and relatively stable surface patterns. These are clues to the art of infrared image interpretation, and practical experience is necessary for consistently successful performance.

Under the more humid conditions characteristic of the open ocean, channel 3 has been observed to provide the best detection of the horizontal structure of the sea surface temperature field (Van Woert 1982). It is also best for detecting unusually hot features of the earth's surface, such as fires, volcanic terrains, etc. (Matson and Dozier 1981). For absolute values of sea surface temperature, the different temperature values in the different channels are used to calculate an estimate. A number of regression algorithms have been developed by various workers using simultaneous surface and satellite measurement combinations. These algorithms have been empirically derived by regressing radiation temperatures and channel temperature differences against the surface measurements. The resulting multichannel sea surface temperature algorithms are of two kinds: the split window and the dual window. The split window refers to the 10.5-11.5 and 11.5-12.5 μm channels in the 10-13 μm window region (Figure 5). The dual window refers to the use of a channel in the 3.7 μm atmospheric window and another in the 10-13 μm window. It has been found empirically useful to use separate algorithms for daytime and nighttime temperature retrievals. The two practical problems in making these measurements are: (1) excluding clouded or cloud contaminated pixels from consideration; and (2) the fact that there are several special problems with channel 3 data. These problems are: poor data due to excessive noise levels; and contamination due to reflected solar radiation (the sun emits a large amount of energy in this particular portion of the IR spectrum). These problems have been overcome by using data from the short periods of good sensor performance; by filtering or averaging of the noisy data; and by either using nighttime passes only or daytime channel 3 data in the anti-sun side of the satellite subtrack (Bernstein 1982, Lynn and Svejkovsky 1984, Van Woert 1982). Channel 5 data have not been available on TIROS-N or NOAA 6, 8, and 9 satellites, so algorithms using this channel have had more limited applications. Table 3 presents algorithms for calculating sea surface temperature. These have been developed by various workers and tested for the specified regions. Since the atmospheric effect is airmass dependent, the user should calibrate and recalculate regression algorithms for the particular region and conditions of his or her work.

The bulk of atmospheric water vapor is in the lowest levels of the atmosphere, particularly the marine boundary layer over oceanic and coastal areas. Thus cloud top emitted radiation is subject to less atmospheric attenuation than sea or land surface radiation. The uncorrected channel 4 temperatures are used in combination with an atmospheric temperature profile to estimate the height of a cloud

Table 3. Algorithms for calculating sea surface temperatures. (1)

Algorithm	Note
$1.0346\ T_4 + 2.58\ (T_4 - T_5) - 283.2$	(2)
$1.0170\ T_4 + 0.97\ (T_3 - T_5) - 276.58$	(3)
$1.3826\ T_3 - 0.31\ T_4 + 1.72$	(4)

(1) T is the radiation temperature of either channel 3, 4, or 5.
(2) Daytime sea surface temperature, in K, based on global buoy data (Strong and McClain 1984).
(3) Nighttime sea surface temperature, in K, based on global buoy data (Strong and McClain 1984).
(4) Sea surface temperature, in °C, based on northern Pacific ship data (Bernstein 1982).

through the use of a temperature versus altitude (skew-t) plot.

Land surface temperatures are not only made through the bulk of the atmosphere (except for mountain regions), but have large compositional differences from place to place that yield emissivities of between 0.5 and 0.9. Emissivities of agricultural and vegetation surfaces have been found to range between 0.98 and 1.0 (Sutherland and Bartholic 1977). Temperatures of most continental regions cycle through a wide range of values during the solar diurnal cycle. Regression analyses of satellite estimated land temperatures (single channel in the 10-13 µm range) versus temperature measured at agricultural weather instrument shelters have found errors of 1-2 °C, which is adequate for agricultural applications.

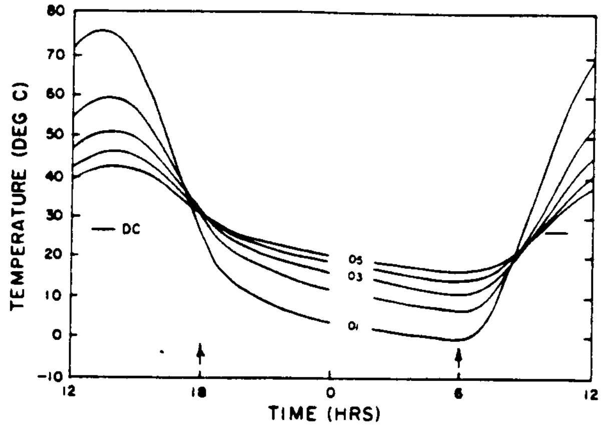
Repeated satellite infrared radiometry and the diurnal land surface temperature cycle lends itself to another kind of analysis, that of thermal inertia (Goddard Space Flight Center 1980, Short and Stuart 1982). Thermal inertia is the resistance of a material (or surface) to temperature change during the full 24 hour solar heating/nocturnal cooling cycle. More exactly, the thermal inertia, P, is defined as:

$$P = (K\ C\ \rho)^{1/2}$$

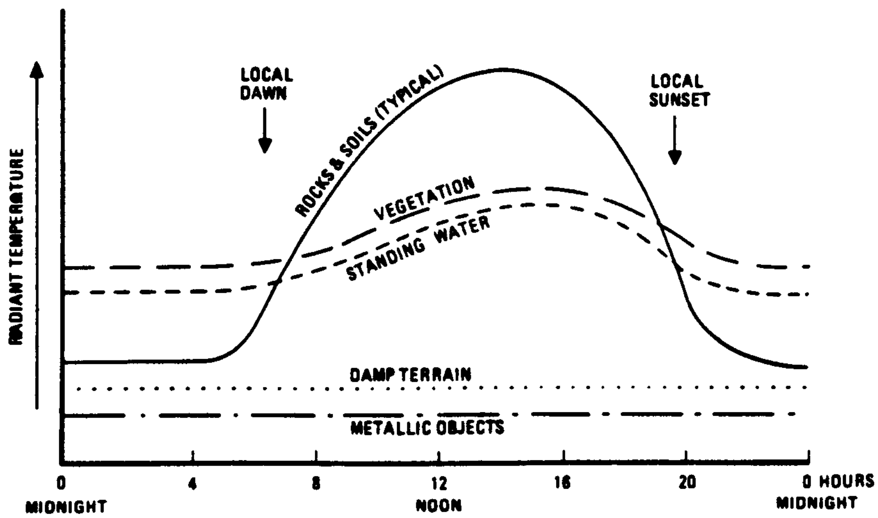
where K = thermal conductivity (watt/[m K])
C = specific heat of surface materials (J/[kg K])
ρ = density of surface materials (kg/m^3)

This equation yields thermal inertia in units of watt $sec^{1/2}/(m^2\ K)$. Some typical values are 0.01-0.05 for soils and 0.05-0.10 for rock bodies in the earth's crust. The variation in soils is largely due to variations in moisture content, whereas that of rock is due to differences in thermal conductivity of the mineral assemblage.

Thermal inertia is a measure of the heat transfer rate across the land surface. Those surfaces with high P values resist temperature fluctuations at the surface boundary. They show less temperature variation per solar diurnal cycle than surfaces with a lower thermal inertia. Solar diurnal radiation temperatures of natural surfaces and temperature cycles for various values of thermal inertia are shown in Figure 7.



(a)



(b)

Figure 7. Thermal inertia. a) Variations of surface temperatures during a 24 hour period relative to different thermal inertia values (Source: Watson, K. Geologic Applications of Thermal Infrared Images. Proceedings of the IEEE 63(1): 128-137. Copyright © 1975 IEEE). b) Changes in radiant temperatures of five surface cover types during a 24 hour thermal cycle (Source: Sabins, F. F., Jr. Remote Sensing Principles and Interpretation. W. H. Freeman and Company. Copyright © 1978).

A true thermal inertia is not available from NOAA satellite data due to the fact that satellite radiometric temperatures are not acquired at the exact times necessary and also since atmospheric effects cannot be completely excluded. An apparent thermal inertia (ATI) that has many of the important properties of true thermal inertia (i.e., surface material discrimination capabilities) can be calculated from satellite data. The ATI is calculated using the visible and the thermal infrared data from a daytime pass and the thermal infrared from the nighttime pass.

The two thermal infrared images are acquired, one near the temperature maximum and one near the temperature minimum of the day. If these images are geometrically registered to each other and subtracted from each other (daytime minus nighttime) a temperature difference image results. It will show those areas which have changed the most (lowest thermal inertia) versus those which have changed temperature the least (high thermal inertia). These temperature difference values can be used to calculate the ATI with the following equation:

$$ATI = N\,C\,(1 - a) / \Delta T$$

where ATI = apparent thermal inertia
N = integer scaling factor to bring results into an 8-bit range (0-255)
C = solar irradiation measure, described below
a = apparent albedo, described below
ΔT = temperature difference between nighttime and daytime pass

C is a measure of the solar irradiation on a spherical earth, and is equal to the following:

$$\sin\theta \sin\phi(1 - \tan^2\theta \tan^2\phi)^{1/2} + \cos\theta \cos\phi \arccos(-\tan\theta \tan\phi)$$

where θ = earth latitude (-90° to +90°)
ϕ = solar declination (-23.5° to +23.5°; see Goddard Space Flight Center [1980] for table of values)

The apparent albedo, a, is obtained from the daytime channel 1 reflectivity measurement and is defined as:

$$a = K\, r \,/\, (\sin\theta \sin\phi + \cos\theta \cos\phi \cos\delta)$$

where $K = (1 + 0.0167 \sin(2\pi(\text{Julian Day} - 93.5) / 365))^2$
r = reflectivity
δ = local hour angle of the sun (this only applies with a spherical earth, i.e., no relief)

The ATI measure has been shown to discriminate between rock and soil types, soil moisture levels, regions of recent rainfall, and vegetation vitality. Fortunately the afternoon orbit of the NOAA satellite (ca. 1500/0300 LST) nearly coincides with the temperature maximum and minimums of the diurnal cycle, making this approach feasible with these satellites. Although thermal inertia is a valuable tool for the analyst, there are several important problems to be considered. These include the difficulty of obtaining two clear sky passes in a row and also the fact that diurnal changes in the temperature and humidity structure of a humid atmosphere can more radically alter radiation temperatures than the actual surface temperature changes. The important complicating factors in interpretation of the thermal imagery and use of thermal inertia are:

(1) Clouds have high albedo and reflect solar energy back to space during the daytime. However, they do trap earth emitted radiation near the surface both day and night.

(2) Surface winds strip heat from the earth's surface (sensible and latent) by turbulent convection, resulting in a lower surface temperature than exists under windless conditions.

(3) Evaporation results in a lower temperature of a moist surface. Vegetation creates the same effect, and vegetation tends to hide the soil/rock surface from radiometric measurement.

(4) Absorption and reemission of earth emitted radiation by atmospheric water vapor and aerosols cause a discrepancy between radiometric measurements made by satellite and the equivalent fluxes observed at ground level.

In addition, surface emissivity, directional characteristics of reflecting materials, topography, slope orientation, subpixel structure of the surface temperature field, and prior temperature and humidity history of the observed material all affect radiometric data. The apparent thermal inertia technique should be considered, however, and used in combination with other data sources where possible.

In summary, the thermal infrared sensors can be used to detect surface temperature patterns, calculate an estimate of absolute surface temperature, estimate cloud heights, and to calculate areal variations in apparent thermal inertia which is related to important bulk properties of the land surface.

ORIGINAL PAGE
BLACK AND WHITE PHOTOGRAPH

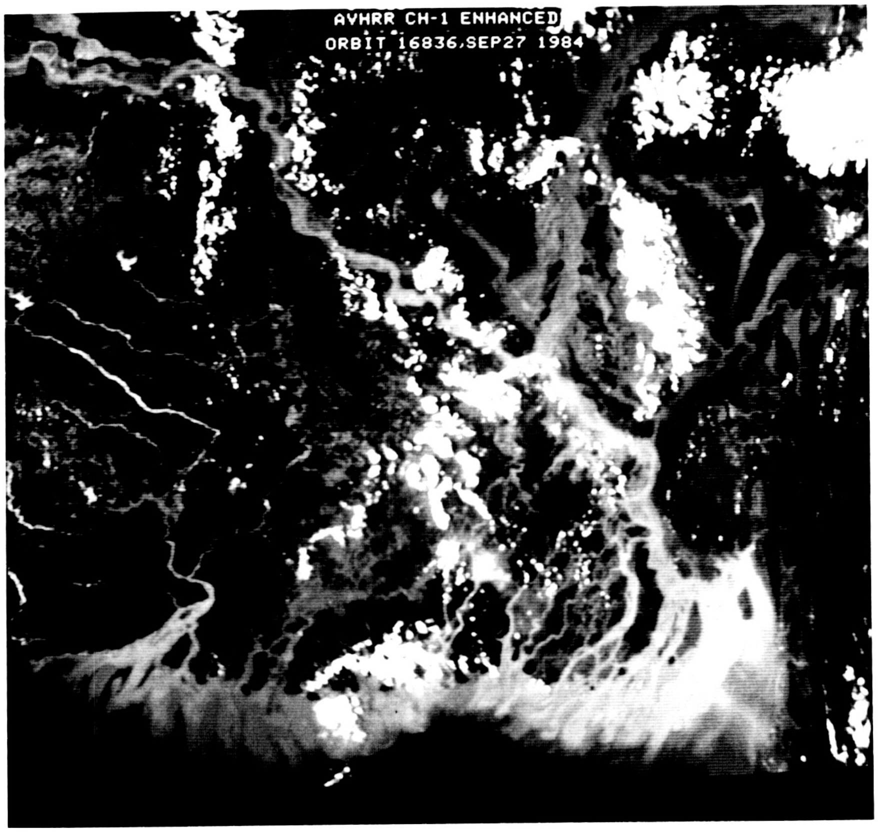


Figure 10. Turbidity of surface waters (image processed by A. Ali and D. A. Quadir, SPARRSO, Bangladesh).

Land/Water Boundary Mapping

Land/water boundary mapping has application in detecting changes of coastline, river channels, and areas covered by standing water. Water covered regions may have high reflectances in the visible portions of the spectrum, particularly if they are highly turbid or if the shallow sea floor is highly reflective. In these cases they can be difficult to separate from other surface features. In the NIR, however, water is a strong absorber and these wavelengths are completely absorbed within the upper 2-10 cm of the surface (Figure 8). Herein lies the principle behind land/water boundary detection with the NOAA AVHRR: the albedo difference between land and water is so great in channel 2 radiances that careful enhancement normally creates a strong and clear land/water separation. This channel is also less affected by atmospheric water vapor and so this separation is often useful when the channel 1 data are degraded by the atmosphere. Only the denser portions of clouds show up in the channel 2 data. Use of NIR (channel 2), therefore, ordinarily provides an excellent separation of land and water along coastlines, lakes, and other bodies of water. An exception to this is under conditions of extreme turbidity; surface waters will then show up with relatively high reflectances in the NIR as well. Often streamlines will show up with the NIR since high levels of turbidity do vary considerably from one portion of a water body to another.

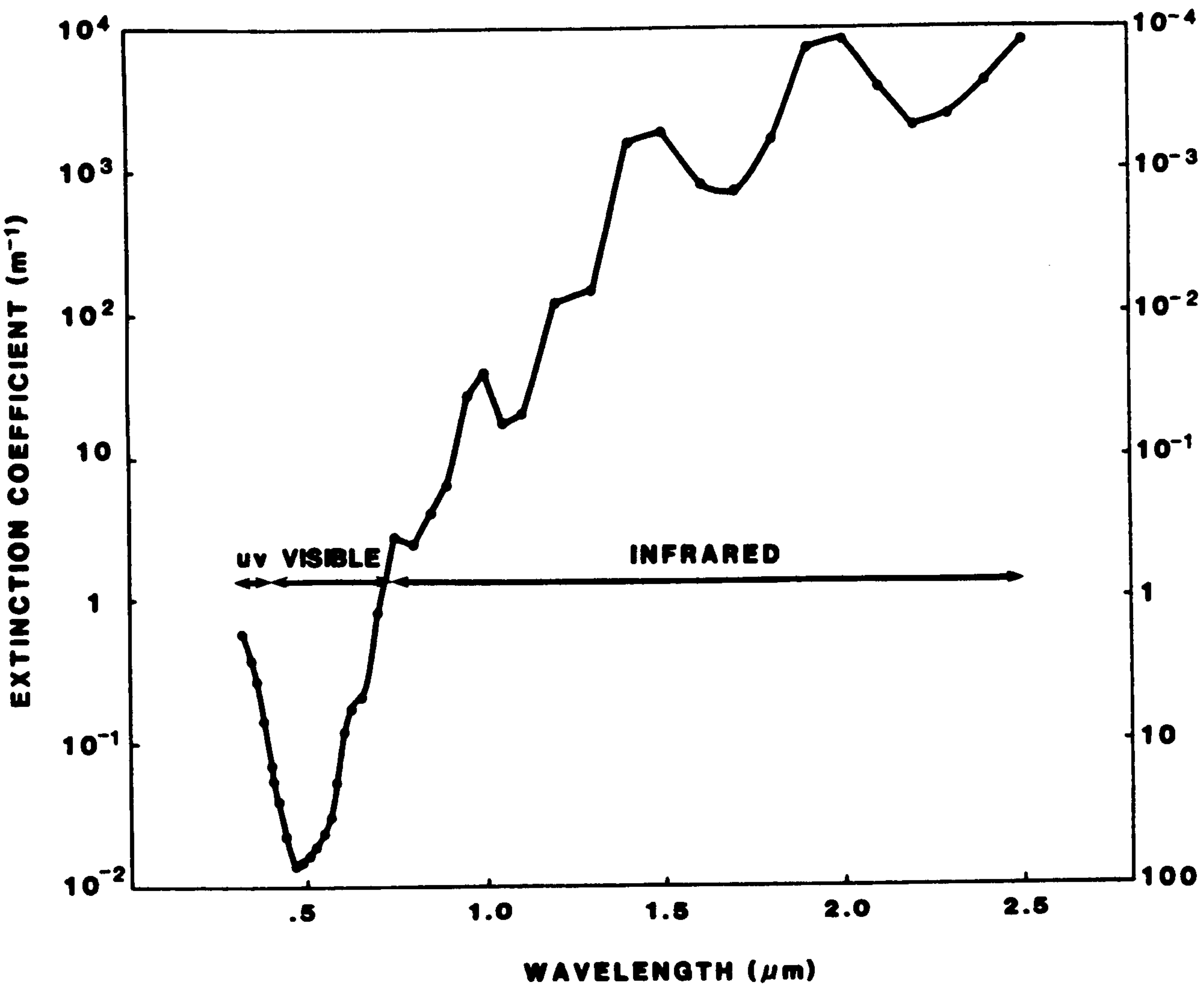


Figure 8. Transparency of water with respect to ultraviolet, visible, and near infrared solar radiation.

The land/water edge separation procedure therefore entails the careful enhancment of AVHRR channel 2 data, in which the reflective brightness values are clearly distinct (Figure 9). Ambiguity will only occur when waters are extraordinarily turbid and clues to this condition will be detectable in the other channels. The section on thresholding (pgs. 57, 176) discusses the use of this technique on the I^2S system.

ORIGINAL PAGE
BLACK AND WHITE PHOTOGRAPH

Figure 9. Land/water edge detection (image processed by A. Ali and D. A. Quadir, SPARRSO, Bangladesh).

Mapping Turbidity in Riverine, Estuarine, and Oceanic Waters

Suspended sediment content in river, estuarine, and ocean surface waters strongly affects the albedo of these bodies. Patterns of high and low albedo and suspended sediment content are useful to detect areas of excessive erosion, water pollution, bottom sediment resuspension, and the fluid flow patterns indicative of water movement. Water quality studies have been conducted primarily using Landsat MSS bands 4 (0.5-0.6 μm) and 5 (0.6-0.7 μm). Only the blue and green wavelengths (0.4-0.6 μm) penetrate below 20 m, whereas red and NIR are absorbed in the upper meter or few centimeters, respectively. Radiation in the range of NOAA AVHRR channel 1 penetrates to a depth of 3-6 m (Schneider et al. 1981). Studies of NOAA AVHRR data over shallow waters have shown that shallow, turbid bodies of water have the greatest channel 1 albedo and a large difference between channel 1 and channel 2. Clear, deep bodies of water have low radiances in channels 1 and 2 and small differences between channel 1 and 2. Ocean and fresh waters over a hundred meters deep and free of suspended sediment particles absorb almost all solar radiation except weak, diffuse backscattered (Rayleigh) blue light. Surface waters thus become increasingly reflective with increased concentrations of sediment and/or plankton. Water body reflection is also influenced by surface conditions, with wind waves increasing reflectance of solar radiation. These same visible wavelengths which are so sensitive to suspended sediment content are also affected by the aerosol content of the atmosphere. With shallow water bodies, reflectance of the bottom also contributes to reflected solar radiation. Because of these difficulties, the Coastal Zone Color Scanner on the Nimbus 7 satellite was designed with special spectral bands and an anti-sun tilting sensor to detect suspended sediment and chlorophyll-a.

In summary, the NOAA AVHRR channel 1 is used for detecting surface waters rich in suspended sediment (Figure 10). Channel 2 will detect only the highest concentrations of suspended sediment near and at the surface (Figure 11). This is because solar radiation in this wavelength is completely absorbed in the upper 10 cm of the water column. It is merely a question of careful interactive enhancement of channel 1 and channel 2 data to discern high albedo patterns (largely suspended sediment) and low albedo patterns in surface water bodies. Quantitative use of the NOAA AVHRR for this purpose is not yet totally feasible nor well established, but important and valuable observations can be made through semiquantitative study of these data.

ORIGINAL PAGE
BLACK AND WHITE PHOTOGRAPH

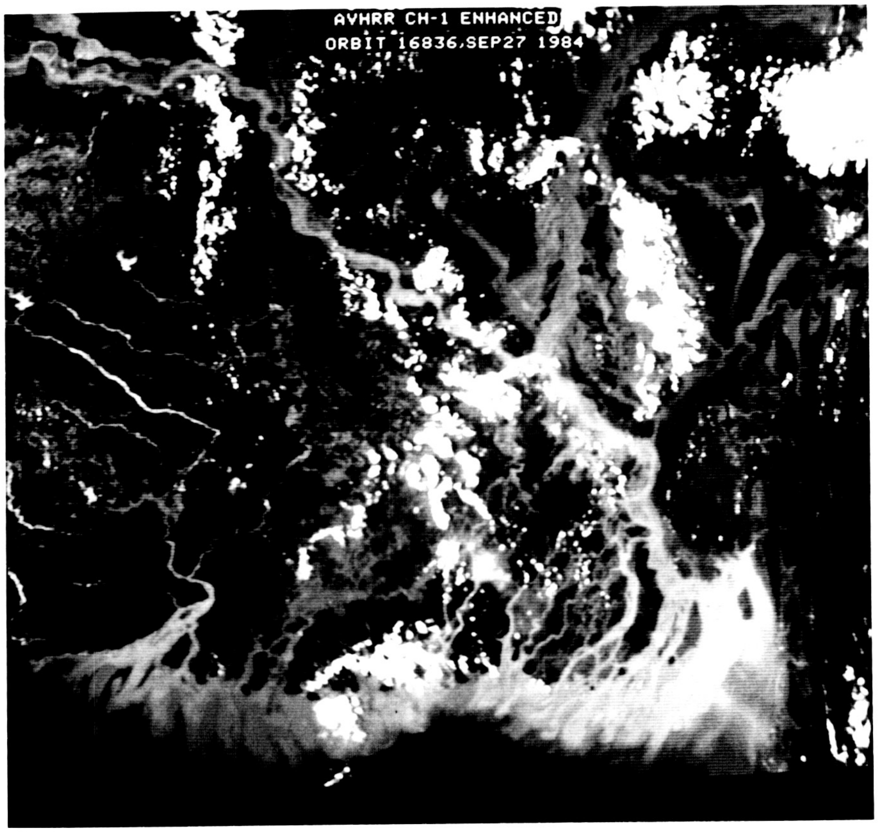


Figure 10. Turbidity of surface waters (image processed by A. Ali and D. A. Quadir, SPARRSO, Bangladesh).

ORIGINAL PAGE
BLACK AND WHITE PHOTOGRAPH

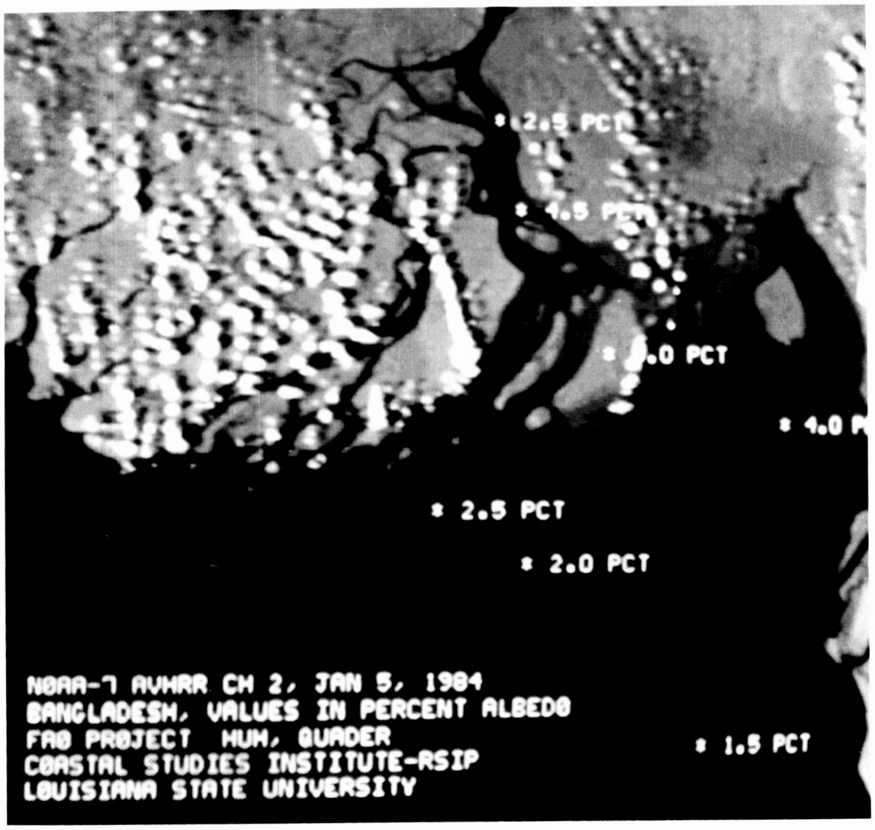


Figure 11. Albedo of highly turbid surface waters.

Snowfield Mapping

The satellite monitoring of snow cover is an example of how rapid regional synoptic surveys by satellite are significantly more cost effective and technically feasible than any other approach. Data on snow cover and its changes are needed for a host of applications, including hydrologic studies, water storage estimates, water supply forecasts, runoff predictions, radiation balance calculations, soil moisture forecasts, and hydroelectric power budgeting.

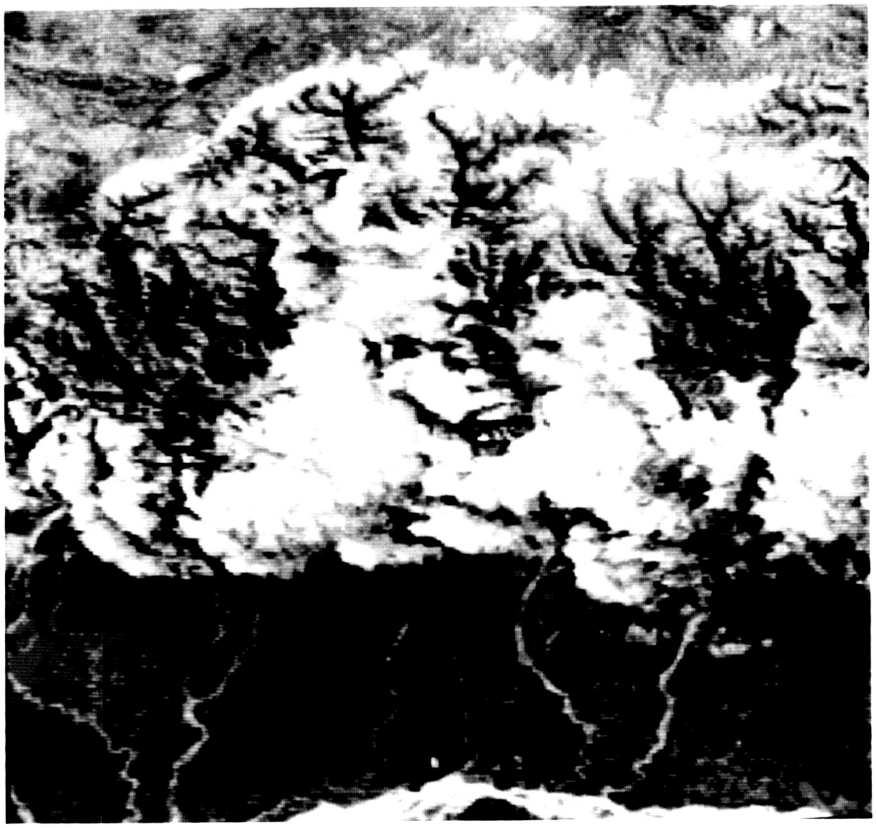
Schneider et al. (1981) found that areal snow cover measurements are best made with the visible sensor (channel 1). The principle behind snow field monitoring is simply that snow has much higher albedo in the visible portion of the spectrum than foliage, rock, or soil. Thus careful enhancement will create a snow/non-snow image. The NIR (channel 2) may possibly provide additional useful information on the condition of the snowpack. This suggests that simple enhancement will separate snow from non-snow covered regions. Under ideal conditions this is true, but in the real world a number of practical difficulties arise. First, clouds have a similar high albedo and at present there is no completely reliable method of separating them. An indirect method such as monitoring rapid time changes can be used but this is not totally satisfactory. Secondly, certain non-snow features of the earth surface (e.g., salt flats or white rock types) may also have similar albedo values. At present, operational snow mapping is done with the NOAA AVHRR in the U.S. and several other nations. Of course it is not possible to remotely acquire any measure of snow depth, so only areal coverage is possible.

Although originally attempted with Landsat, the NOAA AVHRR has proven to be a significantly superior snow monitoring system. First off, Landsat sensors were not designed to make measurements of such high albedo features. Bands 4 (0.5-0.6 μm) and 7 (0.8-1.1 μm) of Landsat saturates at 36 and 48% albedo, levels that result in complete saturation over snow. The NOAA AVHRR sensor was designed for meteorological applications and has saturation values in channels 1 and 2 of 79.1 and 93.1% (ITT 1980). These saturation levels allow NOAA AVHRR to usefully monitor snow fields. The 9-18 day repeat cycle of Landsat (versus once or twice daily with NOAA AVHRR) is also a problem, since there may not be enough passes (particularly when clouded out). The third difficulty of using Landsat in this application is the relatively small area coverage per tape, often requiring several tapes (raising the cost) on a short response time to analyse a single drainage basin.

The use of NOAA AVHRR to detect snow cover varies with terrain type and plant canopy. In an alpine region where erosional topography is very distinct, the high albedo regions conform to topography with a clear cut snowline. In low relief regions where there is no altitude snowline, albedo changes as the snow becomes patchy and underlying surfaces affect the measurements. For example, Norwegian engineers found that when spring melting of the snow reduces cover by about 20%, changes in satellite measured albedo was capable of yielding valuable meltwater data (Ostrem et al. 1981). In the shallow snow pack of the grassy plains of North America it was observed that the greatest difference between channels 1 and 2 occurs over the brightest (deepest) snow cover (Schneider et al. 1981). But as snow cover thinned and grasses influenced readings, channel 1 and 2 values converged.

Studies at LSU on 8-bit NOAA AVHRR data of Himalayan snow fields have shown that channel 1 is brighter over obvious snow regions than channel 2 by 8 to 10 counts (Figure 12); this difference diminishes to nearly zero over clouds. A difference image works well for snow detection, except for the sunglint side of the cumulus clouds where values closely approach those of snow. This observation in a single case may be strictly a function of the snow conditions on that particular day, but such inquiry should continue.

Operational use of NOAA AVHRR data for snow field monitoring has been underway since the early 1960's in the U.S., using lower resolution satellite sensors. Use of the 1.1 km resolution multispectral NOAA AVHRR data has promise that is being actively explored and developed.



ORIGINAL PAGE
BLACK AND WHITE PHOTOGRAPH

Figure 12. Image enhancement for snowfield patterns (image processed by A. Ali and D. A. Quadir, SPARRSO, Bangladesh). Cloud contamination in central portion of scene has not been removed (see text).

Vegetation Index

The NOAA AVHRR scanner can be used to detect the condition of natural vegetation and crops by measuring "greenness" of the earth scene. Such maps are used for agricultural monitoring, crop yield modeling, and for early warning of crop stress (Tarpley et al. 1984). The principle behind radiometric vegetation monitoring is that chlorophyll in green leaves is highly absorptive (reflectance less than 20%) in the visible band (0.5-0.7 μm), but is highly reflective (about 60%) in the NIR (0.7-1.3 μm) range (Hoffer and Johannsen 1969). This differential reflectance has been widely used with Landsat MSS data to estimate crop acreage, detect stress, and classify land cover (Myers 1983, Tucker 1978, Tucker 1979). NOAA AVHRR channels 1 and 2 are similar to bands 5 and 7 of the Landsat MSS. Various mathematical combinations of these radiometric measurements have been found to be sensitive indicators of vegetation greenness and as such are referred to as vegetation indices (Tarpley et al. 1984).

The two most commonly used indices are the simple vegetation index and the normalized vegetation index. The simple vegetation index is calculated according to the following equation:

$$VI = CH_2 - CH_1$$

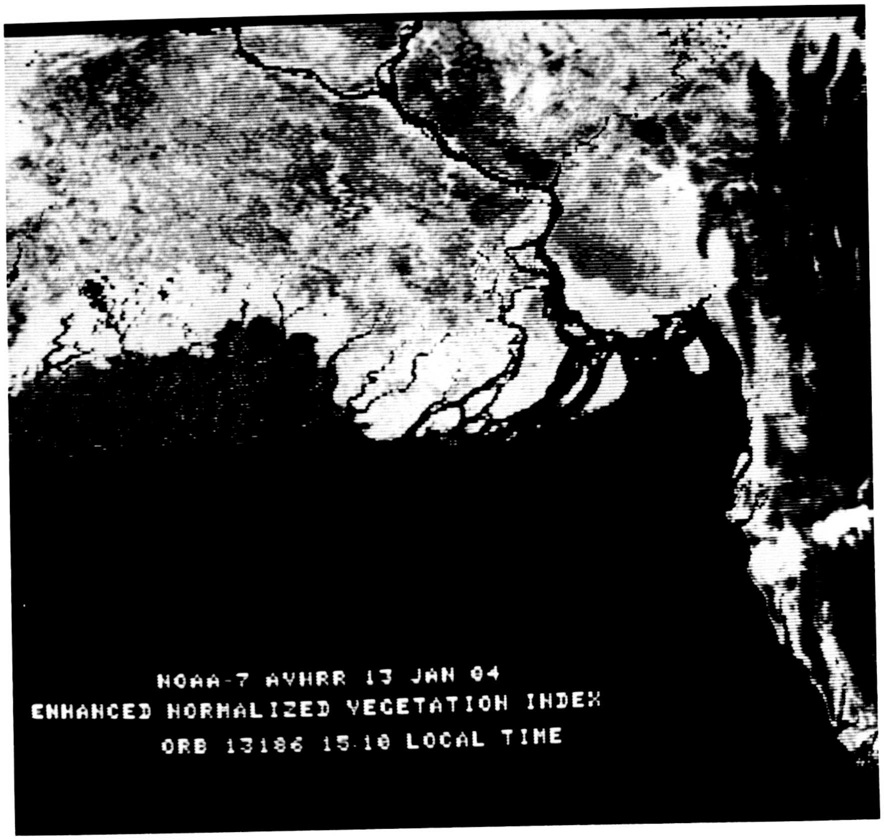
The equation for the normalized vegetation index is:

$$NVI = \frac{CH_2 - CH_1}{CH_2 + CH_1}$$

These indices are calculated using data in the form of the unitless 8-bit integer values that are convertible into albedo values, or the albedo values themselves. The original 10-bit data are normally truncated to 8-bit precision and mapped for calculation of the VI or NVI. Both resulting images are arbitrarily scaled to 8-bit precision for display.

Of the two indices, the normalized vegetation index (Figure 13) is preferred for vegetation monitoring, since it partially compensates for changing scene illumination, surface slopes, and viewing aspects. In the earth scene, clouds, water, and snow have larger reflectances in the visible than in the NIR, and so the NVI becomes negative. Rock and soil have nearly the same reflectances in both parts of the spectrum and so they have vegetation indices near zero. Vegetation has NVI values ranging between 0.1-0.6, as calculated by NOAA (Tarpley 1983). Higher values represent increased chlorophyll density.

The radiometric values are influenced by atmospheric effects and scanning geometry. Rayleigh (wavelength dependent) scattering by dust, aerosols, and water droplets or ice crystals as well as subpixel sized clouds all have the effect of increasing the channel 1 radiances relative



ORIGINAL PAGE
BLACK AND WHITE PHOTOGRAPH

Figure 13. Normalized vegetation index (image processed by A. Ali and D. A. Quadir, SPARRSO, Bangladesh).

to channel 2. These effects reduce the measured vegetation index. As the radiometer scans away from nadir, it views the earth through greater than one atmospheric thickness and, thus, increases the influence of the atmosphere. That portion of the scan line in the direction of the sun also receives more "airlight" (backscattered shortwave solar radiation) in the channel 1 signal. It is important to realize that neither channel 1 nor channel 2 have inflight calibration (pgs. 19-20); thus sensor drift over time may also affect vegetation indices. Thus, changes in atmospheric influence, viewing geometry, and sensor calibration must all be considered in a rigorous analysis of vegetation greenness. The normalized vegetation index based on 1.1 km NOAA AVHRR data is a valuable tool for monitoring spatial and temporal changes in patterns of vegetation greenness. Such qualitative and semiquantitative data are extremely valuable in the development of the art of satellite monitoring of crops and vegetation condition. Observers and analysts will be able to provide more skilled interpretations with time and experience.

Image Classification

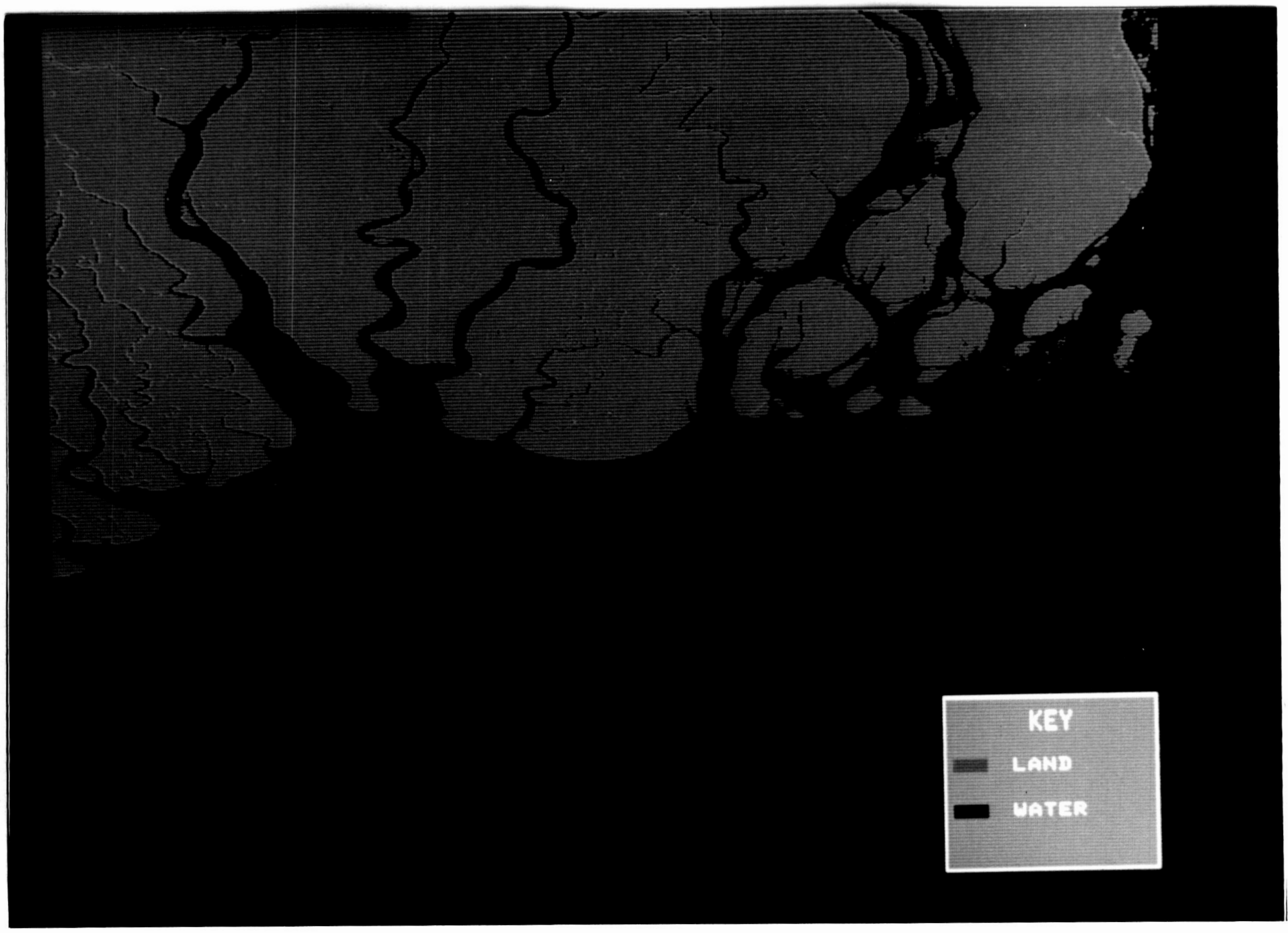
For many resource inventory applications, it is necessary to convert a raw data image (e.g., one containing reflectance values) into a classified image where the pixel values refer to land categories. Three types of classification techniques will be discussed in this section: thresholding, unsupervised classification, and supervised classification.

Thresholding

It is often possible to separate different land classes based solely upon their spectral values. For example, land and water are spectrally separable in the near infrared since water has very low values in these wavelengths while land values are usually much higher (pgs. 41-43). Thermal data can also be used to separate land and water since water cools and warms more slowly than land (pgs. 35-36). When different land classes are spectrally separable, thresholding can be performed to produce a classified image. The BINARY'IMAGE function (Figure 14) is useful for performing this kind of classification. The use of this function is discussed in Chapter 8 (pg. 176).

Unsupervised Classification

Unsupervised classifiers perform land classifications according to some statistical criterion such as clustering. These techniques require no human intervention. This



ORIGINAL PAGE
COLOR PHOTOGRAPH

Figure 14. Binary image illustrating thresholding (Landsat MSS data).

contrasts with supervised classification, where the user selects homogeneous areas representative of the different land types for the computer to "train" on. A supervised classifier operates by comparing pixel values with the values of the different training areas. Whereas the supervised classifier possesses a priori knowledge of the statistics of the different land types, an unsupervised classifier does not have this information and can only look for statistical trends.

The I^2S function used for unsupervised classifications is CLUSTER. This program executes quickly and can often produce output superior to the supervised classifiers (Figure 15). Thus the user should experiment with this function before trying a supervised classification. The CLUSTER function is discussed in Chapter 8 (pgs. 177-178).

Supervised Classification

As was discussed previously, during a supervised classification the user prepares training fields from which class statistics are generated. These statistics are then used to classify the image. A supervised classification consists of three steps on the I^2S system: preparing the training fields with the TRAIN function, generating the training field statistics with the PREPARE function, and classifying the image either with CLASSIFY (Figure 16) or C'CLASSIFY. These functions are discussed in detail in Chapter 8 (pgs. 179-182).

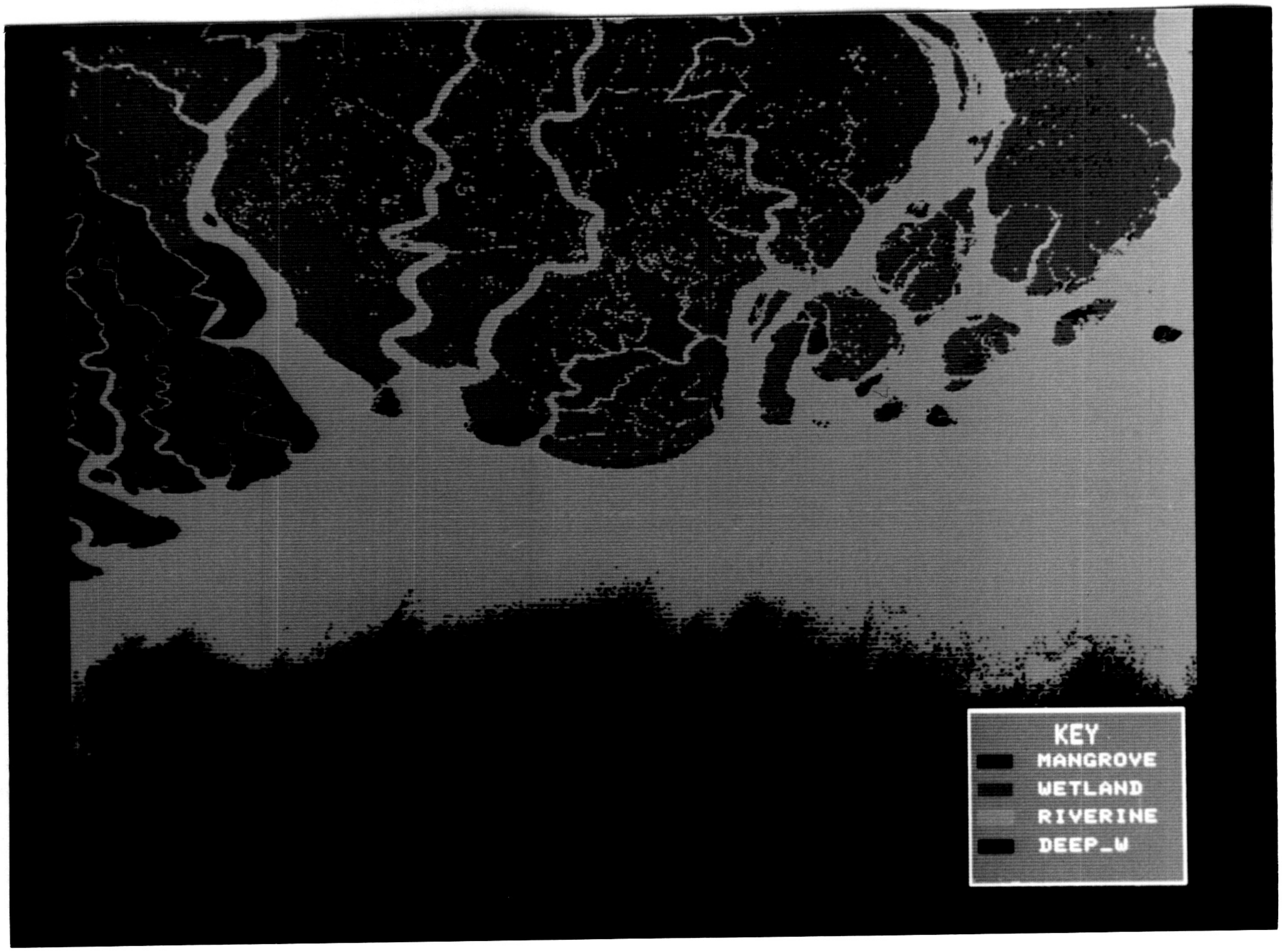


Figure 15. Unsupervised classification using CLUSTER (Landsat MSS data).

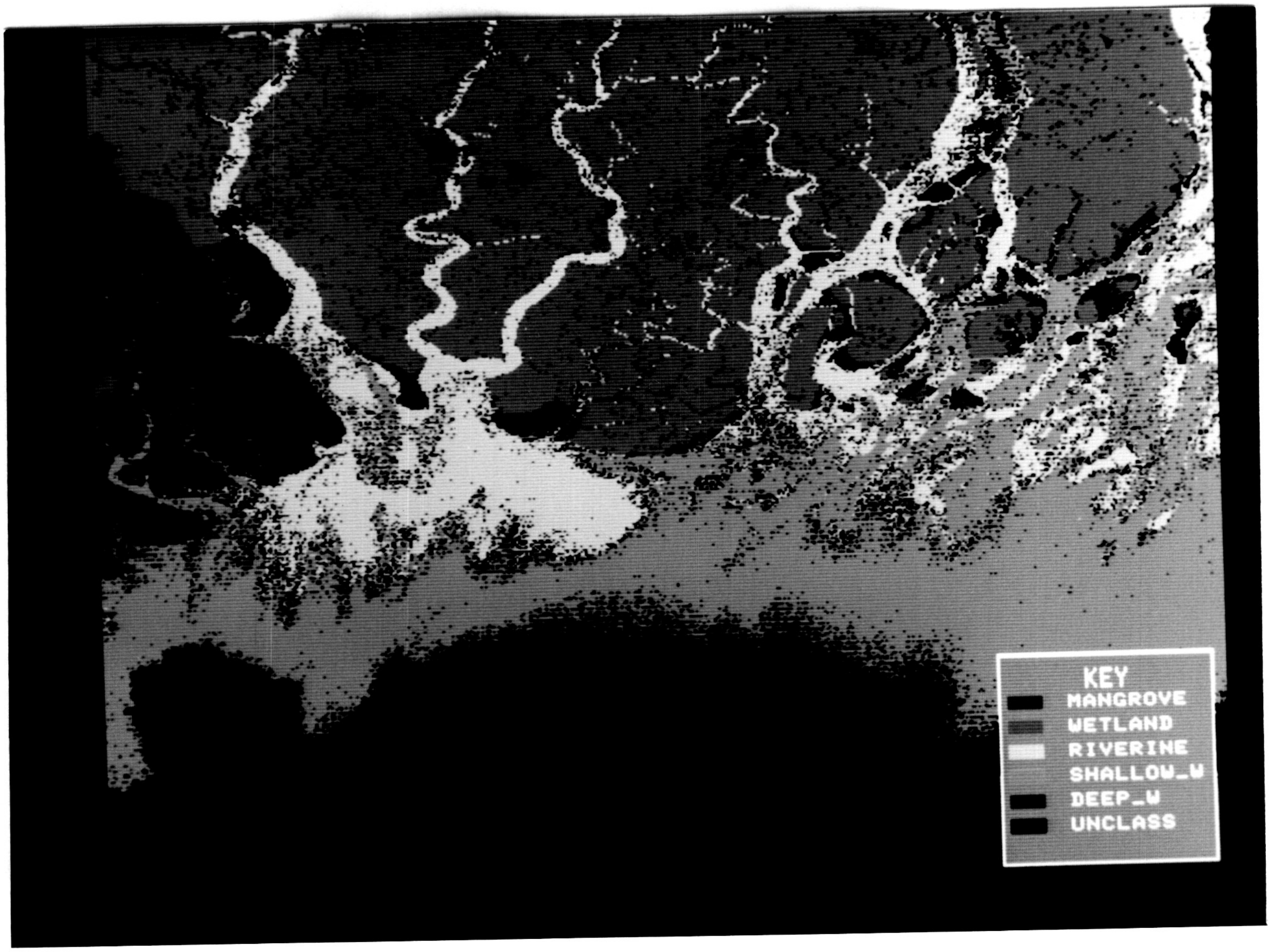


Figure 16. Supervised classification using CLASSIFY (Landsat MSS data). Training regions for this image are shown in Figure 38.

Specialized Data Bases

The World Data Bank II

Satellite mapping of the earth's surface and its change through time requires comparisons between successive images and between images and a standard base or reference chart. A digital file of world geography, the World Data Bank (WDB) II, and portions of the Cartographic Automatic Mapping Program (CAM) have been adapted to the I^2S system to provide computer cartographic capabilities. This section of the User's Guide will briefly summarize these capabilites. The interested reader should refer to the references provided with these systems (CIA 1977a, 1977b).

WDB II is a digital representation of world geographic features that is divided into five volumes by geographic area: North America (Volume 1), South America (Volume 2), Europe (Volume 3), Africa (Volume 4), and Asia (Volume 5). For each area the following data are included: coastlines, islands, lakes, rivers, and international boundaries. The WDB II maps were digitized at scales ranging from 1:1,000,000 to 1:4,000,000, with a nominal scale of 1:3,000,000. Figure 17 contains sample output from the World Data Bank II for Bangladesh and surrounding regions.

The MAP function was adapted from the CAM program to provide access to the WDB II on the I^2S system. This program can plot geographic data in any one of 5 standard

ORIGINAL PAGE
COLOR PHOTOGRAPH

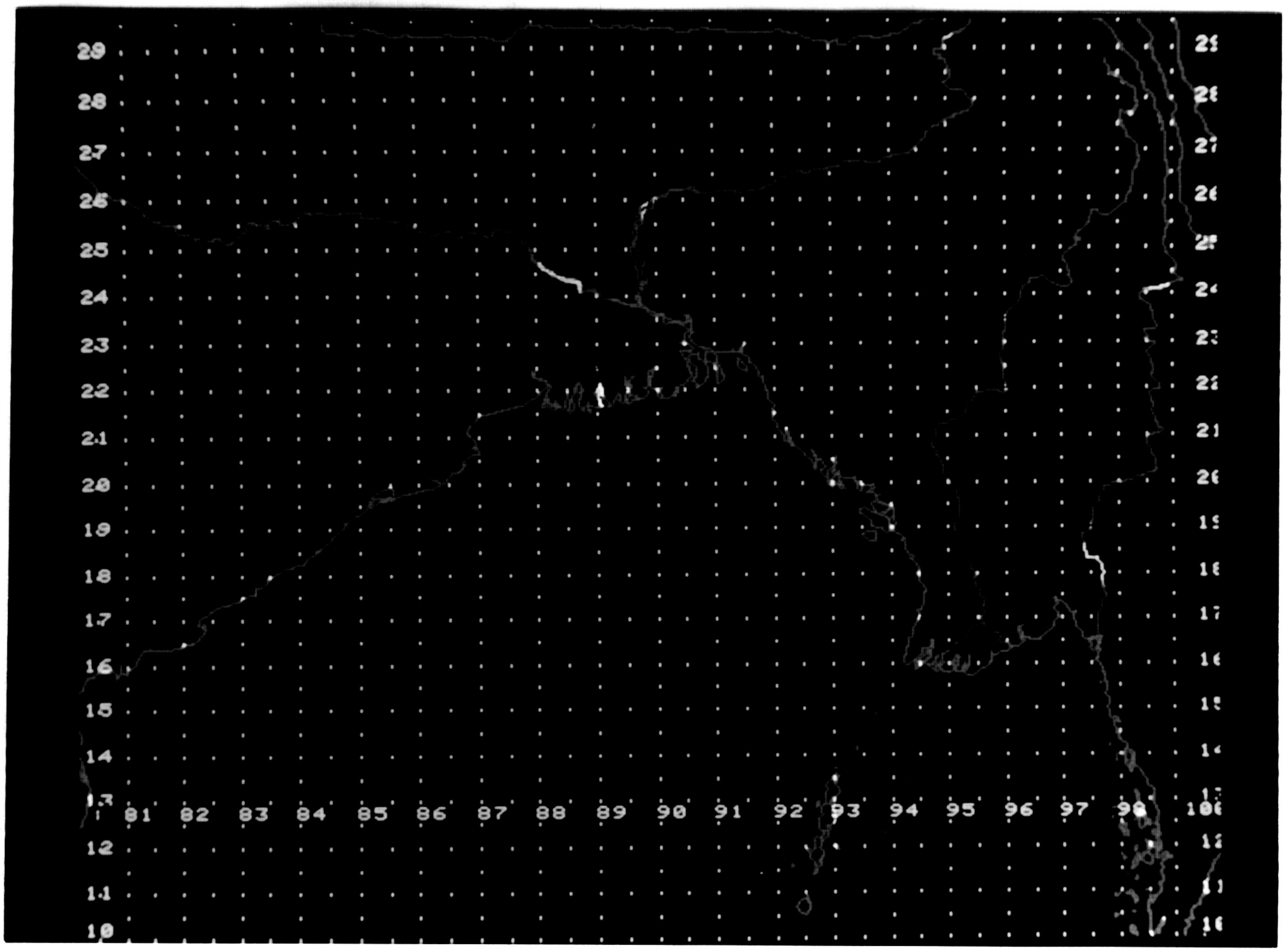
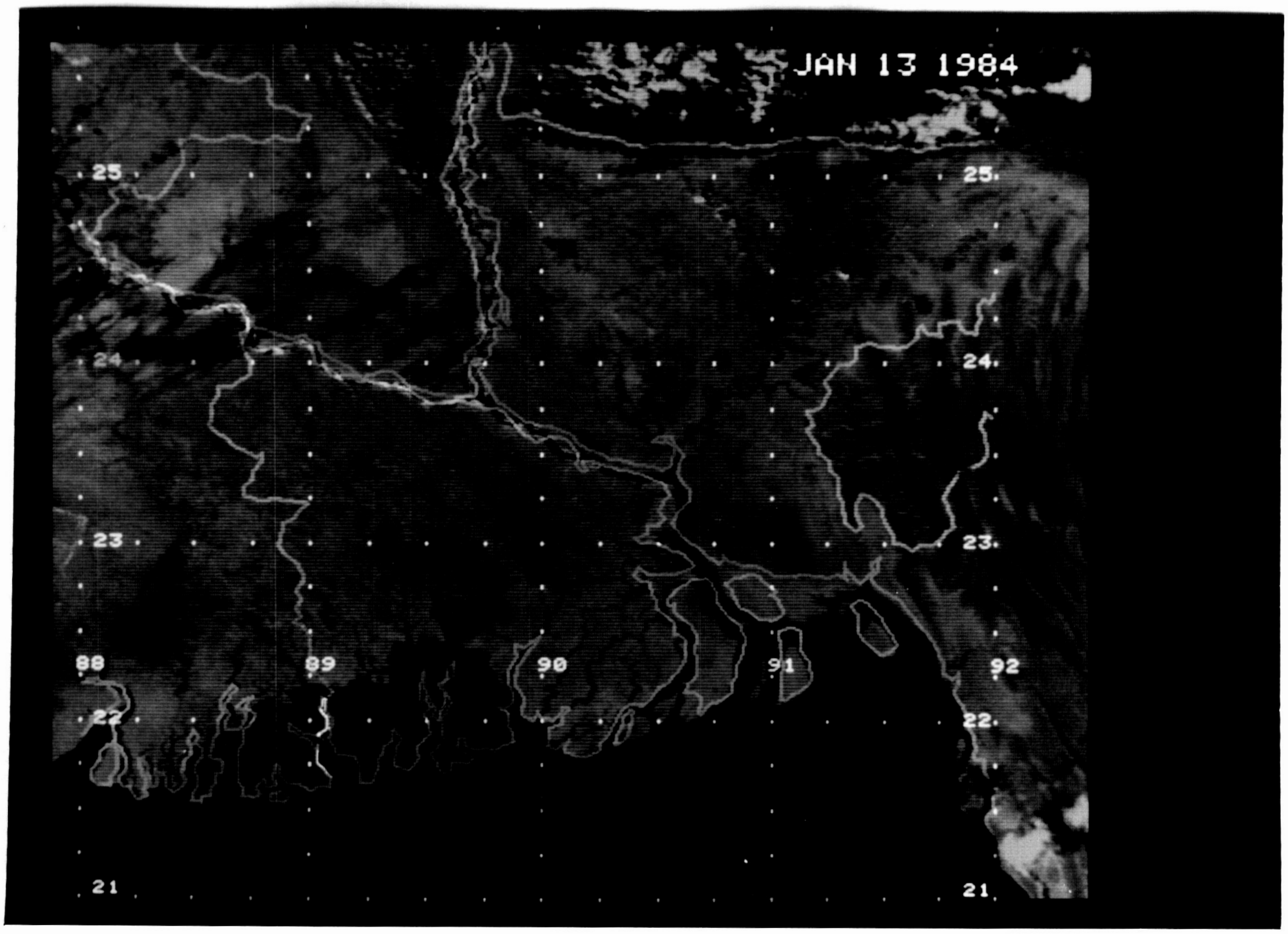


Figure 17. Example of World Data Bank II output. a) Output for entire Bangladesh region. Coastlines are blue, rivers are green, and political boundaries are red.



ORIGINAL PAGE
COLOR PHOTOGRAPH

Figure 17 (continued). b) Subportion overlayed on NOAA AVHRR image. Changes in river course, island area, and coastline can be detected by comparing the image and the WDB II output.

cartographic projections (equirectangular, Mercator, transverse Mercator, gnomonic, and stereographic) and optionally overlay a lat/long grid over the data. The operation of the MAP program is described in Chapter 8 (pgs. 183-184).

Caution should be used in fitting satellite data to WDB II products since rivers change course, islands and coasts erode and change shape, glaciers melt back, etc. (Figure 17.b). The dynamic character of certain geomorphic features may render parts of the WDB II obsolete for certain regions. Location of important geographic features and international boundaries make this system valuable in the analysis of NOAA satellite data.

The Synthetic Bathymetric Profiling System (SYNBAPS)

The Synthetic Bathymetric Profiling System (SYNBAPS) is a software package and data base of low resolution deep water bathymetric data. It provides for rapid random access of water depths around the world and for interactive depth profiling along arbitrarily selected surface tracks. It is possible to contour depths of an oceanic area of the globe in the data base with the I^2S image processing system. Contoured water depths are useful for understanding the behavior of ocean currents, upwelling, and atmospheric cooling of ocean waters. Profiles are needed for plotting sea surface temperatures and physical ocean data for oceanographic studies.

The SYNBAPS program is completely automatic and only requires the input of the latitude and longitude of the end points of the profile to extract and plot the depths. The smallest cell size in the data is a 5 minute square grid of latitude and longitude. The final profile is produced by orienting a cubic spline algorithm along the profile track and interpolating the depth values from the 5 degree squares falling on the path.

Resolution of the data is 5° latitude and longitude horizontally, and 200 m vertically. Contour lines can be generated by interpolation to 1/10th of the data or 20 meters. The coastline cannot be plotted directly, but the 1 m contour can be interpolated as an approximation. Since the data are of very coarse resolution, best use of this system for most countries would be to alter the data base in the region of primary interest in order to have higher spatial and vertical resolution (particularly in shallow coastal areas). The original data for SYNBAPS was digitized from charts at a scale of 1:1,000,000. Van Wyckhouse (1973) provides further information on the SYNBAPS program. Chapter 8 (pgs. 185-188) contains detailed information on the use of SYNBAPS.

REFERENCES

Anding, D, and R. Kauth. 1970. Estimation of Sea Surface Temperatures from Space. Remote Sensing of Environment 1: 217-220.

Barnes, J. C., and M. D. Smallwood. 1982. TIROS-N Series Direct Readout Services User's Guide. Environmental Research and Technology Inc., Massachusetts. 118 pp.

Bernstein, R. L. 1982. Sea Surface Temperature Estimation Using the NOAA 6 Satellite Advanced Very High Resolution Radiometer. Journal of Geophysical Research 87(C12): 9455-9465.

Central Intelligence Agency. 1977a. CAM - Cartographic Automatic Mapping Program Documentation, 5th Edition. PB-270 304, National Technical Information Service, Washington, D. C. 121 pp.

Central Intelligence Agency. 1977b. World Data Bank II: General User's Guide. PB-271 869, National Technical Information Service, Washington, D. C. 7 pp.

Goddard Space Flight Center. 1980. Heat Capacity Mapping Mission (HCMM): Data User's Handbook for Applications. Explorer Mission-A(AEM). National Aeronautics and Space Administration, Washington, D. C. 150 pp.

Hoffer, R. M., and C. J. Johannsen. 1969. Ecological Potentials in Spectral Signature Analysis. In "Remote Sensing in Ecology," P. L. Johnson, ed. University of Georgia Press. Pp. 1-16.

Huh, O. K. 1976. Detection of Oceanic Thermal Fronts off Korea with the Defense Meteorological Satellites. Remote Sensing of Environment 5: 191-213.

Huh, O. K., and D. DiRosa. 1981. Analysis and Interpretation of TIROS-N AVHRR Infrared Imagery, Western Gulf of Mexico. Remote Sensing of Environment 11: 371-383.

Huh, O. K., L. J. Rouse, and P. F. Twitchell. 1982. Outbreaks of Polar Continental Air: Windows on the Mesoscale Structure of the Upper Ocean. Naval Research Reviews XXXIV(1): 26-39.

International Imaging Systems, Inc. 1984. System 575 Software - Digital Image Processing System, Version 3.1: User's Manual. Product No. 777-0039-001. 1500 Buckeye Drive, Milpitas, CA 95035.

ITT Aerospace/Optical Division. 1980. Alignment and Calibration Data Book, AVHRR/2. Contract NAS 5-23400, Ft. Wayne, Indiana. 77 pp.

Kidwell, K. B. 1984. NOAA Polar Orbiter Data, User's Guide. U. S. Dept. Commerce, National Climatic Center, Satellite Data Services Division, Washington, D. C. 168 pp.

Lauritson, L. G., G. Nelson, and R. Z. Porto. 1979. Data Extraction and Calibration of TIROS-N/NOAA-6 Radiometers. NOAA Tech. Memo NESS 107, U.S. Dept. Commerce, Washington, D. C. 58 pp.

Lynn, R. J., and J. Svejkovsky. 1984. Remotely Sensed Sea Surface Temperature Variability Off California During a "Santa Ana" Clearing. Journal of Geophysical Research 89(C5): 8151-8162.

Matson, M., and J. Dozier. 1981. Identification of Subresolution High Temperature Sources Using a Thermal IR Sensor. Photogrammetric Engineering and Remote Sensing 47(9): 1311-1318.

Maul, G. A. 1983. Zenith Angle Effects in Multichannel IR Sea Surface Remote Sensing. Remote Sensing of Environment 13: 439-451.

McClain, E. P. 1980. Multiple Atmospheric-Window Techniques for Satellite-Derived Sea Surface Temperatures. In "Oceanography from Space," J. F. R. Gower, ed. Plenum Press, New York. Pp. 73-85.

McMillin, L. M., and D. S. Crosby. 1984. Theory and Validation of the Multiple Window Sea Surface Temperature Technique. Journal of Geophysical Research 89(C3): 3655-3661.

Myers, V. I. 1983. Remote Sensing Applications in Agriculture. In "Manual of Remote Sensing: Vol. II (Second Edition)," R. N. Colwell, ed. American Society of Photogrammetry, Falls Church, Virginia. Pp. 2111-2228.

Ostrem, G., T. Anderson, and H. Odegaard. 1981. Operational Use of Satellite Data for Snow Inventory and Runoff Forecast. In "Satellite Hydrology," M. Deutsch, D. R. Wiesnet, and A. Rango, eds. American Water Resources Association, Minniapolis, Minnesota. Pp. 230-234.

Prabhakara, C. G. Dalu, and V. G. Kunde. 1974. Estimation of Sea Surface Temperature from Remote Sensing in the 11 to 13 μm Window Region. Journal of Geophysical Research 79: 5039-5044.

Schneider, S. R., D. F. McGinnis, Jr., and J. A. Gatlin. 1981. Use of NOAA/AVHRR Visible and Near-Infrared Data for Land Remote Sensing. NOAA Tech. Report NESS 84, U. S. Dept. Commerce, Washington, D. C. 48 pp.

Schwalb, A. 1978. The TIROS-N/NOAA A-G Satellite Series. NOAA Tech. Memo NESS 95, U. S. Dept. Commerce, Washington, D. C. 75 pp.

Schwalb, A. 1982. Modified Version of the TIROS-N/NOAA A-G Satellite Series (NOAA E-J) - Advanced TIROS-N (ATN). NOAA Tech. Memo NESS 116, U. S. Dept. Commerce, Washington, D. C. 23 pp.

Selby, J. E. A., and R. A. McClatchey. 1975. Atmospheric Transmittance from 0.25 to 28.5 μm: Computer Code LOWTRAN 3. Air Force Cambridge Research Laboratories, AFCRL-TR 75-0255, Hanscom AFB, Massachusetts. 109 pp.

Short, N. M., and L. M. Stuart, Jr. 1982. The Heat Capacity Mapping Mission (HCMM) Anthology. National Aeronautics and Space Administration, Washington, D. C. 264 pp.

Smith, W. L., H. M. Woolf, C. M. Hayden, A. J. Schreiner, and J. M. LeMarshall. 1983. The Physical Retrieval TOVS Export Packages. Presented at the First International TOVS Study Conference, Igls, Austria.

Strong, A. E., and E. P. McClain. 1984. Improved Ocean Surface Temperatures from Space - Comparisons with Drifting Bouys. Bulletin of the American Meteorological Society 65(2): 138-142.

Sutherland, R. A., and J. F. Bartholic. 1977. Significance of Vegetation in Interpreting Thermal Radiation from a Terrestrial Surface. Journal of Applied Meteorology 16: 759-763.

Tarpley, J. D. 1983. Global Vegetation Index User's Guide. Satellite Data Services Division, National Climatic Data Center, NESDIS, U. S. Dept. Commerce, Washington, D. C. 6 pp.

Tarpley, J. D., S. R. Schneider, and R. L. Money. 1984. Global Vegetation Indices from the NOAA-7 Meteorological Satellite. Journal of Climate and Applied Meteorology 23: 491-494.

Tucker, C. J. 1978. A Comparison of Sensor Bands for Vegetation Monitoring. Photogrammetric Engineering and Remote Sensing 44: 1369-1380.

Tucker, C. J. 1979. Red and Photographic Infrared Linear Combinations for Monitoring Vegetation. Remote Sensing of Environment 8: 127-150.

Van Woert, M. 1982. The Subtropical Front: Satellite Observations During FRONTS 80. Journal of Geophysical Research 87(C7): 9523-9536.

Van Wyckhouse, R. 1973. Synthetic Bathymetric Profiling System (SYNBAPS). Tech. Report TR-233, U. S. Naval Oceanographic Office, Washington, D. C.

APPENDIX

FACT SHEET ON TIROS-N/NOAA A-G SERIES SATELLITES

I. Orbit

(a) Type: sun synchronous, near polar orbit
(b) Altitude: 830 and 870 km (+ or - 30 km)
(c) Period: ca. 102 minutes
(d) Orbital inclination (to earth's pole): 98.8°
(e) Number of orbits per day per satellite: 14.2
(f) Nodes (equatorial crossings): daytime - ascending
nighttime - descending
(g) Local Sun Time of overpasses: ca. 0300 and 1500
ca. 0730 and 1930

II. The Advanced Very High Resolution Radiometer (AVHRR)

(a) Resolution: satellite subpoint - 1.1 X 1.1 km
limb - 1.5 X 4.0 km
(b) Channels: Ch 1 - 0.58-0.68 μm
Ch 2 - 0.725-1.1 μm
Ch 3 - 3.5-3.9 μm
Ch 4 - 10.5-11.5 μm
Ch 5 - 11.5-12.5 μm
(c) Swath Width on earth's surface: 2,700 km
1,600 naut. mi
(d) Look angle from nadir (max): 55.4° from nadir
110.8° limb to limb
(e) Digital radiometric data
- five channels of 2048 data pixels per scan line (limb to limb), for a total of 10,240 pixel values
- Sampled every 40 pixels along scan line at points 25, 65, 105, ..., 1985, 2025 for a total of 51 possible solar zenith and lat/long values per scan line. Solar zenith angle stored as one byte; scaled with seven integer bits and one fractional bit. Lat/long values stored in two byte fields in 1/128th degree (east positive, west negative).
- Thermal calibration: sensor views space for zero reference and views temperature controlled housing with three platinum resistance thermometers for exact thermal value which is appended to each scan line. Count to radiance relationship scaled to slope/intercept values (coefficients) in units of milliwatts/(m^2 steradian cm^{-1}) per count. Energy measured by sensor (channel X) is computed as a linear function of the input data value, i.e.:

$$E_X = S_X C + I_X$$

where E_X is the energy of channel X, C is the input count (0-1023), and S_X and I_X are scaled slope and intercept values (calibration coefficients). Energy is converted to radiation temperature using the inverse of Planck's function.
- Visible/NIR calibration: channel counts are related

to percent albedo values by slope and intercept. Percent albedo measured by sensor is computed as a linear function of the input data as follows:

$$A_X = S_X C + I_X$$

where A_X is percent albedo and the other variables are as were defined above. Uses preflight calibration coefficients only; no onboard targets are used.

III. Data Products

(a) Map of earth surface temperature field in channels 3, 4, and 5.

(b) Map of estimated surface temperatures using the multichannel sea surface temperature calculation and tested algorithms.

(c) Map of land/water boundaries, river channels, flooded areas, lakes, and coastal inundation.

(d) Albedo mapping of land, waters, and cloud tops from channels 1 (visible) and 2 (NIR).

(e) Mapping of vegetation greenness using channels 1 and 2 for the simple and normalized vegetation indices.

(f) Snow/cloud discrimination mapping using all channels.

(g) River hydrology, river deposition, and land/water change.

PART II

PROCESSING AND ANALYSIS OF AVHRR IMAGERY

Scott G. Leibowitz
Research Associate
Remote Sensing and Image Processing Lab

Donald DiRosa
Research Associate
Remote Sensing and Image Processing Lab

and

John M. Hill
Associate Professor
Department of Civil Engineering

Louisiana State University
Baton Rouge, Louisiana 70803

CHAPTER 4

THE I^2S IMAGE PROCESSING SYSTEM

Hardware Environment

The I^2S Model 75 Image Processing System is a state-of-the-art processing package that utilizes an advanced hardware architecture to perform most processing in real time. In order to process images with the I^2S system, it is necessary that the user be at least conceptually familiar with some of this hardware. In the following section, the basic components of the I^2S hardware are introduced. For a more complete, technical description, refer to I^2S (1983).

Refresh Memory

The I^2S Model 75 can contain up to 16 refresh planes, where each plane is 512 elements by 512 lines by 8 bits (the actual number of planes is installation dependent). These refresh planes serve as the system's "fast memory," and much of the processing in the I^2S system takes place in these refresh planes. The refresh memory is loaded with data either from disk or from tape. Since each refresh plane is only capable of storing one 8-bit 512 X 512 image, a 3 band 512 X 512 multispectral image would require 3 refresh planes, a one band 1024 X 1024 image would require 4 refresh planes, and a 16-bit 512 X 512 image would require 2 refresh planes.

In addition to 8-bit image data, the refresh planes can handle 1-bit graphics data. Depending on the installation's configuration, either one or two of the refresh planes are reserved for graphics data, thereby providing either 8 or 16 graphics planes (each 8-bit refresh plane is equivalent to eight 1-bit graphics planes). Graphics planes are used for annotation, polygon and line drawing, creation of logical masks, etc.

> NOTE: data in refresh memory are lost when the I^2S session is terminated. To save a processed refresh image, the data must be transfered to disk. For further details, see the discussion of the SAVE function (pgs. 99-100).

Pipeline Processing Channels

The I^2S system has three parallel pipeline processors, each of which controls one of the three monitor color guns (red, green, and blue). The pipeline is the hardware device which transfers data from the refresh memory to the display monitor. Each pipeline can receive data from any or all of the refresh planes.

In addition to transferring data to the display monitor, the pipeline also performs many of the I^2S arithmetic functions. Operations such as addition, subtraction, multiplication, and division are done by the pipeline as the data are transferred to the display monitor.

Thus these operations are performed in real time (i.e., at a 30 frame per second refresh rate).

If a three band, 512 X 512 image were being processed, each of the three pipelines would transfer data from one of the refresh planes to the monitor. Since each pipeline controls a different color gun, the data on the first band would be displayed in red, data on the second band would be displayed in green, and the third band's data would be displayed in blue. Thus the resulting image would be a three band, false-color image. If the three bands represented data from red, green, and blue wavelengths, this configuration would simulate the natural scene. The three bands need not be assigned to the guns in this order, however. For instance, the red band could be assigned to the green gun, the green band to the blue gun, and the blue band to the red gun (Figures 21 and 22 illustrate alternate color gun assignments). This topic is discussed in detail in the section on the DISPLAY function (pgs. 105-110).

To create a false-color display, the number of input bands must be equal to three. While each band in a three band image is uniquely assigned to one of the three pipelines, an image with less or more than three bands has each one of the bands assigned to all three of the pipelines. Thus, when processing other than three bands, a black and white image results where pixel intensities are equal to the sum of the intensity in each band (black and white images can be colored by using "painting" functions, such as COLORS and PALETTE; however these procedures assign colors to the resulting aggregate image, and do not differentiate between different bands).

Look-Up Tables

Each of the three pipelines in the I^2S Model 75 has eight 1024 X 12-bit look-up tables (LUT). An LUT is a device that maps an input grey level onto some output grey level. The particular mapping function is determined by the user. Once the function is loaded into the LUT, the data stream passes through the LUT at the system's 10 MHz data rate. Two examples of the types of functions that can be loaded into an LUT are logarithms and spectral corrections. For example, multiplication on the I^2S consists of loading an LUT with logarithms, adding, and then passing the sum through an LUT containing anti-logarithms (exponentials). Histogram stretching (contrast enhancement) consists of passing the original grey levels through an LUT containing the transformed values. The presence of look-up tables allows the Model 75 to perform many traditionally time consuming functions in real time. In fact, since data streams pass through the LUTs so rapidly, many of the I^2S functions do not alter the original data but instead pass the input data through the LUT during each refresh cycle. Transforming the input data in this way is easier on the I^2S than replacing the original input data with transformed values.

In addition to the pipeline look-up tables, the I^2S also has a programmable input function memory that can transform data as they are loaded into the refresh planes. Thus raw disk or tape data can quickly be transformed as they are loaded into refresh memory. Each pipeline also has an output function memory that transforms the final pipeline outputs into color gun values for monitor display.

Hardware Zoom, Hardware Scroll, and Videometer

The I^2S Model 75 includes hardware devices for zooming (image magnification by factors of 2, 4, or 8; see Figures 34-36, pgs. 161-163) and scrolling (spatially shifting the scene in either X and/or Y direction). It also contains a videometer that calculates histograms from data as they leave the pipeline. Since all three of these functions are hardware implemented, their execution occurs in real time. The speed of these and many of the other I^2S functions allows the user to perform many tasks which were previously time consuming or even infeasible.

Trackball and Footpedal

In order to manually position the cursor in the X-Y plane, the I^2S Model 75 comes equipped with a trackball unit (Figure 18). The user can move the cursor either by rolling the trackball or by using the four cursor positioning buttons that appear on the trackball unit (these four

ORIGINAL PAGE
BLACK AND WHITE PHOTOGRAPH

Figure 18. I^2S trackball unit and footpedal.

buttons move the cursor up, down, left, and right at a speed that is determined by a switch on the trackball unit). Cursor positioning is important in many I^2S functions such as reading pixel values, drawing polygons, scrolling, etc.

In addition to cursor positioning, the trackball unit also comes with 15 function buttons which are used to select amongst the options of the various I^2S functions. For example, the magnification factor in the ZOOM function is selected through these function buttons. In addition to being described in the I^2S User's Manual (I^2S 1984), a description of each button's function is also written to a graphics plane when a particular application is entered (Figure 19). This graphics plane can be selected by pressing the footpedal switch. Thus, an on-line menu is available for many of the I^2S functions.

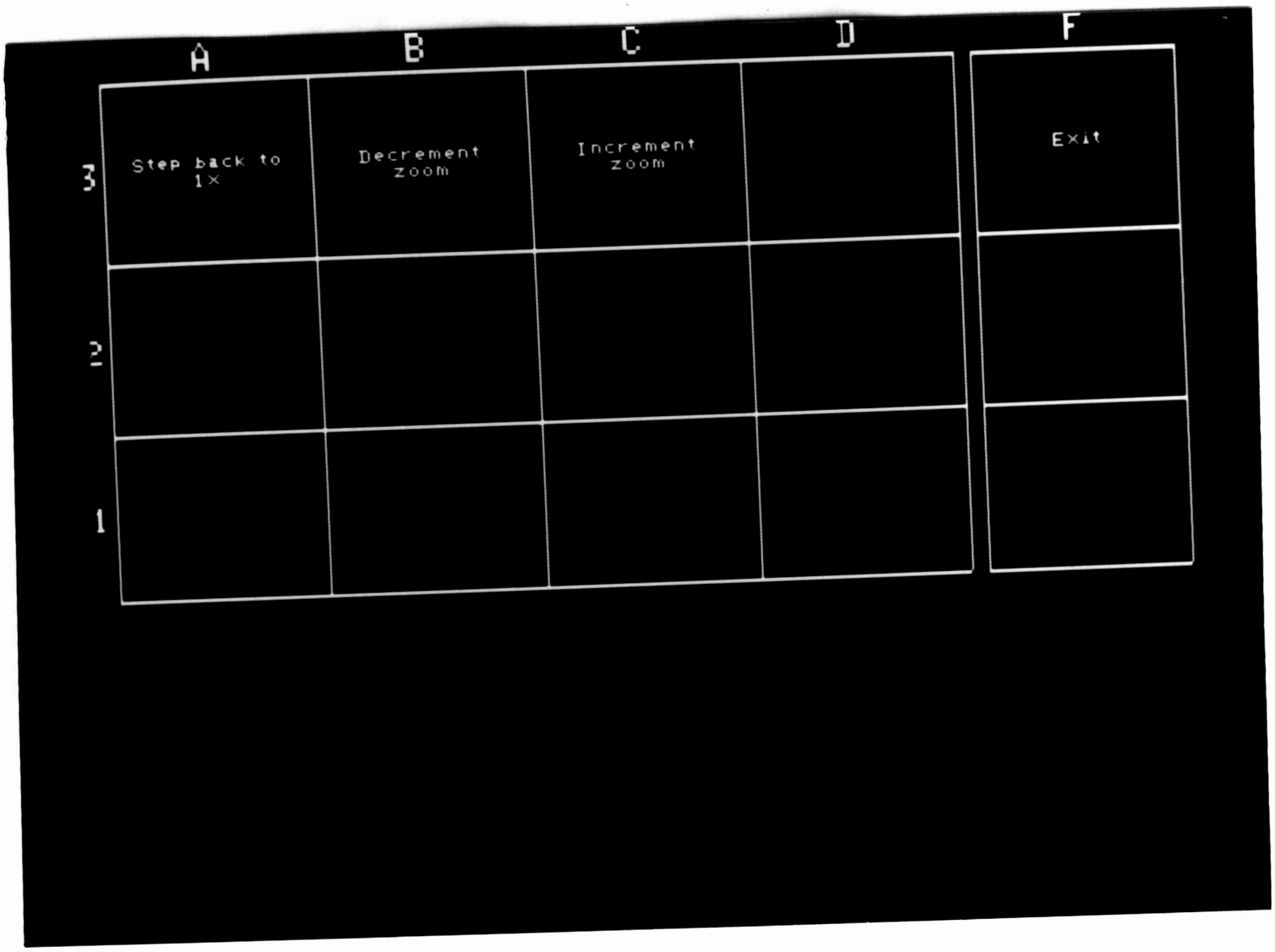


Figure 19. Example of on-line menu for I^2S software functions. Example shown is for ZOOM function.

Command Syntax

The I^2S command syntax is composed of three basic components: input image and modifiers; applications function and parameters; and output images. The general command syntax is:

IN'IMAGE > FUNCTION'NAME > OUT'IMAGE

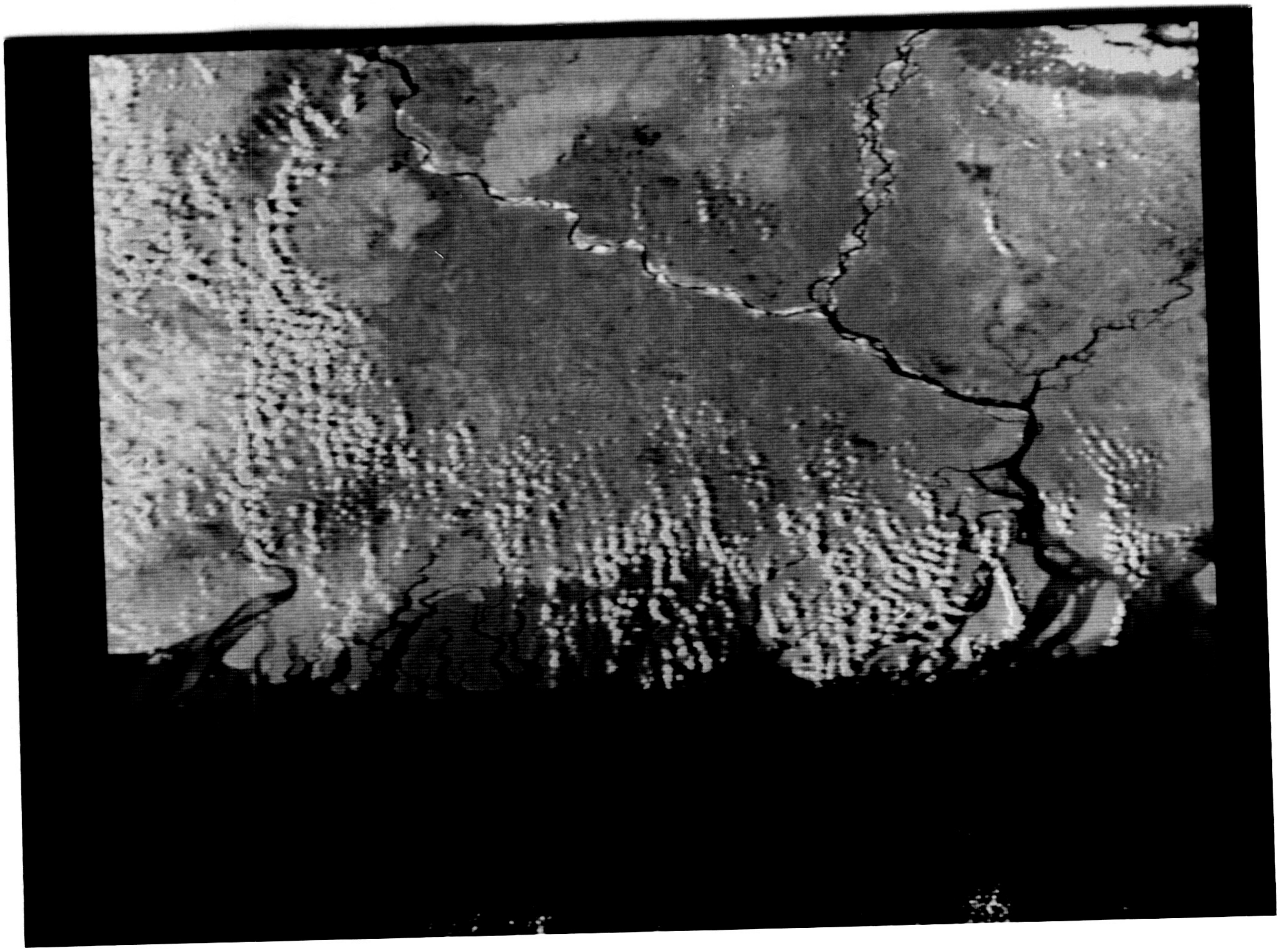
Where the ">" sign is a delimiter. These three components will be described briefly below. For a full description, see I^2S (1984).

Input Files

I^2S functions accept two types of input data - data residing on disk and data residing in refresh memory. Data residing on refresh memory have names preceded by the dollar sign ("$") (e.g., $IMAGE1 would be a refresh image while IMAGE2 would be a disk image). In general, disk data can be either unsigned 8-bit data (byte), signed 16-bit data (integer), or 32-bit floating point data (real). Refresh images are restricted to byte or integer data.

Input data files are of several functional types. The most common file type is the image file, which represents some spatial scene. This is the file type that the user is most often analyzing. Some of the other file types produced through various I^2S functions are stats files (statistical data produced from a training function such as CLASSIFY), stash files (a file that preserves the status of the system's look-up tables, color assignments, and other system parameters so that a particular environment can be restored), control point files (known landmarks used for georeferencing images), and verts files (a file containing the coordinates of training field polygons). The remainder of this discussion will focus on image files.

I^2S data images are essentially three dimensional arrays, where the array is indexed by sample, line, and band. Sample is the number of pixels in the horizontal direction; line is the number of pixels in the vertical direction; and band refers to the types of different information available at each X-Y location (e.g., multispectral or multitemporal data). As was discussed in the section on hardware, a display image is normally a 512 sample X 512 line image (larger arrays can be displayed using the ROAM function, but many I^2S functions assume a 512 X 512 image). Thus, it is often necessary to somehow reduce the image size to 512 X 512. This is accomplished by subsectioning or subsampling. In subsectioning, every element of some smaller portion of the larger image is used (Figure 20). Reading in the upper left hand corner of a 1024 X 1024 image would be an example of subsectioning. In subsampling the entire image is sampled, but the sampling increment that is used is greater than one. For example a 1024 X 1024 image could be reduced to a 512 X 512 image by reading in every other element and line. If the original



ORIGINAL PAGE
COLOR PHOTOGRAPH

Figure 20. Example of subsectioning. Image is from same scene as that shown in Figure 23.

image varied in a smooth manner, then the resulting subsampled image would visually resemble the original image. The spatial resolution, however, would be reduced. Figure 23 (pg. 123) uses subsampling to include an entire 2048 X 2048 scene on a 512 X 512 display.

In addition to subsectioning and subsampling, an input image may also be subbanded. In this case, fewer bands than are available are used. For example, a NOAA AVHRR image contains 5 different bands. If the user only wanted to work with three of those five bands, subbanding would be performed.

The following syntax is used for input data images:

```
IMAGE (SS SL NS NL SI LI : BANDS)

where        SS       is the starting sample number
             SL       is the starting line number
             NS       is the number of samples
             NL       is the number of lines
             SI       is the sample increment
             LI       is the line increment
             BANDS    are the bands to be used
```

As an example, assume a 5 band, 1024 X 1024 image named INPUT. The following input formats would lead to the following results:

```
INPUT (1 1 512 512:1 2 5)
  Read in the upper left hand corner of bands 1, 2,
  and 5.  Note that SI and LI default to 1.

INPUT (513 1 512 512:4)
  Read in the upper right hand corner of band 4.
```

INPUT (1 1 1024 1024 2 2)
Read in the entire image subsampled by 2 in the X and Y direction. By default all 5 bands are read in.

INPUT (:4 2 1)
Read in bands 4, 2, and 1 of the entire image (no subsampling or subsectioning by default).

INPUT
Read in all 5 bands of the entire image.

Many I^2S functions allow multiple image input. The general syntax for this is:

IMAGE1 IMAGE2 ... IMAGEn > FUNCTION'NAME

where IMAGEi is a disk or refresh image. Modifiers may be used to restrict input.

Application Functions

The I^2S system has over 250 functions for processing images. Nearly all of these functions have parameters that are used to specify how that function is to operate (if not specified, parameter values default to system supplied values). The general form of the function statement is:

> FUNCTION'NAME (PARAMETER1 PARAMETER2 ... PARAMETERn)

The parentheses are optional and may be omitted. A complete description of I^2S functions, their parameters, and parameter defaults is given in I^2S (1984).

There are two different types of parameters in the I^2S software. Keyword parameters are logical parameters that are either "true" or "false." The default value for keyword parameters is usually "false." The parameter must be explicitly requested if its value is to be set to "true." For example,

> HISTOGRAM (LP)

requests that the results of a histogram operation be sent to the line printer. Note that the value of a keyword parameter is just the name of the parameter itself.

The other type of I^2S parameter is a value parameter. As the name suggests, value parameters are assigned some value, either integer, floating point, or character. The parameter is assigned this value using the equal sign ("=").
For example,

> CONSTANT (OUTPUTCONSTANT = 0 0 255)

requests that a "constant" image (one where all pixel values are identical) be produced with red, green, and blue gun intensities of 0, 0, and 255, respectively (note that a value parameter can require more than one value).

In the previous two examples, parameter values were specified explicitly in the function statement. There is also a query mode whereby the system prompts the user for any parameter values. To be queried, the function is entered in the following manner:

> FUNCTION'NAME ?

Specifying a question mark after the function name causes the system to prompt the user for all parameter values on that function's parameter list. For each parameter, the system will then print the parameter and its default value(s). The user either enters a carriage return to use the default value, or enters a new value. For keyword parameters a "T" or "F" is entered for true or false. For value parameters an integer, floating point, or character string (enclosed in quotes) is entered, depending upon the variable type.

It is possible to specify function names and parameter names in shorthand notation, using the minimum number of letters that uniquely identifies that command. For example, CONS is sufficient to specify the CONSTANT function, since no other function begins with this prefix. CON, however, would not be sufficient, since several other function names begin with this prefix (in this case, an "ambiguous function name" error would occur).

Although most I^2S functions require either a disk or refresh image input, some functions do not require an input. The CONSTANT function referred to above is one example. Another example is HISTOGRAM, which will produce a histogram of the image currently displayed if no input is given. In cases where no input is specified, the function name should still be preceeded by the delimiter (e.g., ">") so that the system will recognize the variable as a function name rather than an image name.

Output Files

Many of the I^2S functions produce output data files. As in the case of input files, output data may be stored on disk or refresh memory. File names consist of up to sixteen characters composed of letters, numbers, the apostrophe ("'"), and the underscore ("_"). Refresh files must always begin with the dollar sign ("$").

The most common type of output file is the image file. As was the case with input files, however, other file types do occur. For more information on file types, see the discussion under input files (pgs. 82-83) or refer to I^2S (1984).

A function may produce more than one output file. This usually occurs when more than one input file was specified. The general form for multiple file input and output is

```
IN1 IN2 ... INn > FUNCTION'NAME > OUT1 OUT2 ... OUTm
```

Miscellaneous Syntax

In addition to the greater than sign (">"), the back slash ("\") and the up arrow ("^") are also legal delimiters. To stack more than one command on a single line, a semicolon (";") should be placed between commands. Commands requiring more than one line are continued by the use of the hyphen ("-"). A comment may be added to the right of any command by placing an exclamation point ("!") between the command and the comment.

System Functions

There are seven important system commands that are a part of almost every I^2S processing session. Because of the importance of these functions, they will be briefly discussed here.

CI

The command interpreter ("CI") is the software file processor that interprets user supplied commands. "CI" is also the command used to enter I^2S from the host environment. Once inside the I^2S environment, "CI" appears as a prompt after each command.

ACQUIRE

Once the user has entered the I^2S environment, the monitor must be acquired by the system if the user intends to display data. This is done once, usually at the beginning of a session. This function can be set up in a START command file so that it is automatically invoked after entering the system through CI (for further information refer to the discussion on command files in I^2S [1984]).

It is possible to perform functions that do not require image display without invoking ACQUIRE. However, any attempt to display data without first invoking this function will result in an error.

LIST'CATALOG

LIST'CATALOG (or LIST, for short) is one of the most often used I^2S functions, since it allows one to list the data files that are currently in the image directory. Entering the command LIST without any parameters will cause the system to print a list of all current file names (refresh files are preceeded by a "$"). Following the LIST command with the keyword parameter LONG will produce a list that includes the image name, the number of samples, the number of lines, the number of bands, the file type (e.g., image, stats, stash, etc.), the refresh plane number or the data type (byte, integer, or real) depending on whether it is a refresh or disk file, and the name associated with that file on the host system. One inconvenience with LIST LONG is that refresh files are not preceeded with the dollar sign when listed this way. However, it is still possible to distinguish refresh files from disk files when using LIST LONG by noting whether the refresh plane/data type column contains the number of a refresh plane or a data type.

LIST can also be followed by the parameter DISK, indicating that only disk files are to be listed. Similarly, LIST DISPLAY will produce a list of display (refresh) files. Either of these parameters can also be combined with LONG to produce a long list (e.g.,

> LIST DISPLAY LONG

produces a long list of all refresh files).

SESSION

The SESSION command causes a session history file to be created. This file will contain all of the commands entered by the user (NOTE: The session history only contains commands entered explicitly. Parameter values entered through queries by using > FUNCTION'NAME ? are not stored by SESSION). This file is valuable because it can be used with the REDO function (see below), and because it serves as documentation of how an image was produced. This history can also be useful for error analysis.

The session history file exists only for the duration of a session. Terminating the I^2S session through END (see below) or cancelling the session history through > SESSION KILL will cause the session file to be deleted. Thus, if the user desires a printout of the session history, it must be printed before the file is destroyed. This can be done through the following command:

> SESSION LP

This causes the entire session file to be routed to the line printer. If only a portion of the file is desired, the user can enter:

> SESSION FROM=C1 TO=C2 LP

This will cause commands C1 through C2 to be printed to the line printer.

When the SESSION command is invoked, the normal "CI" prompt is replaced by the command number. This command number starts at "1" and is incremented each time a command is entered. This is useful not only in specifying FROM and TO, as was discussed above, but also for use with the REDO command.

REDO

The REDO function allows commands in the session history to be edited and repeated. To use this facility, the SESSION command must first be invoked (see above). With REDO, a previously entered command can be modified by having its characters replaced or deleted, or by having characters inserted. To use this facility, the following command is entered:

> REDO COMMANDS=C1 C2 ... C16

where the C's are the command numbers of up to sixteen previously entered commands (omitting the COMMANDS parameter causes the most recently entered command to be repeated).

Once REDO has been entered, the system repeats the specified line. The user can then edit this line using any of the editing features (for a list of these editing commands and examples of their use, see I^2S [1984]). After the command is altered to the user's satisfaction, a carriage return is entered to reexecute the modified instruction. Thus, the REDO function is useful not only for correcting erroneously entered commands, but can also be used to repeat commands.

HELP

The S575 software comes with a set of on-line documentation for its processing functions. This documentation is accessed through the HELP command. When the > HELP command is entered, all topics for which documentation exists are listed. The user then selects one of these topics or enters a carriage return to return to CI. If a topic is selected, the system will provide a general description of the command and then print a list of subtopics for which further information is available. These subtopics usually include descriptions of the different parameters, syntax, and examples.

Help for a particular topic or a particular subtopic can also be explicitly requested in the HELP statement (e.g.,

> HELP DISPLAY

will give information on the DISPLAY function, while the command

> HELP RECOVER SYNTAX

prints a help listing on the syntax of the RECOVER command).

For further information on HELP and its sub-commands, see I^2S (1984), or enter the command > HELP HELP.

END

The END command is used to terminate the I^2S session and return to the host environment. As was discussed earlier, END will cause the session history file to be deleted if SESSION was invoked. Also, any refresh images are lost upon entering the END command.

CHAPTER 5

DATA ENTRY ON THE I^2S IMAGE PROCESSING SYSTEM

There are three types of storage media utilized by I^2S software: refresh, disk, and tape. Processing usually takes place on refresh memory, while disk memory is used to save data from session to session (data in refresh memory are lost after the session is terminated). Magnetic tape is used for backup, long term storage, and for reading in raw data products.

When processing an image on I^2S, the usual transferal route is from magnetic tape to disk, and then from disk to refresh. After an image is processed, the refresh image is usually saved on disk before the session is terminated. A backup copy of the disk image is often made on tape. In this section, the various functions used to transfer data between different media types are discussed.

Data Transferal to Disk

Tape to Disk

Imagery products purchased from agencies such as NOAA are usually in tape form. Thus, a newly acquired image must be transferred from tape to disk before it can be processed by I^2S. The AVHRR function, described in detail in Chapter 6 (pg. 122), reads in images from NOAA AVHRR format 1B tapes, calibrates them, and writes the data to disk. In addition, the AVHRR function will also create a control point file from latitude and longitude information supplied with the data. The syntax for the AVHRR function is:

> AVHRR ? > IMAGE'FILE CNTP'FILE

where CNTP'FILE is the name of an optional control point file. The question mark signals the system to prompt for all parameter values. Subsectioning can be performed through the NSAMPS and/or NLINES parameters, and subbanding is done through the OUTBANDS parameter. The AVHRR function cannot subsample an image.

I^2S functions are also available for reading in Landsat Multispectral Scanner (LANDSAT or NEWLAND), Landsat Thematic Mapper (THEMATIC'MAPPER), and Coastal Zone Color Scanner (CZCS'ENTER) imagery. These tape input functions are covered in detail in I^2S (1984).

Another tape entry function that is often used is ENTER. This function reads in tape images that were created from disk by the TRANSFER function (see below). In other words, TRANSFER is used to write disk images out to tape and ENTER is used to bring these tape images back to disk. ENTER can read in more than one tape image, and thus multiple output files are allowed.

One nuance concerning the creation of disk images with subbanding needs to be mentioned here. If a tape image is read onto disk with subbanding, the band numbers on the disk image need not correspond with the band numbers on the tape image. For example, assume AVHRR was used to transfer bands 2, 4, and 5 from tape to disk. The following command would be entered:

```
> AVHRR OUTBANDS=2 4 5 > IMAGE1
```

If this command were entered, IMAGE1 would contain 3 bands, with disk bands 1, 2, and 3 corresponding to tape bands 2, 4, and 5, respectively. As another example, consider the following command:

```
> AVHRR OUTBANDS=5 1 2 > IMAGE2
```

In this case bands 1, 2, and 3 of IMAGE2 would correspond to bands 5, 1, and 2 of the original tape image, respectively. These two examples illustrate that the numerical identity of the original bands can be lost through subsampling. It is, therefore, essential that the user pay close attention to what bands are actually represented by the bands on disk images.

Refresh to Disk

As was mentioned earlier (pg. 75), refresh data are lost when the I^2S session is terminated. Thus any processed images that the user wishes to retain must be stored on disk. SAVE is the function that is used for this. The format for SAVE is:

```
$IMAGE1 > SAVE > IMAGE2
```

where $IMAGE1 is the name of the refresh image and IMAGE2 is the name of the file to be created on disk. This function has no parameters.

In addition to SAVE, there are two other important functions for creating disk files. G'SAVE is the function used to save graphics data to disk. The form of this function is:

```
> G'SAVE > DISP'FILE
```

If G'SAVE is used with no parameters, all of the graphics planes will be stored by default. The user can explicitly specify which planes to store by using the PLANES parameter. For example,

```
> G'SAVE PLANES=1 3 4 > GFILE
```

will write graphics planes 1, 3, and 4 to a disk file called GFILE.

In many cases, I^2S image processing consists of assigning values to the look-up tables, registers, and different hardware devices. For example, the function PALETTE "paints" an image by reassigning LUT values according to some coloring scheme. To save the system parameters defining a particular environment, the STASH function is used. The format for this function is

$IMAGE > STASH > STASH'FILE

where $IMAGE is the name of the image whose environmental status is to be saved. This function has no parameters.

Data Transferal to Refresh Memory

Disk to Refresh

Since nearly all I^2S image processing occurs in refresh memory, the transferal of data from disk to refresh is one of the most basic I^2S operations. The function used to perform this is DISPLAY. The general form of DISPLAY is:

IN1 IN2 ... INn > DISPLAY > $OUT1 $OUT2 ... $OUTn

where INi is a disk image and $OUTi is a refresh image. Input images can be modified by subsectioning, subsampling, and/or subbanding. Note that all refresh image names must begin with the dollar sign.

There are several DISPLAY parameters that are worth mentioning. The SIXTEEN'BIT parameter is used when the disk image is composed of 16-bit data. As was mentioned previously, each refresh plane is only capable of storing 8-bit data. The SIXTEEN'BIT parameter causes the image to be split into two 8-bit images, with the 8 least significant bits stored in one refresh plane and the 8 most significant bits stored in a second refresh plane. This arrangement allows 16-bit images to be displayed.

The MINCLIP and MAXCLIP parameters allow the user to specify the count values on the disk image that are assigned to the minimum (0) and maximum (255) count values on the refresh image. This results in a linear stretching, compression, or shifting of the image if values other than 0 and 255 (the defaults) are used. For example, consider a disk image composed of integer data (16-bit), with data values ranging from 0-1023 (NOAA AVHRR data have 10-bit precision). One way of reading this image into refresh would be to use the SIXTEEN'BIT parameter, as was just described. Another way of loading this image into refresh would be to reduce the data to an 8-bit range. This could be done in two ways. First, the 0-1023 count values of the disk image could be compressed into the 0-255 values of the refresh image, using the following command:

```
DISP'IN > DISPLAY MAXCLIP=1023 > $REFRESH'OUT
```

In this case, each count value on the refresh image would represent the aggregate sum of four disk count values.

It is also possible to read in a portion of the range of values, rather than compressing the entire 16 bits. For example, if it was known that the data of interest had count values between 700 and 955, the data read in could be limited to the values of interest. This would be done as follows:

```
DISP'IN > DISPLAY MINCLIP=700 MAXCLIP=955 > $REFRESH'OUT
```

With this command refresh count value 0 would represent disk count value 700 and refresh count value 255 would represent disk count value 955.

A complication can arise when using DISPLAY to clip multispectral images whose bands have nonoverlapping data ranges. If the image consists of only one band or if the multispectral data have the same data ranges, using the MINCLIP and/or MAXCLIP parameters to reduce the data to an 8-bit range is straightforward. For example, visible NOAA AVHRR data normally have reflectance values within a range of 0-255. The default DISPLAY clip values would, therefore, be sufficient to reduce a visible image to 8-bit data. For thermal data, values normally lie outside of the 0-255 range and thus the MINCLIP and MAXCLIP parameters of DISPLAY would be used.

Problems occur, however, when DISPLAY is to be used on a multispectral image containing both visible and thermal data. If this function is used with the default clip values, the visible data will be in range but the thermal data will not. Conversely, if MINCLIP and MAXCLIP are used to put the thermal data in range, then the visible data will be out of range. To get around this, first DISPLAY the visible and thermal bands separately, using the default clip values for the visible bands and the MINCLIP and MAXCLIP parameters for the thermal bands. These separate refresh images are then combined using the MERGE function.

Another DISPLAY parameter that is useful is QUIET. When an image is loaded by DISPLAY, it is also by default sent through the pipeline to the monitor. Thus, DISPLAY clears any previously displayed image. It is sometimes desirable, however, to load images into refresh without displaying them. This is accomplished by the QUIET parameter. An example of when this would be desirable is during a demonstration. To make the demonstration smoother and less time consuming, images could be loaded into refresh in QUIET mode while the user was discussing an image already displayed on the monitor. Thus, the QUIET parameter allows image input to occur "behind the scenes."

As was discussed in the section "Pipeline Processing Channels" (pg. 76), a refresh image with three bands will be displayed in color. Thus, DISPLAY is one of the functions used to assign bands to the color guns (the other function is SELECT). The first band in the refresh image is assigned the red gun, the second band to the green gun, and the third band to the blue gun. For example, consider a three band disk image called DISK, where bands 1 through 3 represent visible, near infrared, and thermal data. If the user desired these three bands to be displayed as red, green, and blue, respectively (Figure 21), then the data would be loaded by using the following command:

```
DISK > DISPLAY > $OUT'IMAGE
```

In other words, the default band assignment would be used since DISPLAY normally assigns band 1 to red, band 2 to

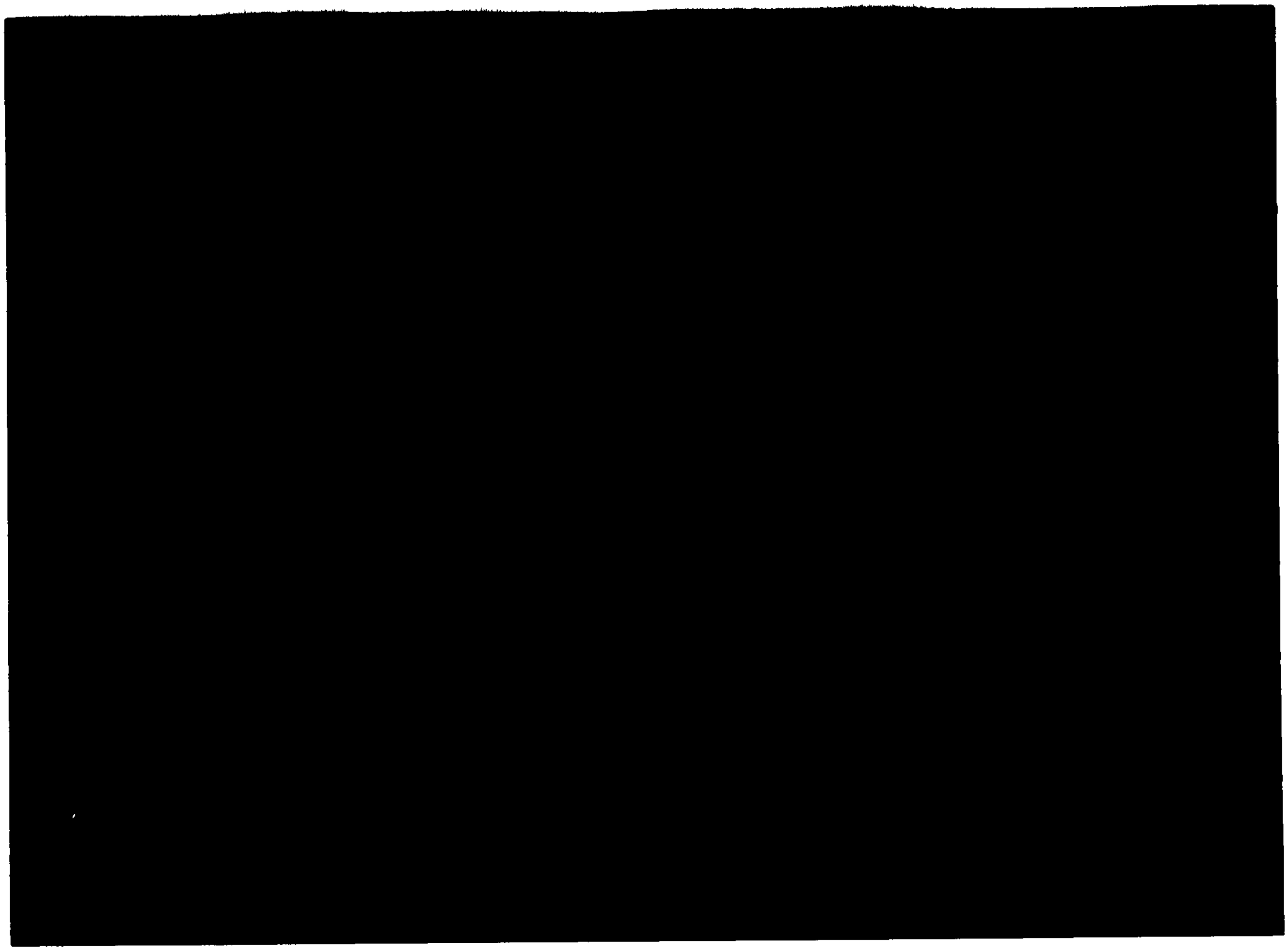
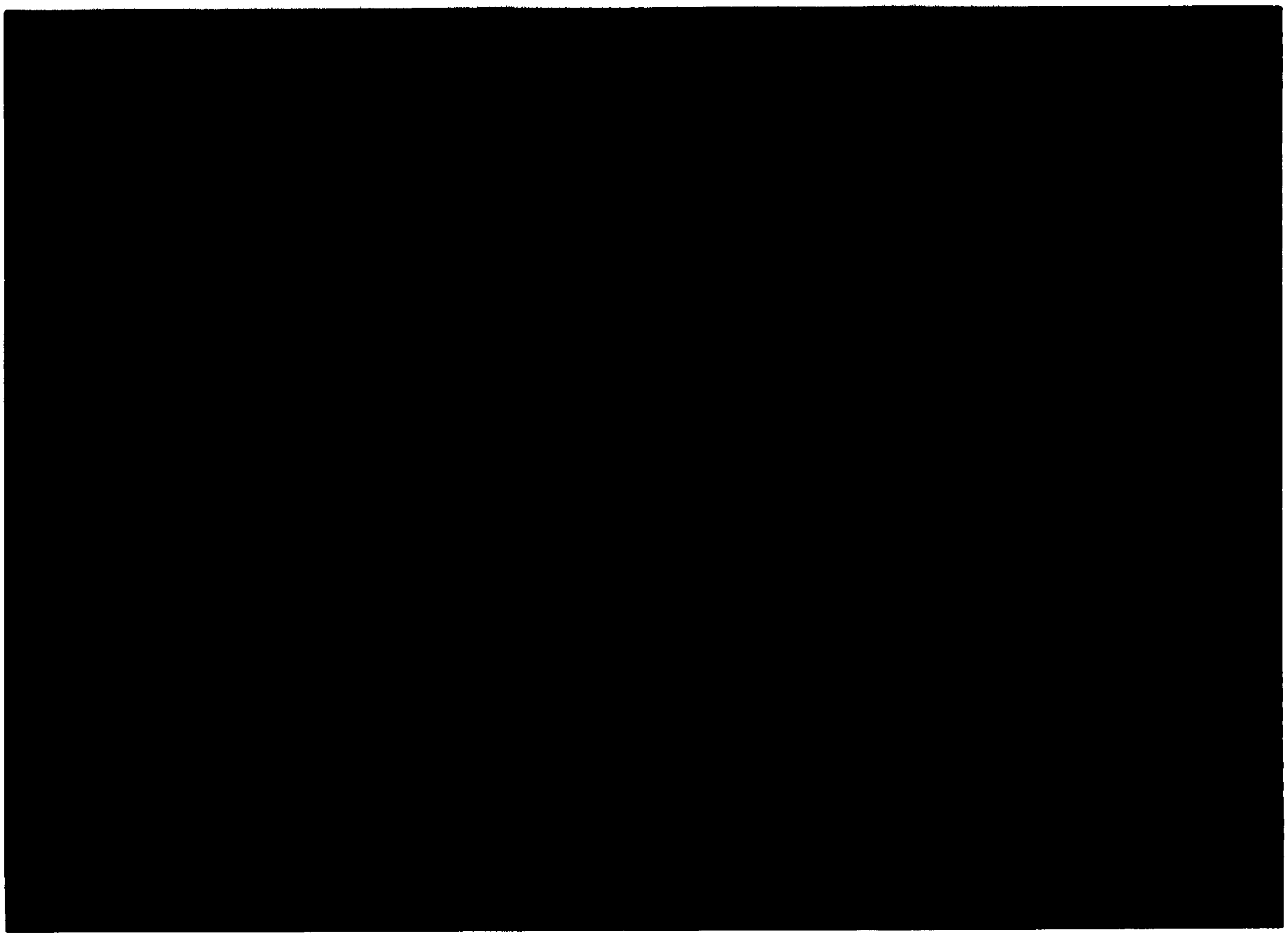
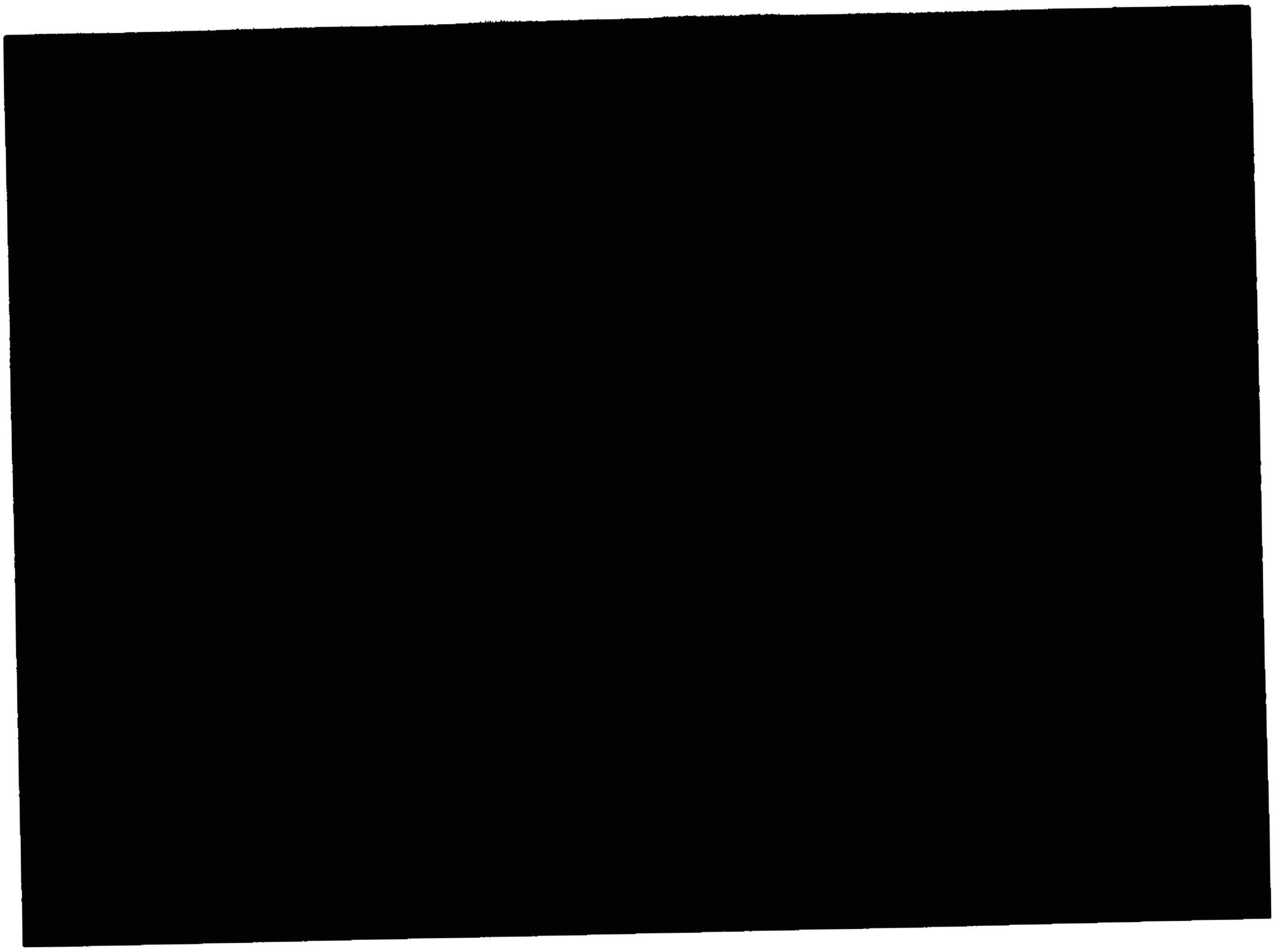


Figure 21. False color composite image (bands 1, 2, and 5 assigned to red, green, and blue color guns). a) Band 1.



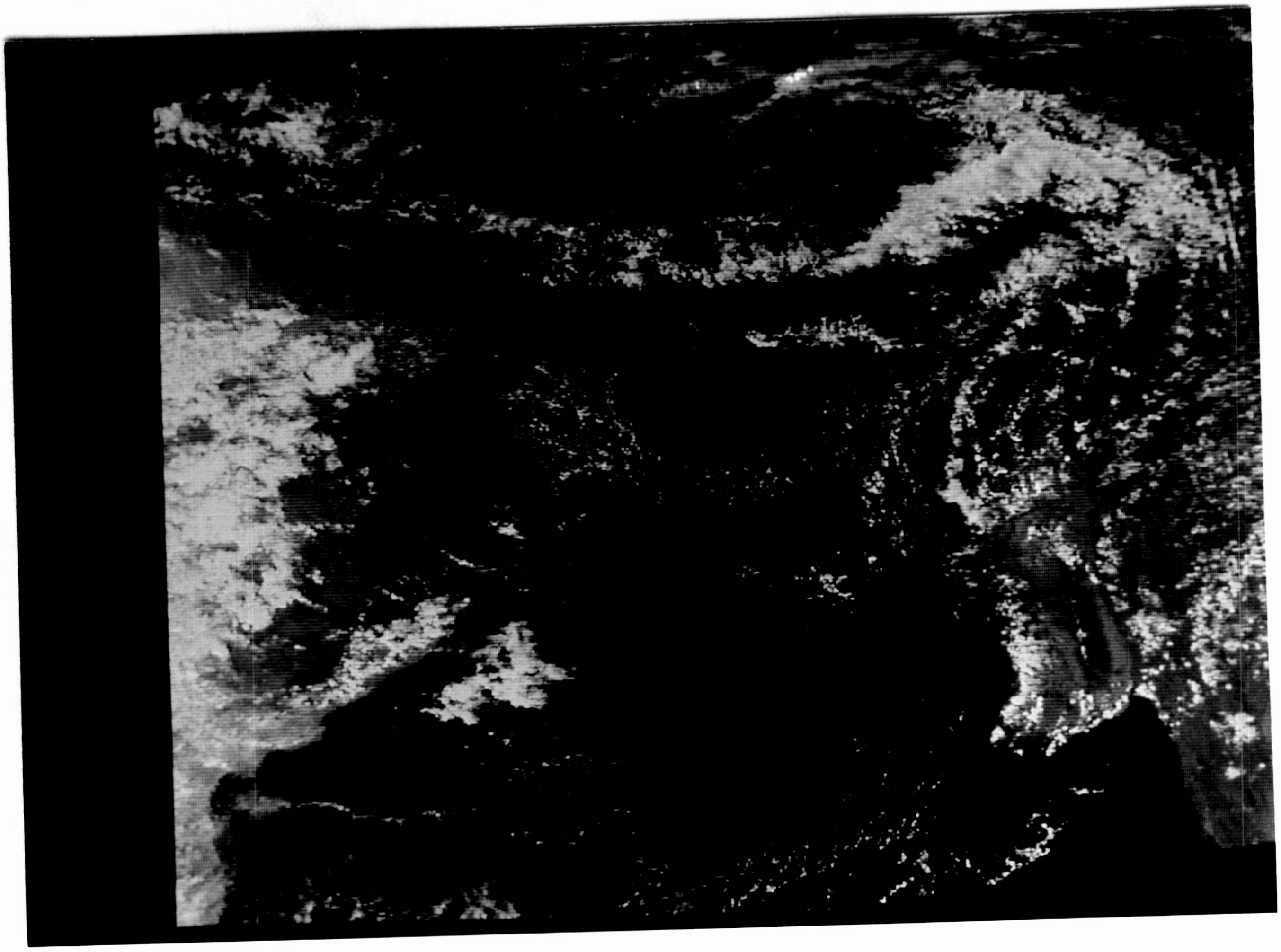
ORIGINAL PAGE
COLOR PHOTOGRAPH

Figure 21 (continued). b) Band 2.



ORIGINAL PAGE
COLOR PHOTOGRAPH

Figure 21 (continued). c) Band 5.



ORIGINAL PAGE
COLOR PHOTOGRAPH

Figure 21 (continued). d) Composite.

green, and band 3 to blue. If, however, the user wanted to assign the visible band to the green gun, the near infrared band to the blue gun, and the thermal band to the red gun (Figure 22), the following would be entered:

```
DISK (:3 1 2) > DISPLAY > $OUT'IMAGE
```

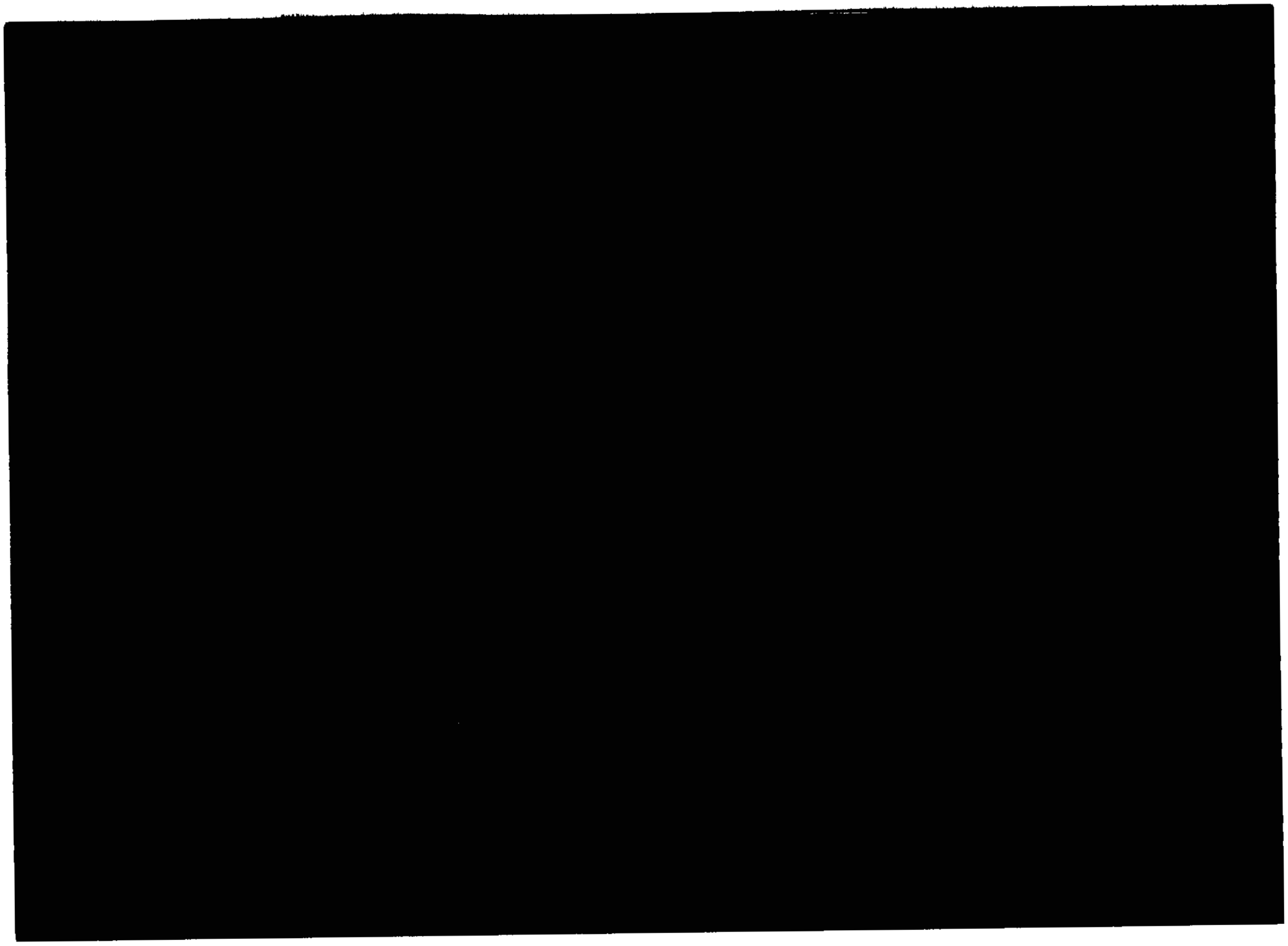
As this example also illustrates, the band assignment in a refresh image need not correspond with the band assignment in the disk image. This was discussed earlier in the section on "Data Transferal to Disk" (pgs. 98-99). Consider the following example:

```
> AVHRR OUTBANDS=1 2 5 > DISK'IMAGE

DISK'IMAGE (:3 2 1) > DISPLAY > $A
```

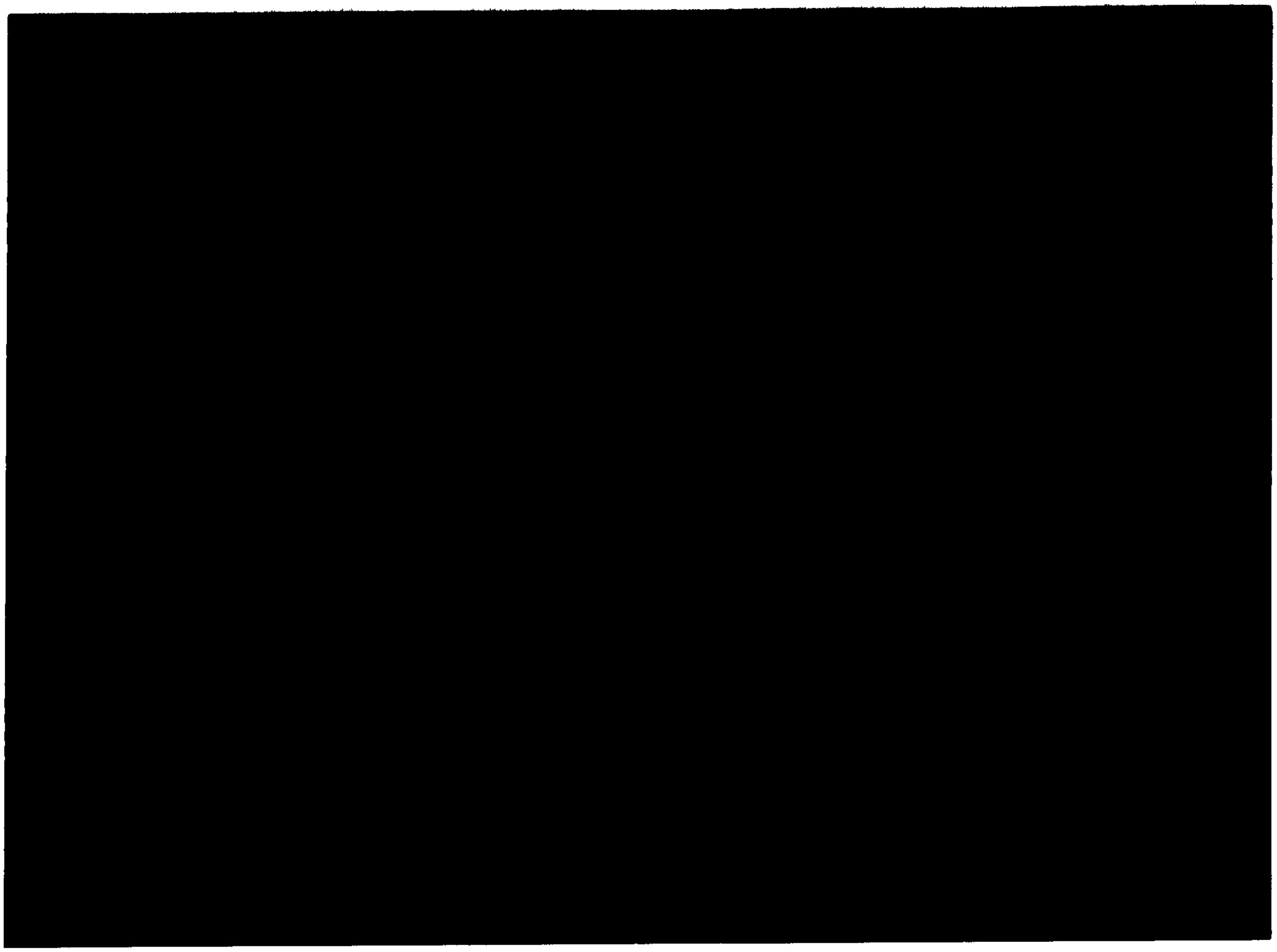
In the original NOAA AVHRR tape image bands 1, 2, and 5 correspond to the visible, NIR, and thermal infrared bands, respectively. These three bands are now designated bands 1, 2, and 3 on DISK'IMAGE. Because they were loaded into $A in reverse order, however, band 1 of $A consists of thermal infrared data, band 2 the NIR, and band 3 contains data from the visible band. This illustrates the importance of keeping track of the information that a particular band represents.

As was mentioned in the section "Refresh to Disk" (pg. 100), graphics planes can be saved to disk by using the



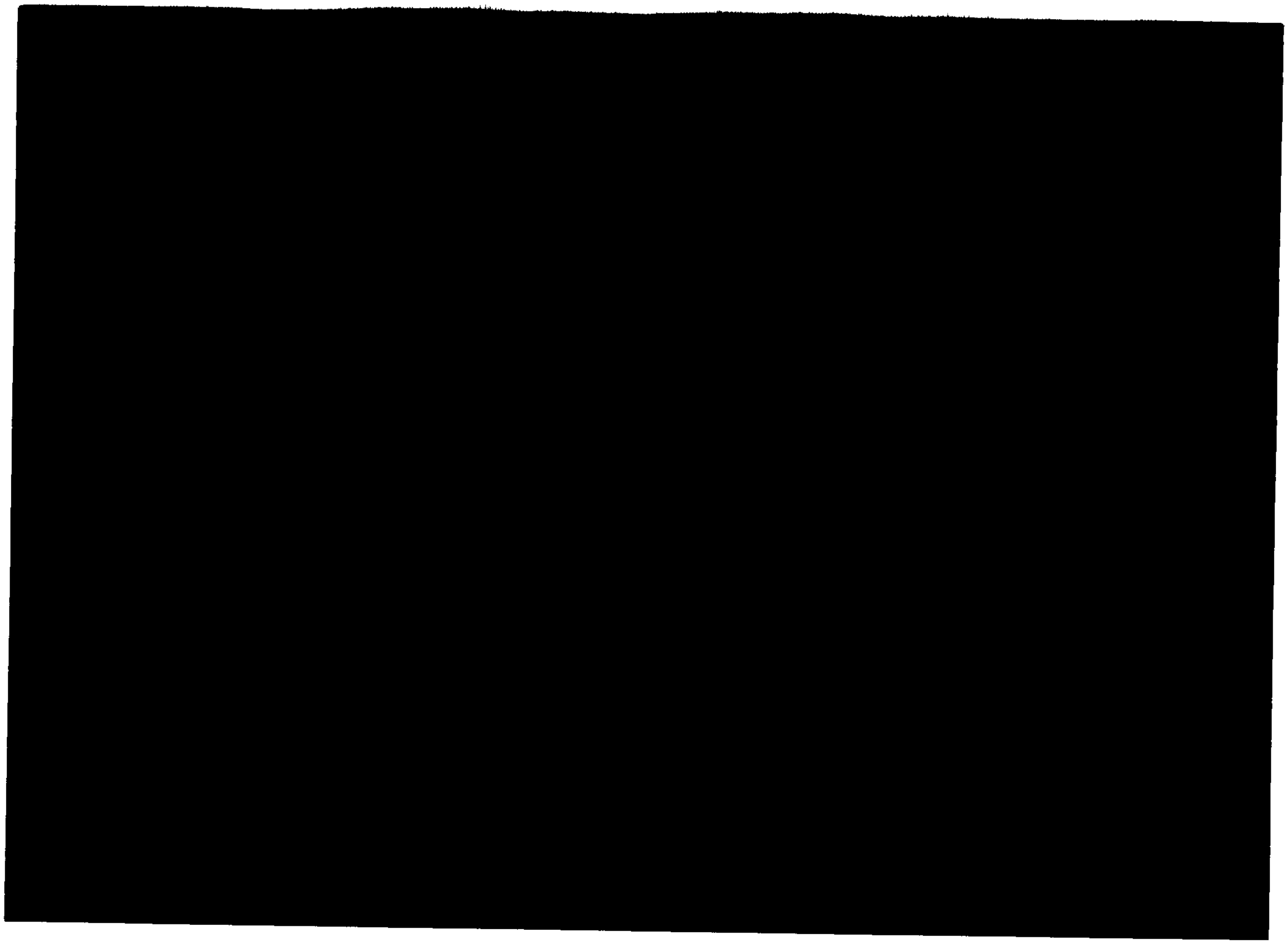
ORIGINAL PAGE
COLOR PHOTOGRAPH

Figure 22. False color composite image (bands 1, 2, and 5 assigned to green, blue, and red color guns). a) Band 1.



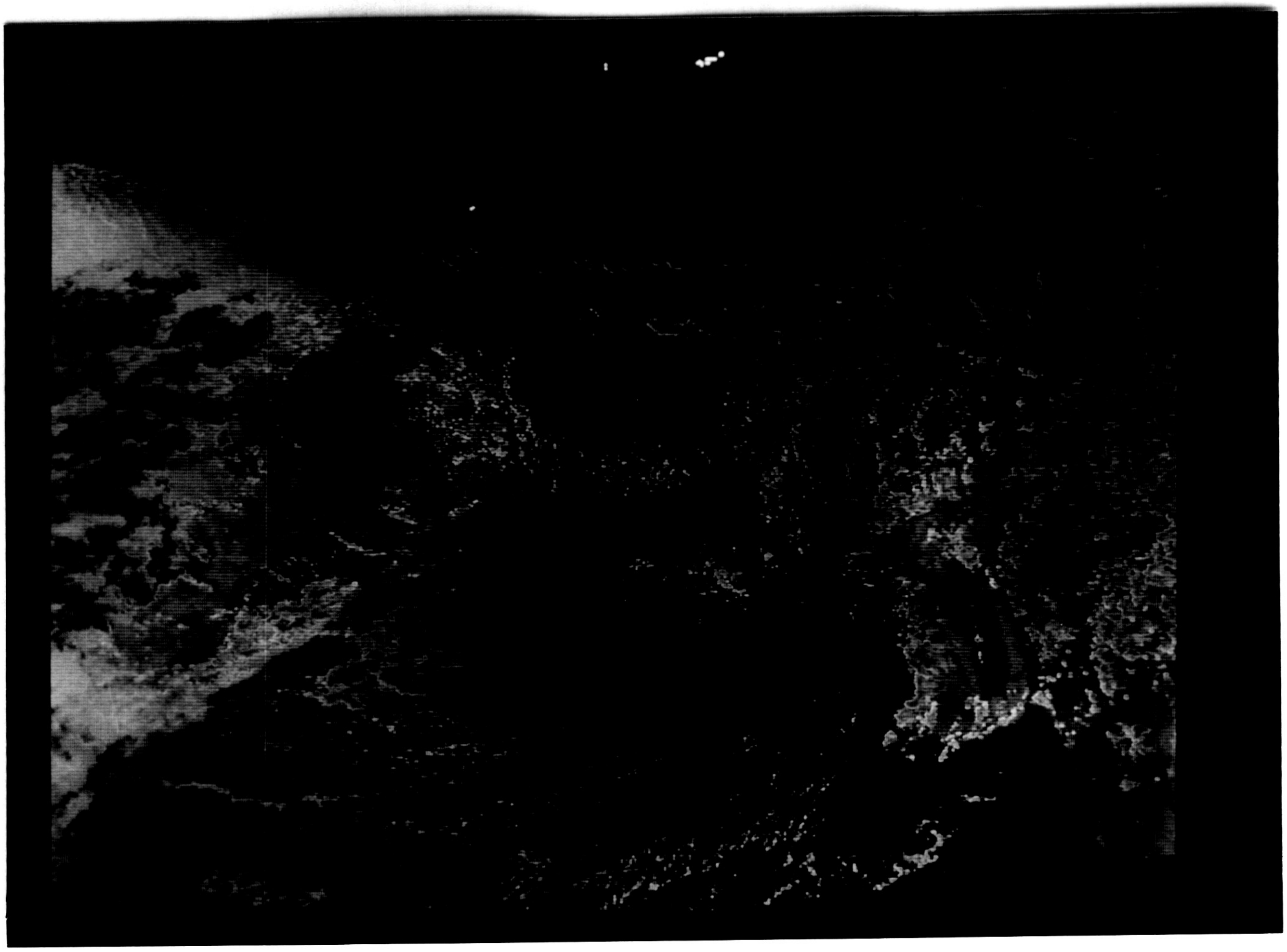
ORIGINAL PAGE
COLOR PHOTOGRAPH

Figure 22 (continued). b) Band 2.



ORIGINAL PAGE
COLOR PHOTOGRAPH

Figure 22 (continued). c) Band 5.



ORIGINAL PAGE
COLOR PHOTOGRAPH

Figure 22 (continued). d) Composite.

G'SAVE function. The inverse of this function is G'WRITE, which reads the graphics disk image to refresh. The format for this function is:

GDISK > G'WRITE

where GDISK is the name of the graphics disk image. If no parameters are specified, all the planes in the disk image are read in by default. Input planes can be limited by using the INPUTPLANES parameter.

The FETCH function is used to restore an I^2S environment that was saved by STASH. The syntax for FETCH is:

$IMAGE STASH'FILE > FETCH

where $IMAGE is the input image whose environment is being restored, and STASH'FILE is the file created by STASH.

Pipeline to Refresh

Many of the functions on the I^2S system operate on pipeline parameters, and not on the original data (this was discussed in the section on I^2S hardware, pgs. 75-76). As an example, consider the H'NORMALIZE function, which normalizes the histogram of a refresh image. This is accomplished by loading look-up tables with values that transform the original histogram into a normalized histogram. Because the transformation is through the look-up tables, the original refresh image is unaltered. The normalized image "exists" only in the pipeline and is not stored in refresh memory. For many applications, it is necessary to copy this pipeline "image" into a refresh plane. The FEEDBACK function is used for this. The format is:

> FEEDBACK > $IMAGE

Using FEEDBACK in this fashion produces a one band refresh image. If the image is composed of three bands, the COLOR parameter should be included.

Data Transferal to Pipeline

The DISPLAY function, discussed earlier, takes a disk image, stores it in refresh memory, and also passes the image through the pipeline and onto the monitor (the image is not transferred through the pipeline if the QUIET parameter is used). Thus, one way to display data on the monitor is through the DISPLAY function. The input to DISPLAY, however, must be a disk image. In order to display refresh images, the SELECT function must be used. The format for SELECT is:

```
$IMAGE > SELECT
```

As an example, consider the following sequence:

```
DISK1 > DISPLAY > $A

DISK2 > DISPLAY > $B
```

The first command causes DISK1 to be loaded into $A. It is also displayed on the monitor. When the second command is entered, $A is cleared from the monitor, DISK2 is loaded into $B, and $B is displayed. If the user wants to display $A on the monitor again, the following command is entered:

```
$A > SELECT
```

The SELECT command is similar to DISPLAY in that it can also be used to specify the assignment of bands to color guns. Consider the example above, where DISK1 is read into

$A. If DISK1 were a three band image, bands 1, 2, and 3 of DISK1 would correspond to bands 1, 2, and 3 of $A (this is the default assignment if the subband modifier is not used). When $A appeared on the monitor through the DISPLAY command, bands 1, 2, and 3 would be assigned to the red, green, and blue guns, respectively. Suppose the user then entered the following command:

```
$A (:3 1 2) > SELECT
```

This would cause bands 3, 1, and 2 to be assigned to the red, green, and blue guns. Thus, it is not necessary for the user to retain the original color gun assignment.

CHAPTER 6

CALIBRATION AND GEOGRAPHIC RECTIFICATION OF NOAA AVHRR IMAGERY

A set of three programs have been developed by NORDA and modified at LSU for calibrating and georeferencing NOAA AVHRR images. The package runs under the I^2S System 575 software environment and is fully compatible with other S575 functions.

The typical starting point in processing a NOAA AVHRR image is a level 1B raw data tape. This tape is first converted to standard I^2S format by running the AVHRR program, which can also radiometrically calibrate the image and select ground control points from the level 1B tape. Calibrated images contain pixel count values that represent either radiometric temperature for thermal channels or percent albedo for visible images. The process of producing a georeferenced image is somewhat more complicated. The wide field of view characteristic of the AVHRR scanner produces severe geometric distortions in the imagery, which must be corrected before an accurate mapping function can be calculated. The LCOPY program corrects for the geometric distortion by effectively stretching the image in the horizontal direction; its output is then passed to the AWARP program, which performs the final mapping. These three programs are described in detail in this chapter; a command summary of these programs is given in Appendix A.

Level 1B Tapes

The designation "level 1B" refers to a standard format for NOAA AVHRR data tapes obtained from SDSD (Satellite Data Services Division). These tapes contain one or more AVHRR data files, each of which consists of a 360-byte header record followed by many 7400-byte data records. GAC data contain 2 scan lines per tape record, while LAC data require 2 tape records for each image scan line.* Ground control points and radiometer calibration coefficients are embedded in each scan line of a level 1B image. Information about the exact format of a level 1B tape can be obtained in Kidwell (1984). The TBM (Terra-Bit Memory) data set name contains a wealth of information about the image (orbit number, time of day, date, etc.) and the reader is urged to look up its format in Kidwell (1984).

An annoying characteristic of level 1B data tapes is that there is frequently no way of knowing how many lines of AVHRR data are contained in a given file. Diligent users will note that there is a field in the level 1B format header that defines the number of scans contained in the

---

*There are two types of imagery available from NOAA AVHRR that differ only in their spatial resolution. LAC (Local Area Coverage) data have a nominal resolution of 1.1 km at nadir, and contain 2048 pixels in a single scan line. GAC (Global Area Coverage) data are created from LAC data by averaging adjacent pixels and subsampling scan lines, yielding a spatial resolution of 4.0 km at nadir. See Chapter 2 (pgs. 21-23) for further details.

data, but this number reflects the entire acquired pass (typically 4500 lines) and it is not updated for selective copies. The most reliable way to determine the size of the level 1B image is to count the number of records using one of the host computer's utilities. The number of scan lines can be calculated by subtracting 3 from the record count (for the 3 header records) and dividing by 2 for LAC or multiplying by 2 for GAC.

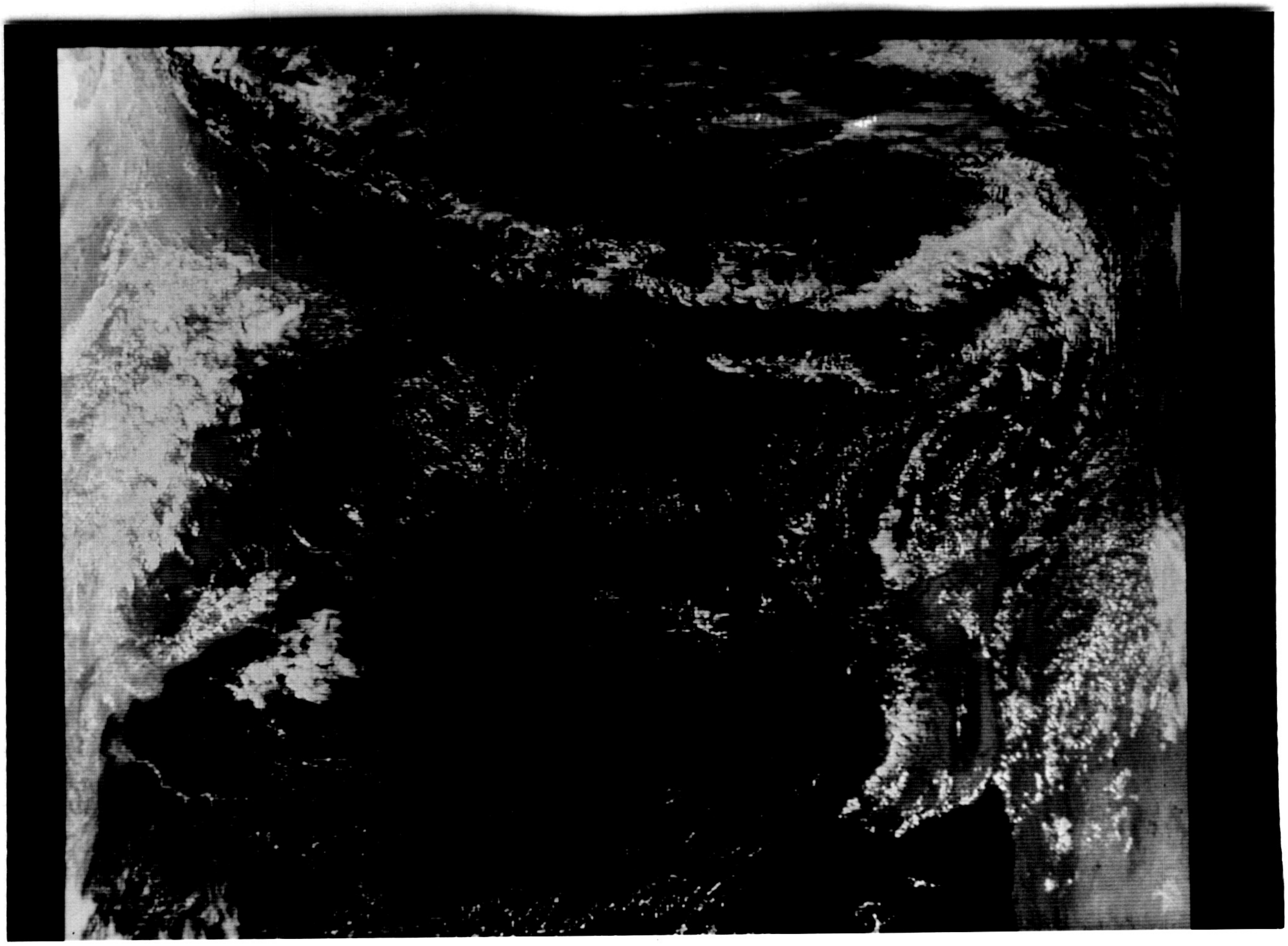
The AVHRR Calibration Program

The AVHRR program reads a level 1B tape and reformats the image data into a standard I^2S disk image file (Figure 23). At the user's option, the program can also radiometrically calibrate the image and extract ground control points from the level 1B tape for use in georeferencing the image. The map projection and grid cell size of the final georeferenced image is determined solely by the AVHRR program parameters, requiring the user to think ahead when specifying these parameters.

The AVHRR program produces a 16-bit (integer) output image, whether calibration is requested or not. Calibrated images contain pixel count values that are a linear function of either temperature or percent albedo. Thermal IR image temperatures start at -50 °C, for count value 0, and increase 0.1 degree C per count value. Thus, for any pixel value the corresponding temperature is given by:

$$\text{TEMPERATURE } (^\circ\text{C}) = (\text{COUNT} * 0.1) - 50.0$$

Calibrated visible channels start at 0% albedo for count value zero, and increase 0.1% for each pixel count value. Thus the maximum count value expected in a visible image is 1000 for 100% albedo. Uncalibrated visible and thermal IR images contain the raw 10-bit AVHRR pixel values transferred to the 16-bit output pixels. The maximum count value expected for uncalibrated images is 1024 (2^{10}).



ORIGINAL PAGE
COLOR PHOTOGRAPH

Figure 23. Sample output from AVHRR function.

Note that the 16-bit output images are excellent for data with a wide range of count values, like NOAA AVHRR data, but they are painfully slow to transfer to the display and they take up twice as much disk space as do 8-bit images. Therefore, it is recommended that the output image be stored on tape, then converted to 8-bit data for further processing. Two methods of converting 16-bit data to 8-bit data are truncation of range (which preserves resolution), and reduction of resolution (which preserves the dynamic range). Both methods can be implemented when the data are sent to the display for inspection, and the 8-bit display image can then be saved back to disk using the MOSAIC program, assuming the image is larger than 512 X 512 (see I^2S 1984 for details on MOSAIC). Converting data to an 8-bit range with the DISPLAY routine involves judiciously setting the MINCLIP and MAXCLIP parameters and omitting the SIXTEEN'BIT parameter. For visible images the pixel reflectances rarely exceed 25.5% albedo, so a MINCLIP of 0 and a MAXCLIP of 255 will truncate the range adequately. Remember that the range 0 to 255 contains 256 levels, and a range of 0 to 511 contains 512 levels; zero levels must be counted. For thermal IR images a MINCLIP of 500 and a MAXCLIP of 755 will produce a truncated range starting at 0 °C and ranging up to 25.5 °C. Similarly clipping from 500 to 1011 will reduce the resolution to 0.2 degrees per count value, resulting in a range of 0 to 51.1 °C.

The AVHRR program parameters allow the user to process any subset of lines, elements, and channels of the raw level 1B image. The SSAMPLE and SLINE parameters specify the sample and line number of the upper left corner of the desired subimage, and the NSAMPLES and NLINES parameters specify the size of the subimage. Note that there are only 2048 elements in a NOAA AVHRR scan line (409 elements for GAC). The OUTBANDS parameter selects the particular NOAA AVHRR bands to be processed. The CALIBRATE parameter enables image calibration on all selected channels when it is TRUE. There is no way to mix calibrated and uncalibrated channels in the same run.

The AVHRR program will not subsample the images that it processes. This is because the LCOPY program is better suited to perform subsampling, due to the scanning geometry distortions of the AVHRR sensor.

Several practical considerations will influence the selection of NOAA AVHRR channels for processing. Nighttime images will naturally contain little useful information in the visible channels, leaving only channels 3, 4 and 5 as candidates. The Greenwich Mean Time of any given orbit can be found in the TBM data set name. A hardware problem with the NOAA satellites has historically caused channel 3 to be noisy, with the noise level increasing as the satellite ages. The noise problem and the effect of solar reflection on channel 3 in daytime images causes that channel to be of limited use, and channel 3 is usually omitted from processing in the interests of economy or time.

The AVHRR program can process up to 8 images in a single run, providing that all images reside on the same input tape. Each subimage selection parameter will have multiple entries for the multiple output images. Unfortunately, all images processed in a multiple run will have the same channel numbers and calibration flag settings. Nevertheless, the use of the multiple run feature can save considerable wear and tear on the source tape, since it is rewound after each run, and it is recommended that this option is used whenever possible.

Sometimes a NOAA AVHRR image can be split across two level 1B tapes, with no indication of this condition other than an abrupt end-of-tape in the middle of the data. To address this situation, a MULTI-REEL parameter has been provided that, when set, causes the program to attempt to continue the run on another tape should an unexpected end-of-data be encountered. The program will rewind the current tape and request that the user mount the next tape and enter RETURN when ready.

Each NOAA AVHRR scan line contains the scan line number embedded in the housekeeping data. These scan line numbers are monitored by the AVHRR program to detect any scan lines that are missing from the data set, due to signal dropouts or other causes. Missing scan lines are filled with pixel value FFFF (hex) in order to preserve the spatial relationships of the AVHRR image. When missing lines are detected, the program will report "FILMSG ENTERED" upon entering the gap filling routine, and then report "EXIT" when the gap is filled. The scan line numbers should always be found in increasing order, and the program interprets an out-of-order scan as an error condition, reporting "NEGATIVE LINE NUMBER INCREMENT" and terminating the processing of that image. This error is often due to a tape I/O error, and while the finished scan lines will remain intact, the rest of the output image will contain garbage.

The specification of the ground control point parameters to the AVHRR program provides the sole determination of the spatial resolution and map projection of the georeferenced image that is output by the AWARP program in later processing. The ground control point parameters will not appear in the prompts if the CONTROL'POINTS parameter is FALSE. The NUM'CONTROL'POINTS parameter selects the desired number of control points and defaults to 128, which is the maximum number that the AWARP program can handle. The AVHRR program will attempt to distribute the control points evenly throughout the image, sometimes resulting in an output of slightly fewer control points than was requested. The ground control points will be calculated for a equirectangular map projection, unless the MERCATOR flag is TRUE, in which case a transverse Mercator projection will be used. The DELLONG and DELLAT parameters specify the spatial resolution to which the lat/long coordinates of the ground control points will be quantized. This will in turn be the resolution of the georeferenced image that is output by AWARP. The default control point spatial resolution is 0.01 degrees of longitude and latitude, which is about 1.1 km for the lower latitudes. For each image the AVHRR program will print the geographic boundaries of the ground control points calculated, which should represent a slightly smaller area than that covered by the image.

GAC data users may wish to specify a coarser resolution for the control point output. The default resolution will cause AWARP to expand the GAC image to LAC proportions, resulting in a blocky output image (this may be what is wanted). A resolution of 0.05 degrees per pixel would be more in line with true GAC resolution.

Tape Management

Level 1B tapes are expensive and possibly irreplaceable, and a word about tape handling and management is perhaps in order. All tapes should be kept in a temperature and humidity controlled environment, free from dust, magnetic fields, or direct sunlight. A tape library log book should be kept with one or more separate pages for each tape. New tapes should be logged as soon as they are obtained, and a log entry should be made every time the tape is used. Such a usage entry should contain the date and time of use, the portion of tape that was read (file and line numbers), and a notation of the success or failure of the read. Tape logs of this type are invaluable for gaging tape wear, and for diagnosing possible tape drive failures.

The 6250 bpi tapes and tape drives are particularly sensitive to contamination. The tape drives should be cleaned on a frequent and regular schedule (e.g., every other day), lest they deposit contaminants on the tape itself. Tapes must be handled with meticulous attention to avoiding dust, fingerprints, and especially cigarette smoke. Even the tape leader, which is never used for data, should be kept as clean as possible or it may deposit filth on the tape drive which can then be transferred to the tape's data area.

Tapes do wear out, and strategies that minimize source tape usage will pay off in the long run. Level 1B tape images should be calibrated only once (if possible) and the results stored on an output tape, which will be used from that time on. A notation should be made in the tape log that a given image has been calibrated, and resides on another library tape. This procedure can save both time and data.

There are also many hidden sources of magnetic fields in the average computer room which can cause havoc with tapes unless they are conscientiously avoided. Disk drives and line printers employ several high power magnets, and tapes should be kept away from them at all times. Also any electric motor is a potential danger, as are magnetic screwdrivers and keychains.

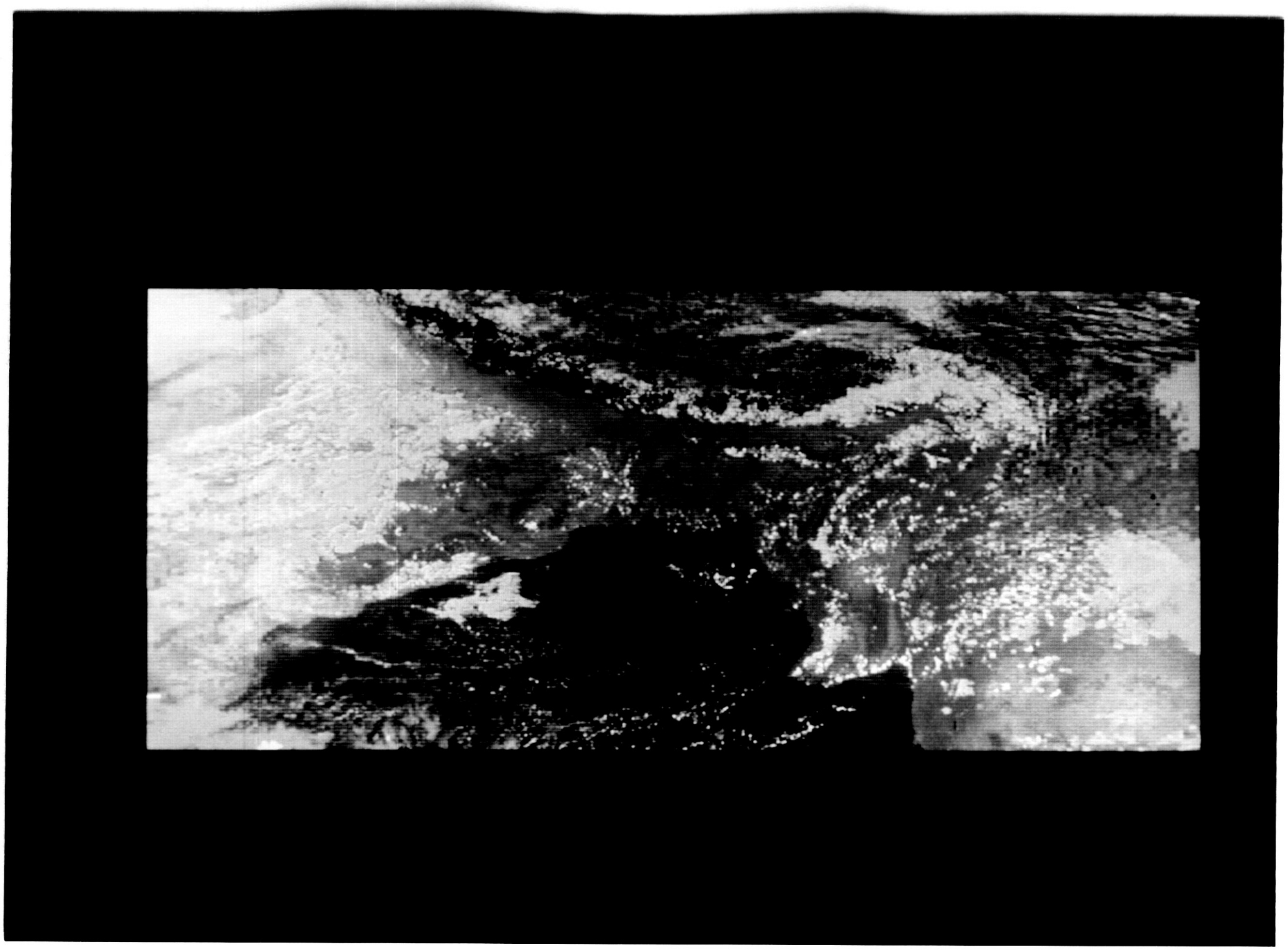
Any output that took more than 5 or 10 minutes to generate is a candidate for storage on tape. The AVHRR program consumes enough CPU time, wall clock time, and user patience to virtually mandate that its output images be stored on tape. Furthermore, the level 1B source tape can be spared from future wear and tear if users can find the image they want already calibrated on the more expendable output tape.

One most unfortunate feature of the standard I^2S image tape format is the fact that no header information of any kind is stored with the image on tape. This makes it mandatory that the user maintain impeccable records of the image size (lines and elements and channels), position on the tape (file number), and a description of the image contents for each image on tape. A log book should be kept for the tape library, with at least one sheet per tape containing all conceivable information about the image (or non-image) files on that tape. One method for managing the tapes in the library is to have one tape for each satellite orbit that has been analyzed, with the calibrated image and all subsequent products (e.g., 8-bit images, LCOPY images, AWARP images, etc.) stored on the same tape.

The LCOPY Program

The LCOPY program removes most of the spatial distortion that is due to the scanning geometry in NOAA AVHRR images (Figure 24). These spatial distortions, documented in Legeckis and Pritchard (1976), result from the combination of a wide scanning angle and the curvature of the earth, and appear in the image as a foreshortening of the scene at the image sides. The LCOPY program stretches the image in the horizontal direction by an amount that increases as the distance from nadir increases. Correcting the scan distortion before georeferencing the image can reduce the average fitting error from 5-6 pixels to less than one pixel.

The first five LCOPY parameters define the scanning geometry of the sensor, and the defaults are set to reflect the characteristics of LAC data. The SALTITUDE parameter is the satellite altitude, and the MAXSCANANGLE is the maximum scan angle of the sensor. The KM/PIXEL is the number of kilometers in the horizontal direction between the center points of adjacent pixels at the nadir position. The AVHRR sensor oversamples its data along the scan line and the KM/PIXEL is 0.8, even though the instantaneous field of view resolves 1.1 km pixels at nadir. The TILTANGLE parameter is intended for use with CZCS and defaults to zero. The TSAMPLES parameter is the total number of samples in an entire scan line, as encompassed by MAXSCANANGLE



ORIGINAL PAGE
COLOR PHOTOGRAPH

Figure 24. Sample output from LCOPY function.

degrees. The exact values of all of these parameters (with the possible exception of TILTANGLE) are not critical because the final georeference depends only on the control points (from AVHRR), and not on the exact dimensions of the LCOPYed image.

The scan geometry parameters must be redefined in order to process GAC data with LCOPY. GAC data have one-fifth the resolution of LAC data, with one-fifth as many pixels in the same sensor scan angle. Thus, for GAC data, KM/PIXEL = 4.0 and TSAMPLES = 409.

One of the most important parameters to set in the LCOPY program is the SSAMPLE (starting sample) parameter, which represents the location of the left edge of the AVHRR output image (the input to LCOPY) with respect to the left edge of the full NOAA AVHRR scan line. This is the same SSAMPLE number that was used to run AVHRR. If a subset of the NOAA AVHRR image is being processed, SSAMPLE still refers to the edge of the entire input image file. The amount of LCOPY edge stretching is dependent on the pixel position within the scan line, and an incorrect value for SSAMPLE will cause distorted output images. Unfortunately, the burden of remembering the starting sample number of the calibrated image falls on the user, since the S575 software assumes that all images start at line and element one. This SSAMPLE number should be written down in the tape log whenever a calibrated image is stored on tape.

The LCOPY program can subsample AVHRR images, but the syntax of the subsampling operation requires explanation. When the user specifies a subsampled image in the command line that invokes a program, that program receives the image data already subsampled from the image read subroutine. In the case of LCOPY, the scan line geometry has already been defined for the full resolution image, not a subsampled image. Thus, it is inappropriate for the user to specify subsampling in the elements direction when entering an LCOPY command line and the program states this if it is tried. Since LCOPY performs no geometric corrections in the lines direction, it is permissible to specify line subsampling in the LCOPY command line. The proper way to perform subsampling with LCOPY is to specify the SINCREM and LINCREM parameters for the desired subsampling. LCOPY will use any line increment specification given in the command line as the default value of LINCREM. An example of subsampling by 4 is as follows :

```
INPUT (1 1 2048 2048 1 4) > LCOPY SINCREM=4 > OUTPUT
```

This command would produce a 512 X 512 output image with the same number of channels as the file INPUT.

The user may notice that the output image from LCOPY looks blocky near the edges of the AVHRR scan line. This is completely normal, reflecting the fact that the edge pixel values represent the energy averaged over a much larger area than at nadir.

If the image is to be georeferenced, the LCOPY program must also be used to stretch the control point file, output by the AVHRR program. The reason is that the control point locations must be adjusted to match the same positions on the stretched image. The control point file is specified in the command line just after the corresponding image file is specified. Similarly, an output control point file name must be provided after the output image file name. The control point file contains coordinate pairs that refer to line and element numbers in the AVHRR output file. If a portion of the AVHRR output image is copied to another file, then the control point coordinates will not be correct for the subimage file (unless the subimage happens to start at line and element 1 of the AVHRR file). Therefore, LCOPY should always be run directly from the AVHRR image file that matches the ground control point file. Image subsections should always be done by using the input image modifiers in the LCOPY command line.

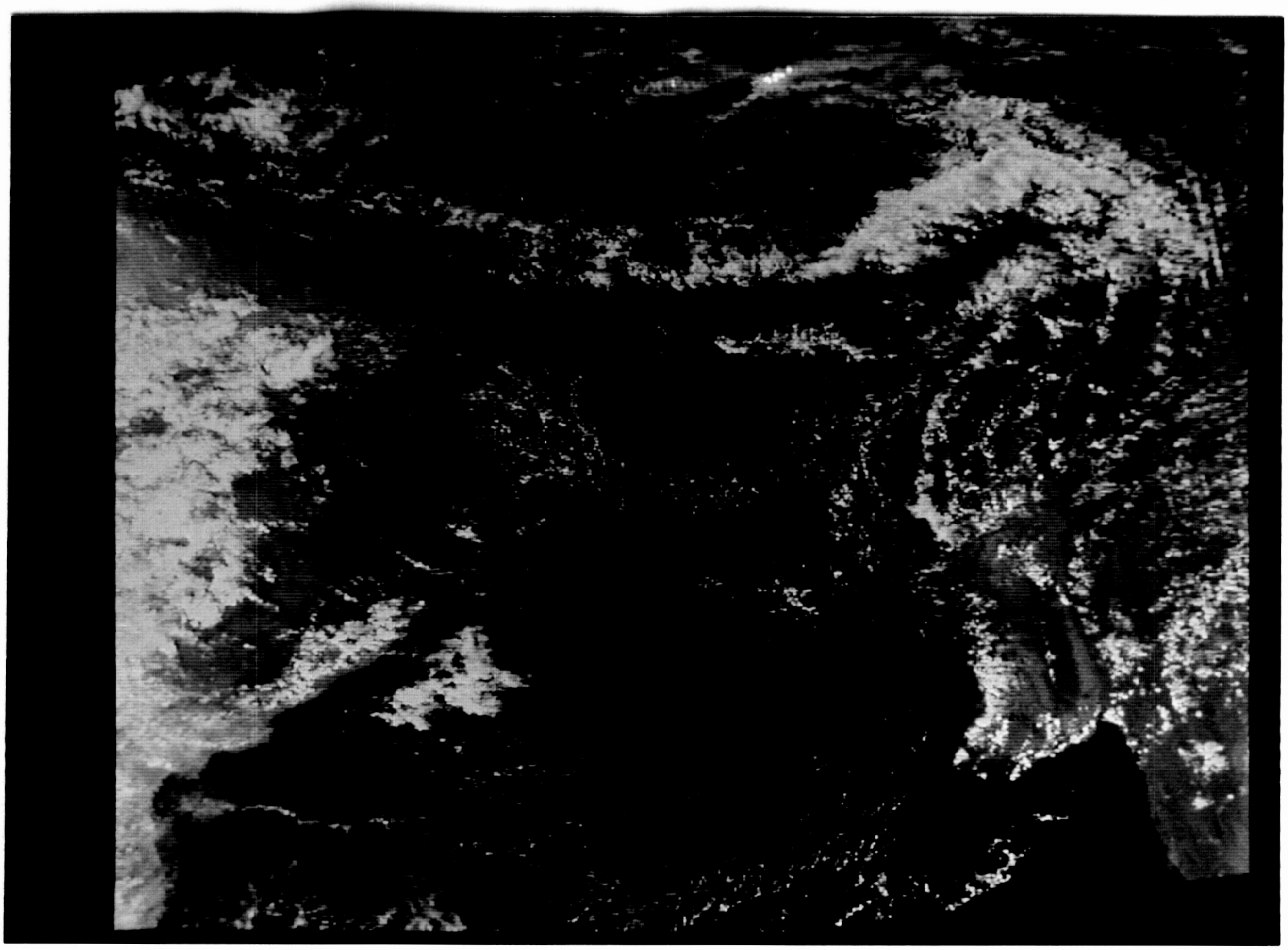
The control point file takes practically no time to process, and it can be regenerated, if lost or corrupted, by rerunning LCOPY and specifying that only one image line be processed. The single line image output is then thrown away, leaving the regenerated control point file. It is imperative that the user specify the same subimage for control points regeneration that was used to LCOPY the original image, otherwise the control points will not make sense with that image.

The AWARP Program

The AWARP program attempts to warp its input image to fit a set of control points that are either contained in a control point file or entered by hand (Figure 25). Both the input and output images must reside on the display. This limits the size of the image that can be warped, but it greatly increases the speed of the warping process. Adjacent sections of an image can be separately AWARPed and then pieced back together into a larger output image.

The S575 software tends to assume that all display images are 512 pixels square, and this forces the user to correctly input several vital parameters relating the input image size and position. The first case that warrants attention is that in which a subsection of an LCOPYed image is loaded into the display to be AWARPed. The ground control point file contains pixel coordinates that are defined relative to the upper left corner of the LCOPY image, and unless they are adjusted to match the image subsection on the display they will be inaccurate. The SSAMPLE and SLINE parameters are used to indicate the coordinates of the upper left corner of the displayed subimage, relative to the entire LCOPY image. If they are inaccurate, the image will not warp correctly.

The INAREA parameter is used any time the input image is larger than 512 pixels in either direction. The software



ORIGINAL PAGE
COLOR PHOTOGRAPH

Figure 25. Sample output from AWARP function.

will assume that any multiplane image is a 512 X 512 multichannel image, unless the INAREA parameter specifies otherwise. Thus, an INAREA = (1 1 1024 1024) prevents a four-plane image from being interpreted as a 512 X 512 image with 4 channels. For multichannel images, the true number of image channels is found by dividing the total number of planes that the image occupies by the number of planes required for a single channel. If the image in the previous example occupied 8 planes, it would be interpreted as a 2 channel 1024 X 1024 image.

The accuracy of the ground control points in a level 1B tape is enough to match a pair of images to within 3 pixels over a 512 X 512 pixel image. The I^2S REGISTER program can be used to slide two images together for the best match. This step must be done by hand, since the S575 software does not maintain the absolute lat/long coordinates of a georeferenced image. AVHRR channel 2, with its crisp land/water separation, is ideal for registration of daytime images. Naturally, image registration should be evaluated based on the most stable land features that can be found. One good method of checking image registration is to assign one image to the green color gun, and the other image to the red gun. The misregistrations will appear clearly as red and green outlines on the yellow composite image.

If greater registration accuracy is desired, then the regular WARP program can be used either to match the two (or more) images to each other or, more practically, to match each image to a World Data Bank map of the area. The CONTROL'POINTS program is very useful for picking control points for this extra warping step, and it is also useful for examining the level 1B control point files as they progress through the AVHRR and LCOPY programs.

Note that the AWARP output image will most often be larger than 512 X 512 pixels, and the MOSAIC program must be used in these cases to save the image from the display to disk. This is due to the same display image size ambiguity that necessitates using the INAREA parameter in the AWARP program. If only one 512 square subimage is required, the desired subimage can be found with ROAM, and then FEEDBACK and SAVE can be used to store it on disk.

CHAPTER 7

PREPROCESSING IMAGES ON THE I^2S

Image processing usually consists of two somewhat distinct phases: preprocessing, where the data are radiometrically and geometrically corrected and then enhanced for image examination; and product generation, where specific thematic maps are created. Chapter 6 (pg. 119) describes radiometric calibration and geometric correction of NOAA AVHRR data using the AVHRR, LCOPY, and AWARP programs. In this chapter image enhancement and image coloring techniques are described. Also included is a discussion of functions used for examining and labelling images.

Image Enhancement

Image enhancement is one of the more important processes performed in image analysis since it allows the user to bring out features of interest that were not previously visible. There are two general classes of image enhancement techniques: point operators and neighborhood operators. This discussion will focus on the use of I^2S point operation functions. Neighborhood operators will only be mentioned briefly.

Point Operators

A point operator is any function that replaces a specific pixel reflectance value with a new reflectance value solely based upon some property of that single pixel. This contrasts with a neighborhood operator, where the new reflectance value is a function of not only that pixel but a neighborhood of surrounding pixels. An example of a point operator would be adding a constant to each pixel's reflectance value, whereas an example of a neighborhood operator would be replacing a pixel value with the average value of the 3 X 3 surrounding neighborhood.

Point operator enhancement techniques allow the user to change the relationship between data reflectance values and monitor intensity values. Consider 8-bit data, where the reflectance values range from 0-255. The I^2S monitor is capable of displaying 256 grey shades, or intensity values, from 0 (black) to 255 (white). Normally the data are displayed on the monitor using a linear function with a slope of one, so that the reflectance values are identical to the monitor intensity values. In this case a reflectance value of zero would have an intensity value of zero and a reflectance value of 255 would have an intensity value of 255. It is possible, however, to use some other function for mapping reflectance values onto intensity values. For example, an inverse linear function could be used so that reflectance values of zero are given an intensity value of 255, reflectance values of 255 are given an intensity of 0, etc.

There are many reasons why a user would want to perform this kind of operation. For example, thermal data are often displayed with an inverse linear function so that the cold clouds (low data values) are displayed in white (high intensity values). Another reason for altering the relationship between data and intensity values would be to highlight a low reflectance feature. Further, this type of operation can increase the contrast of a scene. For example, consider some image where the reflectance values range from 0-63. In this case, a linear mapping with a slope of one would result in a low contrast image, since only dark values would be displayed. If, however, the display mapping function multiplied each reflectance value by four and displayed the product, the data would then vary over the entire range of 256 display values. Multiplication of reflectance values thus "stretches" the image and increases contrast.

As was just discussed, an image is by default displayed using a linear mapping function with a slope of one. The TLM function allows the user to alter the slope of this mapping function (Figure 26). TLM operates on refresh images by altering LUT values. This function can operate on the entire image or on a subregion defined by a blotch plane (blotch planes are discussed with the BLOTCH function, pgs. 171-173). TLM allows the user to change the slope of the mapping function by manipulating the trackball. For multispectral images, bands can be transformed separately or

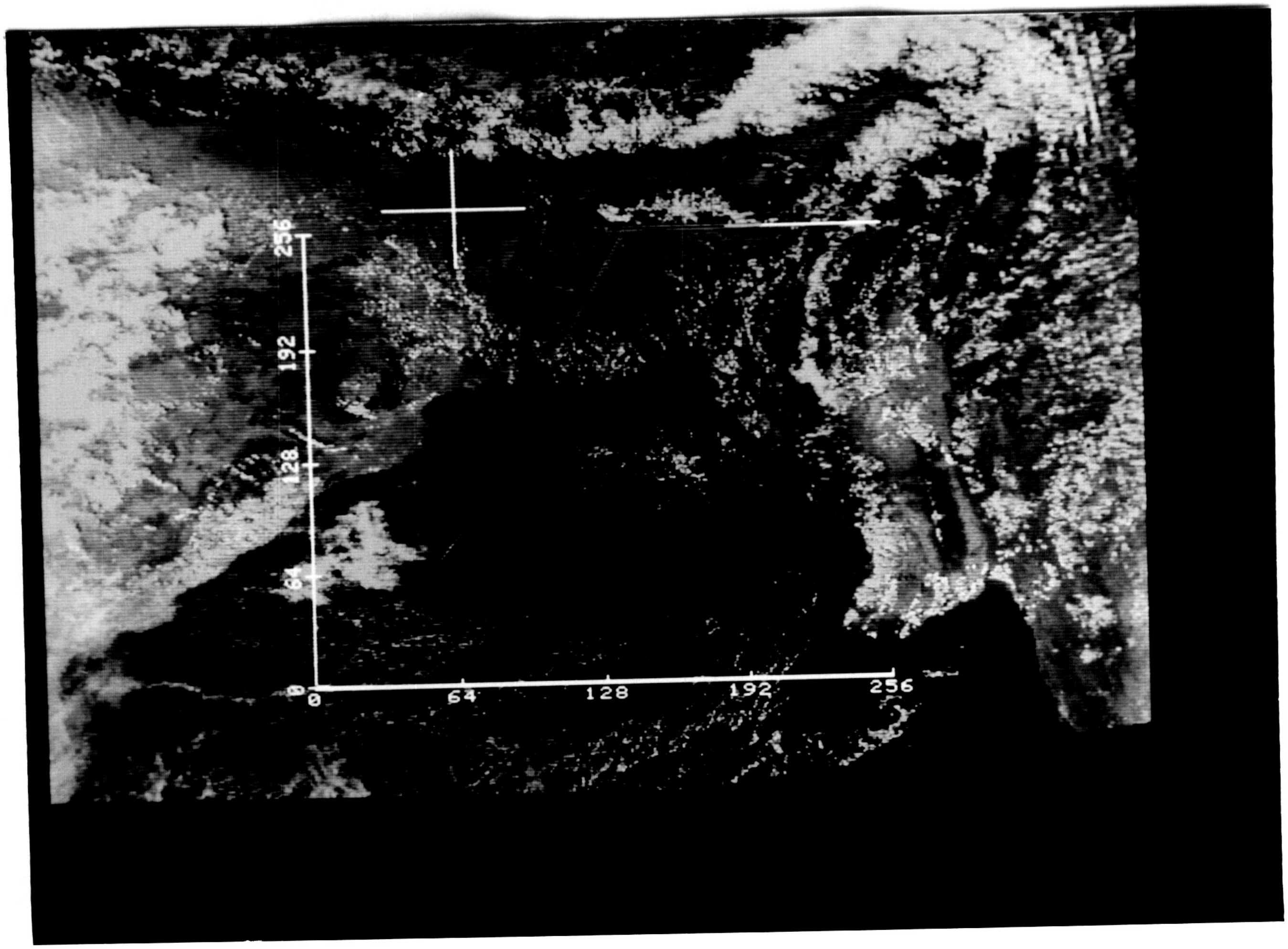


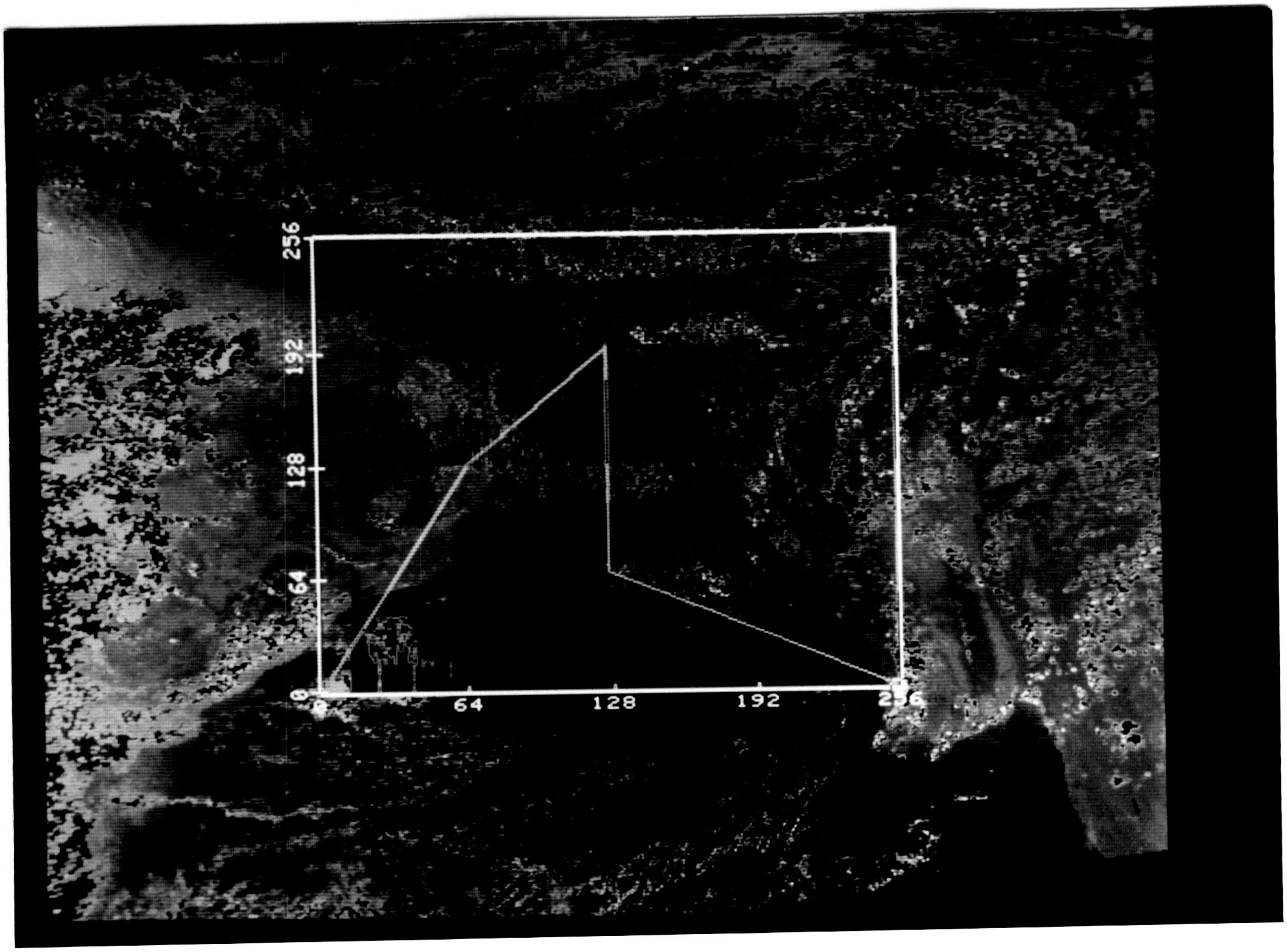
Figure 26. Image enhancement with the TLM function. Note that the red, green, and blue bands have been enhanced with separate linear functions.

together by use of the function buttons on the trackball unit. The format for TLM is:

$IMAGE > TLM

Since TLM only affects the system's look-up tables, the original refresh image is not altered. Thus, if the user wishes to save the transformed image, the FEEDBACK function must be used (pgs. 115-116). The necessity of using FEEDBACK to save transformed images is true for all of the I^2S enhancement functions discussed in this section.

The PIECEWISE'LINEAR function also allows the user to alter the mapping function. However, in this case the mapping function need not be linear. Instead, the user can design a mapping function composed of many small linear segments. For example, reflectance values of 0-64 could be mapped onto intensity values 0-128, reflectance values of 65-128 could be mapped onto intensity values 128-192, and reflectance values of 129-255 could be mapped onto 64-0 (a negative mapping). This is illustrated in Figure 27. The PIECEWISE'LINEAR function achieves this by allowing the user to define "breakpoints," where each breakpoint pair defines a separate linear segment. As was the case with TLM, PIECEWISE'LINEAR can operate on the whole image or on a subregion defined by a blotch plane. It can also operate on bands simultaneously or independently. The format for this function is:



ORIGINAL PAGE
COLOR PHOTOGRAPH

Figure 27. Image enhancement with the PIECEWISE'LINEAR function. Data values 0-64 are stretched onto display values 0-128, data values 65-128 are displayed as 128-192, and data values 129-255 are displayed as 64-0. Lower histogram is for raw data and upper histogram is for transformed data. Only green band (band 2) has been transformed; red and blue bands (bands 1 and 5) are unchanged.

$IMAGE > PIECEWISE'LINEAR

Clipping functions are another way to stretch (or compress) data values. For example, using a low clip value of 100 and a high clip value of 200 will stretch reflectance values of 100-200 onto a 0-255 intensity range. Clipping can be performed with the MINCLIP and MAXCLIP parameters of DISPLAY (pgs. 102-104). The SCALE function also performs image clipping. Here the clipping parameters are CLIPLOW and CLIPHIGH. The difference between clipping with DISPLAY and SCALE is that DISPLAY clips the image as it is stored in refresh, and thus the refresh image contains clipped data. SCALE, however, clips a refresh image by changing look-up table values. The refresh image is not affected. The syntax for SCALE is:

$IMAGE > SCALE CLIPLOW=VAL1 CLIPHIGH=VAL2

where VAL1 is the minimum value to be displayed and VAL2 is the maximum.

The remaining two functions are histogram operations. The histogram is a frequency plot of reflectance values (Figure 28). A histogram can be printed by entering the following command:

$IMAGE > HISTOGRAM

HISTOGRAM assumes the image is a three band, color image unless the BW (black and white) parameter is

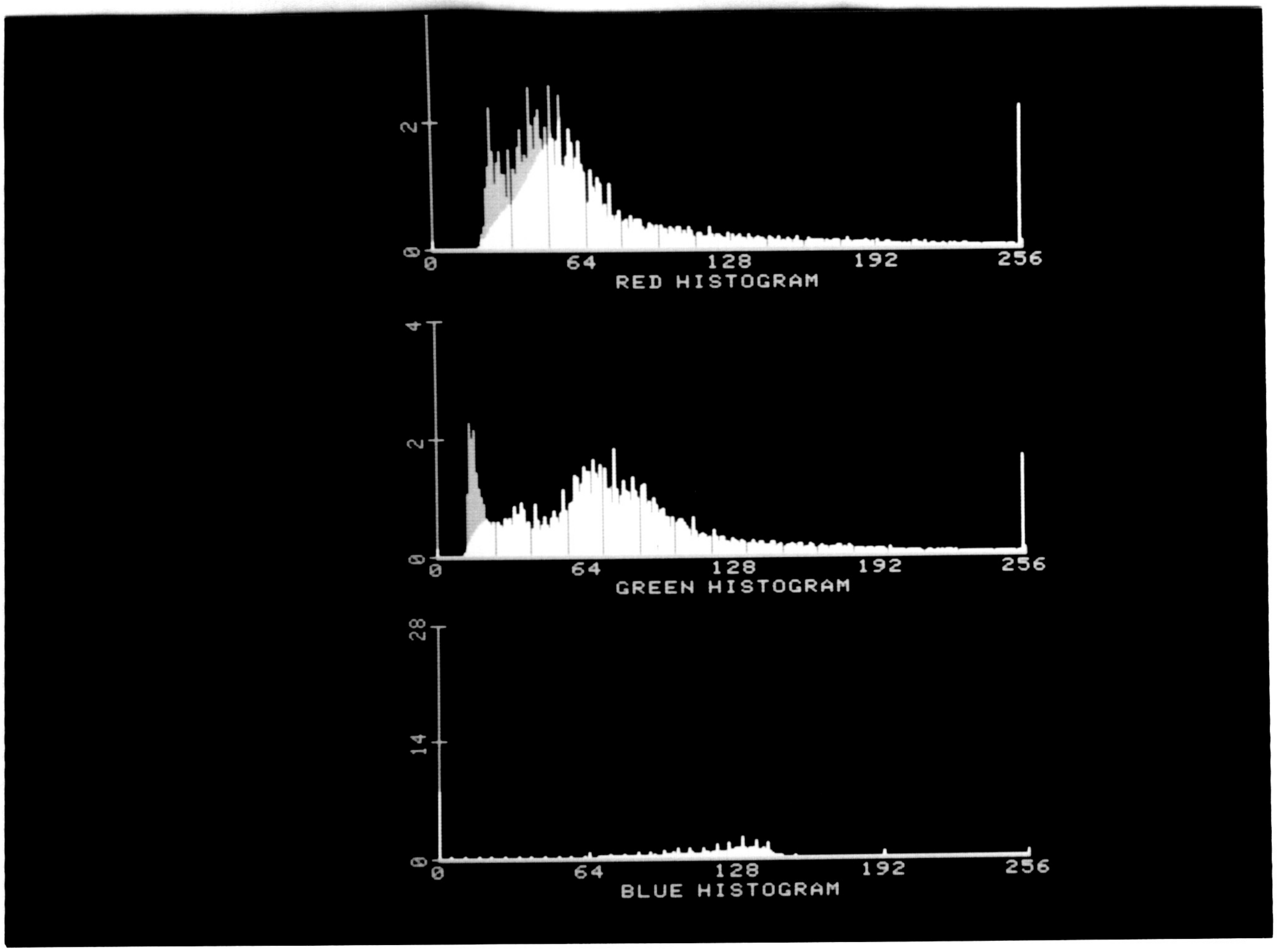


Figure 28. Histogram of unenhanced image (Figure 25). Blue curve is cumulative histogram.

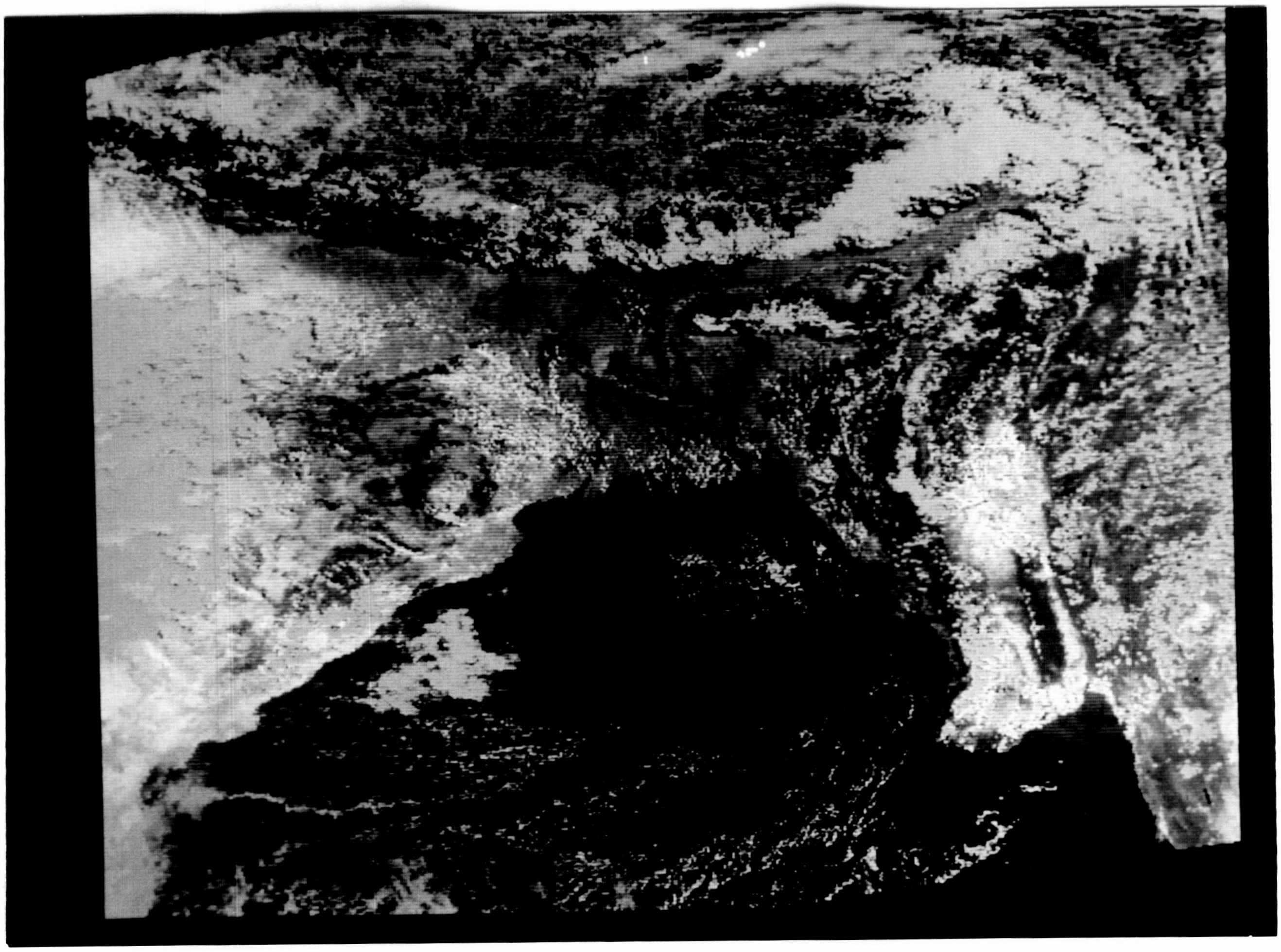
specified. If the CUMULATIVE parameter is included, the cumulative histogram will also be plotted (the value of the cumulative histogram at data value X is equal to the total number of pixels having a data value less than or equal to X).

The H'EQUALIZE function "equalizes" the histogram of an image (Figure 29). This means that the data values are transformed so that the resulting image has a linear cumulative histogram (Figure 30). The result of this transformation is to compress areas of the original histogram that are sparsely populated and to stretch those areas that are densely populated. H'EQUALIZE is one of the quickest and most effective enhancement functions available on I^2S. It is simple to use and dramatically brings out features that might not have been obvious. The format for this function is:

$IMAGE > H'EQUALIZE

H'EQUALIZE can operate on the entire image or on a blotch defined subregion.

The H'NORMALIZE function performs a histogram normalization (Figure 31). This function is similar to H'EQUALIZE, except the function attempts to transform the histogram into a normal (i.e., gaussian) histogram (Figure 32). As with all of the enhancement functions discussed here, H'NORMALIZE can operate on an entire image or a



ORIGINAL PAGE
COLOR PHOTOGRAPH

Figure 29. Image enhancement with the H'EQUALIZE function.

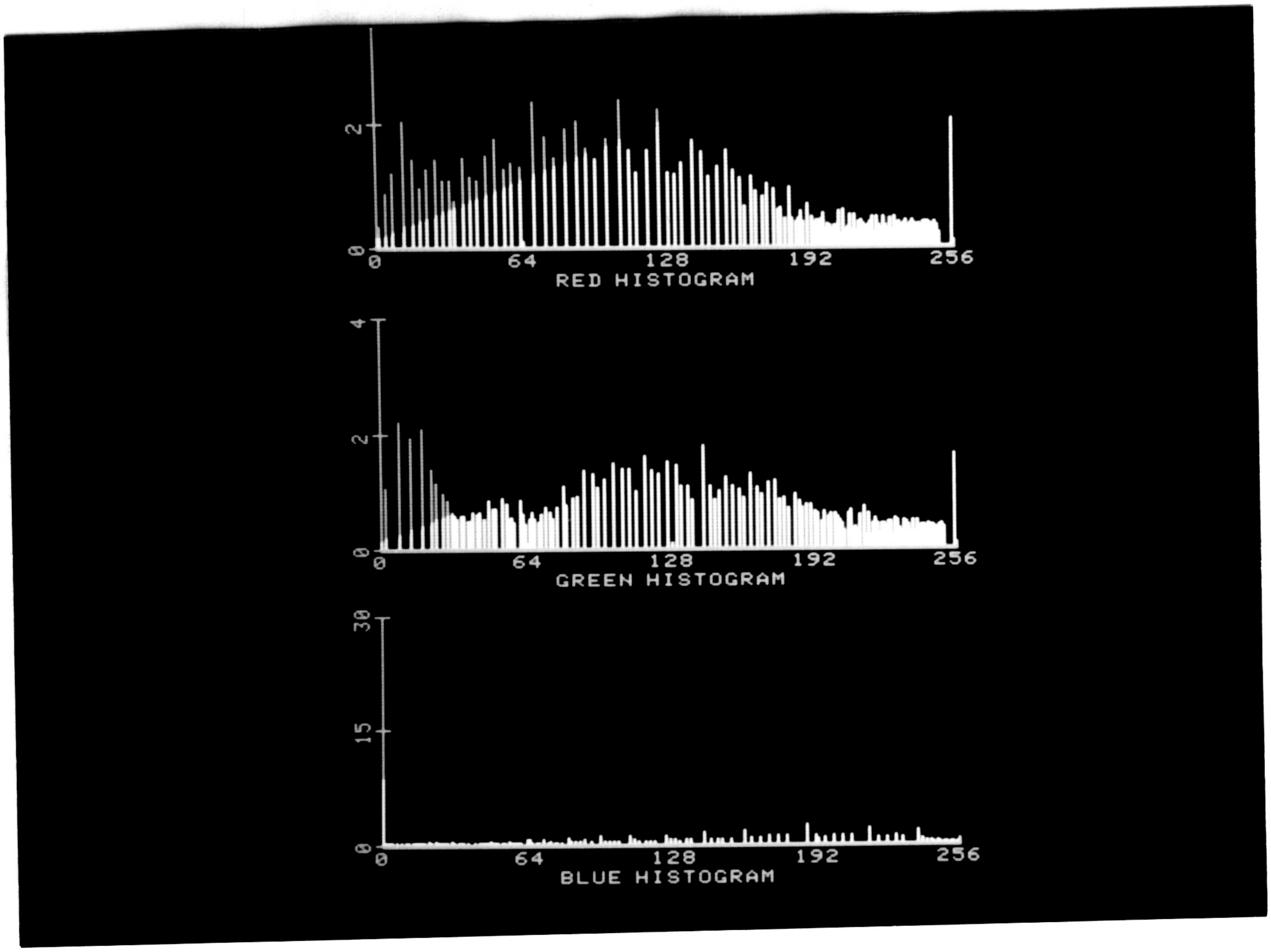


Figure 30. Histogram of image produced with H'EQUALIZE (Figure 29). Blue curve is cumulative histogram.

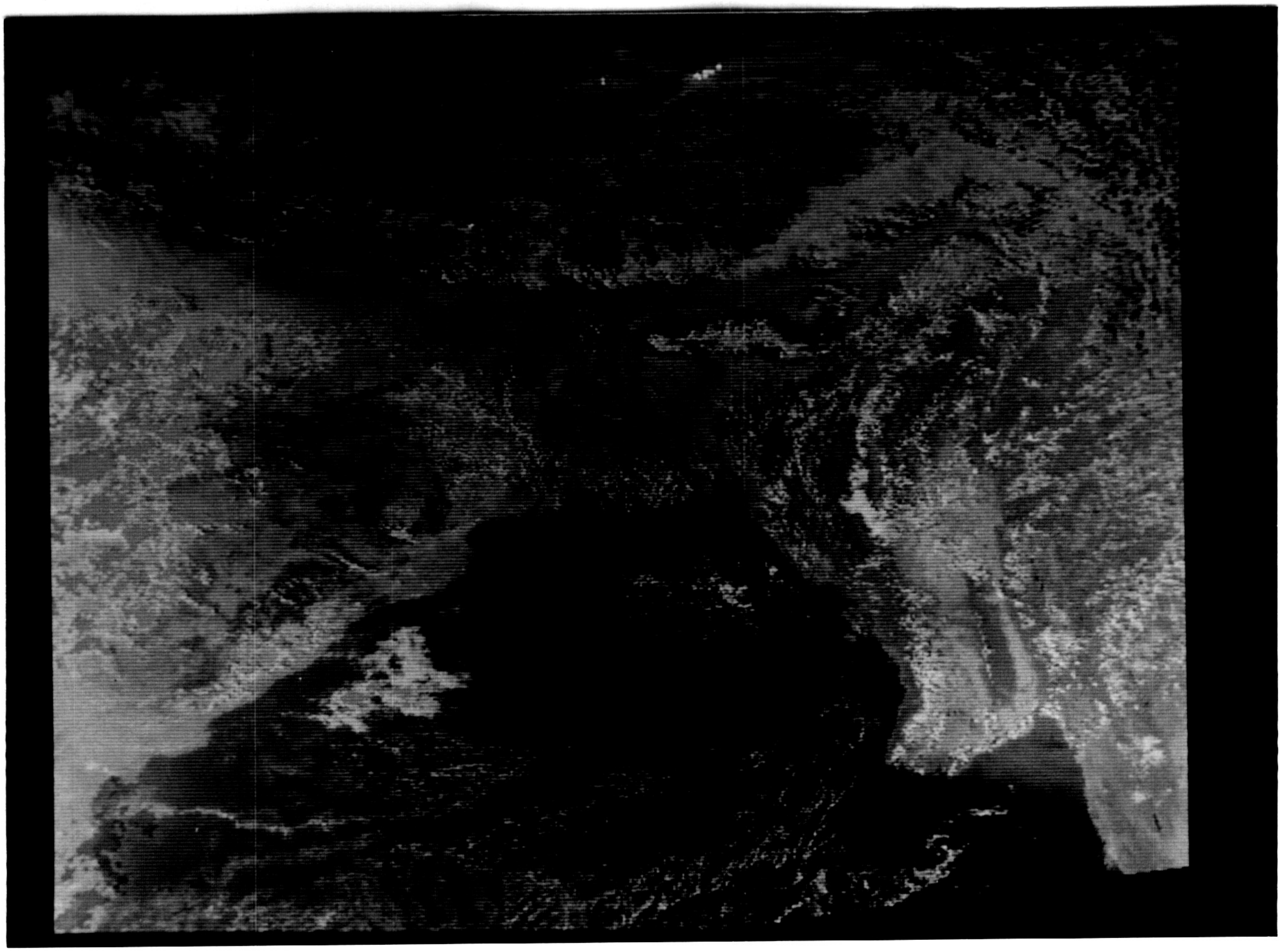


Figure 31. Image enhancement with the H'NORMALIZE function.

ORIGINAL PAGE
COLOR PHOTOGRAPH

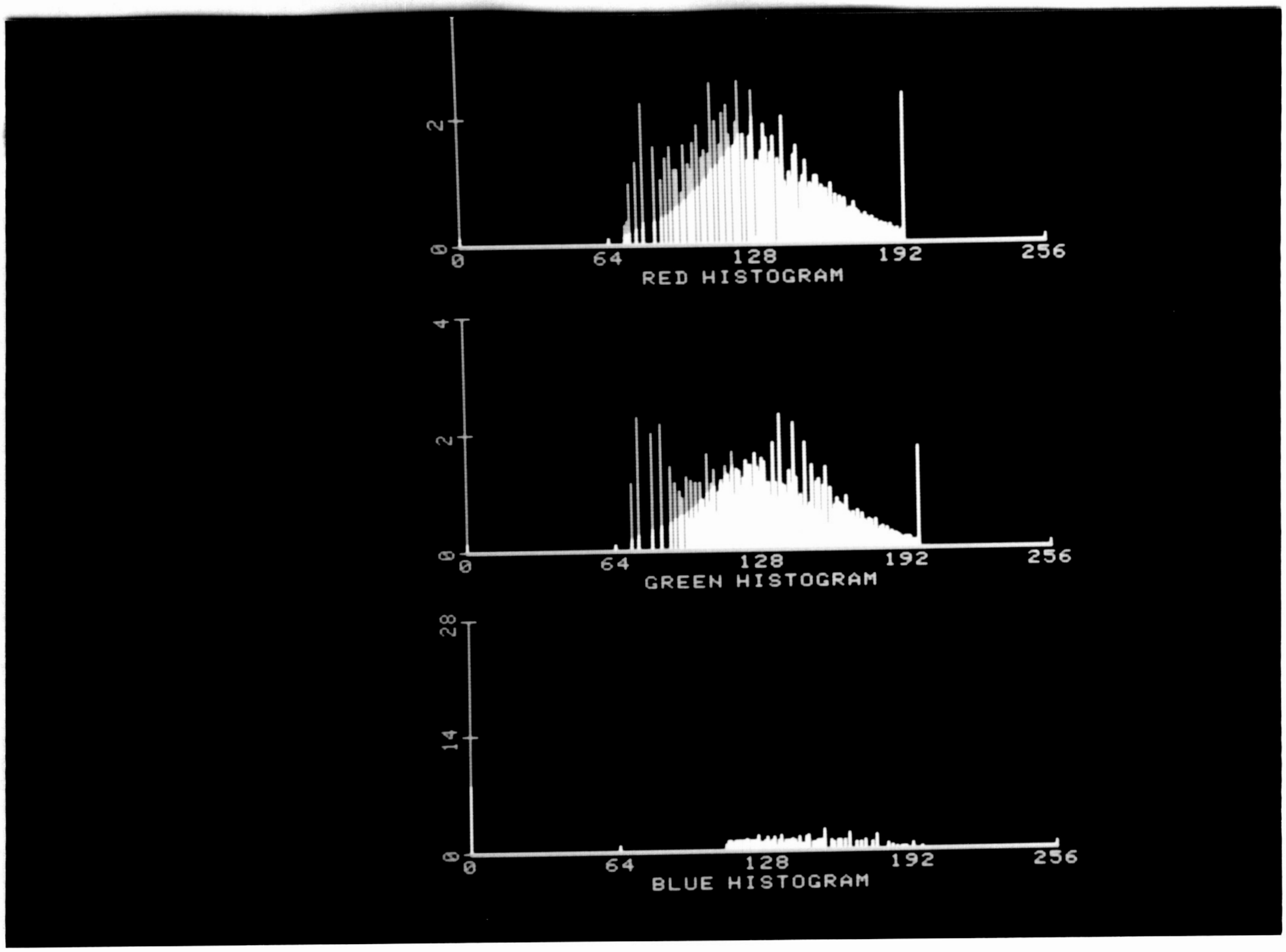


Figure 32. Histogram of image produced with H'NORMALIZE (Figure 31). Blue curve is cumulative histogram.

subregion. The format for this function is:

$IMAGE > H'NORMALIZE

The STANDARDEVIATION and MEAN parameters allow the user to specify the standard deviation and mean of the resulting normal histogram. Default values are 32 and 127.99, respectively.

Neighborhood Operators

As was previously discussed, a neighborhood operator produces an output value that is a function of the pixel and a group of surrounding neighbors. Neighborhood operators are generally used for noise removal, data filtering, and edge detection. The use of these operators is beyond the scope of this manual. However, a partial list of some of the I^2S neighborhood operators follows: COMPASS, CONVOLVE, FFT2D, GAUSS'FILTER, IFT2D, LAPLACE, and WALLIS. Details on these functions are available in I^2S (1984).

Image Coloring

It is often desirable to add color to a black and white image. Three ways of image coloring will be discussed in this section: the COLORS function, the PALETTE function, and using the CONSTANT function to add dummy bands that turn a single-banded (monochrome) image into a three banded (color) image.

The COLORS function allows the user to add user defined colors to specific data values. Each color can be assigned one or several data values. The user must enter the number of color classes to be created through the NSTEPS parameter. Colors can be added to the entire image or to blotch regions. The syntax for this command is:

```
$IMAGE > COLORS NSTEPS=N
```

where N is the number of color classes to be entered. Once entered, the program will prompt the user to enter the color weights for the first class. Values for the red, green, and blue guns range from zero to one. After entering the color weights, the user is prompted for the data values to be assigned to that color. This can be one value or many. Ranges of values are specified with the slash ("/") symbol. For example, to assign the color green to data values 10, 50, and 75-120, "0 1 0" would be entered for the color weights and "10,50,75/120" would be entered for the values. The system will repeat this procedure until all of the

NSTEPS color classes have been entered. Figure 14-16 (pgs. 58, 60-61) were all colored with this function.

One idiosyncrasy of COLORS is that it will not color data values of zero. Therefore, if a zero valued class is to be colored, the value either has to be transformed or a different coloring method must be used.

Whereas the COLORS function operates by commands entered through the terminal, the PALETTE function allows the user to interactively use the trackball in a paintbrush fashion. PALETTE is capable of operating on the entire image or a BLOTCH subregion. The syntax for this command is:

$IMAGE > PALETTE

Once PALETTE is entered, the trackball and cursor are used to interactively assign colors to pixels. Colors are selected through the trackball unit function buttons, and are assigned in either of two ways: in the first method, the cursor is positioned over a specific pixel and function button 3C is then pressed. This colors the underlying pixel and all other pixels having the same data value. In the second method, the cursor is positioned over a greyscale bar lying beneath the image, and a particular grey value is assigned that color by pressing button 3C.

Once one color has been "attached" to the scene, adding a second color can be done in two different ways. If function button 3C is selected, pixels of the target value are all assigned the new value. If button 2C is selected, however, a different effect is achieved. Here the first color and data value and the second color and data value act as endpoints, and pixels with intermediate data values are assigned intermediate color values. For example, suppose the color blue is assigned to pixels with a data value of zero using button 3C. If the color red is then attached to pixels of value 128 using button 2C, the result will be that pixels of value 0 will be blue, pixels of value 128 will be red, and the color of intermediate valued pixels will vary from blue to red. If green were then assigned to pixels with a value of 255 using the 2C button, pixels with values from 128-255 would vary from red to green.

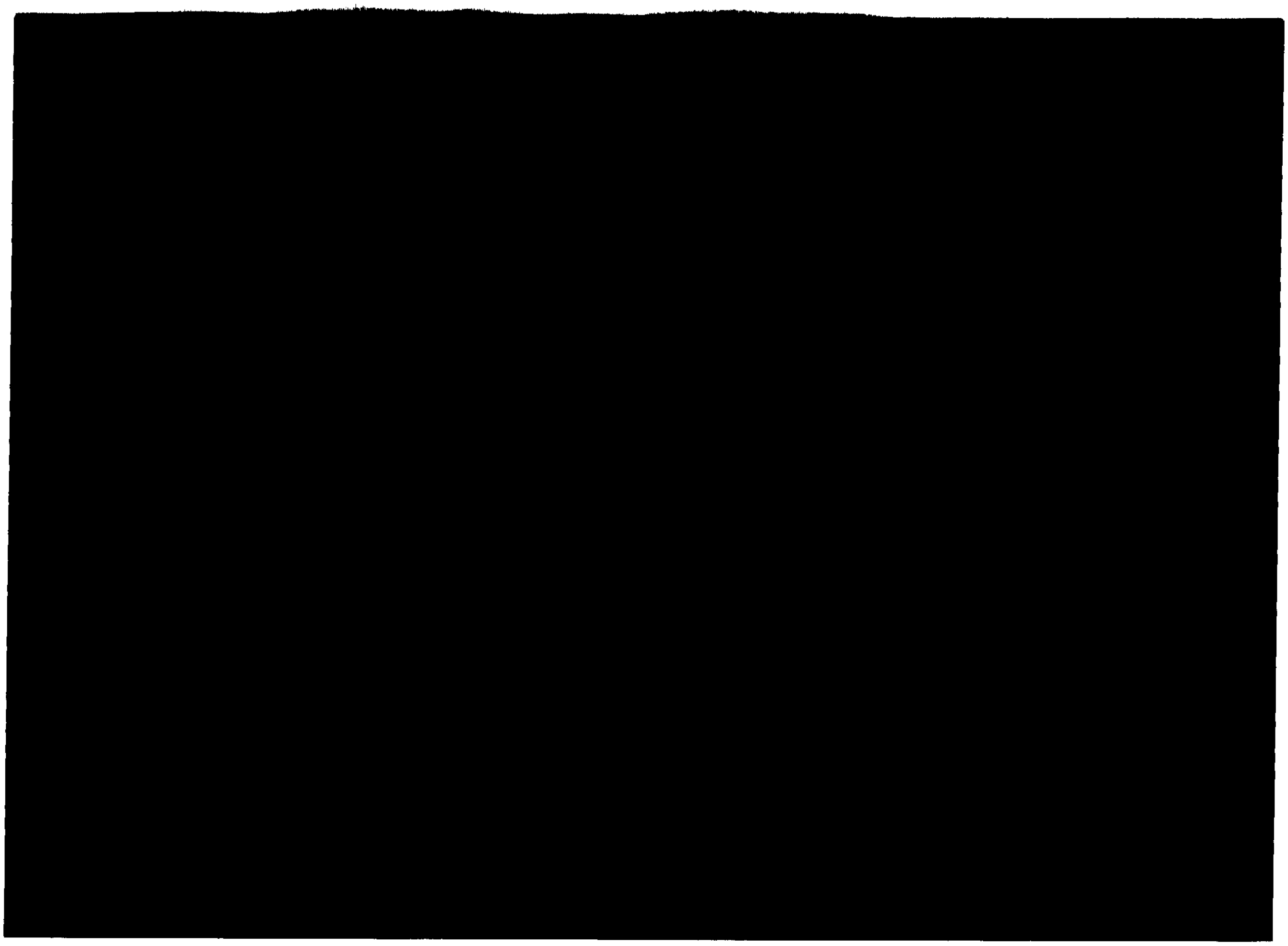
The PALETTE function also allows the user to use the trackball to slide and stretch the color table that has been applied to the image. Since the operation of this function is difficult to describe verbally, it is recommended that the user experiment with PALETTE to learn the subtleties of its use.

For both the COLORS and PALETTE function, image coloring is performed internally by setting look-up table values. If the user wishes to save the color scheme, the STASH function must be used (pgs. 100-101).

The final coloring method to be discussed here is using dummy bands to create a three band color image. Consider a one band image that the user wishes to display in blue shades rather than grey shades (i.e., low data values assigned dark blue shades and high data values assigned bright blues). This could be done by creating two blank bands using the CONSTANT function, and then using SELECT to send the data and the two dummy bands to the monitor. Since three bands are displayed, they would appear in color with each color gun assigned to a different band. There would be no red or green contribution, however, since the red and green color guns are displaying zero valued data. Thus, the outcome would be the original data displayed with the blue gun (Figure 33). The following sequence of commands would be used to implement this technique:

```
> CONSTANT OUTPUTCONSTANT=0 0 0
> FEEDBACK COLOR > $DUMMY
$DUMMY (:1) $DUMMY (:2) $IMAGE > SELECT
```

In this example, $IMAGE is the original one band data. The first two steps create the dummy bands, and the third step displays the 3-banded image. Note that $IMAGE is specified last so that it is assigned to the blue color gun. By creating different dummy data values with CONSTANT, the user can create different color shades.



ORIGINAL PAGE
COLOR PHOTOGRAPH

Figure 33. Display of a monochrome image with blue color gun, using dummy bands (see text for explanation).

Image Examination

In the following section, some of the I^2S functions that are used for image examination are presented. These functions allow the user to magnify a portion of the image or allow the user to determine the actual data value of a pixel. Thus, the main use of these functions is in the study of specific pixel features.

The ZOOM function allows the user to magnify portions of the image by a factor of 2, 4, or 8X (Figures 34-36). The magnification values are selected through the trackball unit function buttons. Rolling the trackball moves the image viewing window. One of the nicer features of ZOOM is function button 3A, which temporarily steps the image back to a 1X magnification. In addition, the previous viewing window is outlined with a box on the 1X image. The magnification returns to the previous setting as soon as the trackball is moved. This feature allows users to orient themselves on the whole picture when using high magnifications. The format for ZOOM is:

$IN'IMAGE > ZOOM > $OUT'IMAGE

where $OUT'IMAGE is optional. Normally ZOOM returns to the original 1X magnification when the function is terminated. If $OUT'IMAGE is specified, the magnified image will be saved as a refresh image. ZOOM assumes that $OUT'IMAGE is black and white unless the COLOR parameter is specified.

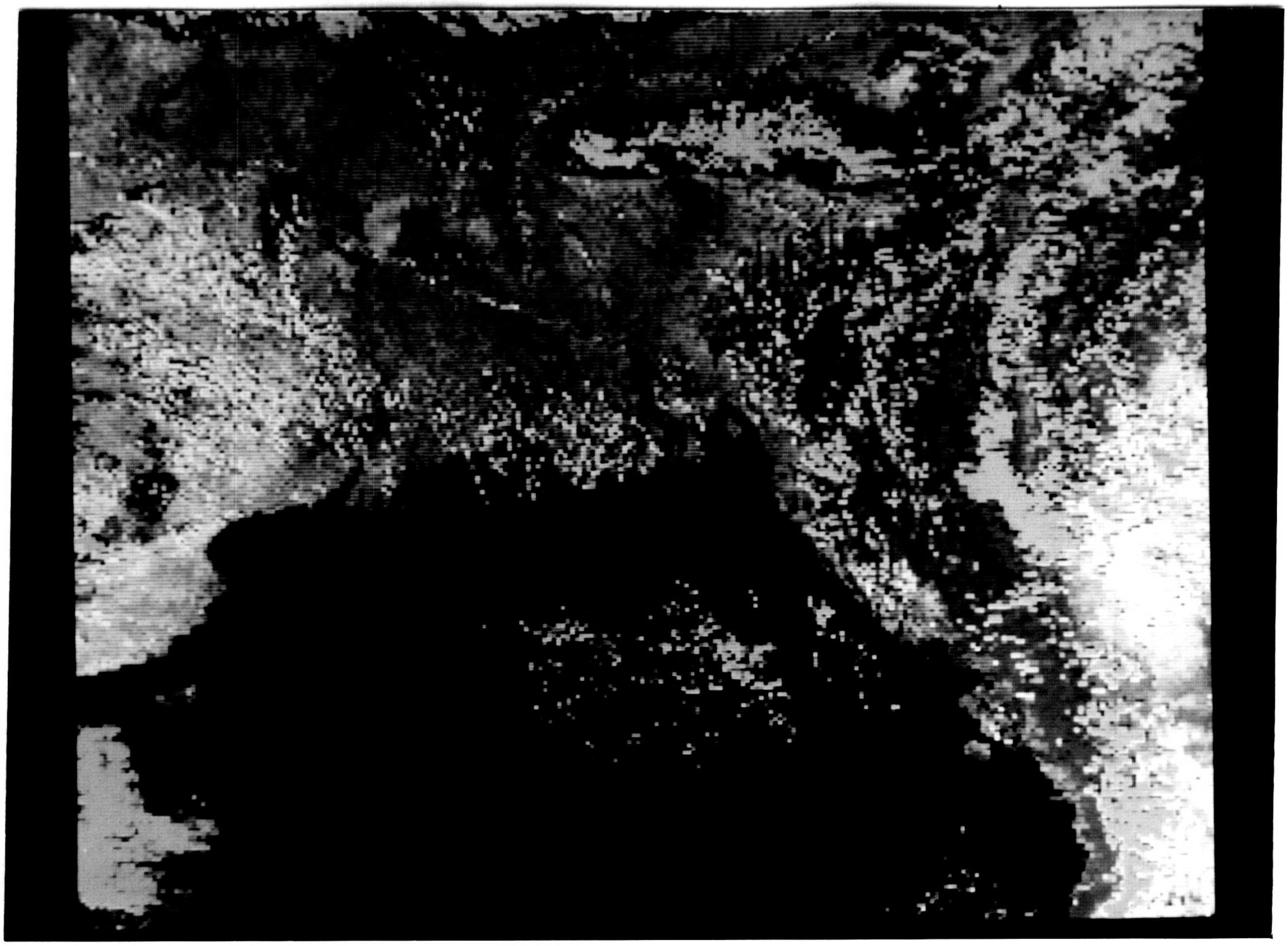
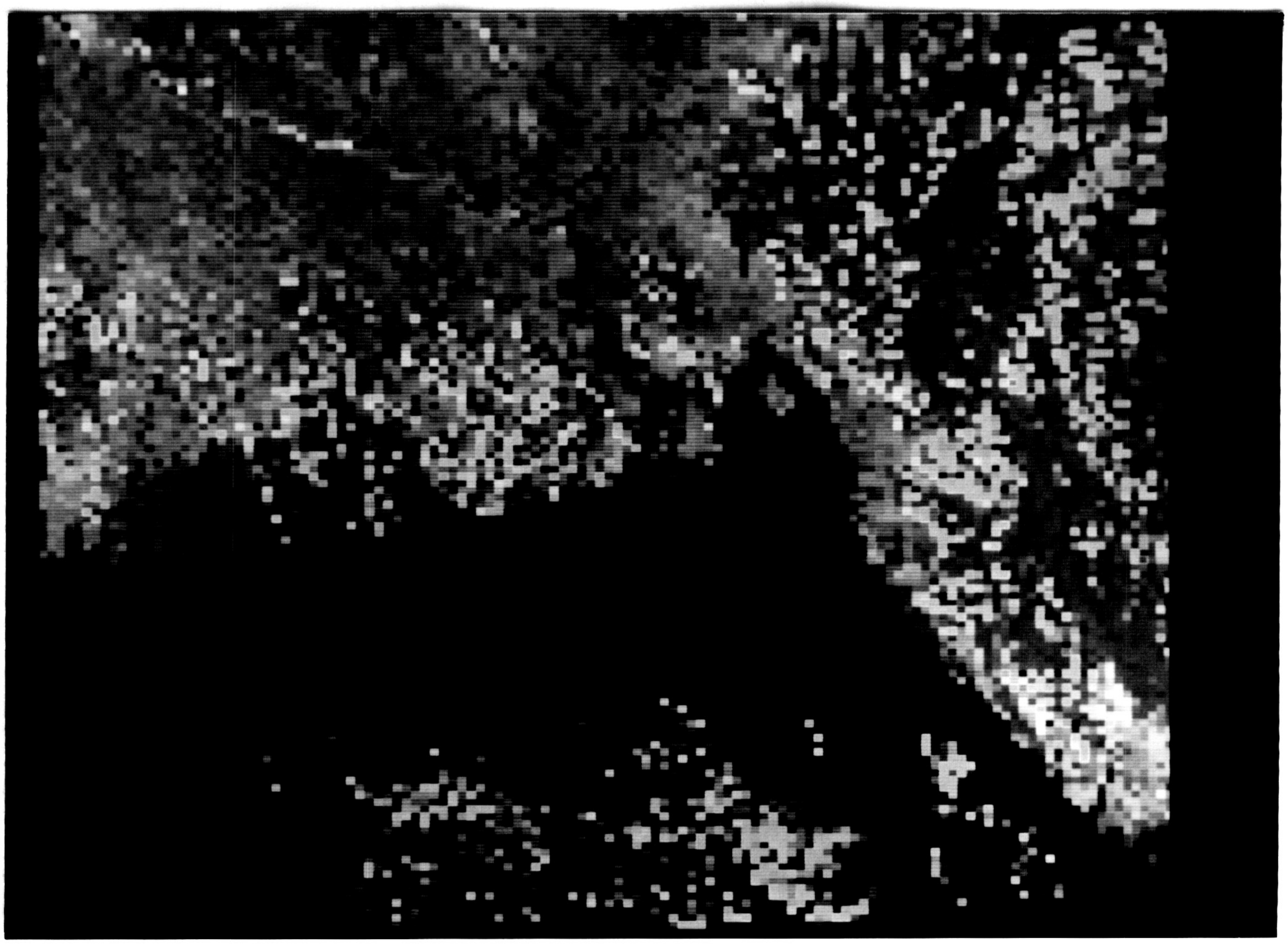


Figure 34. Image magnification (2X).



ORIGINAL PAGE
COLOR PHOTOGRAPH

Figure 35. Image magnification (4X).

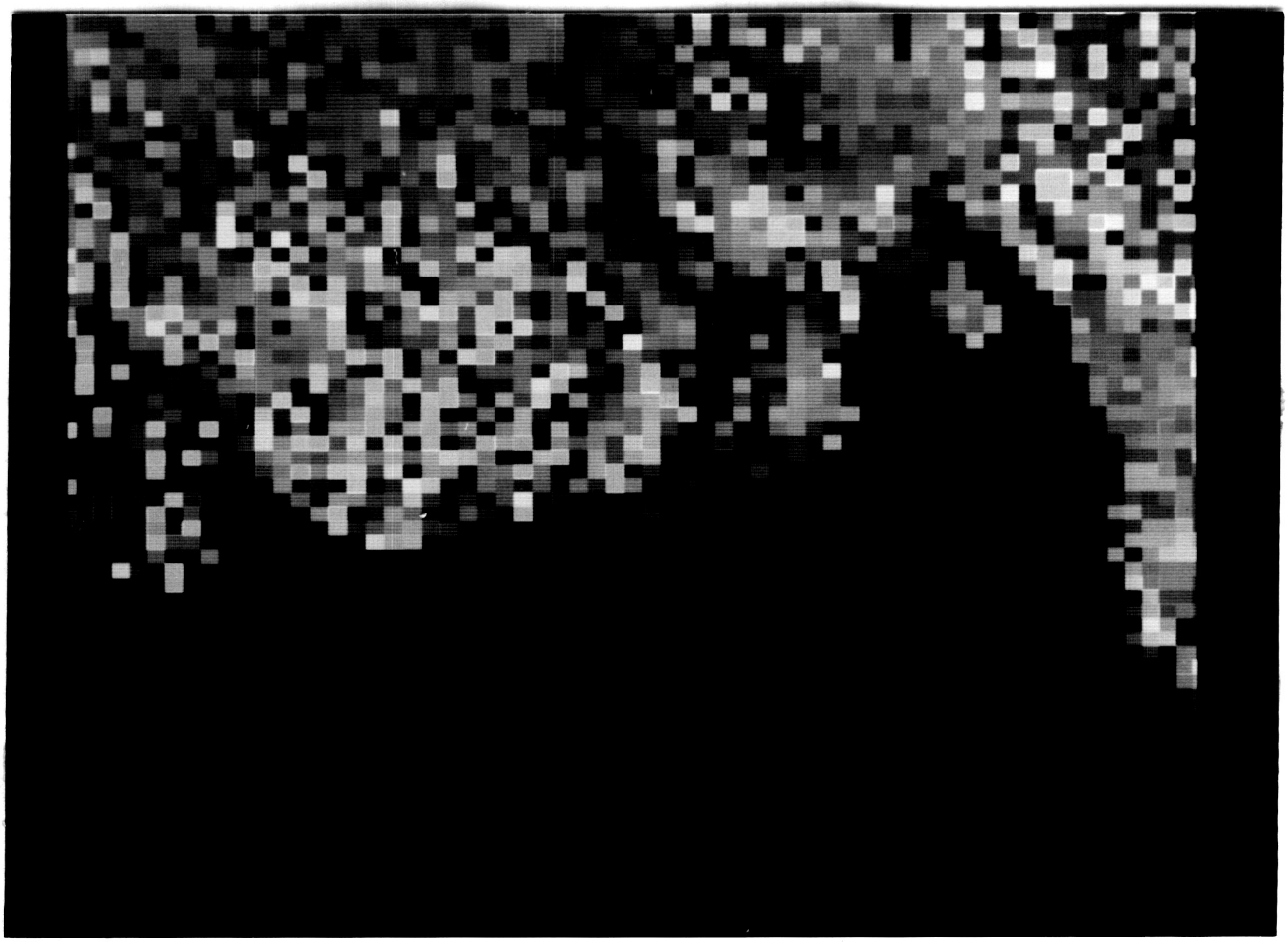


Figure 36. Image magnification (8X).

When the ZOOM function is used, the entire image is magnified by the user selected viewing factor. With LOUPE, a trackball controlled "magnifying glass" (or loupe) can be moved around the image to magnify selected portions. The default size of the magnifying glass is 80 X 80 pixels, and can be changed with the GLASSIZE parameter. The user can also interactively change the glass size by using the trackball unit function buttons. LOUPE allows magnifications of up to 8X. The format for LOUPE is:

$IMAGE > LOUPE

ROAM allows the user to roll a 512 X 512 window through a larger image by using the trackball. For example, using ROAM with a 1024 X 1024 image would allow the user to roll a 512 X 512 viewing window across the image. Although the user can only view a 512 X 512 portion at any one time, this window can be interactively changed in real time. ROAM can also magnify the image up to 8X with the function buttons. In addition, it has the same step back feature as was described for ZOOM. The syntax for ROAM is:

$IMAGE > ROAM PICSACROSS=N1 PICSDOWN=N2

where N1 and N2 are the number of 512 X 512 images in the horizontal and vertical planes, respectively. More than one input image may be specified for ROAM.

The POINTS, BOXPRINT, and PROFILE functions print out data values for specific pixels. With the POINTS function, the cursor is moved over the scene with the trackball and the following information is supplied for the underlying pixel (Figure 37): sample number, line number, and intensity value (if the image is a three band color image, the red, green, and blue intensity values are displayed). The format for POINTS is:

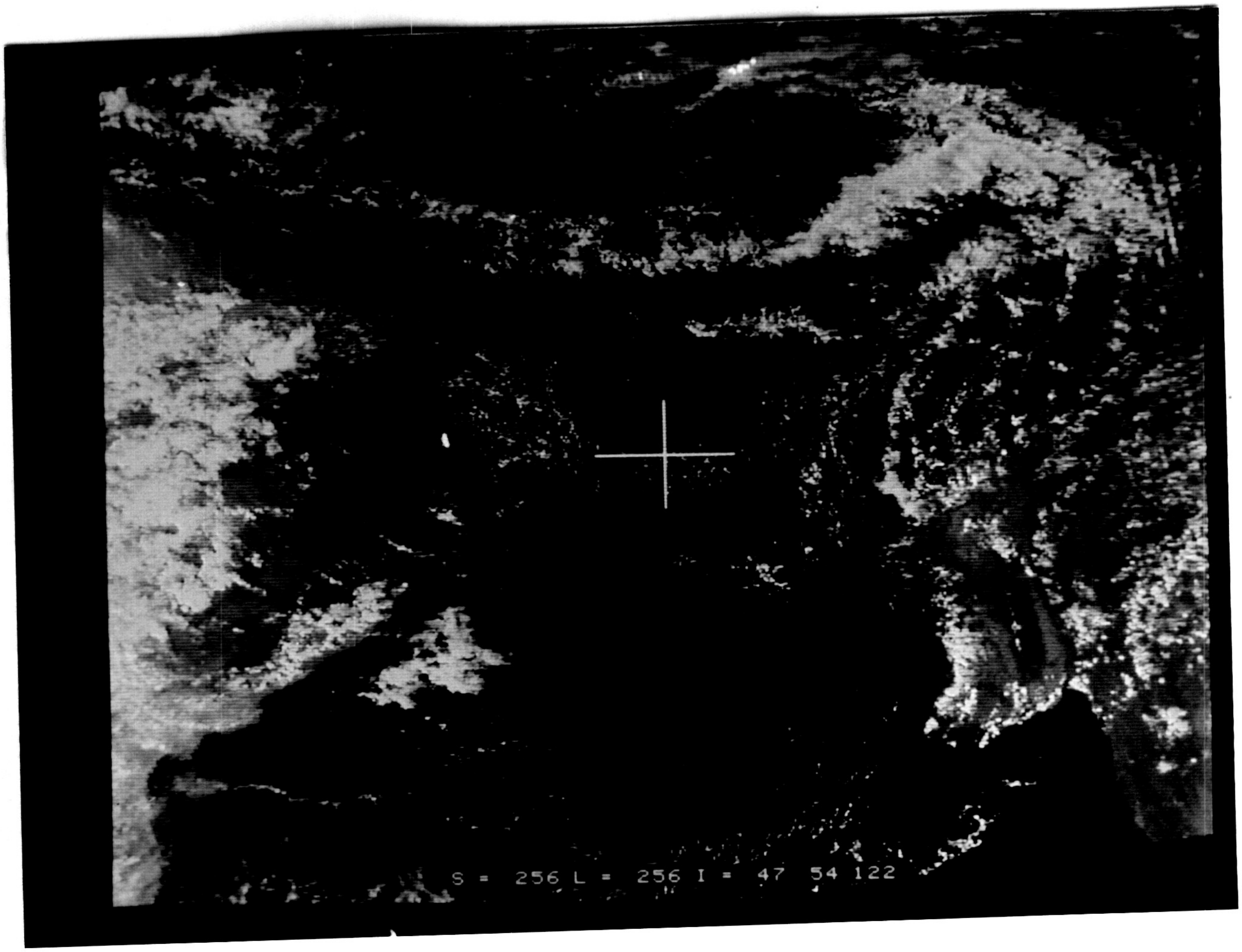
```
$IMAGE > POINTS
```

If the image contains 16-bit data, the SIXTEEN'BIT parameter should also be included. The SECTION parameter is used if the displayed scene was subsectioned or subsampled. For example, if a 1024 X 1024 image was subsampled by two and analyzed by POINTS, this function would assume a 512 X 512 image and print out sample and line values ranging from 1-512. To have POINTS print the sample and line numbers corresponding to the original image, the following command would be entered:

```
$IMAGE > POINTS SECTION=1 1 1024 1024 2 2
```

If this were entered and the cursor pointed to the bottom right pixel, POINTS would print a sample and line number of 1024 rather than 512.

POINTS can also perform a 4X magnification through the use of the trackball unit function buttons. This allows the user to more precisely aim the cursor position.



ORIGINAL PAGE
COLOR PHOTOGRAPH

Figure 37. Illustration of POINTS function. Information provided for the pixel underlying the cursor is sample number, line number, and intensity values for red, green, and blue channels.

The BOX'PRINT function prints a 16 X 16 window of values to the terminal and optionally to the printer. Sample and line numbers are provided on the outside of the box. One value is printed for each pixel if the image is black and white, while three values are printed for color images. The syntax for this command is:

```
$IMAGE > BOX'PRINT
```

To dump the window to the line printer, the LP parameter is included.

The PROFILE command allows the user to plot data values for pixels lying between user specified vertices. The user picks vertices on the image with the trackball and the intensity values of all the points between these vertices are then plotted. The syntax for PROFILE is:

```
$IMAGE > PROFILE
```

As was the case with POINTS, PROFILE also allows the user to zoom in on portions of the image by using the trackball unit function buttons.

Modifications to the POINTS and PROFILE functions were performed at LSU during the NASA/ACEMP training project. These changes are documented in Appendix A.

Image Labeling and Graphics

The S575 software comes with several functions that are used to label images. Two of the more useful functions are ANNOTATE and KEY'CLASSES. These programs are discussed in this section. In addition, the BLOTCH function and some graphics management functions are also covered in this section.

ANNOTATE allows the user to add character text to an image for labeling purposes. The label can be produced on a graphics plane or "burned into" the refresh image (thus replacing the data in the underlying pixels). Parameters exist to control the color, size, and orientation of the text. The text is positioned on the image through the use of the trackball. The format for this function is:

```
$IMAGE > ANNOTATE ANNOTATION="TEXT"
```

where TEXT can contain up to 64 characters and must be enclosed in quotes if more than one word is used or if lower case characters are included. By default, ANNOTATE will burn the annotation into the refresh image unless the GRAPHICS parameter is included. If GRAPHICS is selected, the default is to first erase whatever graphics previously existed. Including the KEEP parameter causes the label to be merged with any preexisting graphics.

The BACKGROUND parameter is used to specify the color of the background box surrounding the label. The default value is "0 0 0," where the three values refer to the color gun settings. Values can range from 0 (black) to one (white). The COLOR parameter is used to specify the color of the text. The default value for this parameter is "1.0 1.0 1.0," or white. These two previous parameters are only used to color foreground and background if GRAPHICS is selected. If annotating on refresh, the BOX'CHARVALUES parameter is used. The default values for this parameter are "0 255," resulting in a black background and a white foreground.

The HEIGHT'VSPC'HSPC parameter controls character height, in pixels, and the number of pixels between characters. The default is "11 2 2" where the first value is the height of the characters plus blank pixels above and below, the second value is the number of blank pixels above and below the text, and the third value is the number of blank pixels between characters. The ASPECT parameter controls the aspect ratio of the characters. The default, one, produces normal characters while values less than one produce wider characters and values greater than one produce taller characters. The SKEW parameter controls both the slant angle of the characters and their rotational angle. The default values are "0 0."

Once the ANNOTATE function and the text have been entered, the cursor will appear on the image (if the GRAPHICS parameter is not included, the text itself replaces the cursor). The user then positions the text with the trackball. After positioning the cursor over the desired location, pressing any of the trackball unit function buttons completes the annotation. The user then has the option to add a new annotation, to change some of the character parameters, or to change the color parameters.

After a raw image has been classified and color-coded, the KEY'CLASSES function can be used to add an explanatory key to the image. The key is burned into the refresh image. For each class, the key includes a colored box and the class name. The format for this function is:

```
$IMAGE > KEY'CLASSES CLASSNAMES=CLASS1 CLASS2 ... CLASSn -
                BOXVALUE=VAL1 VAL2 ... VALn
```

where CLASSi is the name of the ith class and contains up to 16 characters, and VALi is the brightness value of the ith class. This should be the same brightness value that is used to display that class on the image. Up to 128 classes can be included with KEY'CLASSES. Figures 14-16 (pgs. 58, 60-61) were all annotated using KEY'CLASSES.

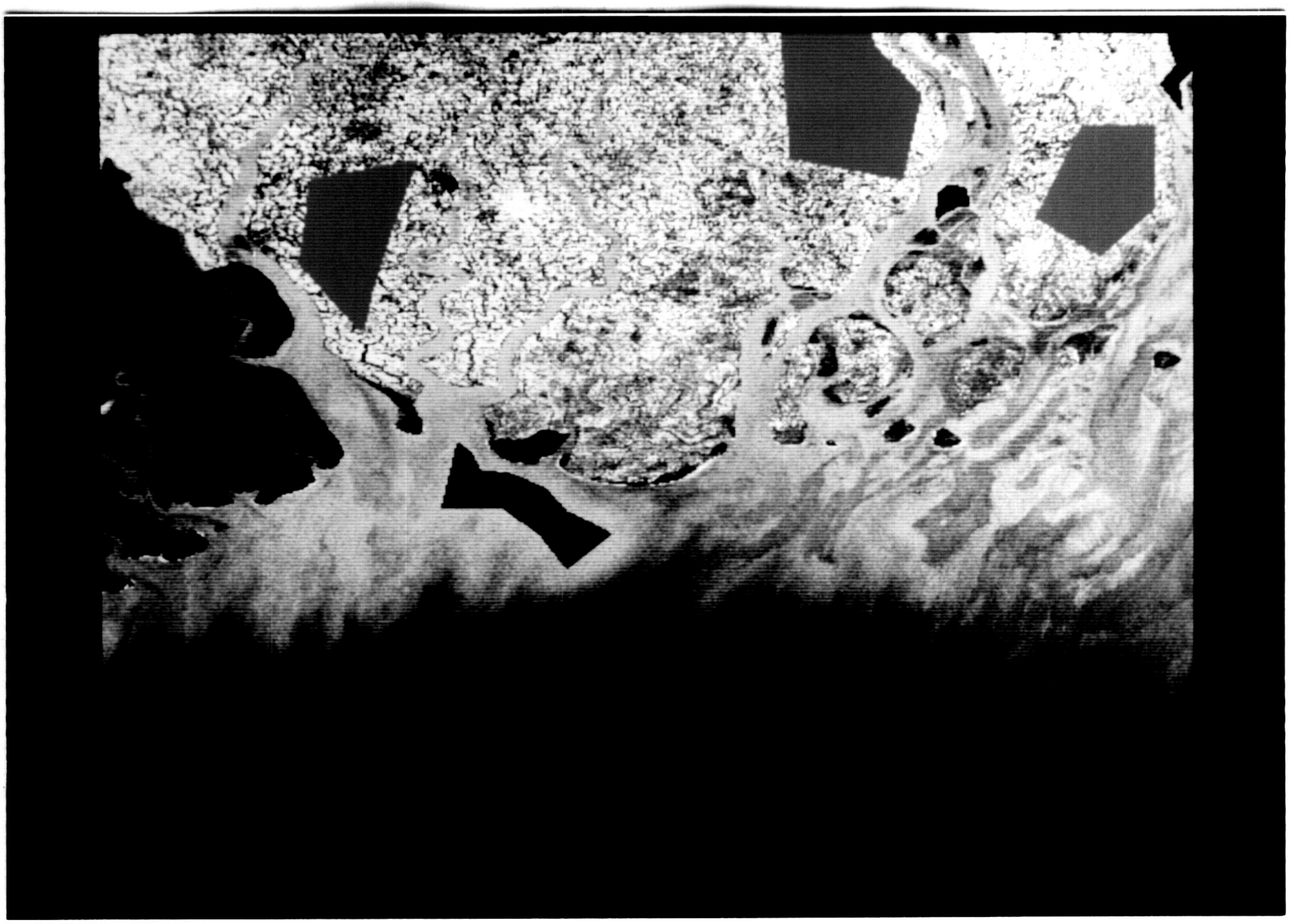
The location of the key is controlled by the KLOCATION parameter. The default values for KLOCATION are "2 80 100 350," where the first two values are the starting sample and starting line of the key and the second two values are the number of samples and number of lines. The key is placed in a box whose background color is by default grey. This can be changed using the BOXVALUE parameter (values range from 0-255). The key outline and class names are by default white, and can be changed with the LABELVALUE parameter.

The BLOTCH function allows the user to define subregions of an image by interactively creating vertices with the trackball (Figure 38). These blotch polygons are stored on the graphics planes and can be used with many of the I^2S functions as logical masks. For example, invoking the HISTOGRAM function with the BLOTCH parameter will cause the histogram to include only those pixels inside the blotch region. Thus BLOTCH is a powerful function that allows the user to analyze selected portions of the image. The syntax for BLOTCH is:

```
$IMAGE > BLOTCH
```

where the input image is optional. By default, graphics plane five is the blotch plane, unless the PLANE parameter specifies otherwise. The previous contents of the graphics plane are erased unless the KEEP parameter is included.

Once this function has been entered, the trackball is used to draw vertices on the blotch plane. After positioning the trackball, function button 1A is pressed to pick vertices. The previous vertex can be deleted with button 2A (for RSX-11M hardware units, function button 1B is used to delete; see I^2S [1984]). Once the polygon is complete, function button 1D is used to close the region.



ORIGINAL PAGE
COLOR PHOTOGRAPH

Figure 38. Subregions defined with the BLOTCH function (Landsat MSS data).

The user can then add another region to that same blotch plane, or can use function button 1F (2F on the RSX-11M) to change to a new plane. The user can magnify the image up to 4X by pressing function button 2B (1C on the RSX-11M). Since the assignment of the function buttons are unit dependent, it is recommended that this function be used with the footpedal driven menu.

The G'ON and G'OFF functions are used to turn graphics displays on or off. The format for G'ON is:

> G'ON

G'OFF has the same syntax. By default, both functions affect all graphics planes. To operate on a different combination of planes, the PLANES parameter is used. For the G'ON function, the display colors of the graphics planes can also be specified with the COLORS parameter. This parameter requires 3 values (for the red, green, and blue gun) with a 0-1 range for each plane that is turned on.

CHAPTER 8

PRODUCT GENERATION ON THE I^2S

In Chapter 3 (pg. 24), the theoretical basis for various NOAA AVHRR applications was discussed. In this chapter, a detailed explanation is presented for creating these images using the I^2S software.

Sea Surface Temperature

As an example of how to use the I^2S system for creating a sea surface temperature image, consider the first equation of Table 3 (pg. 34). This equation simplifies to the following:

$$3.6146\ T_4 - 2.58\ T_5 - 283.2$$

Thus channel 4 values must be multiplied by 3.6146 and channel 5 values by - 2.58. These values plus the constant must then be added. The I^2S ADD function allows channels to be added and also lets the user weight any values by a coefficient. Before this function can be used, a constant image must first be created so the third term can be added. Assume that the channel 4 data are in refresh image $C4 while $C5 contains the channel 5 data. Then the following set of commands would be used to create the sea surface temperature image:

```
> CONSTANT OUTPUTCONSTANT = 1 1 1

> FEEDBACK COLOR > $K

$C4 $C5 $K (:1) > ADD WEIGHTS = 3.6146 -2.58 -283.2 NOSCALE
```

The first command causes a constant image (every pixel contains a value of one) to be displayed. The FEEDBACK function then stores this as a refresh image. Note that only one channel of the constant image is used. The third step then adds the channel 4, channel 5, and constant images, using the appropriate coefficients. The NOSCALE parameter must be included to insure that the values are not scaled.

To save this sea surface image, FEEDBACK would be used to create a refresh image. SAVE would then be used to store this on disk:

```
> FEEDBACK > $TEMP

$TEMP > SAVE > SST
```

BINARY'IMAGE

BINARY'IMAGE employs a user provided threshold to produce an image having only two data values (Figure 14, pg. 58). All pixels having values less than the threshold are assigned a value of 0, whereas pixels with values greater than or equal to the threshold are assigned a value of 255. The threshold can be entered with the THRESHOLD parameter (default value of 127) or interactively by moving the cursor in the X direction. This interactive mode is selected through the CURSOR parameter. The format for this function is:

$IN'IMAGE > BINARY'IMAGE > $OUT'IMAGE

$OUT'IMAGE is an optional refresh image. If it is not supplied, however, the binary image will replace $IN'IMAGE. The user can use the OUTMIN and OUTMAX parameters to produce an image with values other than 0 and/or 255.

CLUSTER

The CLUSTER function is used for creating unsupervised classifications of raw spectral data (Figure 15, pg. 60). The syntax for CLUSTER is:

$IN'IMAGE IN'STATS > CLUSTER > $OUT'IMAGE $CONF OUT'STATS

where $IN'IMAGE is the image to be classified and $OUT'IMAGE is the classified output. IN'STATS is an optional statistics file prepared either through a previous CLUSTER run or from PREPARE or CLASSIFY (see below), and OUT'STATS is an optional statistics file containing the statistics from the last iteration. This file can be used as IN'STATS during a later run. The $CONF file is an optional confidence map having dark values for pixels that were classified with low confidence and bright values for pixels classified with a high confidence.

CLUSTER has a number of parameters that are used to control its operation. The INPUTSTATS parameter is used to signal that an input statistics file is being included. NCLASSES specifies the number of initial classes that the algorithm starts with (the default value is 16). MAXCLASSES is used to limit the maximum number of classes that CLUSTER will produce (the default is 32). The ITERATIONS parameter allows the user to specify the number of iterations that the algorithm will perform (default value of 8). The MIGRATIONQUIT parameter allows a threshold to be established for terminating the function. The default for this parameter is one. For a complete discussion of CLUSTER and its parameters, refer to I^2S (1984).

Supervised Classification

TRAIN

Using the TRAIN function to create training fields is similar to creating blotch planes with BLOTCH (pgs. 171-173). The trackball is used to create polygons that enclose homogeneous areas representative of a land type (Figure 38, pg. 172). Unlike the BLOTCH function, however, TRAIN only saves the polygon vertices. Because there is no function for displaying verts files, the user cannot redisplay the training fields after they have been constructed. This injects a certain amount of difficulty into the training process, since it is not possible to later edit the training fields. The syntax for the TRAIN function is:

```
$IMAGE > TRAIN NCLASSES=VAL > VERTS'FILE
```

where VAL is the number of land classes that are to be included and VERTS'FILE is the name of the file that will be used with PREPARE. For each land class, a maximum of 5 training fields can be constructed unless the NREGIONS parameter specifies otherwise.

Once TRAIN has been entered, the user will be prompted to enter the name of the first land class. The polygons are then drawn with the trackball as is done with BLOTCH. Once a polygon is completed, another polygon in the same land class can be drawn, or the user can go onto the next land class.

PREPARE

The PREPARE function takes the verts file output from TRAIN and generates a statistics file containing the class statistics. The format for PREPARE is:

IMAGE VERTS'FILE > PREPARE > STATS'FILE

where IMAGE is the disk image, VERTS'FILE is the output from TRAIN, and STATS'FILE is the file to be used by the classifier. Note that TRAIN operates on a refresh image while PREPARE operates on the disk image. It is often desirable to use TRAIN with an enhanced version of the image so that training areas are more visible. PREPARE, however, should operate on the raw spectral data. PREPARE will print the training field statistics to the terminal if the BOTH parameter is included, whereas entering the LP parameter causes the statistics to be printed on the line printer.

CLASSIFY

Once the STATS'FILE has been created, two different functions can be used to create the classified image. CLASSIFY performs the classification through a minimum distance algorithm (Figure 16, pg. 61). The syntax for CLASSIFY is:

$IN'IMAGE IN'STATS > CLASSIFY STATSFILE > $OUT'IMAGE -
$CONF OUT'STATS

where $IN'IMAGE is the image to be classified, $OUT'IMAGE is the classified image, and IN'STATS is the file created through PREPARE or a previous CLASSIFY run. OUT'STATS is an optional statistics file that is produced by the algorithm and can be used as input for future runs. $CONF is an optional confidence map having the same characteristics as the $CONF file produced by CLUSTER (pg. 177).

Two of the more important parameters that go along with CLASSIFY are TAXICAB, which causes the algorithm to use a "city-block" distance measure instead of euclidian distance; and THRESHOLD, which controls the threshold distance value beyond which a pixel remains unclassified.

As described thus far, CLASSIFY uses an input statistics file to receive the training field statistics. Two other methods are available for including the statistics, however. If IN'STATS is not included and if the INPUTSTATS parameter is specified, the system will prompt the user to enter the data by hand. This allows the user to create training fields with the BLOTCH function (so that the polygons can be recalled at a later time) and then enter by hand the statistics prepared by the STATISTICS function. If the TRAIN parameter is included, however, the program goes through a simplified training session to allow the training fields to be picked during the CLASSIFY function.

C'CLASSIFY

The C'CLASSIFY function uses a maximum log-likelihood algorithm to perform the image classification. The format for this function is:

IN'IMAGE STATS'FILE > C'CLASSIFY > OUT'IMAGE

where IN'IMAGE is the disk copy of the input file, OUT'IMAGE is the classified image, and STATS'FILE is the file produced by PREPARE. The inclusion of STATS'FILE is mandatory and is the only way to pass the training statistics to this function. C'CLASSIFY has only two parameters: THRESH is a threshold, in standard deviations, beyond which a pixel is not classified; and SWEIGHTS can be used if IN'IMAGE is a three-banded image to assign weighting factors to the three bands. The default is "1.0 1.0 1.0," and values can range from -1.0 to +1.0.

Although CLUSTER was described as an unsupervised classifier, it is also capable of accepting a PREPARE statistics file. Thus, this classifier can also be used as an optional supervised classifier.

MAP

As was described in Chapter 3 (pgs. 62-65), The MAP program was adapted from the CAM program to provide access to the WDB II on the I^2S system. This program can plot geographic data in either equirectangular, Mercator, transverse Mercator, gnomonic, or stereographic projections, and optionally overlay a lat/long grid over the data. These products are displayed on the monitor as a binary digital overlay. The plot can contain all overlays (coastline, islands, lakes, rivers, and international boundaries), or rivers and international boundaries may be excluded. Through the RANK parameter the river detail can vary from least amount of detail (major rivers only; RANK=1) to most amount of detail (minor rivers and streams; RANK=8).

The WDB program is accessed on the I^2S system by entering the following command:

```
> MAP ?
```

Including the question mark causes the system to prompt for parameter values (these parameters are described in detail in Appendix A). Once the command is entered and parameter values supplied, the requested overlays are written to the graphics planes (Figure 17.a, pg. 63). To save these maps, the G'SAVE function (pg. 100) is then used. To overlay the WDB map on an AWARPed image (Figure 17.b, pg. 64), the image is first displayed and then the disk file containing the graphics is loaded to the graphics planes with G'WRITE (pgs. 110, 115). If the image does not line up correctly with the graphics, the ZOOM function can be used to realign the image with the following command:

$IN'MAGE > ZOOM > $OUT'IMAGE

If $IN'IMAGE is a color image, the COLOR parameter should also be included.

The SYNBAPS Bathymetry Program

The SYNBAPS bathymetry data base provides georeferenced images containing pixel values that represent the ocean depth in meters at each location. The data base covers the entire globe and any area can be selected and gridded to the desired spatial resolution in either equirectangular or Mercator map projections. These bathymetry images can be overlaid with NOAA AVHRR images for deep water oceanographic studies.

The SYNBAPS program reads bathymetry data from the SYNBAPS data base and produces a 16-bit (integer) image of a selected area, with the pixel count values representing the ocean depth at each position. Either equirectangular or Mercator map projections are available for the output image, and the spatial resolution of the output image is user selectable (defaulting to 0.01 degrees/pixel for AVHRR). Once a bathymetry image has been created for an area, it need not be regenerated since the SYNBAPS data base does not change, and it is a prime candidate for the tape archives.

The SYNBAPS data base is gridded into 5-minute cells (12 cells per degree of lat/long) and it covers most areas of the globe. Areas where the bathymetry is subject to constant changes, due to sedimentation or other causes, are apparently not included in the SYNBAPS data base, and the transition to the voided areas is usually quite abrupt.

SYNBAPS cell values are positive numbers that represent the ocean depth, in meters, with areas above sea level represented by pixel value 0. The SYNBAPS data base itself is contained in a non-standard disk file (called SYNBAPS.DAT/S on the Perkin-Elmer) that contains 2160 records (180 degrees of latitude times 12), each having 8640 bytes (360 degrees of longitude X 12 halfwords).

The SYNBAPS parameters allow the user to select the location, size, and map projection of the output bathymetry image (a command summary for SYNBAPS appears in APPENDIX A). The upper left corner of the desired output is specified in the NORTH'LAT and WEST'LONG parameters, using negative numbers to represent south or east coordinates. The output map resolution is given as DEG/PIXEL, and the total output image size is given GRID'SIZE in units of pixels. Thus, the total output coverage in degrees (latitude or longitude) is DEG/PIXEL multiplied by GRID'SIZE, and the number of output lines always equals the number of output elements. The default DEG/PIXEL of 0.01 is intended to match AVHRR LAC images, but a lower resolution is usually appropriate for GAC data. The MERCATOR parameter selects the Mercator projection for output instead of the default equirectangular projection.

The SYNBAPS program provides a choice of three resampling techniques for changing the resolution of the SYNBAPS data. Resampling involves interpolating values for new pixels in positions between the original SYNBAPS data base samples. Each technique represents a tradeoff between processing speed and smoothness of the resampled output image. The default resampling technique is nearest neighbor, where the new pixel value is simply taken as the value of the original pixel that is closest to the new position. Nearest neighbor is the fastest algorithm, but the output images will look "blocky" since each small neighborhood of pixels will have the exact same values. Bilinear interpolation (selected by the INTERP parameter) linearly interpolates new pixel values from the four surrounding original samples, producing a smoother output image. This method uses more computer time than nearest neighbor, and since interpolation is essentially a filtering operation, some loss of image detail will result. Cubic convolution (selected by the CUBIC parameter) consumes still more computer time, but does not cause the loss of detail characteristic of bilinear interpolation.

The last parameter is NUM'CNTL'POINTS, which specifies the number of internal control points that will be used for warping the SYNBAPS data to the output map projection. These control points are not available to the user. Since the SYNBAPS data base is already in equirectangular projection, a large number of control points (not to exceed 255) is more useful for Mercator output projections than for equirectangular projections.

The SYNBAPS command takes no input files, and requires a scratch file in addition to the image output. The scratch file is usually small and is used only to hold the extracted portion of the SYNBAPS data base that is being used. It is usually deleted by the user after the SYNBAPS run. An example of a typical SYNBAPS run is:

```
> SYNBAPS NORTH=93 WEST=120 DEG/PIX=0.01 GRID=1024 -
                 > SCRATCH BATHY'OUT
```

Once the SYNBAPS image is created, it will probably be registered onto a NOAA AVHRR image. This turns out to be tricky, since the bathymetry is stored on a 16-bit image and the coastlines are somewhat vague. The REGISTRATION program of the System 575 software can be used for coarse registration with fairly good success. More accurate registration may be possible by matching the known latitude and longitude of the upper left corner of the bathymetry image against a known lat/long of the georeferences AVHRR image. Once the pixel offsets between the bathymetry and AVHRR images are found, the MERGE routine can be used to produce the registered output image.

REFERENCES

International Imaging Systems, Inc. 1983. Model 75 Image Processor: Product Description. 1500 Buckeye Drive, Milpitas, CA 95035.

International Imaging Systems, Inc. 1984. System 575 Software - Digital Image Processing System, Version 3.1: User's Manual. Product No. 777-0039-001. 1500 Buckeye Drive, Milpitas, CA 95035.

Kidwell, K. B. 1984. NOAA Polar Orbiter Data, User's Guide. U. S. Dept. Commerce, National Climatic Center, Satellite Data Services Division, Washington, D. C. 168 pp.

Legeckis, R., and J. Pritchard. 1976. Algorithm for Correcting the VHRR Imagery for Geometric Distortion due to the Earth Curvature, Earth Rotation and Spacecraft Roll and Attitude Errors. NOAA Tech. Review NESS 77, U.S. Dept. Commerce, Washington D.C. 31 pp.

Van Wyckhouse, R. 1973. Synthetic Bathymetric Profiling System (SYNBAPS). Tech. Report TR-233, U. S. Naval Oceanographic Office, Washington, D. C.

APPENDIX A

COMMAND SUMMARY FOR NOAA AVHRR IMAGE PROCESSING

SOFTWARE ADDED TO THE I^2S PACKAGE*

AVHRR

DESCRIPTION

The AVHRR program calibrates level 1B NOAA AVHRR tapes and provides a set of control points in an I^2S format control point file for georeferencing by the programs LCOPY and AWARP. The output image is an integer image. Any subset of the lines, elements, and channels of the raw level 1B image may be processed. Subsampling, however, is not allowed. If subsampling is desired, LCOPY may be used. It is also important to remember the coordinates of the upper left hand corner of the output image relative to the left edge of the full NOAA AVHRR scan line for later use in LCOPY.

SYNTAX

> AVHRR (parameters) > IMAGE CNTP'FILE

PARAMETERS

SSAMPLE

Type: Integer Default: 1

Sample number of the upper left corner of the desired subimage.

SLINE

Type: Integer Default: 1

Line number of the upper left corner of the desired subimage.

*Command summaries were prepared by Chong Ng of the Remote Sensing and Image Processing Laboratory, Louisiana State University, Baton Rouge.

NSAMPLES

Type: Integer Default: 2048

Number of samples per line in the desired subimage. 2048 is the maximum that can be specified.

NLINES

Type: Integer Default: none

Number of lines in the desired subimage.

OUTBANDS

Type: Integer array Default: none

List of the particular AVHRR channels to process.

CALIBRATE

Type: Logical Default: False

This parameter enables image calibration on all selected channels when it is true. If false an uncalibrated integer image will be produced.

MULTI-REEL

Type: Logical Default: False

A NOAA AVHRR image can sometimes be split across two level 1B tapes. This parameter has been provided so that, when true, the program will attempt to continue the run on another tape.

CONTROL'POINTS

Type: Logical Default: False

When control points are desired, this parameter must be true.

NUM'CONTROL'POINTS

Type: Integer Default: 128

This parameter specifies the maximum number of control points desired.

MERCATOR

Type: Logical   Default: False

When this parameter is true, the ground control points will be calculated for a transverse Mercator projection. Otherwise they will be calculated for an equirectangular map projection.

DELLONG

Type: Real   Default: 0.01

This parameter together with DELLAT, specifies the spatial resolution to which the lat/long coordinates of the ground control points will be quantized. This will be the resolution of the georeferenced image output by AWARP.

DELLAT

Type: Real   Default: 0.01

Description is as for DELLONG.

AWARP

DESCRIPTION

This function is a modification of the I^2S WARP function and was written exclusively for use with the functions AVHRR and LCOPY. It is essentially used for performing a final spatial warp of an output from LCOPY. It is used in the same manner as WARP with the exception of two extra parameters, XLOCATION and YLOCATION. This documentation can be appended to the existing documentation of the WARP function.

PARAMETERS

XLOCATION

Type: Integer Default: 1

The starting sample number of the input image relative to the original output from LCOPY. This is the X-coordinate of the upper left-hand corner of the subsection taken from the original image.

YLOCATION

Type: Integer Default: 1

The starting line number of the input image relative to the original output from LCOPY. This is the Y-coordinate of the upper left-hand corner of the subsection taken from the original image.

The rest of the parameters are as described in WARP.

LCOPY

DESCRIPTION

The LCOPY program removes most of the spatial distortion of NOAA AVHRR images due to the scanning geometry. The program stretches the image in the horizontal direction by an amount that increases as the distance from the nadir increases. Correcting the scan distortion before georeferencing the image can reduce the average fitting error from 5-6 pixels to less than one pixel. This program makes it easier for AWARP to handle the high degree of distortion by making a preliminary stretch on the image.

SYNTAX

```
IN'IMAGE IN'CNTP'FILE > LCOPY (parameters) -
         > OUT'IMAGE OUT'CNTP'FILE
```

PARAMETERS

SALTITUDE

Type: Real Default: 840.37

Satellite altitude.

MAXSCANANGLE

Type: Real Default: 55.29

Maximum scan angle of the AVHRR sensor.

KM/PIXEL

Type: Real Default: 0.79

This parameter gives the number of kilometers between the centerpoints of adjacent pixels in the horizontal direction at the nadir position.

TILTANGLE

Type: Real Default: 0.00

This parameter is intended for use with Coastal Zone Color Scanner data.

TSAMPLES

Type: Integer Default: 2048

This is the total number of samples in an entire scan line, as encompassed by MAXSCANANGLE degrees.

SSAMPLE

Type: Integer Default: 1

This gives the location of the left edge of the AVHRR output image, with respect to the left edge of the full AVHRR scan line.

SINCREM

Type: Integer Default: 1

Sample increment desired in the output image.

LINCREM

Type: Integer Default: 1

Line increment of the desired ouput image.

MAP

DESCRIPTION

MAP is the function used to access the World Data Bank II files on the I^2S system. This program allows graphic overlays from any of the three WDB files (coasts, islands, and lakes; rivers; and international boundaries) to be written to I^2S graphics planes. In addition, lat/long lines are also supplied.

PARAMETERS

PTYPE

Type: Integer Default: 1

This parameter is the projection type. The following projections are available: equirectangular (PTYPE=1), Mercator (2), transverse Mercator (3), gnomonic (4), and stereographic (5).

ILATS

Type: Integer Default: 21

The southern latitude limit.

ILATN

Type: Integer Default: 26

The northern latitude limit.

ILNGE

Type: Integer Default: 93

The eastern longitude limit.

ILNGW

Type: Integer Default: 87

The western longitude limit.

GRID SPACING

Type: Integer Default: 1

The frequency with which the lat/long grid is printed. Specifying one causes a one degree grid to be printed, whereas specifying five would lead to a five degree grid.

CONTINENTS

Type: Character Default: None

The name(s) of the continent file(s) in which the area is found (North America, South America, Europe, Asia, or Africa).

RANK

Type: Integer Default: 1

The map rank, where 1 is the least detail and 8 is the most detail. This refers to rivers only.

COASTLINE ONLY

Type: Integer Default: 0

Allows the user to include all overlays (0) or coastline only (1).

POINTS

DESCRIPTION

A parameter PIPE was added to the function POINTS to enable a user to display an enhanced image while at the same time being able to read the corresponding pixel value from the original raw image. The pixel values of both images are displayed simultaneously at the bottom of the screen. The first set of pixel values correspond to the displayed image while the second set of values correspond to the raw image. Note that the two images considered should have an equal number of channels. Both images also have to be refresh images. For example, if both images are three channel color images, six pixel values will be displayed at the bottom of the screen. The first set of three values belong to the image currently displayed.

SYNTAX

```
$ENHANCED'IMAGE $RAW'IMAGE > POINTS (PIPE [other parameters])
```

PARAMETER

PIPE

Type: Logical Default: False

If true, the pixel values of both images will be displayed but only the first image will appear on the screen.

PROFILE

DESCRIPTION

Two parameters SCALE and DRAW were added to the function PROFILE to enhance the readibility of the graphs drawn by this function. The SCALE parameter allows a scaling of the X-axis of the graph. This is useful when the length of the profile being drawn is short. The DRAW parameter will cause the points plotted on the graph to be connected. In other words, the profile is displayed as a continuous curve.

PARAMETERS

SCALE

Type: Integer Default: 1

This parameter allows a scaling of the profile along the X-axis. For example, a SCALE factor of 2 will double the original distance between the points to be plotted.

DRAW

Type: Logical Default: False

If true this keyword parameter causes the points of the profile being drawn to be connected and displayed as a continuous curve.

A further description of PROFILE can be found in I^2S (1984).

SYNBAPS

DESCRIPTION

The SYNBAPS program reads bathymetry data from the SYNBAPS data base and produces a 16-bit (integer) image of a selected area with the pixel values representing the ocean depth at that position. Either equirectangular (the default) or Mercator map projections are available for the output image, and the spatial resolution of the output image is user selectable (use 0.01 degrees/pixel for NOAA AVHRR data). Once a bathymetry image has been created for an area, it need not be regenerated since the SYNBAPS data base does not change.

SYNTAX

```
> SYNBAPS (parameters) > SCRATCH'IMAGE   OUT'IMAGE
```

PARAMETERS

NORTH'LATITUDE

Type: Real Default: None

This specifies the latitude of the upper left corner of the desired output image. Negative numbers are used to represent south coordinates.

WEST'LONGITUDE

Type: Real Default: None

This specifies the longitude of the upper left corner of the desired output image. Negative numbers are used to represent east coordinates.

DEGREES/PIXEL

Type: Real Default: 0.01

This parameter specifies the ouput map resolution.

GRID'SIZE

Type: Integer Default: None

The total output image size is given as GRID'SIZE in units of pixels. Therefore the total output coverage in degrees (lat or long) is DEG/PIXEL multiplied by GRID'SIZE. The number of output lines always equals the number of output elements.

INTERPOLATION

Type: Logical Default: False

When this parameter is true, a bilinear interpolation is used in the resampling for changing the resolution of the SYNBAPS data. The nearest neighbor resampling technique is the fastest and is the default if type of resampling technique is not specified.

CUBIC'CONV

Type: Logical Default: False

This specifies that a cubic convolution resampling technique be used and requires more computation time than either nearest neighbor or bilinear interpolation.

NUM'CNTL'POINTS

Type: Integer Default: 128

This specifies the number of control points desired. These are internal control points only, and are not accessable to the user.

MERCATOR

Type: Logical Default: False

This parameter selects the Mercator projection for output instead of the default equirectangular projection.

APPENDIX B

COMMAND LISTING FOR IMAGERY PRODUCED FOR THIS MANUAL

Figure 14 (pg. 58)

This figure uses BINARY'IMAGE to create a land/water separation of BANG'LAND band 3 (pg. 222). The legend is provided by KEY'CLASSES, and the image is colored with COLORS (since COLORS cannot color pixels with a value of one, the OUTMIN parameter of BINARY'IMAGE is used to set the minimum value to one). Four classes must be colored: the water, the land, the key box (which is given a data value of 2), and the key labels (which are given a data value of 3).

```
BANG'LAND(1 1 2048 2048 4 4:3) > DISPLAY > $A

$A > BINARY'IMAGE CURSOR OUTMIN=1 > $B

$B > KEY'CLASSES CLASSNAMES=LAND WATER -
SMALLBOXVALUES=1 255 BOXVALUE=2 LABELVALUE=3 -
KLOCATION=400 400 100 100

$B > COLORS NSTEPS=4

[1] Enter color weights : 0.2 0.2 0.8
Enter classes to be assigned color : 1
Continue, repeat or exit, type C, R or E: C

[2] Enter color weights : 0.5 0.5 0.5
Enter classes to be assigned color : 2
Continue, repeat or exit, type C, R or E: C

[3] Enter color weights : 1.0 1.0 1.0
Enter classes to be assigned color : 3
Continue, repeat or exit, type C, R or E: C

[4] Enter color weights : 0.6 0.4 0.2
Enter classes to be assigned color : 255
Continue, repeat or exit, type C, R or E: C
```

Figure 15 (pg. 60)

This figure uses CLUSTER to create an unsupervised classification of BANG'LAND (pg. 222). The legend is provided by KEY'CLASSES, and the image is colored with COLORS. Six classes must be colored: mangrove, wetland, riverine, deep water, the key box (which is given a data value of 5), and the key labels (which are given a data value of 6).

```
BANG'LAND(1 1 2048 2048 4 4) > DISPLAY > $A

$A > CLUSTER NCLASSES=5 MAXCLASSES=7 > $CLASS'MAP $CONF'MAP

$CLASS'MAP > KEY'CLASSES CLASSNAMES=MANGROVE WETLAND -
RIVERINE DEEP_W SMALLBOXVALUES=1 2 3 4 BOXVALUE=5 -
LABELVALUE=6 KLOCATION=400 400 100 100

$CLASS'MAP > COLORS NSTEPS=6

[1] Enter color weights : 0.1 0.5 0.1
Enter classes to be assigned color : 1
Continue, repeat or exit, type C, R or E: C

[2] Enter color weights : 0.6 0.4 0.2
Enter classes to be assigned color : 2
Continue, repeat or exit, type C, R or E: C

[3] Enter color weights : 0.7 0.7 0.4
Enter classes to be assigned color : 3
Continue, repeat or exit, type C, R or E: C

[4] Enter color weights : 0.2 0.2 0.8
Enter classes to be assigned color : 4
Continue, repeat or exit, type C, R or E: C

[5] Enter color weights : 0.5 0.5 0.5
Enter classes to be assigned color : 5
Continue, repeat or exit, type C, R or E: C

[6] Enter color weights : 1.0 1.0 1.0
Enter classes to be assigned color : 6
Continue, repeat or exit, type C, R or E: C
```

Figure 16 (pg. 61)

This figure is a supervised classification of BANG'LAND (pg. 222) using the TRAIN, PREPARE, and CLASSIFY sequence. TRAIN is used to create five training fields for the different land types. Since TRAIN works on a refresh image, it is first enhanced with H'EQUALIZE to make training field selection easier. PREPARE then creates class statistics of these training fields, and CLASSIFY performs the final supervised classification. The image was then given a legend using KEY'CLASSES and colored with COLORS.

```
BANG'LAND(1 1 2048 2048 4 4) > DISPLAY > $A

$A > H'EQUALIZE

> FEEDBACK COLOR > $B

$B > TRAIN NCLASSES=5 > BANG'VERTS

BANG'LAND(1 1 2048 2048 4 4) BANGVERTS -
> PREPARE > BANG'STATS

$A BANG'STATS > CLASSIFY STATSFILE > $CLASS'MAP $CONF'MAP

$CLASS'MAP > KEY'CLASSES CLASSNAMES=MANGROVE WETLAND -
RIVERINE SHALLOW_W DEEP_W UNCLASS -
SMALLBOXVALUES=1 2 3 4 5 0 BOXVALUE=6 LABELVALUE=7 -
KLOCATION=400 380 100 120

$CLASS'MAP > COLORS NSTEPS=7

[1] Enter color weights : 0.1 0.5 0.1
Enter classes to be assigned color : 1
Continue, repeat or exit, type C, R or E: C

[2] Enter color weights : 0.6 0.4 0.2
Enter classes to be assigned color : 2
Continue, repeat or exit, type C, R or E: C

[3] Enter color weights : 0.7 0.7 0.4
Enter classes to be assigned color : 3
Continue, repeat or exit, type C, R or E: C

[4] Enter color weights : 0.4 0.6 0.8
Enter classes to be assigned color : 4
Continue, repeat or exit, type C, R or E: C

[5] Enter color weights : 0.2 0.2 0.8
Enter classes to be assigned color : 5
Continue, repeat or exit, type C, R or E: C
```

```
[6] Enter color weights : 0.5 0.5 0.5
Enter classes to be assigned color : 6
Continue, repeat or exit, type C, R or E: C

[7] Enter color weights : 1.0 1.0 1.0
Enter classes to be assigned color : 7
Continue, repeat or exit, type C, R or E: C
```

Figure 17.a (pg. 63)

This illustration is a sample output from the World Data Bank II for the Bangladesh region.

```
> MAP ILATS=10 ILATN=30 ILNGE=100 ILNGW=80 CONTINENTS=ASIA
```

Figure 17.b (pg. 64)

This figure shows a sample of the World Data Bank II output overlayed on an enhanced image called BANG'JAN13. The ZOOM function, which allows the image to be scrolled in the X-Y plane, was iteratively used to position the image until a satisfactory match was found. Note that the graphics plane is turned off when ZOOM is entered; thus the user must approximate the position of the map, position the image with ZOOM, turn the graphics planes on, review the results, and then repeat the steps if necessary.

```
BANG'JAN13 > DISPLAY > $A

$A > H'EQUALIZE

> FEEDBACK COLOR > $B

> MAP ILATS=21 ILATN=26 ILNGE=92 ILNGW=88 CONTINENT=ASIA

> G'SAVE PLANES=0 1 2 3 > WDB

$B > ZOOM COLOR > $C

WDB > G'WRITE INPUTPLANES=0 1 2 3 OUTPUTPLANES=0 1 2 3

$C > ZOOM COLOR > $D
```

Figure 20 (pg. 84)

This figure uses a subsectioned portion of TEMP'AVHRR, the unclipped version of BANG'AVHRR (pg. 211). To enhance the image, the visible and thermal bands were separately clipped and then merged.

```
TEMP'AVHRR (780 810 512 512:1 2) > DISPLAY MAXCLIP=100 > $A1

   TEMP'AVHRR (780 810 512 512:3) > DISPLAY MINCLIP=600 -
                  MAXCLIP=855 > $A2

        $A1(:1) $A1(:2) $A2 > MERGE > AVHRR'SUBSECT

              AVHRR'SUBSECT > DISP > $A
```

Figure 21 (pg. 109)

The composite image shown in Figure 21.d was created by DISPLAYing the BANG'AWARP image (pg. 213) with disk bands 1, 2, and 3 going to refresh bands 1, 2, and 3 (the default). The color separates were obtained not through software, but by using a camera system that could photograph each color gun separately.

```
BANG'AWARP (1 1 1024 1024 2 2) > DISPLAY > $A
```

Figure 22 (pg. 114)

The composite image shown in Figure 22.d was created by DISPLAYing the BANG'AWARP image (pg. 213) with disk bands 3, 1, and 2 going to refresh bands 1, 2, and 3. The color separates were obtained not through software, but by using a camera system that could photograph each color gun separately.

```
BANG'AWARP(1 1 1024 1024 2 2:3 1 2) > DISPLAY > $A
```

Figure 23 (pg. 123)

This image is the result of using the AVHRR function to read in an image from tape. The size of the input image is 2048 X 2048, and it resides on tape file number three. Only bands 1, 2, and 5 are read in (these will then be bands 1, 2, and 3 on the disk image). The data are calibrated as they are read in, and a control point file is also created. Once on disk, the thermal band (band 3) is separately clipped to bring the 10-bit data into an 8-bit range (pg. 104). This is then merged with the remaining two bands to create the final image.

```
> AVHRR NSAMPS=2048 NLINES=2048 OUTBANDS=1 2 5 FILES=3 -
  CALIBRATEBANDS CONTROLPOINTS > TEMP'AVHRR CNTP'AVHRR

TEMP'AVHRR(:1 2) > DISPLAY > $A1

TEMP'AVHRR(:3) > DISPLAY MINCLIP=550 MAXCLIP=805 > $A2

$A1(:1) $A1(:2) $A2 > MERGE > BANG'AVHRR
```

Figure 24 (pg. 133)

This image is the LCOPYed version of BANG'AVHRR (pg. 211). Lines are subsampled using the input file modifier, and samples are subsampled using the function parameter SINCREM (pg. 135).

```
BANG'AVHRR(1 1 2048 2048 1 4) CNTP'AVHRR > LCOPY -
        SINCREM=4 > BANG'LCOPY CNTP'LCOPY
```

Figure 25 (pg. 138)

This is the AWARPed image produced from BANG'LCOPY and CNTP'LCOPY (pg. 212). Since AWARP works on refresh images, the files must first be read into refresh using DISPLAY. The INAREA parameter is used to pass the image dimensions to the program. The resulting image resides on refresh memory, and must be stored on disk with the SAVE function. Since the final image is 1024 X 1024, it is subsampled for display.

```
BANG'LCOPY > DISPLAY > $A

$A CNTP'LCOPY > AWARP INAREA=1 1 929 512 > $B

$B > SAVE > BANG'AWARP

BANG'AWARP(1 1 1024 1024 2 2) > DISPLAY > $C
```

Figure 26 (pg. 144)

This image illustrates the use of the TLM function to enhance BANG'AWARP (pg. 213). Once the function was entered, trackball function button 3D was pressed to transform the three bands separately.

```
BANG'AWARP(1 1 1024 1024 2 2) > DISPLAY > $A

$A > TLM
```

Figure 27 (pg. 146)

This illustration demonstrates the use of the PIECEWISE'LINEAR function on the BANG'AWARP image (pg. 213). Once the function was entered, trackball function button 3C was pressed to transform the green band only.

```
BANG'AWARP(1 1 1024 1024 2 2) > DISPLAY > $A

$A > PIECEWISE'LINEAR
```

Figure 28 (pg. 148)

This figure illustrates the use of the HISTOGRAM function to produce a histogram and cumulative histogram of the BANG'AWARP image (pg. 213).

```
BANG'AWARP(1 1 1024 1024 2 2) > DISPLAY > $A

          > HISTOGRAM CUMULATIVE
```

Figures 29-30 (pgs. 150-151)

These two figures are the H'EQUALIZEd version of BANG'AWARP (pg. 213) along with the histogram.

```
BANG'AWARP(1 1 1024 1024 2 2) > DISPLAY > $A

$A > H'EQUALIZE

> HISTOGRAM CUMULATIVE
```

Figures 31-32 (pgs. 152-153)

These two figures are the H'NORMALIZEd version of BANG'AWARP (pg. 213) along with the histogram.

```
BANG'AWARP(1 1 1024 1024 2 2) > DISPLAY > $A

$A > H'NORMALIZE

> HISTOGRAM CUMULATIVE
```

Figure 33 (pg. 159)

As described in the text (pgs. 157-158), this illustration shows the use of dummy bands to display a monochrome image (BANG'AWARP band 3; pg. 213) as a blue shaded image.

```
BANG'AWARP(1 1 1024 1024 2 2:3) > DISPLAY > $A

> CONSTANT

> FEEDBACK COLOR > $K

$K(:1) $K(:2) $A > SELECT
```

Figures 34-36 (pgs. 161-163)

These three images illustrate the use of the ZOOM function on the H'EQUALIZEd version of BANG'AWARP (pg. 213). Once ZOOM is entered, trackball function button 3C is pressed once to increase the magnification.

```
BANG'AWARP(1 1 1024 1024 2 2) > DISPLAY > $A

$A > H'EQUALIZE

> FEEDBACK COLOR > $B

$B > ZOOM COLOR > $B'2X

$B'2X > ZOOM COLOR > $B'4X

$B'4X > ZOOM COLOR > $B'8X
```

Figure 37 (pg. 166)

Illustration of the POINTS function on the BANG'AWARP image (pg. 213). Note that the SECTION parameter was not used, and thus the center pixel is read as sample 256 and line 256 rather than sample 512 and line 512.

```
BANG'AWARP(1 1 1024 1024 2 2) &gt; DISPLAY &gt; $A

$A &gt; POINTS
```

Figure 38 (pg. 172)

This figure illustrates the use of the BLOTCH function on a Landsat image. The LANDSAT function is used to read 3 bands of the 1024 X 1024 image from tape.

```
> LANDSAT AREA=501 1 2048 2048 1 1 BANDS=1 2 4 > BANG'LAND

      BANG'LAND(1 1 2048 2048 4 4) > DISPLAY > $A

                  $A > BLOTCH
```