

d

DETECTION AND LOCATION OF RESEAUX
USING PICTORIAL PATTERN RECOGNITION

A Dissertation

Presented to

the Faculty of the School of Engineering and Applied Science
University of Virginia

In Partial Fulfillment

of the Requirements for the Degree
Doctor of Philosophy (Electrical Engineering)

by

James Hiram Aylor

April 1977

Engr Lib.

Diss.

Engr.

495

Copy 2

APPROVAL SHEET

This dissertation is submitted in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy (Electrical Engineering)

James Hiram Ayler
Author

This dissertation has been read and approved by the Examining
Committee:

Edward Parrie Jr.
Dissertation Adviser

E. S. M. Uz

James W. Moore

Genab Cook

E. White

Accepted for the School of Engineering and Applied Science:

J. E. Gibson
Dean, School of Engineering and
Applied Science

April 1977

TABLE OF CONTENTS

ACKNOWLEDGEMENTS	iii
ABSTRACTS	iv
LIST OF FIGURES	v
LIST OF TABLES	viii
LIST OF SYMBOLS	ix
CHAPTER I INTRODUCTION	1
CHAPTER II A RESEAU CORRECTION SYSTEM	6
2.0 System Hardware	6
2.1 System Software	10
CHAPTER III RESEAU MODELLING	12
3.0 Introduction	12
3.1 Signal Model	18
3.2 Noise Model	25
3.3 Model Results	30
CHAPTER IV RESEAU CENTER DETECTION	39
4.0 Introduction	39
4.1 Image Preprocessing	40
4.1.0 Fourier Transform	40
4.1.1 Walsh-Hadamard Transform	46
4.1.2 Regularization and Nonlinear Filtering	49
4.1.3 Bayes Threshold Selection Technique	52
4.1.4 Conclusions of Preprocessing Schemes	72
4.2 Center Location Algorithms	74
4.2.0 Template Matching	74
4.2.1 Center of Gravity Technique	76
4.3 Results and Conclusions of the Center Location Process	79

CHAPTER V	RESULTS AND CONCLUSIONS	81
	5.0 Proposed Correction System	81
	5.1 Conclusions and Recommendations	86
BIBLIOGRAPHY		88
APPENDIX	CORRECTION SYSTEM DEVELOPMENT SOFTWARE LISTINGS	90

ACKNOWLEDGEMENTS

The author would like to express his extreme appreciation to Dr. E. A. Parrish, Jr. for his valuable advise and encouragement as faculty advisor and friend during the course of this research.

A sincere thanks is also extended to all those whose continued support has made this endeavor a reality and especially to my wife, Sherry, who has been very patient and understanding throughout this undertaking.

The author acknowledges the U.S. Army Engineer Topographic Laboratories, Fort Belvoir, Virginia for partial financial support on this project.

ABSTRACT

Plus-like images called reseaux and found on aerial photographs provide the capability of determining the extent of geometric distortion present in these photographs. Accurate location of these images within a photograph is, at present, a manual operation. An automatic system to perform this operation using a minicomputer system and microdensitometer is presented.

A model is generated from the statistics of actual reseau samples to create a test vehicle for the performance of various picture processing algorithms. A method of including various photographic effects into the model is presented.

Various combinations of scene analysis schemes are presented with results for the center location problem. Preprocessing techniques for noise removal are described. Also developed is a scheme for noise removal based on Bayes decision theory taking advantage of the relative slow scanning speed of the microdensitometer. Two methods of center location are described and comparative speed and accuracy results presented.

The operation of the final system is also described.

LIST OF FIGURES

<u>Figure</u>	<u>Page</u>
1.1 Computerized Reseaux Correction System Block Diagram . . .	4
2.1 Model 1050A Microdensitometer System	7
3.1 Measurement Space	14
3.2 Reseau Reproductions Into 16 Gray Shades	15
(a) 4.2 micron aperture	
(b) 8.5 micron aperture	
3.3 Original 32 x 32 Reseau Sample, File Number 4-- Light Background	16
3.4 Reseau Sample Cross-Sections	17
(a) 20th Row	
(b) 20th Column	
3.5 Reseau Cross-Section Models	19
(a) Ideal	
(b) Actual	
3.6 Typical Curve of Density versus Log Exposure	21
3.7 3 x 3 Window Used in the Gradient Calculation	24
3.8 Samples of Histograms for Different Aperture Sizes	27
(a) File Number 1--Light Background	
(b) File Number 5--Dark Background	
3.9 Weighting Matrices for Each 45° of Gradient Rotation	33
3.10 Generated Reseau Cross-Section for the 8.5 Micron Diameter Case Without Noise	34
(a) Ideal--8 samples wide	
(b) Corrupted by nonlinear transform	

3.11	Comparison of Generated and Actual Reseau Cross-Section	35
	(a) Generated reseau	
	(b) Actual reseau	
3.12	Gray Scale Reproduction of a Generated Reseau and An Actual Reseau	36
	(a) Generated	
	(b) Actual	
3.13	Reseau Models	37
	(a) Missing portions	
	(b) Five percent change in transmissivity	
3.14	Reseau Model of the Kostinsky Effect	38
	(a) Cross-section	
	(b) Gray scale reproduction	
4.1	Low Pass Filtering Using the Fast Fourier Transform . . .	44
	(a) Original cross-section	
	(b) Frequency terms removed above fundamental term	
	(c) Frequency terms removed above first harmonic term	
4.2	High Pass Filtering via Fast Fourier Transform	45
	(a) Original cross-section	
	(b) D.C., fundamental and first harmonic terms removed	
	(c) D.C. through second harmonic terms removed	
4.3	Fourier Harmonics, Walsh Functions, and Hadamard Matrices for $n = 8$	47
4.4	Low Pass Filtering via Fast Hadamard Transform	50
	(a) Original cross-section	
	(b) D.C., S_{a1} and C_{a1} terms	
	(c) D.C., 4 S_{a1} and 4 C_{a1} terms	
	(d) D.C., 8 S_{a1} and 8 C_{a1} terms	
4.5	Comparison of Regularization and Nonlinear Filtering . . .	53
	(a) Original	
	(b) 3×3 average	
	(c) 5×1 average (3×3 gradient window)	

4.6	Nonlinear Transformation (X-Direction)	54
	(a) Cross-section of original	
	(b) Cross-section of transformed image	
4.7	Nonlinear Transformation (Y-Direction)	55
	(a) Cross-section of original	
	(b) Cross-section of transformed image	
4.8	Probability Density Functions for a Generated Reseau in Homogeneous Background	62
4.9	A Posteriori Probabilities for a Generated Homogeneous Reseau	63
4.10	Probability Density Functions for an Actual Reseau in Homogeneous Surroundings	64
4.11	A Posteriori Probabilities for an Actual Reseau Sample in Homogeneous Surroundings	65
4.12	Probability Density Functions for an Actual Reseau Sample in Non-Homogeneous Background	66
4.13	A Posteriori Probabilities for an Actual Reseau Sample in Non-Homogeneous Background	67
4.14	Cross-Section of a Generated Sample for Low Contrast Reseau Image	68
4.15	Probability Density Functions for Low Contrast Reseau Sample	69
4.16	A Posteriori Probabilities for Low Contrast Reseau Sample	70
4.17	Samples of Thresholded Images	71
	(a) Generated reseau in homogeneous background	
	(b) Actual reseau in urban background	
	(c) Generated reseau of low contrast	
5.1	Reseaux Correction System Flow Chart	82

LIST OF TABLES

<u>Table</u>		<u>Page</u>
2.1	Effective Aperture Sizes in Microns	8
3.1	Calculation of Selwyn Granularity for Various Sample Data	28
3.2	Chi-Square Results for 5% Level of Significance	31
4.1	Comparison of Center Detection Schemes	77

LIST OF SYMBOLS

a	area of a sampling aperture
\bar{A}	two-dimensional discrete Fourier transform (matrix form)
A_N	area of sampling window
A_w	area of sampling window
$A(m,n)$	two-dimensional discrete Fourier transform
A/D	analog-to-digital
$A-D$	analog-to-digital
a through f	elements of a digital picture
\bar{B}_M	weighting matrix of discrete Fourier transform
\bar{B}_N	weighting matrix of discrete Fourier transform
$B(k;n)$	one-dimensional discrete Fourier transform
$B_k(m,n)$	weighting matrix for modified regularization transform
CCD	charge-coupled device
D	density
D_s	saturation density
D/A	digital-to-analog
$D.C.$	average value
DCRS	Digital Coordinate Readout System
DEC	Digital Equipment Corporation
DFT	discrete Fourier transform
e_j	expected frequency

E_p	picture energy
$E(m,n)$	Euclidean distance
ETL	U.S. Army Engineer Topographic Laboratories
\overline{F}	two-dimensional Hadamard transform
$F(u)$	one-dimensional Hadamard transform
f_x	spatial frequency in x direction
f_y	spatial frequency in y direction
FFT	fast Fourier transform
G	Selwyn granularity
$G(f_x, f_y)$	two-dimensional continuous Fourier transform
$g(i,j)$	composite digital picture function
$g_i(u)$	weighting value of the Walsh functions
\overline{H}_M	Hadamard matrix of dimension M
\overline{H}_N	Hadamard matrix of dimension N
i	index
i_c	center of gravity coordinate
IBM	International Business Machines Corporation
j	index
j_c	center of gravity coordinate
JPL	Jet Propulsion Laboratory
k	number of estimated population parameters
M	number of density range bins <u>or</u> dimension of the Fourier or Hadamard transform
N	number of independent observations <u>or</u> dimension of the Fourier or Hadamard transform
$n(i,j)$	digital noise picture function

o_j	observed frequency
$p(x)$	mixture density function
$p(\bar{x})$	mixture density function vector
$p(x \omega_j)$	class conditional probability density function
$p(\bar{x} \omega_j)$	class conditional probability density function vector
$P(\omega_j)$	a priori probability for class ω_j
$P(\omega_j x)$	a posteriori probability
$P(\omega_j \bar{x})$	a posteriori probability vector
PDS	Photometric Data Systems Corporation
PACS	Pattern Analysis and Computer Systems Laboratory
$R_{gt}(m,n)$	cross-correlation function
$R(\alpha_i x)$	expected loss or conditional risk
$R(\alpha_i \bar{x})$	expected loss or conditional risk vector
S_1, S_2, S_x, S_y	partial derivatives
$s(i,j)$	digital signal picture function
$S(m,n)$	digital picture function
$S_N(i,j)$	modified regularization transform
$S_W(i,j)$	regularization transform
T	transmission value
$t(i,j)$	template picture function
v	degrees of freedom
W_α, W_M, W_N	exponential weighting values of discrete Fourier transform
$w(i,j)$	sampling window
$wal(u,x)$	Walsh function

x	feature measurement
\overline{x}	feature measurement vector
$z(k, \ell)$	a complex number
\overline{z}	an array of complex numbers
$\alpha_i, \alpha(x)$	decision of classifier
$\alpha(\overline{x})$	decision vector of classifier
θ	direction of the gradient
$\lambda(\alpha_i, \omega_j)$	loss function
$\sigma(D)$	standard deviation of the density statistics
χ^2	chi-square statistic
ω_j	symbol for class j

CHAPTER I

INTRODUCTION

For many years, aerial photography has been an important resource for the acquisition of data. It has been found that in time of crises, aerial photography is sometimes the only source of timely information about a military situation [1]. Since the advent of large computer systems, much work has been done on computer processing of aerial photographs. Such applications as target detection, recognition of buildings, crop classification, and contour following have generated many papers in the area of pattern recognition.

Aerial photographs are obtained from both film cameras and television vidicon cameras. There are several significant differences which are manifested in the manner in which the image projected onto the receiving surface is sensed. For example, differences occur in spectral and dynamic sensitivity as well as linearity. Also, film resolution is limited by grain size whereas vidicon resolution is limited by scanning-beam spot size [2].

Photographs from both types of camera systems require corrections in order to make the information obtained accurate. Although the systems do differ, the required corrections are the same. Some examples are geometric correction, photometric correction, and noise removal.

Geometric correction involves the physical straightening of the photographic image. This distortion can be the result of nonlinearities in the optical system or in the electronic system.

Photometric correction involves compensation for nonuniform brightness across the photograph. Photometric distortions include glare introduced by light scattering in the optics, shutter leaks, and various electronic transfer functions.

The noise problem includes both structured noise and random noise. Noise, as is the case in many situations, is derived from many sources. Such things as film grain noise in the case of film cameras and electronic system noise in the case of vidicon cameras constitute structured noise. Random noise can occur from a poor signal-to-noise ratio in television camera systems [2].

This dissertation is concerned with the geometric distortion problem. The removal of geometric distortion is required in order to make accurate determination of spatial relationships in object space. These measurements are important for such things as constructing maps of areas photographed and for locating positions of targets in object space from these maps or photographs. The geometric correction task is facilitated by the use of plus-like marks (+) called reseaux. These reseaux form a pattern which is accurately deposited on the television vidicon target or on a glass plate in front of the film in the case of a film camera. This pattern is thus used to obtain calibration data for the correction task.

A great deal of research has been done on digital picture processing in the past five years. However, from a search of the literature, only the Jet Propulsion Laboratory (JPL), appears to have done extensive work on the problem of geometric correction [2]. The motivation for their work was provided by the television-transmitted photographs from early lunar and Mariner explorations. The computer processing done by JPL used relatively

large dedicated systems such as the IBM 360/44 and the IBM 360/75. Here, the concern is limited to studying the feasibility of a computerized system using a 16-bit minicomputer operating in a closed-loop manner with the digitizing scanner in order to obtain a reasonably priced overall system.

The system block diagram is shown in Figure 1.1. The system consists of a minicomputer, an image scanning device, analog-to-digital (A/D) and digital-to-analog (D/A) converters, and a mass storage device. In operation, an operator grossly positions the scanner on the first reseau since interest is confined to small areas of a photograph at a time. The computer determines the center of the reseau and moves the scan table to center the reseau. By using calibration data from the accurately located reseau grid placed in the cameras, the computer moves the scan table to the next reseau where the centering procedure is repeated. This process continues until all reseaux are found. The reseaux coordinates and the inter-reseaux distances are maintained on the mass storage unit for future use.

Three main problems are encountered in designing such a system. The first is identifying the reseaux among other objects on the photograph. Since films vary in density and noise, the detection technique must be adaptive. The second task is the accurate location of the center of each such reseau. Since the center location is needed for the calibration algorithm, the detection scheme must be accurate. The third concern is to determine the size and speed of a computer system needed and if the use of a minicomputer is feasible.

The correction task involved the investigation of the latest pattern recognition techniques in picture processing. Consideration of the

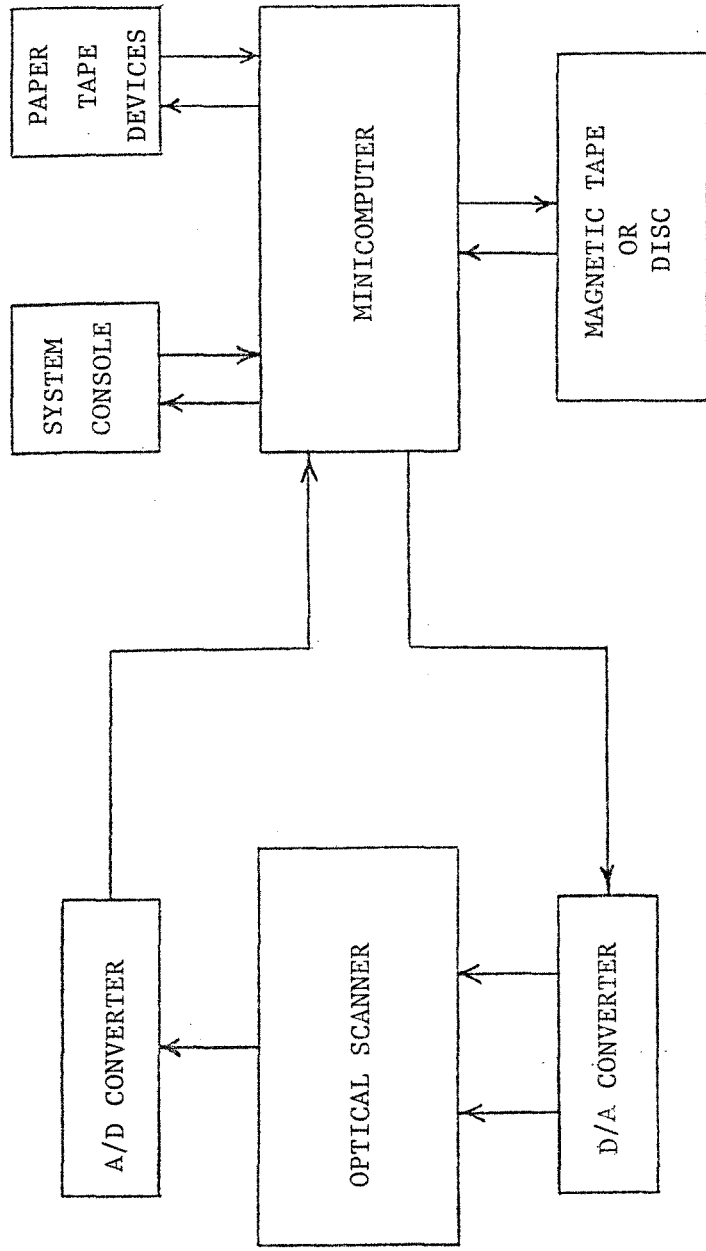


Figure 1.1 Computerized Reseaux Correction System
Block Diagram

processing rates of the techniques and memory requirements are important. Also of interest is the accuracy of such techniques for the center location problem.

This dissertation will present the techniques considered and the applicability of each technique to the geometric correction problem. Chapter II considers both the hardware necessary and an overview of the software that will be considered for the task. Chapter III presents the development of a reseau model which will be used to test the various pattern recognition and picture processing schemes studied. In Chapter IV, various picture preprocessing schemes and center location techniques are evaluated. Comparison of several overall techniques are presented with results. A final reseau correction system is proposed in Chapter V.

CHAPTER II

A RESEAUX CORRECTION SYSTEM

2.0 System Hardware

An optical scanner suitable for the proposed system shown in Figure 1.1 is presently available from Photometric Data Systems Corporation (PDS) of Webster, New York [3]. The scanner is designed to record very accurate density measurements over small areas of photographic film while providing precise locations of these measurements.

A block diagram of the system is shown in Figure 2.1. The Model 1050A Microdensitometer System is comprised of three main subsystems: a Model 1010A Microdensitometer, a Model 2300A Data Acquisition System, and a Model 270 Digital Coordinate Readout System (DCRS).

The microdensitometer is a device which measures film density by passing light through a film sample onto a photomultiplier tube. An optical system on each side of the film sample, each symmetrical to the other, causes the light passing through the film to be highly concentrated. The photomultiplier tube converts the collected light energy into electrical energy which varies proportionally to the light intensity. The 1010A Microdensitometer also provides the capability of reporting either transmission(T) or density(D). The relationship between the two values is

$$D = \log_{10} \frac{1}{T} \quad (2-1)$$

Both circular and rectangular (slits) sampling apertures are available. Effective aperture sizes possible are given in Table 2.1 and are dependent

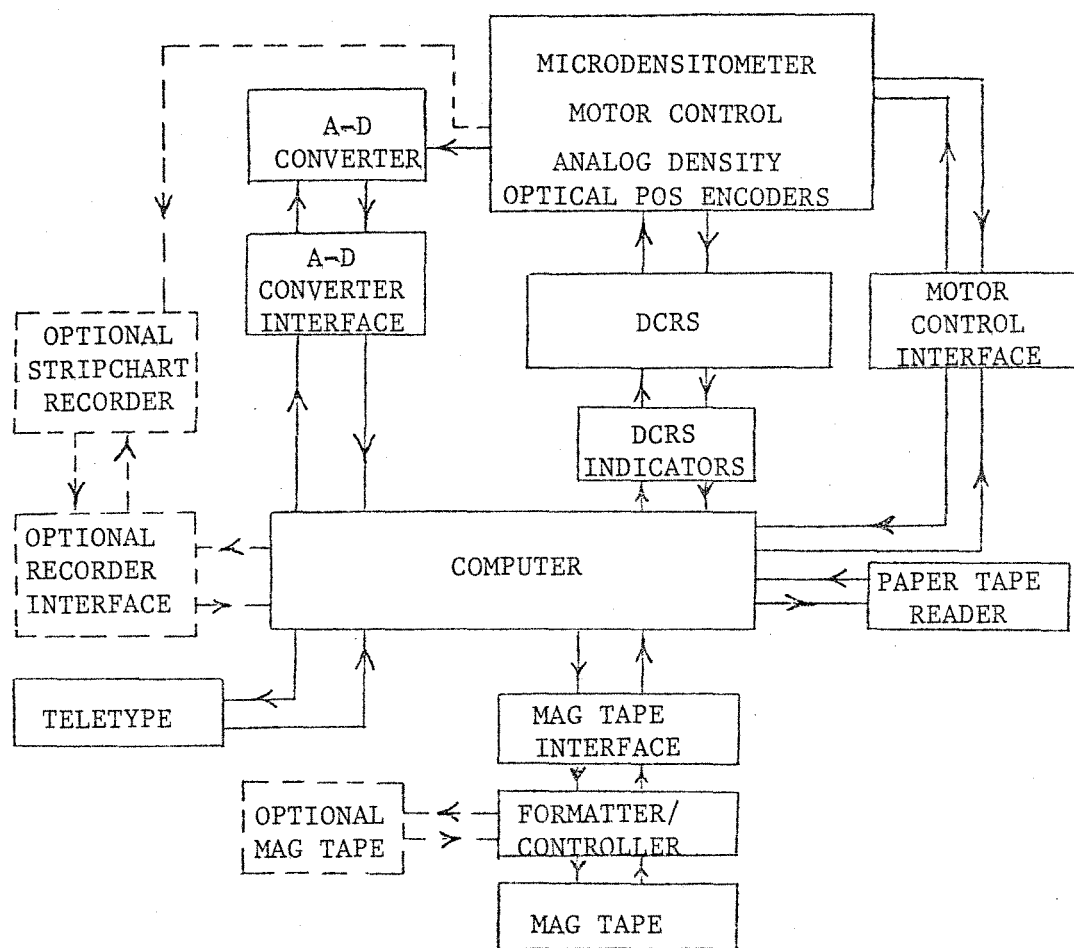


Figure 2.1 Model 1050A Microdensitometer System [3:1-1]

TABLE 2.1
EFFECTIVE APERTURE SIZES IN MICRONS [3:8-12]

EFF. MAG	<u>DISK POSITION</u>							
	A	B	C	D	E	F	G	H
23.2	8.6	17.2	34.5	51.7	86.2	17.2 x 431.0	34.5 x 431.0	86.2 x 431
47.2	4.2	8.5	16.9	25.4	42.4	8.5 x 211.9	16.9 x 211.9	42.4 x 211
60.0	3.3	6.6	13.3	20.0	33.3	6.6 x 166.7	13.3 x 166.3	33.3 x 166
94.4	2.1	4.2	8.5	12.7	21.2	4.2 x 105.9	8.5 x 105.9	21.2 x 105
118.0	1.7	3.4	6.7	10.1	16.8	3.4 x 84.2	6.7 x 84.2	16.8 x 84
192.0	1.0	2.1	4.2	6.3	10.4	2.1 x 52.1	4.2 x 52.0	10.4 x 52

on effective magnification and actual aperture. Neutral density and color filters are also included on the microdensitometer.

The film sample is placed on a 10-inch by 10-inch X-Y platen which can be controlled manually or automatically to within one micron accuracy. The microdensitometer also contains several controls for manual operation and an indicator for measurement value.

The DCRS monitors and displays the X and Y positions of the microdensitometer platen. It also provides an interface between the Data Acquisition System and the optical position encoders. A keyboard is included on the DCRS which enables the loading of information for certain operations such as setting platen travel limit values.

The Data Acquisition System contains a DEC PDP-8/I minicomputer, a 7-track magnetic tape unit, and required interfaces to control and retrieve data from the DCRS and microdensitometer systems. The PDP-8/I is a 12-bit minicomputer and contains 8196 words of core memory. Included are a paper tape reader and teletype. In this system, the analog signal from the photomultiplier tube is converted to a digital signal by a 10-bit A/D converter resulting in a resolution of 1024 gray levels. Platen position information is obtained from the DCRS. The computer is used to control the position of the platen thereby making automatic data acquisition possible. The ultimate output of the system is a magnetic tape containing density measurements and their respective coordinates which can be used for later computer analysis.

Many other types of components could be used in the system described by Figure 1.1 other than those used in the PDS system. The image scan device could be a television camera, phototransistor (or photodiode) or CCD array.

The analog-to-digital and digital-to-analog devices could be of different sizes depending on the resolution needed in both cases. The speed of these devices is of little concern since the scan device and computer software will constitute the larger part of the time needed to perform the task of geometric correction.

The mass storage device could be a disc system instead of a magnetic tape device. One type of mass storage will, however, be necessary since the computer may not be able to hold all measurements simultaneously and since the correction information is needed for further film sample analysis.

The choice of a minicomputer to be used involves no particular restrictions other than possessing enough memory to handle the software needed to perform the correction task. Thus, practically any modern minicomputer will suffice.

2.1 System Software

The heart of the proposed geometric correction system lies in algorithms that can be executed in a minicomputer in a reasonable time period and be sufficiently accurate to determine the geometric corrections needed. Although the PDS Microdensitometer system includes sufficient hardware to perform the correction task, with the possible exception of the computer memory, the software available with the system is very limited. The PDS system only provides the capability for automatic acquisition of density and coordinate information from photographic film and some statistical information on the density measurements in the form of histograms.

The overall objective of the correction system is the accurate location of the center of each reseau. Each reseau sample must first be

preprocessed in order to remove unwanted information. Such preprocessing must not alter the shape of the reseau, because this could easily lead to poor estimation of the center location. Bayes decision theory, Fourier and Hadamard transforms, regularization, spatial differentiation, and non-linear filtering are considered for the preprocessing scheme. Once preprocessing has been performed on the reseau scene, a method of center detection is employed. Several approaches such as center of gravity and template matching will be presented in the sequel.

Reseau samples used in the development of the correction system were extracted from aerial photographs supplied by the U.S. Army Engineer Topographic Laboratories (ETL), Fort Belvoir, Virginia. The samples were digitized and recorded on magnetic tape by a PDS 1050A Microdensitometer System described above and located at the Research Institute of ETL. The correction system was then developed on a Hewlett-Packard 2100A minicomputer at the Pattern Analysis and Computer Systems (PACS), Department of Electrical Engineering, University of Virginia.

CHAPTER III

RESEAU MODELING

3.0 Introduction

Early in the development of the reseau measurement system it was determined that to evaluate effectively the performance of the pattern recognition and scene analysis schemes employed, a model of a reseau was needed. This model could then be used to generate a statistically meaningful number of reseaux whose centers are accurately known. Although a few actual samples of reseaux taken from photographic negatives are available, error analysis is impossible since their exact centers are not known.

This model can also incorporate many types of distortions caused in a reseau by the photographic process. Scenes can be generated using the model containing reseaux which vary in complexity due to such things as inhomogeneous backgrounds and small transmissivity difference between the local background and a reseau.

The first step in the generation of the reseau model is to determine information about actual reseaux. This information has been obtained from three main sources: computer analysis of actual reseau samples, conversation with people presently measuring inter-reseaux distances, and a study of certain aspects of the photographic process. All reseau samples considered were obtained from typical aerial photographs and were taken from areas of varying background conditions.

Each digital image contained 8281 density values arranged in a 91 x 91 matrix array fashion. Samples were obtained with spatial separations of both 4 and 8 microns with circular sampling apertures of 4.2 and 8.5 micron diameters, respectively. Each extremity of a reseau has a thickness of approximately 55 microns and extends to the edge of the measurement domain in each direction. The sample separation was chosen as a function of the sampling aperture diameter in order that the entire measurement space be covered with no sample overlap. The sampling aperture diameters were chosen to provide approximately ten samples across the reseau thickness. Two different aperture sizes were used to examine the effect of varying sampling resolution on preprocessing schemes and overall performance. The difficulty in obtaining these digitized images made the study of an optimal sampling aperture size and a large sample test set impractical.

A sample of a reseau found in a homogeneous background is depicted by Figure 3.1. Reproductions into 16 shades of gray are shown in Figure 3.2. Figure 3.2a is a reseau image in a light, homogeneous background with an effective circular aperture diameter of 4.2 microns. Figure 3.2b has the same background conditions but an aperture diameter of 8.5 microns.

Figure 3.3 is another type of gray scale rendition of a 32 x 32 sample portion of the same image as shown in Figure 3.2a. This image reproduction is also divided into 16 shades of gray with a blank representing the smallest density value and an "F" representing the largest. Figure 3.4 shows typical cross-sections in each direction of a reseau image in a homogeneous background.

Since the high frequency density fluctuations found in the scan of a homogeneous region is directly related to the probability of finding

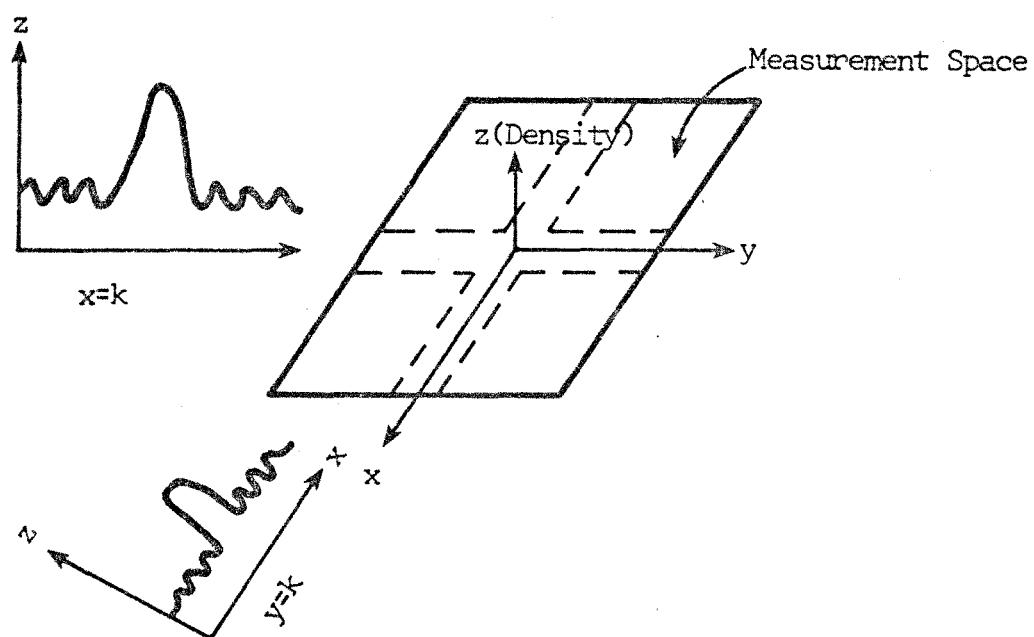


Figure 3,1 Measurement Space

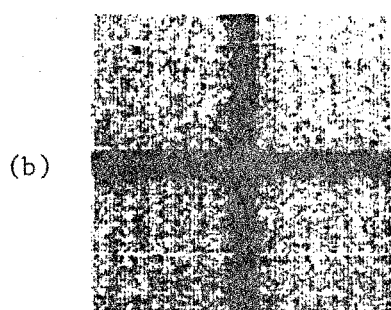
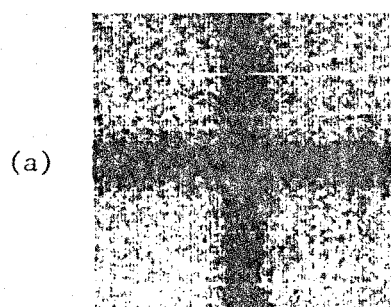


Figure 3.2 Reseau Reproductions
Into 16 Gray Shades

- (a) 4.2 micron aperture
- (b) 8.5 micron aperture

```

+++++
+|      1 12225786321 1 1 1+
+      1 12 1245787321 1 1 1+
+11 1      11212246886541      +
+11 1 2 1 113469A842 11      +
+1      1.112122235975431 21  +
+1 31 12 1 1212447766111 1      +
+1 1 1 2 1111 124467532      +
+ 1      1 11212346664311 1 1  +
+1 111 112 22347765411 111 1 1+
+213111 1111 223468863221 11  +
+1 1111111 112337896521 1111 1  +
+2111 11111124367987421111111 1+
+332211112122323568896231223 1121+
+33232233122214576998553332321221+
+65465453444536889ACB764545633255+
+/36887678778878ABDDA887666657669+
+AA73A999A9BA8CCCCFDBABBB99A9939A+
+ABBA999AAABADCBDDIEEDCBBCABCCA98A+
+93987889868883AACDDBA9A989998898+
+5667464446567577ACCB866565665457+
+44432233334435446998662343233333+
+23212111233233545777434242231222+
+312 1121 11312245698431111 1112 +
+111111 1111 1134667832111121 11 +
+ 11 111 22346785421111 1111+
+ 1 111 3456786431 1 11  +
+2 1 1 1 111 234537643121 1  +
+      11122553A853211211  +
+1 1 11 1 2224489994212 11  +
+ 11 11 1124679A8422111  +
+ 1 1 11124358B8421 1 1  +
+      2467796422 1 1 1  +
+++++

```

Figure 3.3 Original 32 x 32 Reseau Sample
File Number 4--Light Background

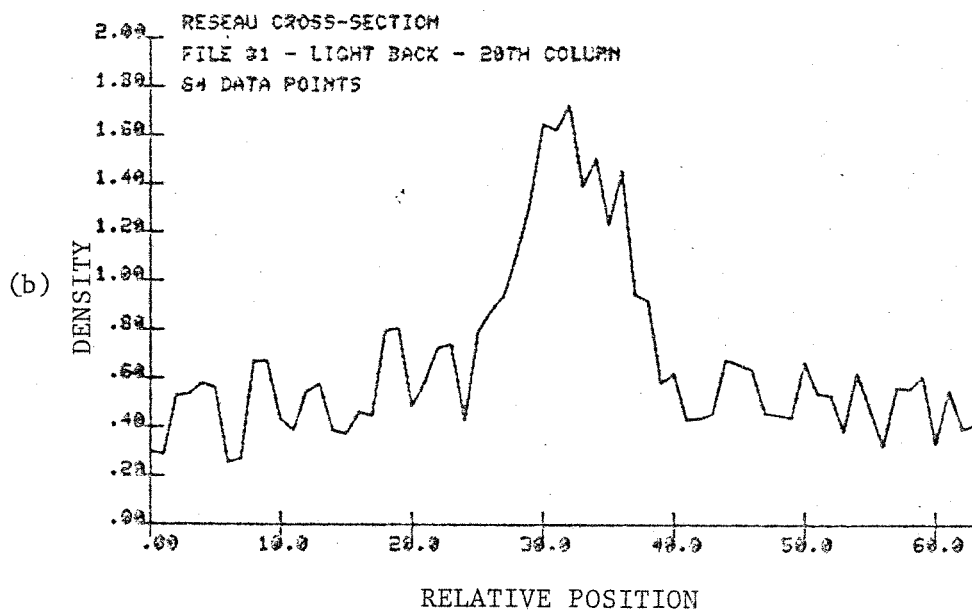
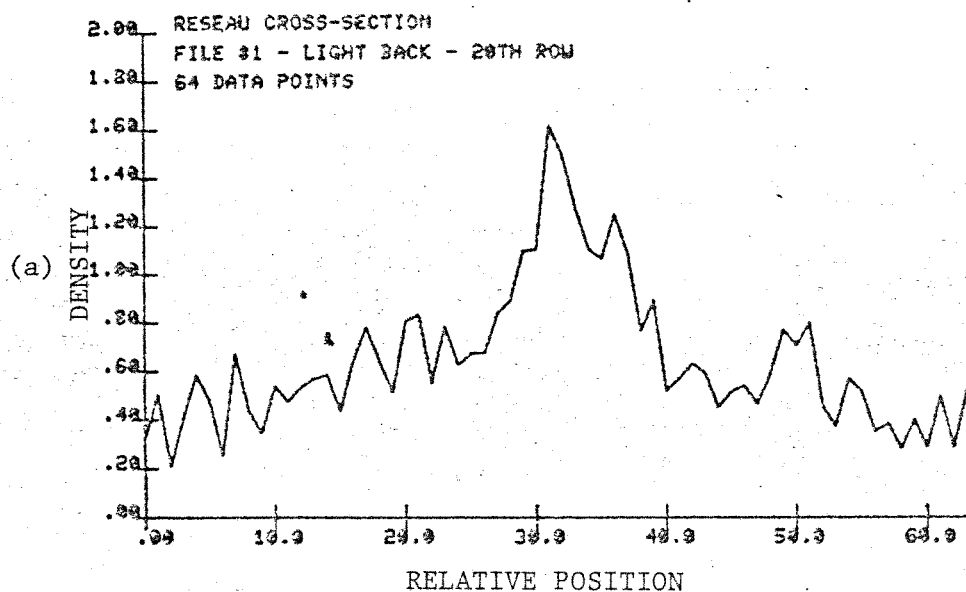


Figure 3.4 Reseau Sample Cross-Sections

- (a) 20th Row
- (b) 20th Column

a silver halide grain and not a function of the image to be recorded on the film negative [5], the digital picture function will be considered a composite of two picture functions; the image of interest and noise. That is,

$$g(i,j) = s(i,j) + n(i,j) \quad (3-1)$$

where

g = composite digital picture function

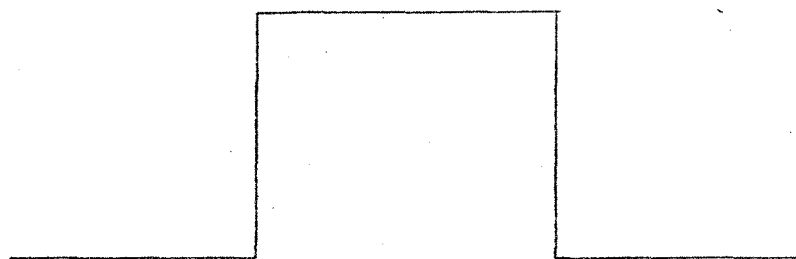
s = digital signal picture function

n = digital noise picture function.

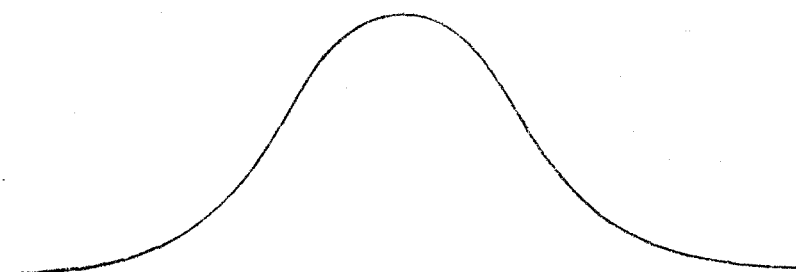
The signal picture function contains the images of interest such as reseaux, buildings, and roads and accompanying distortions because of the photographic process. The granularity of the film will be described by the noise picture function.

3.1 Signal Model [5-9]

The ideal signature of a reseau is a plus-like image containing sharp vertical and horizontal edges. A cross-section of such an image is depicted in Figure 3.5a. The actual signature, however, is corrupted in the photographic process producing an image which has smoothed or rounded edges, such as shown in Figure 3.5b. This phenomenon occurs because of the fact that light incident on a photographic emulsion will reach into a shaded area, such as that produced by the reseau at its edges, through refraction, reflection, diffraction, and scattering of light by the silver halide grains. This property of a material is known as optical turbidity. The particular process, of the four listed above, causing the diffusion



(a)



(b)

Figure 3,5 Reseau Cross-Section Models

- (a) Ideal
- (b) Actual

of light in a given emulsion is a function of the film grain size of that emulsion. This same phenomenon will occur on images of buildings, roads, etc.

The generation of a model of a reseau which includes the diffusion effects is greatly complicated by the fact that the characteristic curve (density versus log exposure) of film is not linear over the entire range. A typical characteristic curve is shown in Figure 3.6. This phenomenon causes the diffusion process to form an image with edges that cannot be duplicated by a simple linear transformation of the ideal reseau model through, for example, low-pass filtering. The shape of the characteristic curve may also be such that the toe and shoulder regions do not have the same curvature, thereby causing the upper and lower portions of the reseau edge to have different curvatures.

The scheme used to transform the ideal model is a modification of the regularization of functions [10]. The regularization process is a smoothing operation that replaces the gray scale value at a given point with the average value of the picture function in the immediate vicinity. For a two-dimensional digital picture function, this operation is defined as

$$S_w(i,j) = \frac{1}{A_w} \sum_{w(i,j)} s(m,n) \quad (3-2)$$

where $w(i,j)$ is a window of area A_w centered at (i,j) . Regularization assumes that each gray scale value within the averaging window is equally weighted. The modification is that of weighting the values within the window in an unequal fashion such that the averaging process produces the

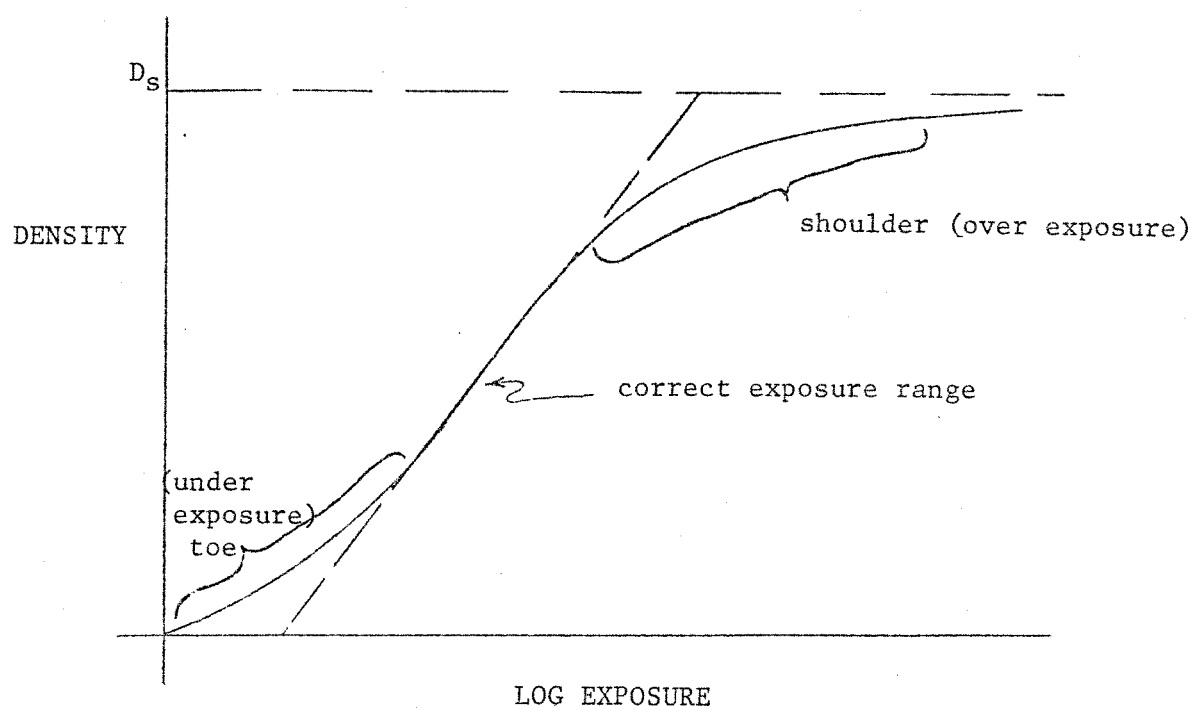


Figure 3.6 Typical Curve of Density
versus Log Exposure [9]

smoothed edges with the slopes desired. In order for this operation to produce the same effect on all edges of the reseau, the weighting matrix must be dynamically oriented depending upon which edge the transformation is forming. This causes the transformation to become nonlinear [11]. The orientation is determined by the direction of the density gradient, since the transformation desired is along the edges of the reseau model. The modified transform is thus defined as

$$S_N(i,j) = \frac{1}{A_N} \sum_{w(i,j)} B_K(m,n) * S(m,n) \quad (3-3)$$

where $w(i,j)$ is a window of total weight A_N and B_K is a weighting matrix whose values are a function of the direction of the gradient. The size of the averaging window and the values of the weighting matrix are determined by a heuristic approach since no two reseaux are identical and a visual comparison seemed sufficient as a performance criterion.

The process of determining the gradient is known as spatial differentiation [10]. The gradient can be estimated if the directional directives are known along two orthogonal directions. A simple approximation to the directional derivatives is given by

$$S_1 = s(i + 1, j + 1) - s(i,j) \quad (3-4)$$

and

$$S_2 = s(i, j + 1) - s(i + 1, j) \quad (3-5)$$

Since the result of the nonlinear regularization process is a function of all the gray scale values within a window, the approximation to the directional derivatives should also be some function of all or a large majority

of these values. Therefore, the approximation used incorporated averaging in the result. Using the example of a 3 x 3 window shown in Figure 3.7, the directional derivatives were approximated by

$$S_x = (c + f + i) - (a + d + g) \quad (3-6)$$

and

$$S_y = (g + h + i) - (a + d + c) \quad (3-7)$$

and the direction of the gradient by

$$\theta = \tan^{-1} (S_y/S_x) \quad (3-8)$$

Results of this transformation and the window and weighting matrices used will be given in Section 3.3.

Modeling reseaux using the nonlinear regularization transformation creates the possibility of simulating, with relative ease, several more attributes of the photographic process. These effects involve the interaction of images with film development phenomena. One effect, known as the Kostinsky effect [5], is an apparent increase in the separation of two small nearby images by a shift in the geometric centers of the images. This effect is produced by an asymmetrical growth of the images. This condition can be produced in the model by simply adjusting the weighting matrices corresponding to opposing edges of the reseau so as to result in an asymmetric image.

Another phenomenon is the physical movement of silver halide grains caused by interacting gelatin effects. This effect, known as the Ross effect [5], results in the movement of two nearby images toward each other.

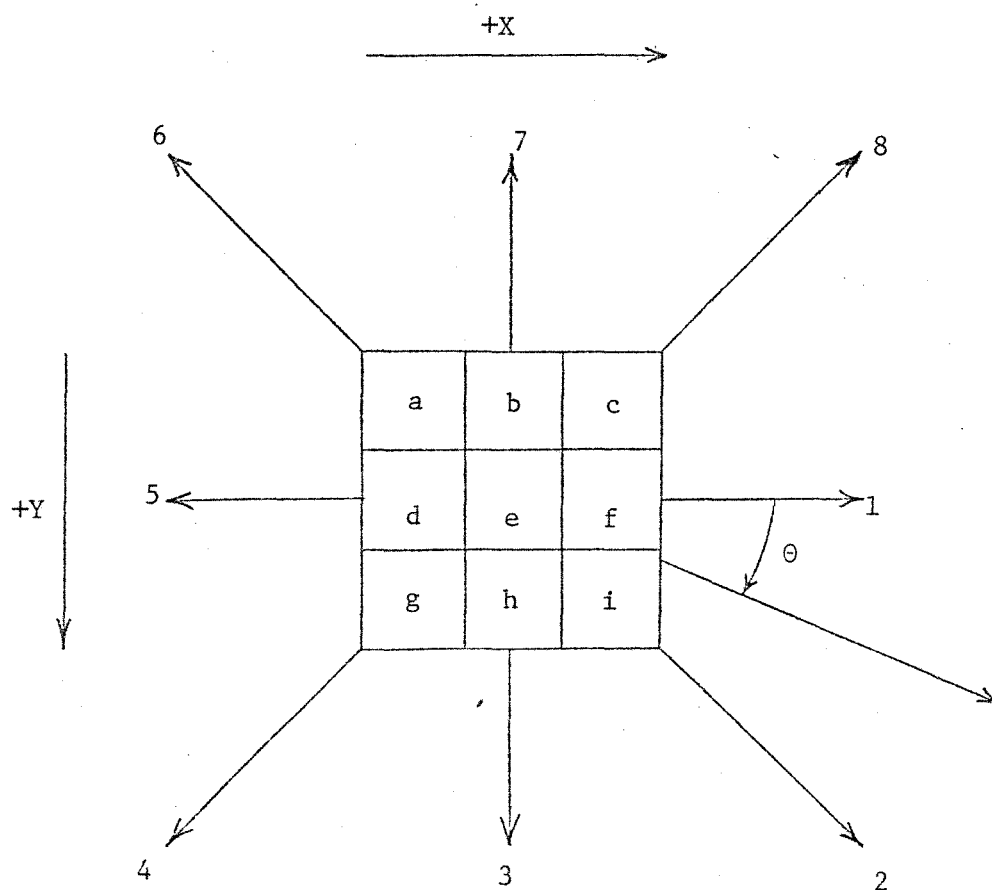


Figure 3,7 3 x 3 Window Used in the
Gradient Calculation

The shifting of any portion of the ideal model before the transformation can be used to simulate this effect.

Other typical phenomena such as absence of a portion of the reseau and different reseau density amplitudes can also be created by modification to the ideal model. Examples will be presented in Section 3.3.

3.2 Noise Model [5-9]

The density fluctuations over a uniformly exposed area are, in general, because of the size of the silver halide grains in the film emulsion. This property is known as graininess or granularity. In some cases, the term graininess is used for both graininess and granularity while in others graininess refers to the subjective response of an observer to the non-uniformity of an image, and granularity designates the objective aspect in terms of spatial variations in the transmitting or reflecting properties of the image.

Many schemes for measuring these quantities have been devised. One such method for assigning a value of graininess to a photographic material is to make a series of enlargements of an image at different magnifications [9]. The viewer then examines these enlargements at a constant distance and determines the maximum magnification at which the graininess is not objectionable. Granularity is measured by scanning a uniformly exposed and developed image and recording the statistics of the scan.

Granularity is the quantity upon which the noise model will be based. Goetz, et. al., [9] determined that the distribution of transmittance and density fluctuations follow a Gaussian law with a discrepancy small enough to be insignificant. Since an exact Gaussian distribution includes an

equal and finite probability for very large plus or minus deviations while the largest minus deviation of a true distribution is equal to the mean, the true distribution cannot be strictly Gaussian. The true distribution demonstrates a degree of asymmetry with the degree varying with the spot size. The Gaussian law, however, is a good approximation and a large number of the granularity measurement systems are based on the Gaussian theory.

Because of the averaging effect of the sampling aperture, the magnitude of the density fluctuations will decrease as the aperture size increases. Since the granularity should be constant for a given film, a proportionality constant which is a function of the aperture size is needed. Selwyn [9] related granularity and density fluctuations by

$$G = \sigma(D) * a^{1/2} \quad (3-9)$$

where $\sigma(D)$ is the standard deviation of the density statistics and a is the area of the scan aperture.

In order to obtain the statistics for the model from the actual reseaux sample data, histograms of 1600 density measurements each were generated from regions of homogeneous background. Examples of two such regions are shown in Figure 3.8. The two sample areas contained different average background levels as is shown by the different mean density values. The effect of the different aperture sizes is very apparent from the different standard deviation since the regions were taken from the same film. Table 3.1 gives the calculated Selwyn granularity values for several different areas of the same film with varying aperture sizes and mean density values.

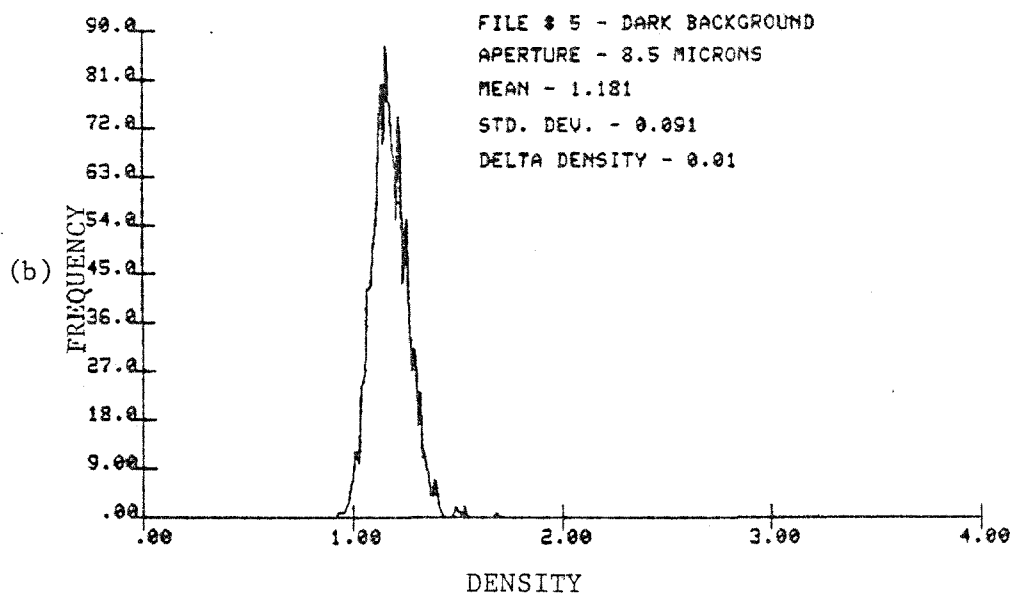
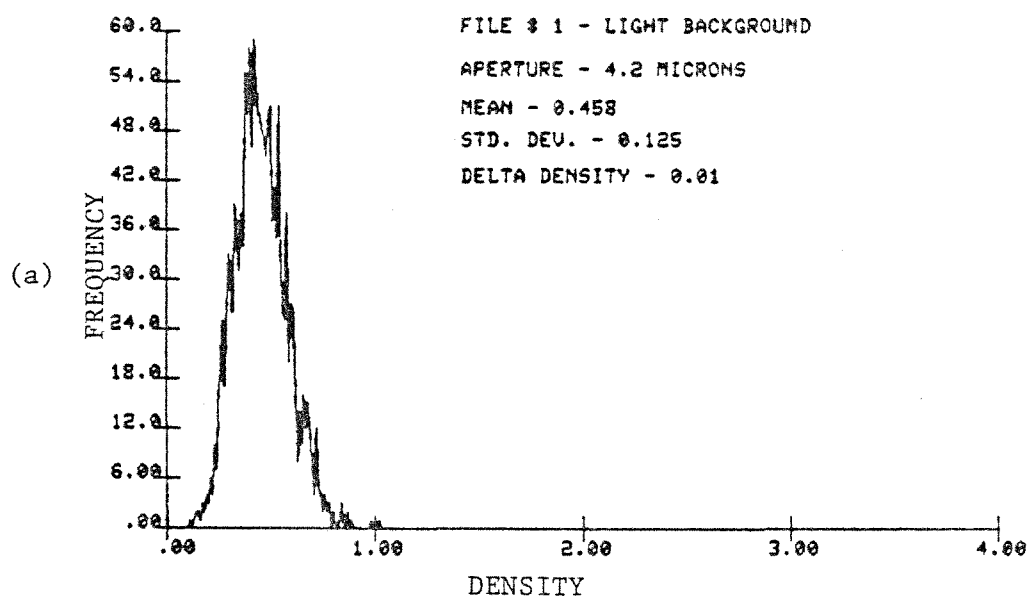


Figure 3.8 Samples of Histograms for
Different Aperture Sizes

TABLE 3.1
CALCULATION OF SELWYN GRANULARITY FOR
VARIOUS SAMPLE DATA

File #	Density Mean	Density Standard Deviation	Aperture Size (Microns)	G
#1	0.458	0.125	4.2	.2562
#2	0.656	0.132	4.2	.2705
#4	0.397	0.075	8.5	.2187
#5	1.181	0.091	8.5	.2653

Although the histograms of Figure 3.8 appear Gaussian, a method of assessing the agreement of these curves to a normal curve was used. The particular test used was the chi-square test [12, 13, 14]. The chi-square statistic, χ^2 , measures the discrepancy existing between the observed and expected frequencies of particular set of k events and is given by

$$\chi^2 = \sum_{j=1}^k \frac{(o_j - e_j)^2}{e_j} \quad (3-10)$$

where o_j is the observed frequency and e_j is the expected frequency. If the events are chosen so as to have expected frequencies of at least five, then the sampling distribution of χ^2 is approximated very closely by the chi-square distribution. The exact chi-square distribution is generated from two sample distributions of which one is drawn from the other. If, therefore, the chi-square statistic falls within a certain region of the mean of the distribution, then the expected and observed values are of the same form.

The shape of the chi-square distribution is a function of the number of independent observations and the number of population parameters which must be estimated from the sample observations. The value is known as the degrees of freedom of a statistic and is given by

$$v = N - k \quad (3-11)$$

where N is the number of independent observations and k is the number of estimated population parameters. In testing the film statistics, the frequencies of the range bins of the histogram are the observations and

the expected frequencies are generated from the sample mean and sample variance of the histogram. The number of estimated parameters is, therefore, two and the number of independent observations is the number of range bins minus one, since the total number of density samples is known. Therefore, the number of degrees of freedom is given by

$$v = M - 3 \quad (3-12)$$

where M is the number of density range bins.

The decision on whether or not the observed data is a good approximation to the expected data is based on whether or not the computed value of χ^2 falls between two critical values of the exact distribution. Two critical values are used to prevent the observed and expected frequencies from not only differing too greatly but also from becoming too similar. This situation could possibly indicate a bias in the measurement process. The critical values are based on the probability of incorrect decision. A five percent level of significance indicates that the null hypothesis would be rejected when, in fact, it should not have been. The null hypothesis states that there is no difference between the two sets of events. Typical values for level of significance are .05 and .01.

Table 3.2 shows the results of the chi-square test on several different histograms. Range bins were combined in order to satisfy the requirement that the frequency be greater than five, therefore, the degrees of freedom varied greatly.

3.3 Model Results

In generating a model for the case of an 8.5 micron diameter aperture

TABLE 3.2

CHI-SQUARE RESULTS FOR 5% LEVEL OF SIGNIFICANCE

File #	Mean	S. D.	Density Inc.	Degrees of Freedom	Critical Values		χ^2
					χ_1^2	χ_2^2	
1	0.458	0.125	.01	50	71.4	32.4	60.62
2	0.656	0.132	.02	26	41.9	13.8	27.00
4	0.397	0.075	.01	20	34.2	9.59	23.50
5	1.181	0.091	.01	36	54.4	21.4	41.87

size, a 5 x 5 averaging window with the weighting matrices as shown in Figure 3.9 was used. The model generating program divided the total possible rotations of the gradient into eight intervals. A different weighting matrix could, therefore, be generated for each 45 degrees of gradient rotation. The number of the weighting matrix given in Figure 3.9 corresponds to the numbered direction of the gradient shown in Figure 3.7. The corresponding matrix was used 22.5 degrees each side of the numbered directions. Figure 3.10b shows the result of the nonlinear transformation on the ideal model of Figure 3.10a.

The noise model was generated using a Gaussian random number generator [15]. Statistics for the noise model were taken from actual homogeneous regions of the film. Figure 3.11 compares a generated reseau cross-section with an actual reseau cross-section. The model was generated by adding the noise model to the signal of Figure 3.10b. The statistics for the noise model were derived from the background area of the same scene from which Figure 3.11b was taken. Figure 3.12 shows a gray scale reproduction of the two images from which the cross-section of Figure 3.11 were taken.

Several artificial reseaux were generated with defects common to actual reseaux samples. Figure 3.13 shows examples of where portions of a reseau did not get recorded on the film for some reason and where the reseau does not vary from the background more than five percent. Figure 3.14 is a model illustrating the Kostinsky effect where asymmetric growth portions of the reseau occur.

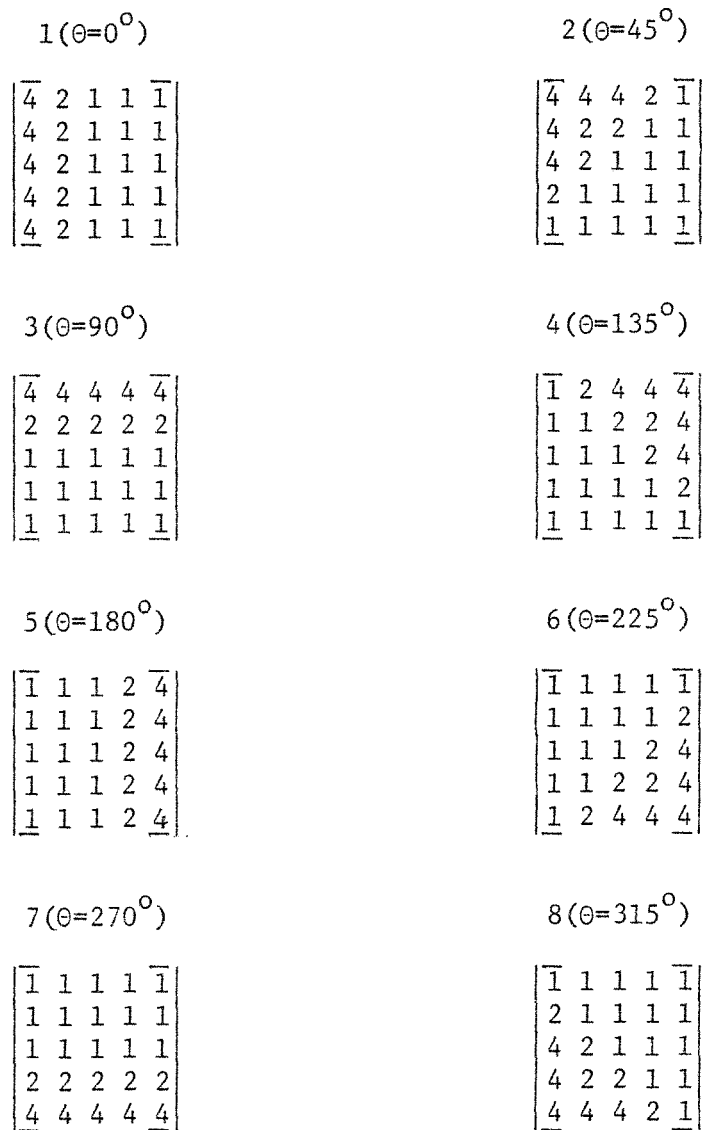


Figure 3.9 Weighting Matrices for Each
 45° of Gradient Rotation

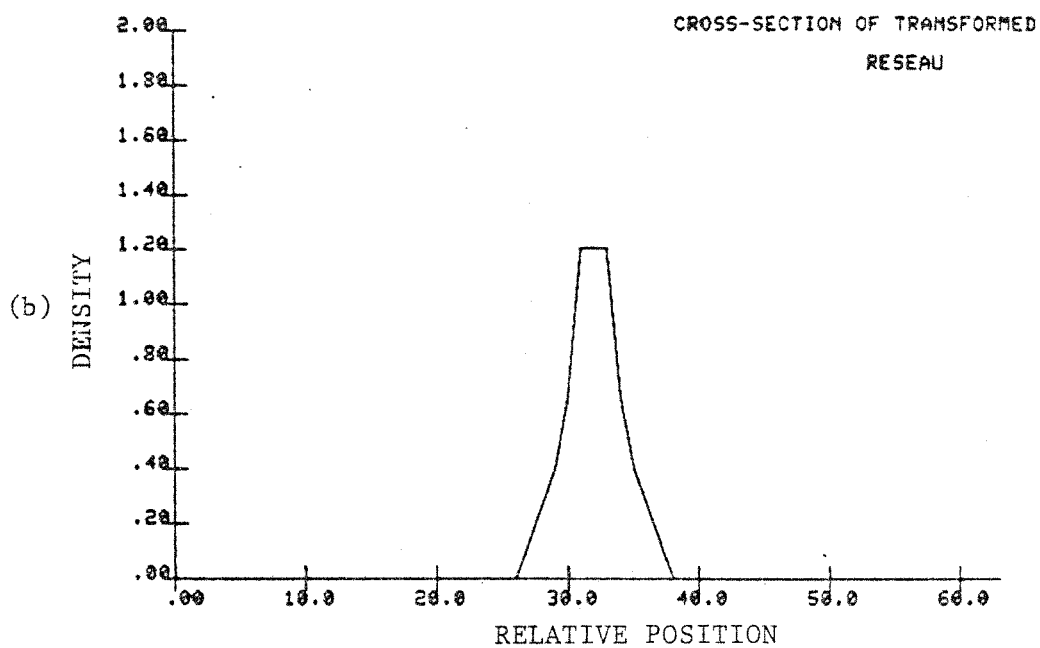
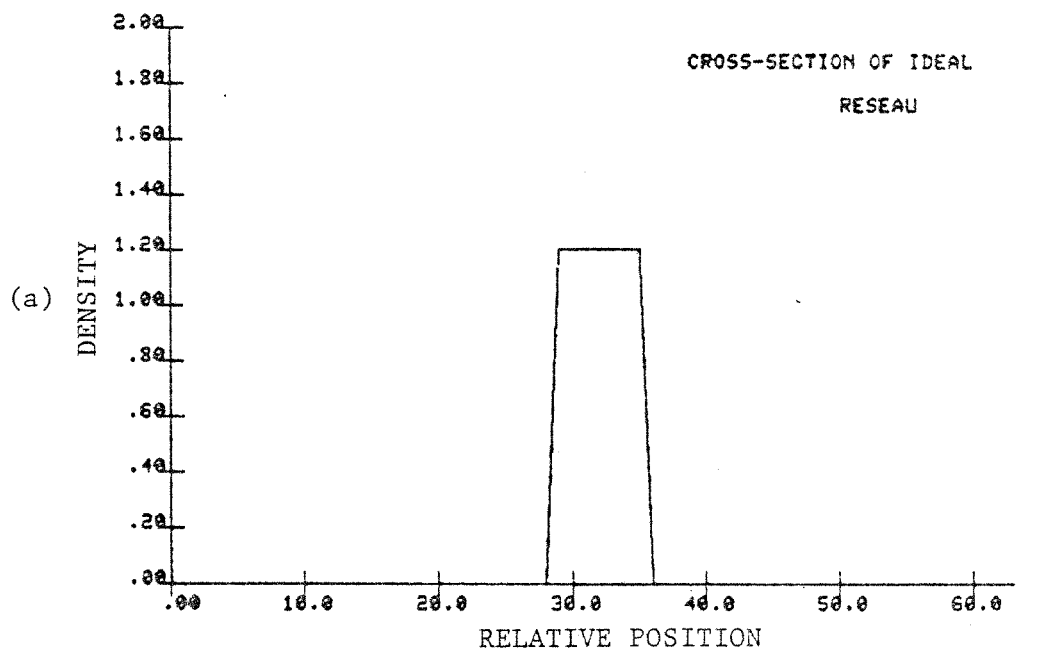


Figure 3.10 Generated Reseau Cross-Section for
the 8.5 Micron Diameter Case Without Noise

- (a) Ideal--8 samples wide
- (b) Corrupted by nonlinear transform

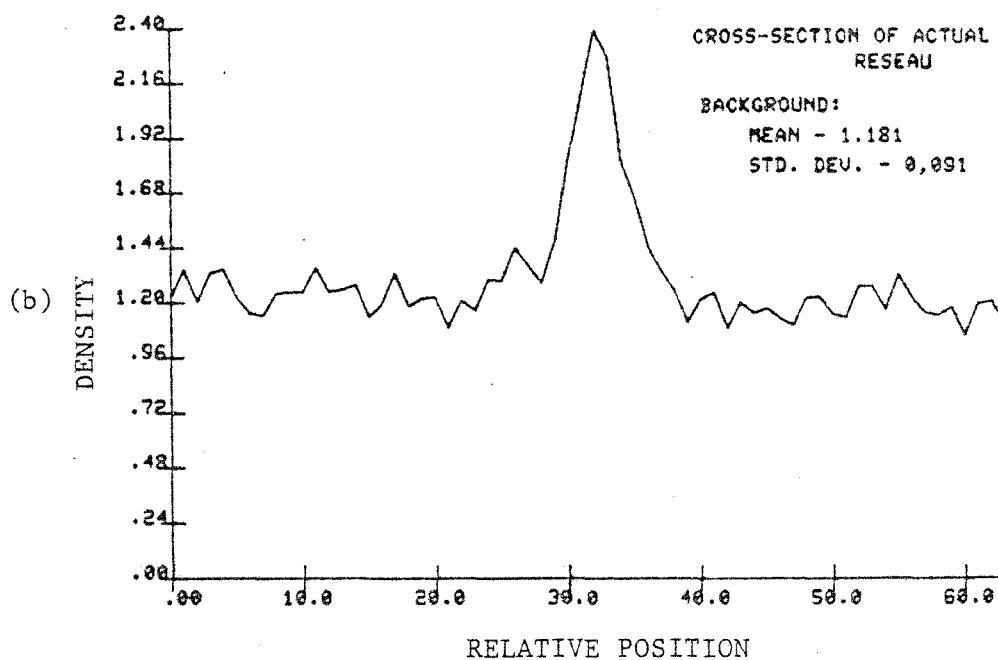
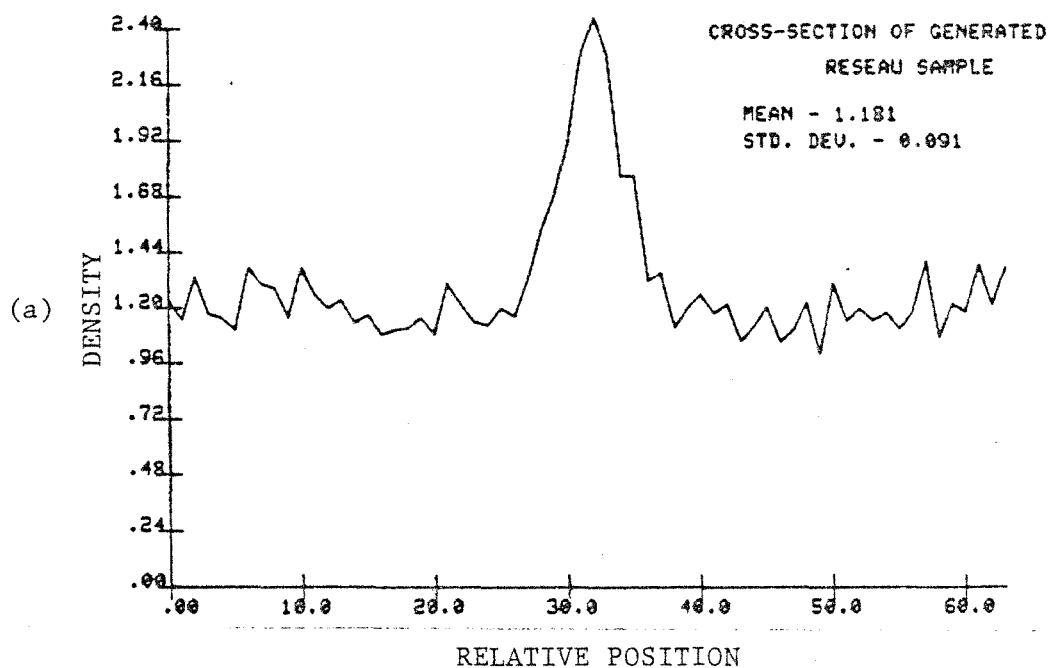
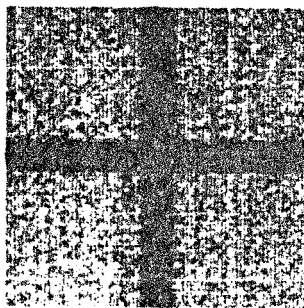


Figure 3.11 Comparison of Generated and Actual
Reseau Cross-Section

- (a) Generated reseau
- (b) Actual reseau

(a)



(b)

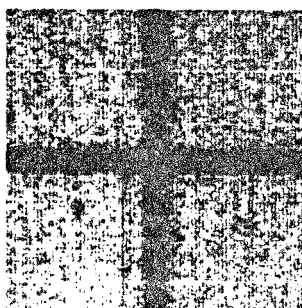
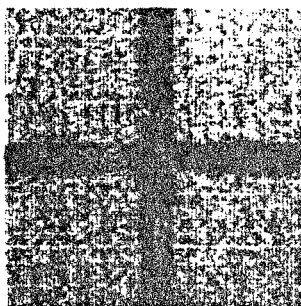


Figure 3.12 Gray Scale Reproduction of a
Generated Reseau and An Actual Reseau

- (a) Generated
- (b) Actual

(a)



(b)

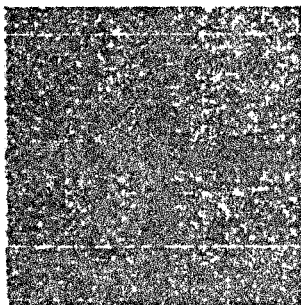
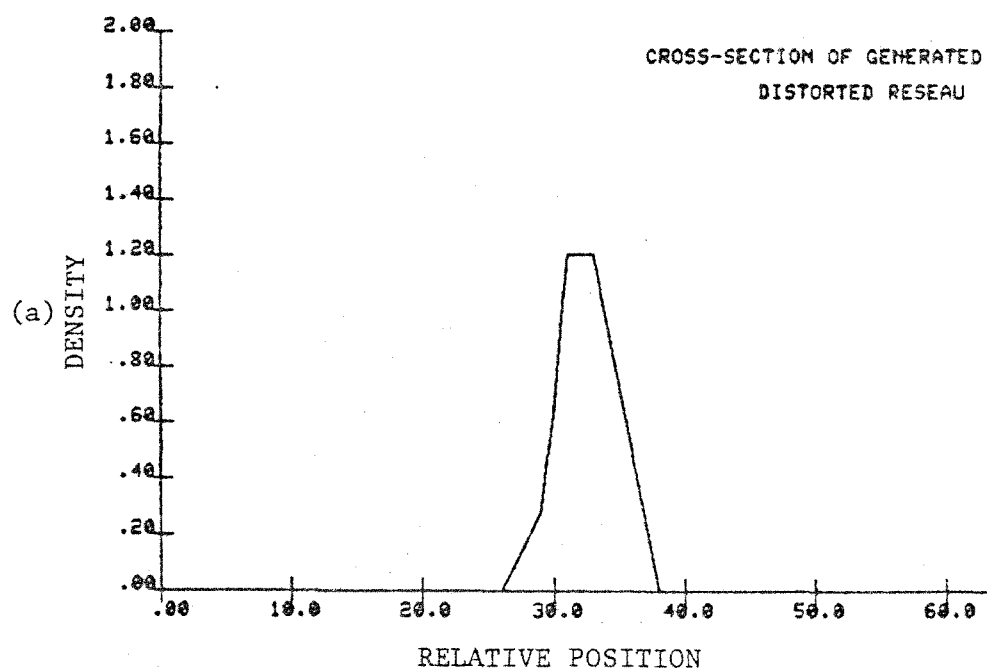


Figure 3.13 Reseau Models

- (a) Missing portions
- (b) Five percent change in transmissivity



(b)

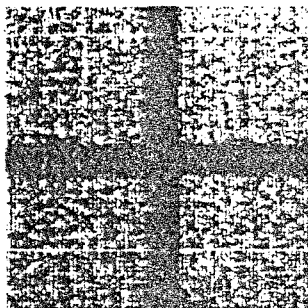


Figure 3.14 Reseau Model of the Kostinsky Effect

(a) Cross-section

(b) Gray scale reproduction

CHAPTER IV

RESEAU CENTER DETECTION

4.0 Introduction

The ultimate goal of accurate location of the center of reseau images places an extra requirement on the performance of pattern recognition and scene analysis techniques. Typical scene analysis schemes that perform such simple tasks as noise removal and edge enhancement now must be thoroughly examined with respect to operations that cause deformation of the actual reseau image to the point where errors in the center detection process are introduced. Also, classical pattern identification techniques such as template matching, when applied to the center detection problem, may be inadequate with respect to center location accuracy.

With these requirements in mind, various image preprocessing schemes were applied to the picture images. Some of the schemes served a two-fold purpose. The main purpose was to create a transformed image that was more "noise-free" and would result in a cleaner image for the center detection process. The secondary purpose was to obtain spatial frequency information about the picture image. Spatial filtering, via the Fourier and Walsh-Hadamard transforms, is an example of this kind of preprocessing. As will be shown, the picture images produced by Fourier and Hadamard filtering contained reseau images which had been greatly modified necessitating consideration of other schemes. These schemes were chosen on the basis of their ability to preserve such things as edges or the image in general.

Various center detection schemes were tried of which two are presented along with the results achieved. The first method employed the classical technique of template matching. Although this scheme is usually applied to finding objects within a picture, the resulting location of the template can be used to determine the center of the image. It will be shown that, because of problems in defining the template accurately as well as limitations in the scheme itself, center location errors can occur.

The second method was a combination of template matching and computation of the center of gravity. Templates were used to find the edges of the reseau image and the center of gravity was then calculated for picture elements or pictels within the reseau boundaries. This scheme required preprocessing to remove noise prior to locating the edges. Both methods are described and comparative results presented below.

It should be noted that only methods that led to automatic operation for both preprocessing and center detection schemes were considered since an automatic system was envisioned.

4.1 Image Preprocessing

4.1.0 Fourier Transform

Given an analog picture function $g(x,y)$ assumed defined over the infinite plane, its Fourier transform $G(f_x, f_y)$ is defined by

$$G(f_x, f_y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} g(x,y) \exp [-2\pi j(f_x x + f_y y)] dx dy \quad (4-1)$$

and its inverse Fourier transform is defined by

$$g(x,y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} G(f_x, f_y) \exp [2\pi j (f_x x + f_y y)] df_x df_y \quad (4-2)$$

Equation (4-2) can be viewed as an expansion of the picture function $g(x,y)$ in terms of a sum of weighted complex exponentials or weighted spatial frequencies [10]. These weights are produced by Equation (4-1) and give the spatial frequency content of the picture function.

Since a digital computer can only operate on finite sets of discrete samples, integration and infinite summations must be approximated by finite summations and continuous functions must be sampled. In the case of the Fourier transform, this process is called the discrete Fourier transform (DFT). The two-dimensional DFT is defined by

$$A(m,n) = \frac{1}{MN} \sum_{k=0}^{M-1} \sum_{\ell=0}^{N-1} z(k,\ell) W_M^{-km} W_N^{-\ell n} \quad (4-3)$$

where $m = 0, 1, \dots, M-1$ and $n = 0, 1, \dots, N-1$. Also, $W_\alpha = \exp \{-2\pi j / \alpha\}$ and $z(k,\ell)$ is an array of complex numbers [19]. Equation (4-3) can be written as

$$A(m,n) = \frac{1}{M} \sum_{k=0}^{M-1} \left[\frac{1}{N} \sum_{\ell=0}^{N-1} z(k,\ell) W_N^{-\ell n} \right] W_M^{-km} \quad (4-4)$$

or

$$A(m,n) = \frac{1}{M} \sum_{k=0}^{M-1} B(k;n) W_M^{-km} \quad (4-5)$$

where

$$B(k;n) = \frac{1}{N} \sum_{\ell=0}^{N-1} z(k,\ell) W_N^{-\ell n} \quad (4-6)$$

and is the one-dimensional transform [19]. The two-dimensional transform is, therefore, the application of the one-dimensional transform on each row (column) with the resulting matrix being transformed by columns (rows).

The two-dimensional DFT can also be expressed in matrix form by

$$\overline{A} = \frac{1}{MN} \overline{B}_M \overline{z} \overline{B}_N \quad (4-7)$$

where \overline{B}_M is a $M \times M$ matrix with its elements at row k and column ℓ given by $W_M^{-k\ell}$ and \overline{B}_N is a $N \times N$ matrix with its respective elements given by $W_N^{-k\ell}$. Various other matrix formulations exist since the sampled function is usually real and the transform array may include different elements representing the real and imaginary components. A similar development can be shown for the inverse DFT. A fast algorithm for calculating the DFT and inverse DFT has been developed and is called the fast Fourier transform (FFT) algorithm [16-19].

It should be noted that only one-half the values of the one-dimensional transform are unique since all the values past $N/2$ are the negative frequency components of the function.

The two-dimensional Fourier transform becomes useful in picture processing as a means of describing a picture by its spatial frequency content. Edges in a picture introduce spatial frequencies along a line in the complex frequency plane orthogonal to the edge. Sharp edges will contribute high spatial frequency components which will lie toward the edge of the frequency plane. More gradual edges introduce components near the origin of the plane. An approximately uniform gray level picture will, for example, contain most of its picture energy near the origin of the

frequency plane or near the average value component, These components have the dimension of cycles per picture cell. Information on the extent of aliasing can be obtained from the higher frequency spectrum coefficients, and changes in the sampling process can be made to correct this condition.

In picture processing, the FFT finds greater use as a mathematical tool to perform other image transformations. Such operations as cross-correlation or template matching are sometimes performed more easily and efficiently using the FFT. Also, filtering operations can be applied to picture functions by multiplication in the frequency domain followed by an inverse transformation. This is a much simpler operation to implement than is the equivalent convolution summation in the spatial domain.

The Fourier transform becomes of interest in the reseau detection problem because picture noise associated with film grains and extraneous spikes contain, in general, much higher spatial frequencies than the reseau image itself. The removal of these high frequency components or low pass filtering eliminates much of the unwanted noise found in the original picture.

Results of low and high pass filtering using the Fourier transform are shown in Figure 4.1 and 4.2. The filter function implemented was that of an ideal filter. Figures 4.1a and 4.2a show the density values for a typical record (row or column) of a reseau picture. Figure 4.1b is the result of low pass filtering with the removal of all frequency components above the fundamental. The cross-section of Figure 4.1c

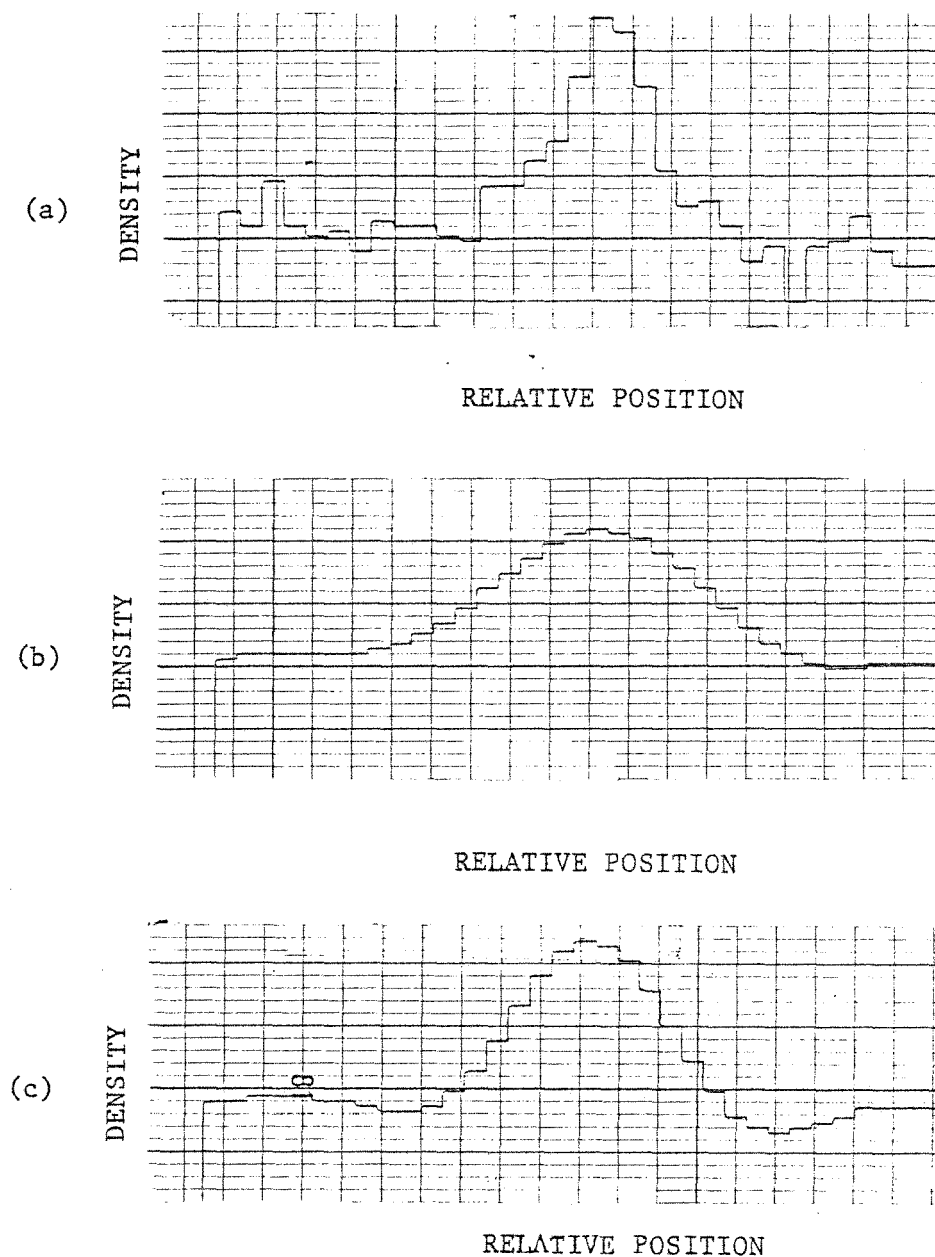


Figure 4.1 Low Pass Filtering Using the
Fast Fourier Transform

- (a) Original cross-section
- (b) Frequency terms removed above fundamental term
- (c) Frequency terms removed above second harmonic term

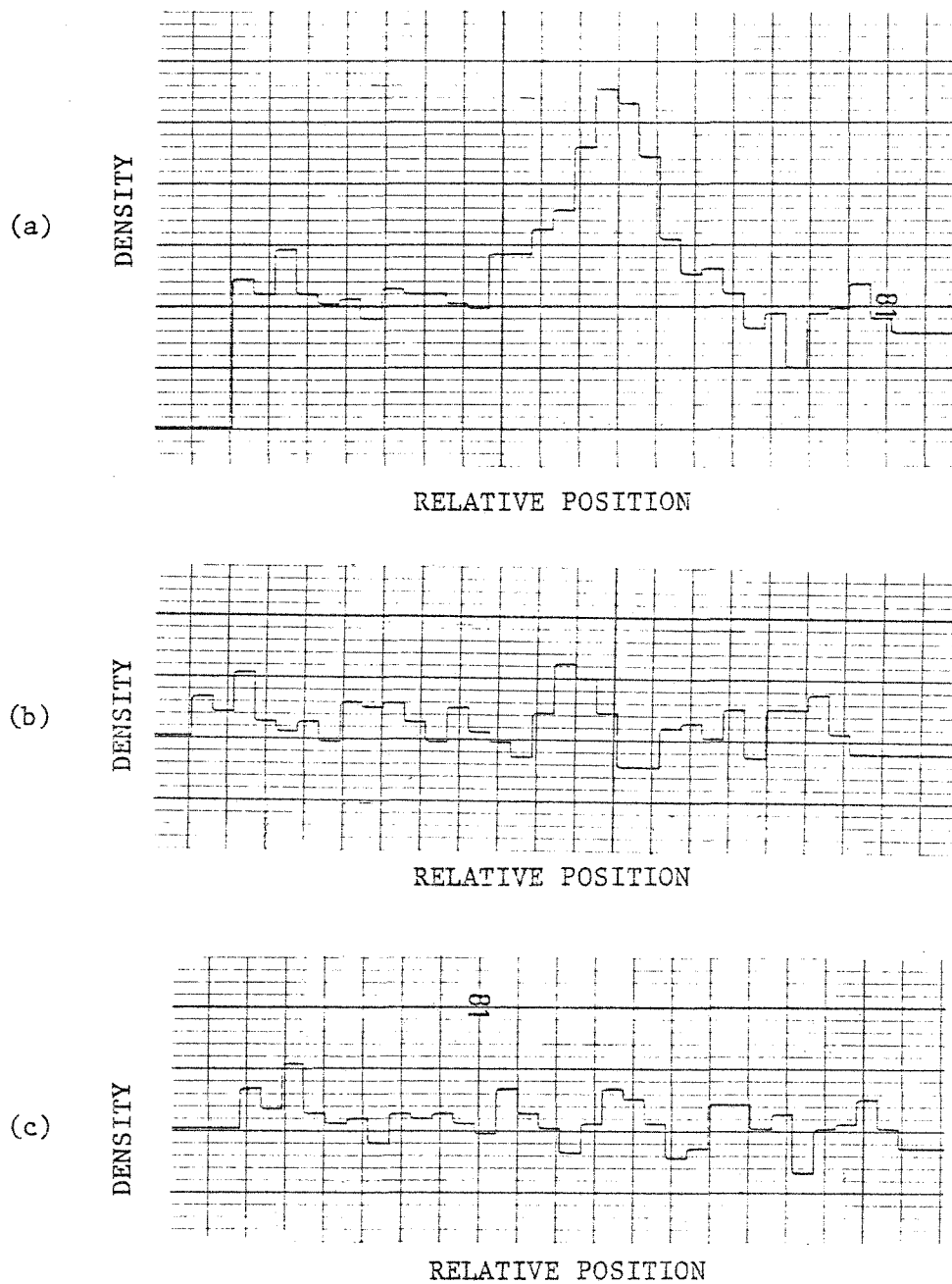


Figure 4.2 High Pass Filtering via Fast Fourier Transform

- (a) Original cross-section
- (b) D.C., fundamental and second harmonic terms removed
- (c) D.C. through third harmonic terms removed

contains components through the second harmonic. Figures 4.2b and 4.2c show the result of high pass filtering. The curve of Figure 4.2b contains all frequency components above the second harmonic with Figure 4.2c including also the third harmonic term. These results demonstrate the validity of the assumption that much of the noise energy lies in higher frequency components than that of the reseau. The examples of Figures 4.1 and 4.2 were taken from a 32 x 32 sample image. In keeping with the sampling theorem, the Fourier transform contains only 16 spatial frequency terms. If an increase in frequency resolution is desired, a corresponding increase in the number of samples must occur. The sampling interval, however, must remain unchanged. If the higher terms are not approximately zero, aliasing has occurred and a decrease in the sampling interval must take place in order to insure no loss of information. Indications of aliasing were not present in the results largely because of the fact that the equivalent of low pass filtering was accomplished optically in the sampling process.

4.1.1 Walsh-Hadamard Transform

In 1923, J. L. Walsh published a set of orthogonal functions which take on the values of ± 1 and are similar in oscillation to the trigonometric functions [19]. The relation between the Fourier harmonics and the Walsh functions are shown in Figure 4-3. Sequency, the analog of frequency, is defined as the average number of zero crossings of the Walsh functions. The Sal function is the Walsh function similar to the sine function and the Cal function is similar to the cosine. By periodic sampling of these Walsh functions, matrices that possess transform properties can be generated. One such ordering of the sampled values produces

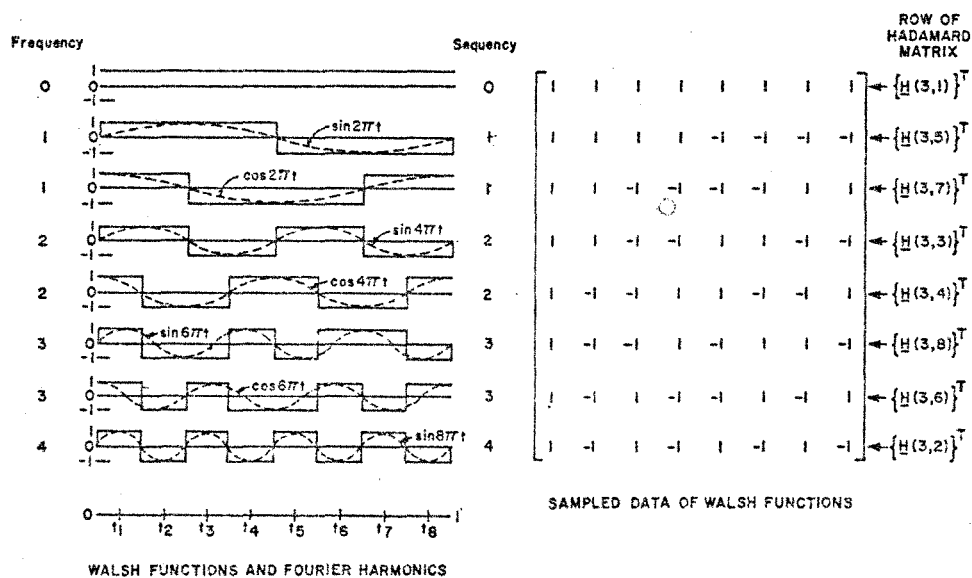


Figure 4.3 Fourier Harmonics, Walsh Functions, and Hadamard Matrices for $n = 8$ [20:226]

the Hadamard matrix [20-21]. A Hadamard matrix for an eight point transform is shown also in Figure 4-3.

The Hadamard matrices can be expressed in either "natural" or "ordered" form. Although the natural form provides the capability of generating large Hadamard matrices from smaller matrices, the ordered form is more useful for interpretation in the sequency domain. In this form the sequency of each row is larger than the preceding row and the transform results are ordered in terms of Walsh functions of increasing sequency.

The one-dimensional Hadamard transform is given in terms of the Walsh functions by

$$F(u) = \sum_{x=0}^{N-1} f(x) \text{wal}(u, x) \quad (4-8)$$

where $N = 2^n$. The Walsh functions can be generated by

$$\text{wal}(u, x) = \sum_{i=0}^{n-1} g_i(u) x_i \quad (4-9)$$

where

$$\begin{aligned} g_0(u) &= u_{n-1} \\ g_1(u) &= u_{n-1} + u_{n-2} \\ g_2(u) &= u_{n-2} + u_{n-3} \\ &\vdots \\ g_{n-1}(u) &= u_1 + u_0 \end{aligned}$$

and the terms u_i and x_i are binary representations of u and x ; that is [21]

$$u = (u_{n-1} \ u_{n-2} \ \dots \ u_1 \ u_0)_{\text{BINARY}} \quad (4-10)$$

The two-dimensional Hadamard transform can be expressed in matrix form by

$$\overline{F} = \overline{H}_M \overline{f} \overline{H}_N \quad (4-11)$$

where \overline{H}_M and \overline{H}_N are symmetric Hadamard matrices of order M and N respectively and \overline{f} is a M x N array of real values [21].

The Walsh-Hadamard transform was used to produce the same end result as the Fourier transform, that of filtering. A fast algorithm for this transform also exists. The big advantage is that the computation time required for the Hadamard transform of a picture function is much less (about 40%) than that required for the Fourier transform of the same function. This results from the replacement of the sine and cosine functions by ± 1 . Also, since the Walsh transform contains only real components, only half the memory necessary for a FFT is required. It should also be noted that all coefficients of transform are meaningful.

The result of Walsh-Hadamard filtering of a typical digital picture function is shown in Figure 4.4. Figures 4.4b and 4.4c are the resulting cross-sections of low pass filtered data with the remaining sequency terms indicated.

4.1.2 Regularization and Nonlinear Filtering

Averaging of sub-pictures within a digital picture is a common method for eliminating noise. This process known as the regularization of functions and used in generating the reseau model is defined by

$$S_w(i,j) = \frac{1}{A_w} \sum_{w(i,j)} s(m,n) \quad (4-12)$$

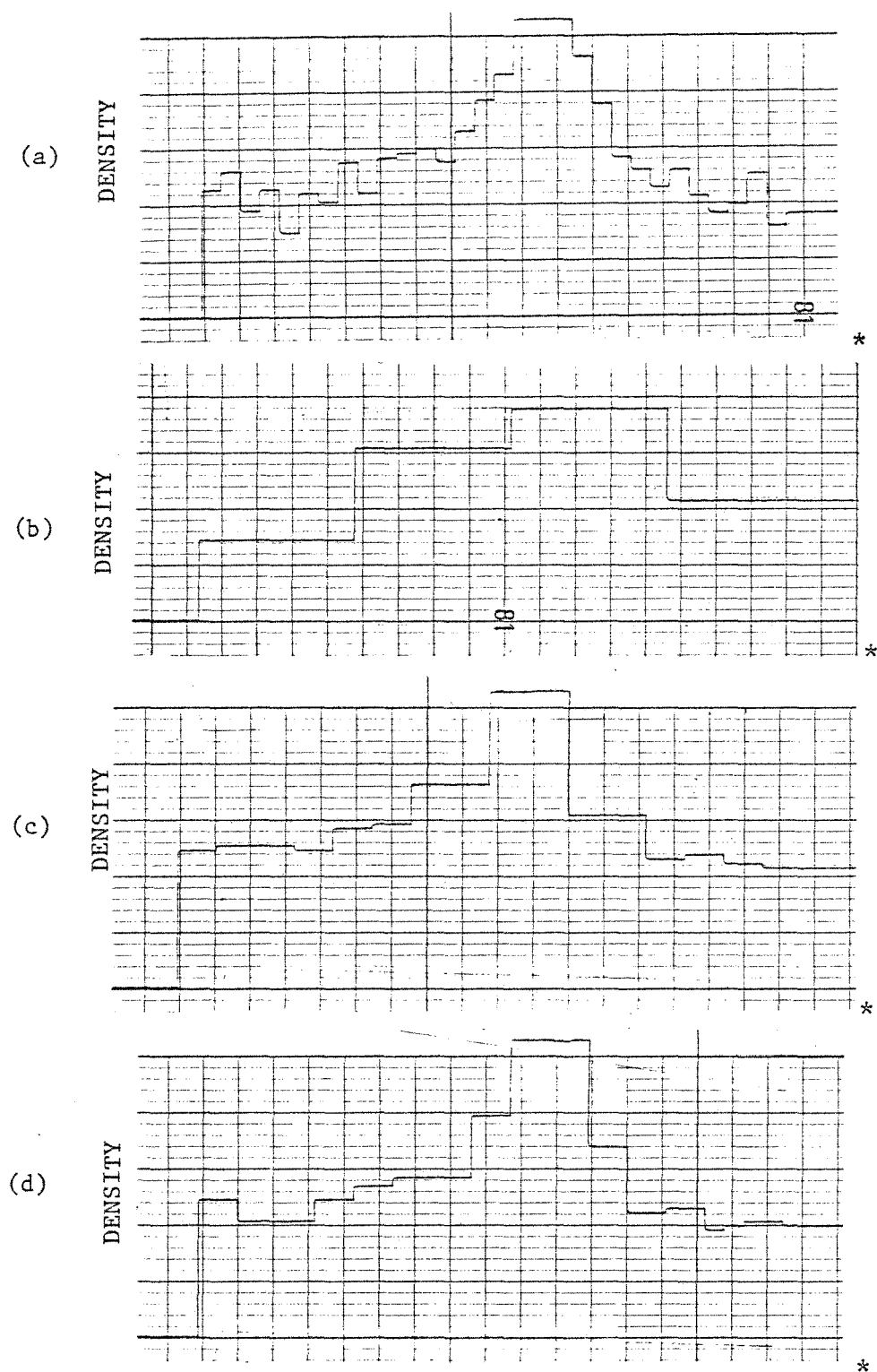


Figure 4.4 Low Pass Filtering via Fast Hadamard Transform

- (a) Original cross-section
- (b) D.C., Sal and Cal terms
- (c) D.C., 4 Sal and 4 Cal terms
- (d) D.C., 8 Sal and 8 Cal terms

* RELATIVE POSITION

where $w(i,j)$ is a window or sub-picture of area A_w centered at (i,j) [10]. Since this process is essentially a low pass filter operation, noise is eliminated but the details of the original picture function (such as edges) are also blurred. The degree of filtering accomplished is determined by the dimension of the averaging window.

Since edges are of extreme importance in the reseau detection problem, this blurring is of principal concern. Although various shaped windows can be used to preserve important characteristics, these windows are often picture orientation dependent. The ideal picture-to-picture transformation is one that smooths noise regions, minimizes degradation of sharp edges and is independent of picture orientation. A nonlinear transformation based on regularization is used in an attempt to satisfy these conflicting demands [11].

The nonlinear transformation consists of a long narrow averaging window that is dynamically oriented perpendicular to the gradient at each pixel. This assures that the window will always be parallel to an edge so that degradation will be at a minimum. In order to insure that the gradient at any point is a good indication of the behavior of the scene and not due to noise, neighboring elements were averaged before calculating the partial derivatives. This approach is described in Section 3.1. The possible orientations of the mask were chosen so that the mask would exactly coincide with grid points. This meant that only four orientations were possible; horizontal, vertical, and both diagonals. If more accuracy is necessary, new density values must be calculated that are interpolations of the grid density values. Independence of picture orientation follows

since as the picture is rotated, so is the mask.

Comparison of regularization and nonlinear filtering is shown in Figure 4.5. Figure 4.5a is a typical cross-section of a reseau picture. Figures 4.5b and 4.5c show the results of regularization and nonlinear filtering, respectively. It can be clearly seen that the regularization process tends to smooth the edges whereas the identity of the edges are much more likely to be preserved by the nonlinear filter. Figures 4.6 and 4.7 show cross-sections from orthogonal directions of the same picture before and after the transformation. Picture orientation is evident from these results.

4.1.3 Bayes Threshold Selection Technique*

The method of slicing gray levels of a picture in order to extract objects from a background has been widely used. A variety of methods for determining an optimum threshold for this procedure has been proposed [25-27]. Many of these methods involve a qualitative judgment on the selection of a threshold [25]. Although this approach is appealing for noise removal from the standpoint of degrading the reseau image, a method of automatic selection of the optimum threshold that is adaptive to each picture is necessary. Presented herein is an analytic process using Bayes decision theory for finding an optimum threshold.

In developing the adaptive process it is convenient to consider the scene of interest to be composed of a grid of $n \times n$ sub-scenes. Feature

*A similar analysis for automatic threshold selection based on minimum error is developed in [23]. This reference appeared subsequent to the author's development. The author's work was first reported in [24].

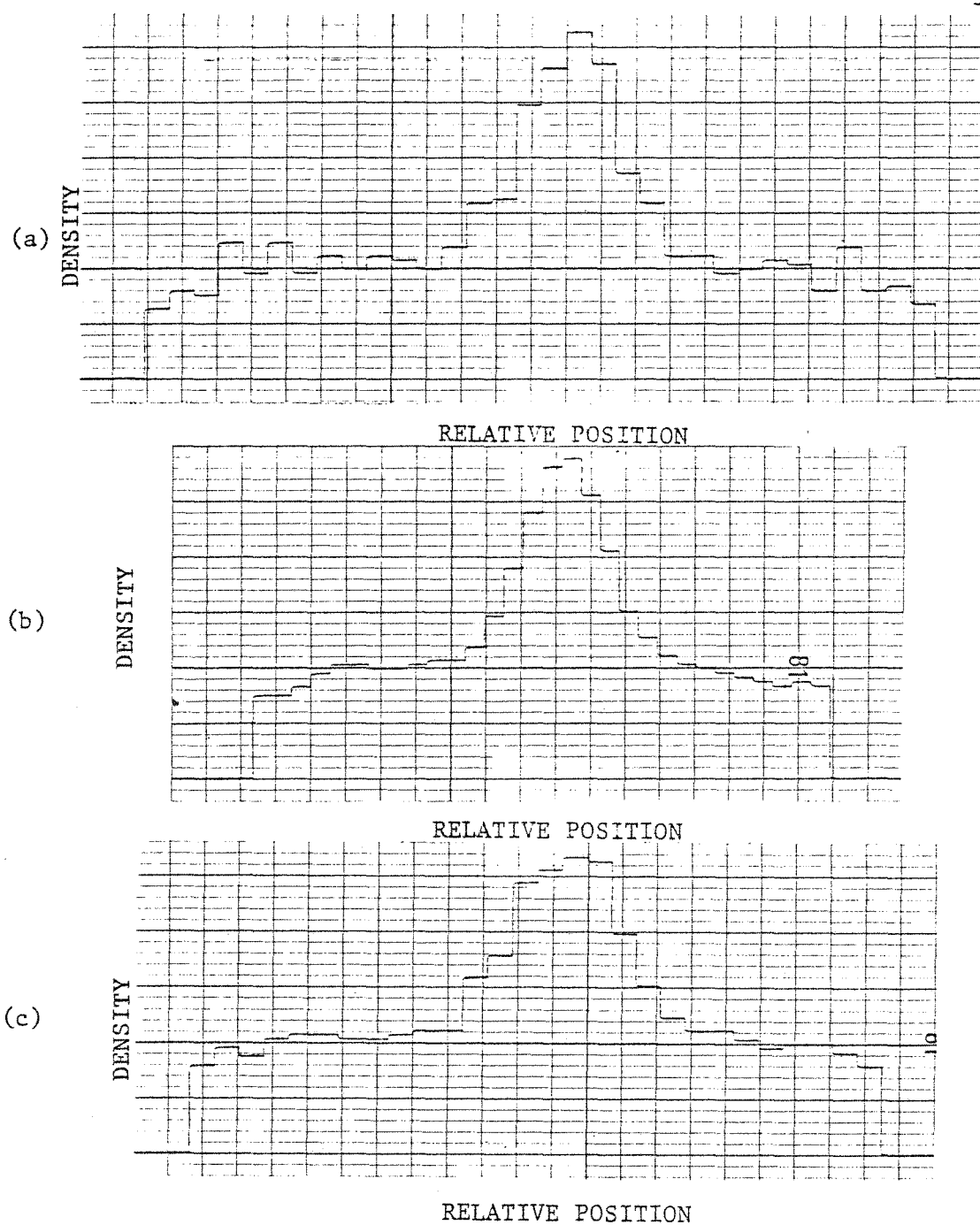


Figure 4.5 Comparison of Regularization and Nonlinear Filtering

- (a) Original
- (b) 3 x 3 average
- (c) 5 x 1 average (3 x 3 gradient window)

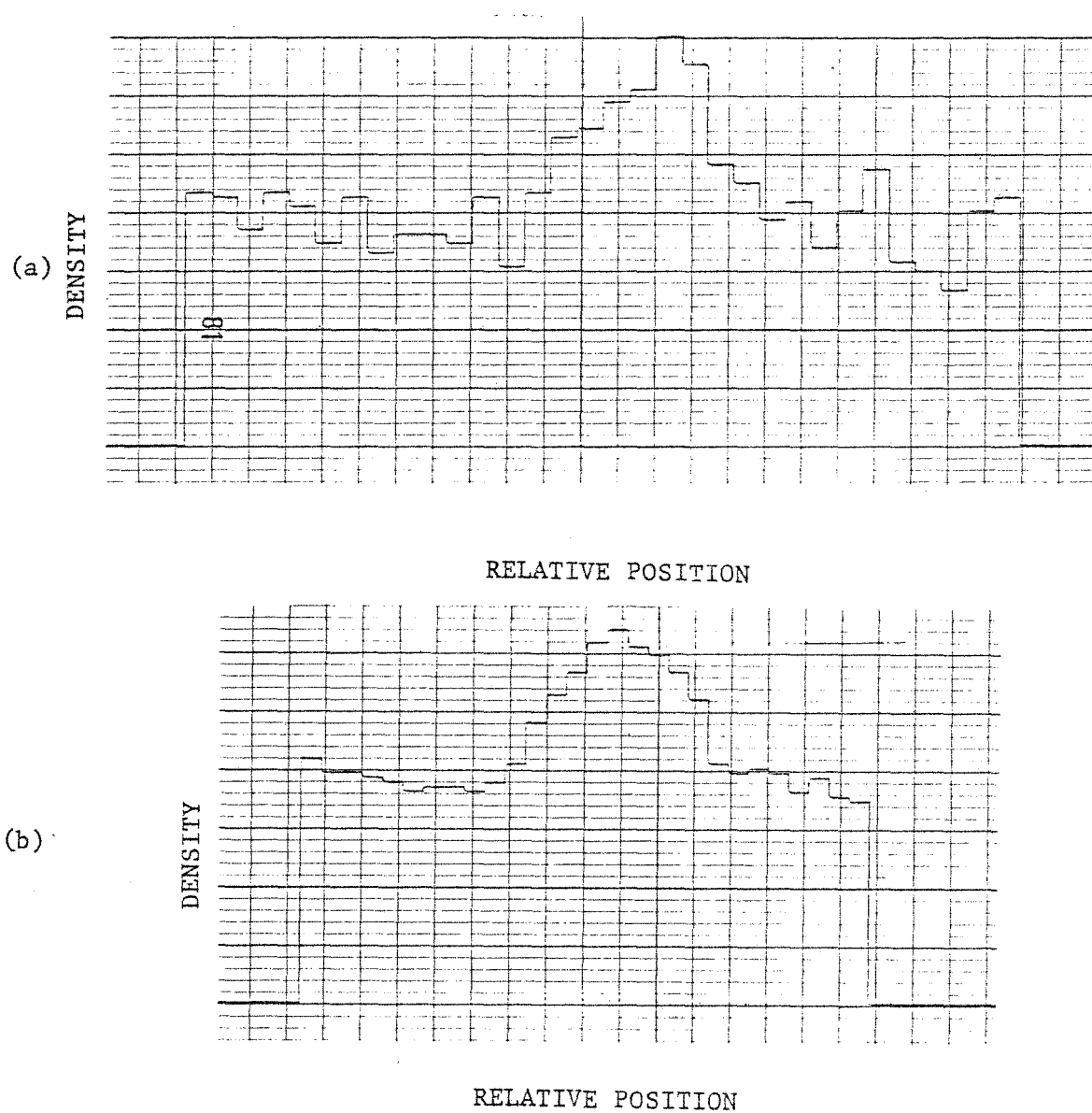


Figure 4.6 Nonlinear Transformation (X-Direction)

- (a) Cross-section of original
- (b) Cross-section of transformed image

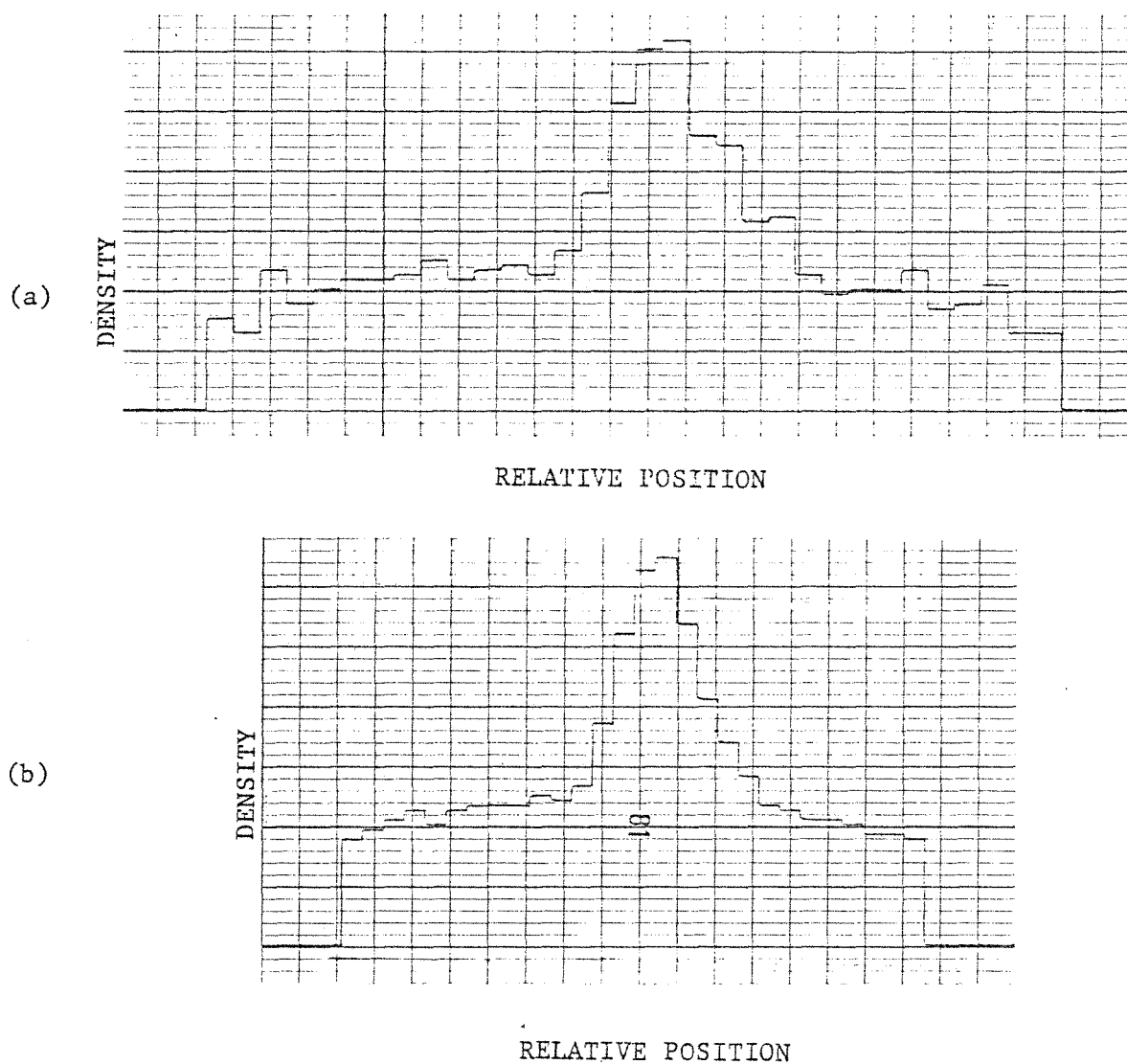


Figure 4.7 Nonlinear Transformation (Y-Direction)

- (a) Cross-section of original
- (b) Cross-section of transformed image

measurements are taken on the sub-scenes and can be assumed to have density functions of the form

$$p(\bar{x}) = \sum_{j=1}^c p(\bar{x}|\omega_j) P(\omega_j) \quad (4-13)$$

where \bar{x} is a vector of m random variables representing the feature measurements, $p(\bar{x}|\omega_j)$ is the class conditional probability density function for each of the c classes, and $P(\omega_j)$ is the a priori probability for each class. A density function of the form of $p(\bar{x})$ is known as a mixture density while $p(\bar{x}|\omega_j)$ and $P(\omega_j)$ are called component densities and mixing parameters, respectively. It is necessary for the c classes to be a mutually exclusive and collectively exhaustive set of events in order for the mixture density representation to hold.

By knowing the component densities and mixing parameters, the a posteriori probability $P(\omega_j|\bar{x})$ can be computed by using Bayes theorem:

$$P(\omega_j|\bar{x}) = \frac{p(\bar{x}|\omega_j) P(\omega_j)}{p(\bar{x})} \quad (4-14)$$

In order to make a decision about a particular class ω_j given a feature measurement, it is necessary to define a function $\lambda(\alpha_i|\omega_j)$, called a loss function. It is defined to be the loss incurred if the decision α_i is made given the class is ω_j . Using this function, the expected loss incurred making the decision α_i from a particular feature measurement is

$$R(\alpha_i|\bar{x}) = \sum_{j=1}^c \lambda(\alpha_i|\omega_j) P(\omega_j|\bar{x}) \quad (4-15)$$

This is also known as the conditional risk. In order to minimize the expected loss for a particular feature measurement, a decision α_i must be chosen that minimizes the conditional risk, $R(\alpha_i|\bar{x})$. By defining a function $\alpha(\bar{x})$ which specifies a decision for every \bar{x} or feature measurement, the overall risk for all \bar{x} can be expressed as

$$R = \int R(\alpha(\bar{x})|\bar{x}) p(\bar{x}) dx \quad (4-16)$$

It can be noted from this equation that by minimizing the conditional risk for each \bar{x} , the overall risk will also be minimized. Therefore, in order to minimize the overall risk, select the α_i decision which minimizes the conditional risk $R(\alpha_i|\bar{x})$ for every \bar{x} . This is known as the Bayes decision rule and is the best performance that can be achieved.

Relating the above theory to the problem of selecting a threshold, consider the feature measurements \bar{x} of each sub-scene to be the density value at the center of each. Therefore, the vector \bar{x} becomes a scalar, or

$$\bar{x} = x \quad (4-17)$$

Each sub-scene is to be contained in one of two pattern classes, ω_1 or ω_2 . One class includes scenes which contain useful or wanted information whereas the other class contains background scenes. These classes will satisfy the requirements of being mutually exclusive, collectively exhaustive, and of having non-zero a priori probabilities needed for Bayes decision theory. The mixture density $p(x)$ now becomes

$$p(x) = p(x|\omega_1) P(\omega_1) + p(x|\omega_2) P(\omega_2) \quad (4-18)$$

It is now necessary to determine the complete probability structure of the scene; that is, the conditional density functions and the a priori probabilities. The structure of the reseau pictures make possible the determination of these parameters. The mixture density, $p(x)$, can be determined by generating a histogram of all the feature measurements over the entire scene. The overall scene must be broken into enough sub-scenes such that the histogram will be a good approximation to the density function. Since the reseau is approximately centered and occupies only a small percentage of the entire scene (approximately 15%), large areas of purely background samples can be found from which feature measurements can be taken to approximate the class conditional probability density function $p(x|\omega_2)$. Since the reseau image size is known and is constant from picture to picture, the a priori probabilities, $P(\omega_1)$ and $P(\omega_2)$, can be approximated. The thresholding scheme will tend to perform more accurately if the scene is largely background, so that the regions that are used to determine $p(x|\omega_2)$ are not near the useful information.

Since exact boundaries of regions of useful information are usually hard to define, it is usually, therefore, impossible to determine the conditional density function for this class. It can, however, be calculated from the mixture density, i.e.,

$$p(x|\omega_1) = \frac{p(x) - p(x|\omega_2) P(\omega_2)}{P(\omega_1)} \quad (4-19)$$

Knowing the statistics of the scene, Bayes rule is used to calculate the a posteriori probabilities,

$$P(\omega_1|x) = \frac{p(x|\omega_1) P(\omega_1)}{p(x)} \quad (4-20)$$

and

$$P(\omega_2|x) = \frac{p(x|\omega_2) P(\omega_2)}{p(x)} \quad (4-21)$$

From these can be obtained the conditional risk functions for the two category case,

$$R(\alpha_1|x) = \lambda(\alpha_1|\omega_1) P(\omega_1|x) + \lambda(\alpha_1|\omega_2) P(\omega_2|x) \quad (4-22)$$

and

$$R(\alpha_2|x) = \lambda(\alpha_2|\omega_1) P(\omega_1|x) + \lambda(\alpha_2|\omega_2) P(\omega_2|x) \quad (4-23)$$

For the two-category case, Bayes decision rule reduces to: decide α_1 if

$$\begin{aligned} \lambda(\alpha_1|\omega_1) P(\omega_1|x) + \lambda(\alpha_1|\omega_2) P(\omega_2|x) < \\ \lambda(\alpha_2|\omega_1) P(\omega_1|x) + \lambda(\alpha_2|\omega_2) P(\omega_2|x) \end{aligned} \quad (4-24)$$

else, decide α_2 .

When the decisions are specifying a particular class for each feature measurement, a decision rule that minimizes the probability of error is intuitively appealing. Therefore, the symmetrical or zero-one loss function is used. That is

$$\lambda(\alpha_i|\omega_j) = 1 - \delta_{ij} \quad (4-25)$$

where

$$\begin{aligned}\delta_{ij} &= 1, i=j \\ \delta_{ij} &= 0, \text{ elsewhere}\end{aligned}$$

The zero-one loss function assigns no loss to a correct decision and unity loss to an incorrect decision. The conditional risk functions reduce then to

$$R(\alpha_1 | x) = P(\omega_2 | x) \quad (4-26)$$

and

$$R(\alpha_2 | x) = P(\omega_1 | x) \quad (4-27)$$

Therefore the Bayes decision rule says that for minimum error rate choose the decision that contains the smallest probability of error, or decide class ω_1 if

$$P(\omega_2 | x) < P(\omega_1 | x) \quad (4-28)$$

else, decide class ω_2 .

This technique provides a completely analytical means of determining an optimal (in Bayes sense) threshold value for scenes whose probability structure can be determined. This scheme has the advantage of being adaptive in the sense that as the probability structure changes from scene to scene, the threshold value changes accordingly.

Results were obtained on the performance of this technique from both generated and actual reseau samples. The a priori probabilities

for the two classes used in the calculation of all the threshold boundaries were based on a reseau image thickness of 7 data samples and the assumption that the image extended to the picture boundaries. This led to a priori probabilities for signal and noise of 0.15 and 0.85, respectively, for the 91×91 matrix picture function of interest.

Figures 4.8 and 4.9 show the results of the technique applied to the model of a reseau in a homogeneous background. Figures 4.10 and 4.11 are the results when applied to an actual reseau sample whose statistics were used to generate the model of the previous two figures. Figures 4.12 and 4.13 show the case where the background was non-homogeneous causing a greater overlap of the conditional density functions. This condition increased the probability that useful information might be lost in the thresholding process.

Figure 4.14 is the cross-section of a generated sample whose reseau image amplitude is only five (5) percent greater than two standard deviations of the noise amplitude. Such a low contrast condition can occur in actual reseau samples and causes a problem for any picture processing scheme [4]. Figures 4.15 and 4.16 show the result of this technique on the low contrast model where the probability of error in the decision was large. Gray scale reproductions of the thresholding process are shown in Figure 4.17 where all density values below the calculated threshold are replaced by the threshold value.

Although the calculation of the total probability of error for each decision is a simple task, the worth of such a process is questionable. The performance of the overall system is to be judged by the error in

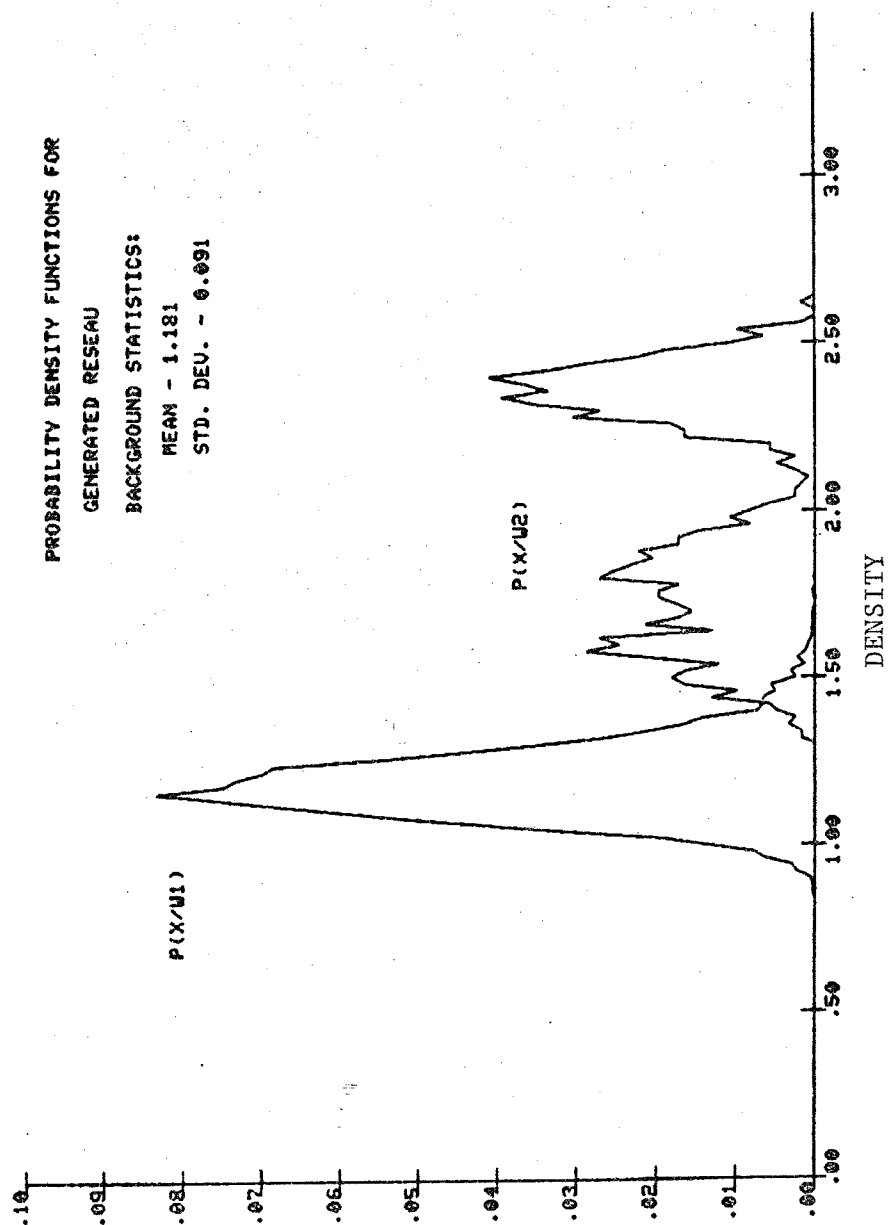


Figure 4.8 Probability Density Functions for a Generated Reseau
in Homogeneous Background

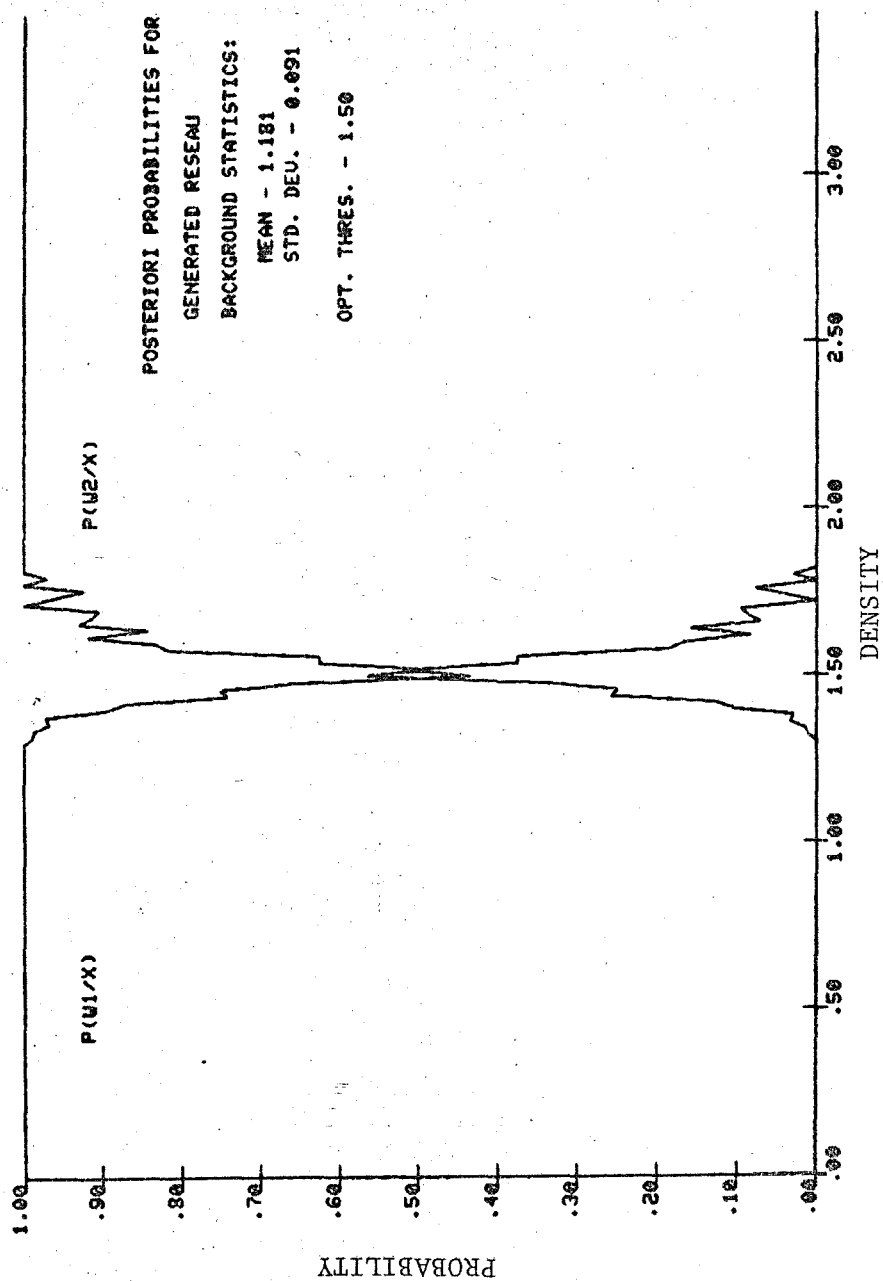


Figure 4.9 A Posteriori Probabilities for a
Generated Homogeneous Reseau

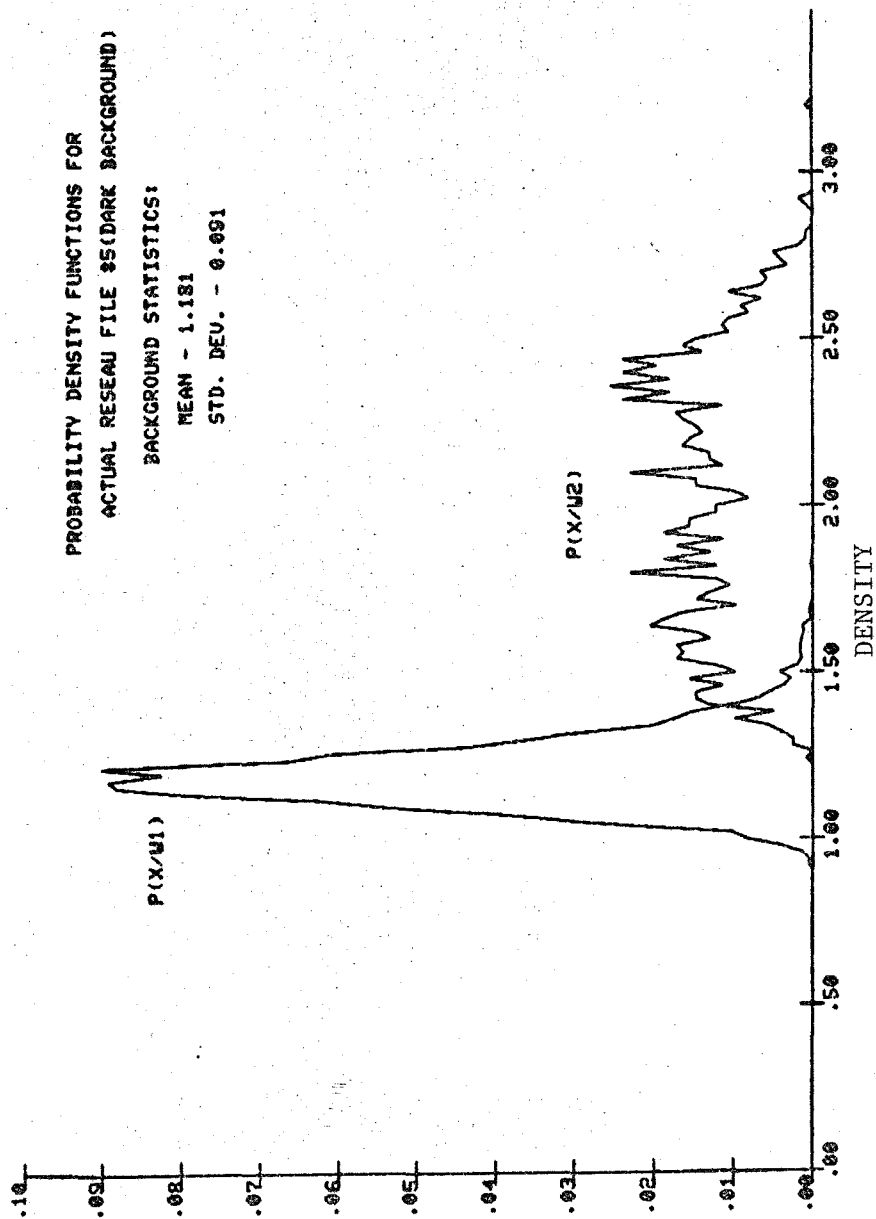


Figure 4.10 Probability Density Functions for an Actual
Reseau in Homogeneous Surroundings

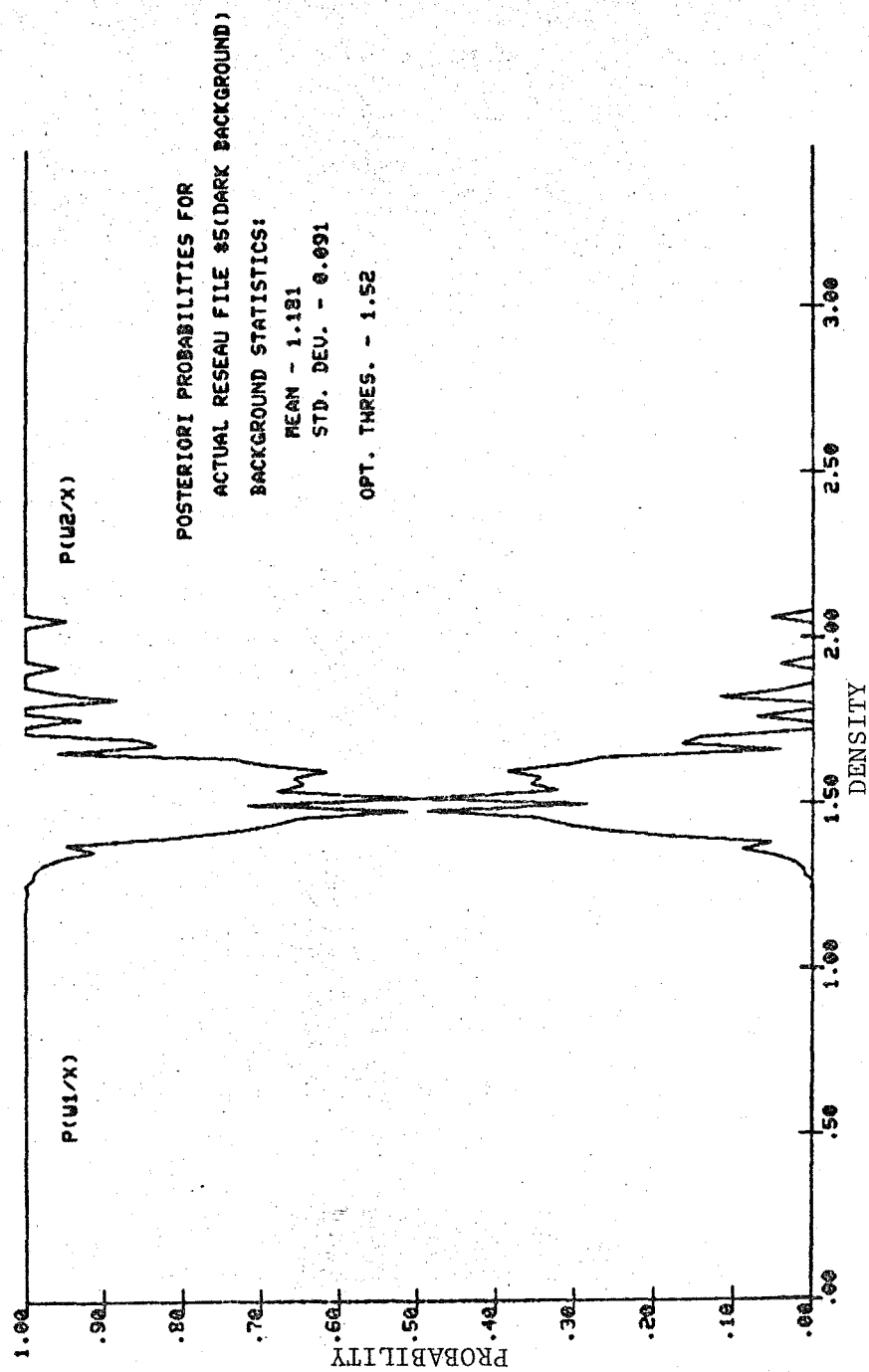


Figure 4.11 A Posteriori Probabilities of an Actual Reseau Sample in Homogeneous Surroundings

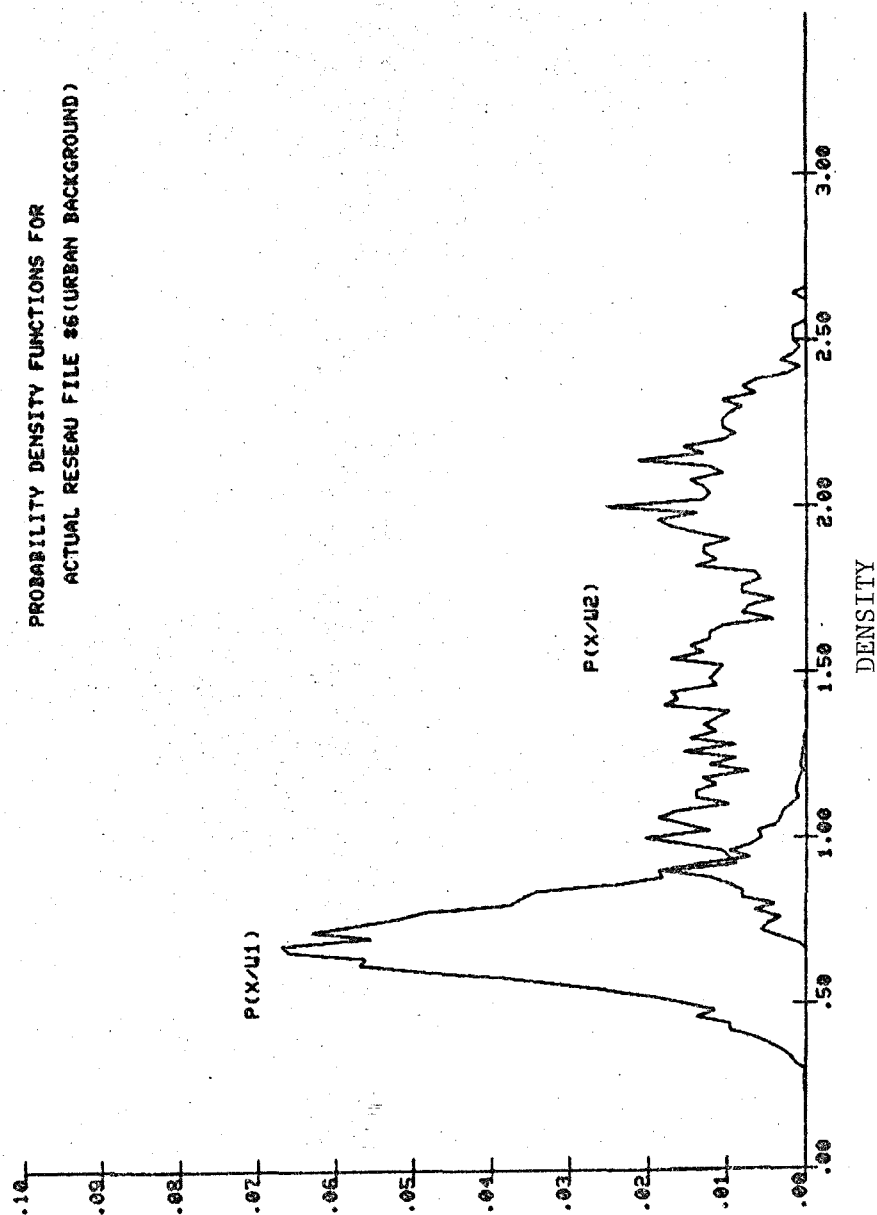


Figure 4.12 Probability Density Functions for an Actual Reseau Sample
in Non-Homogeneous Background

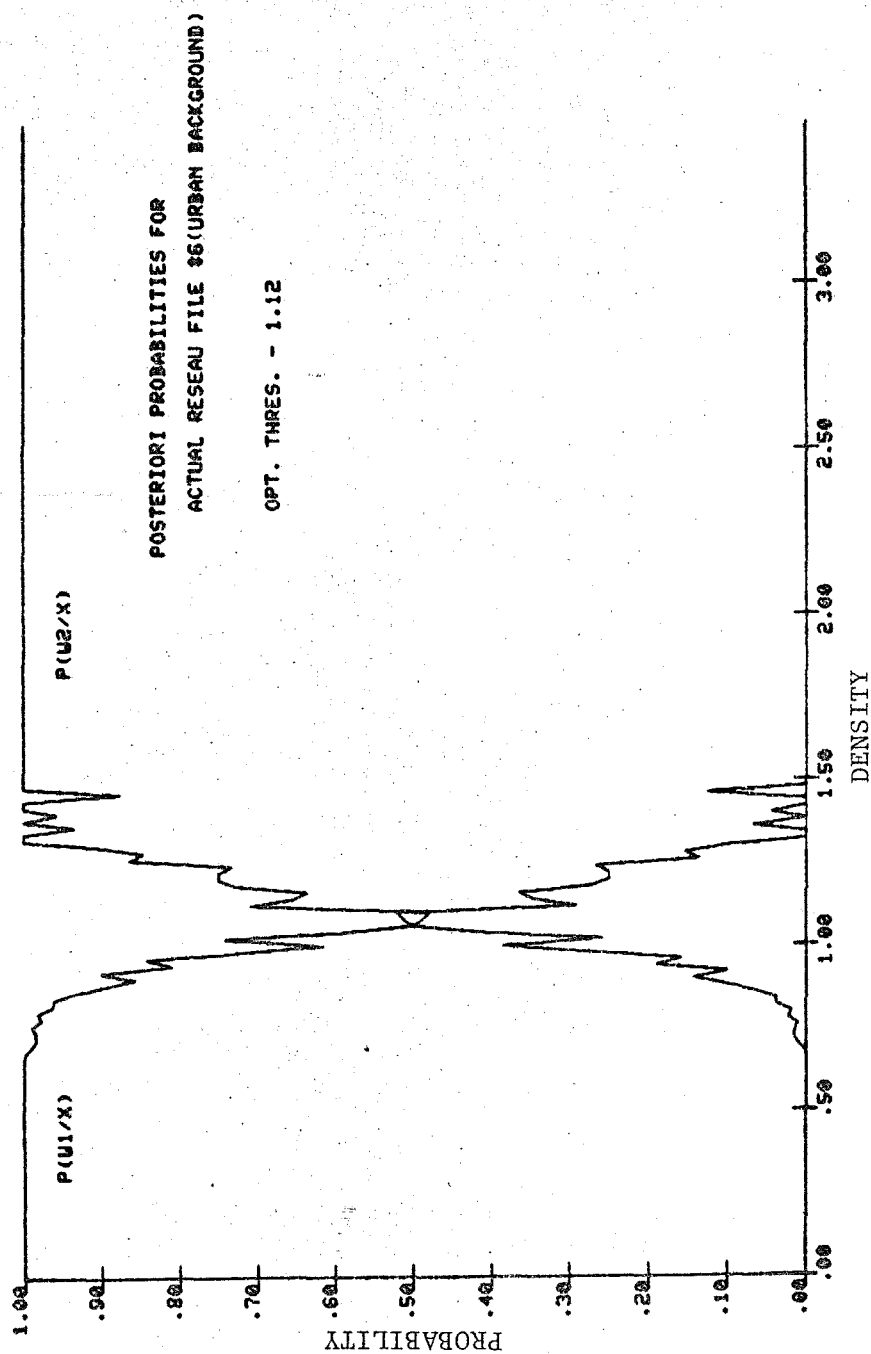


Figure 4.13 A Posteriori Probabilities for an Actual Reseau Sample in Non-Homogeneous Background

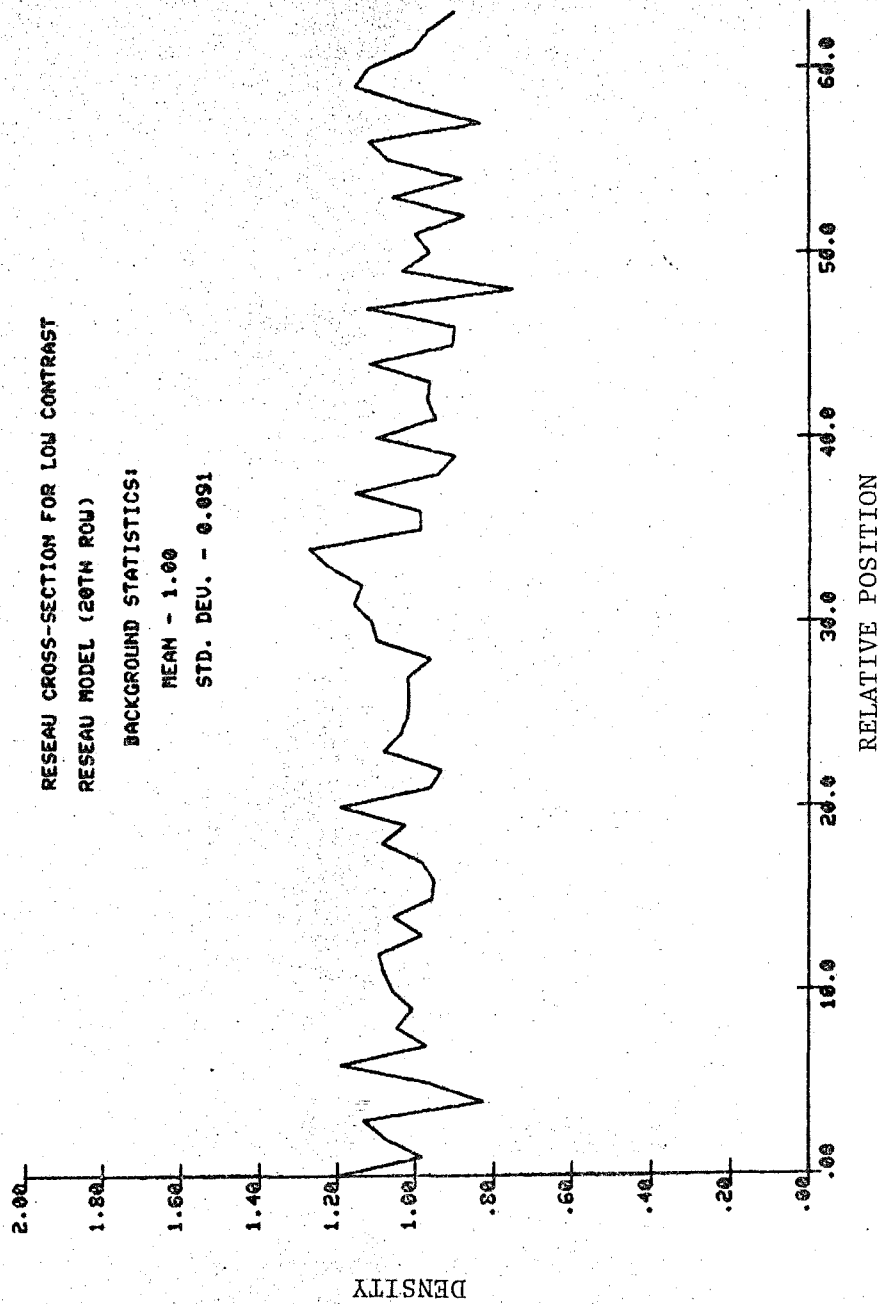


Figure 4.14 Cross-Section of a Generated Sample for Low Contrast Reseau Image

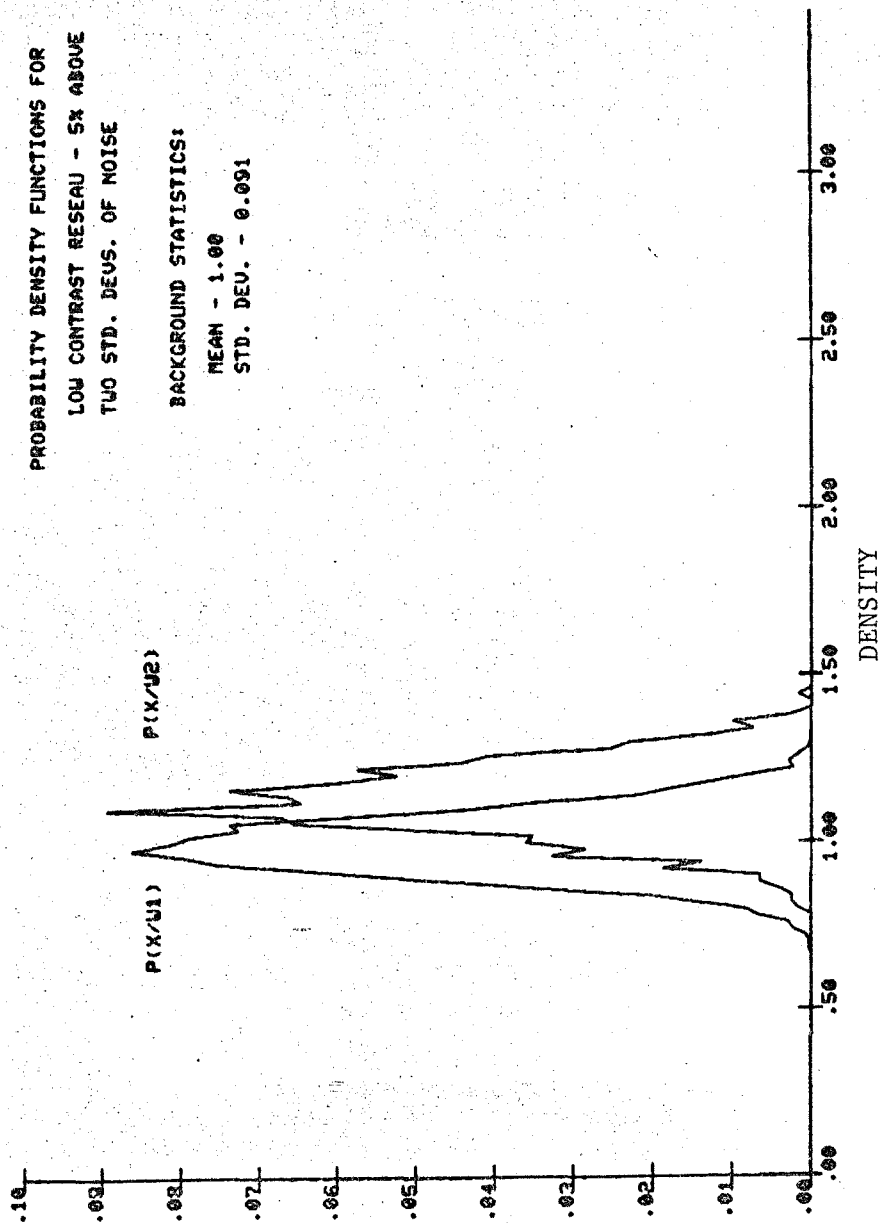


Figure 4.15 Probability Density Functions for Low
Contrast Reseau Sample

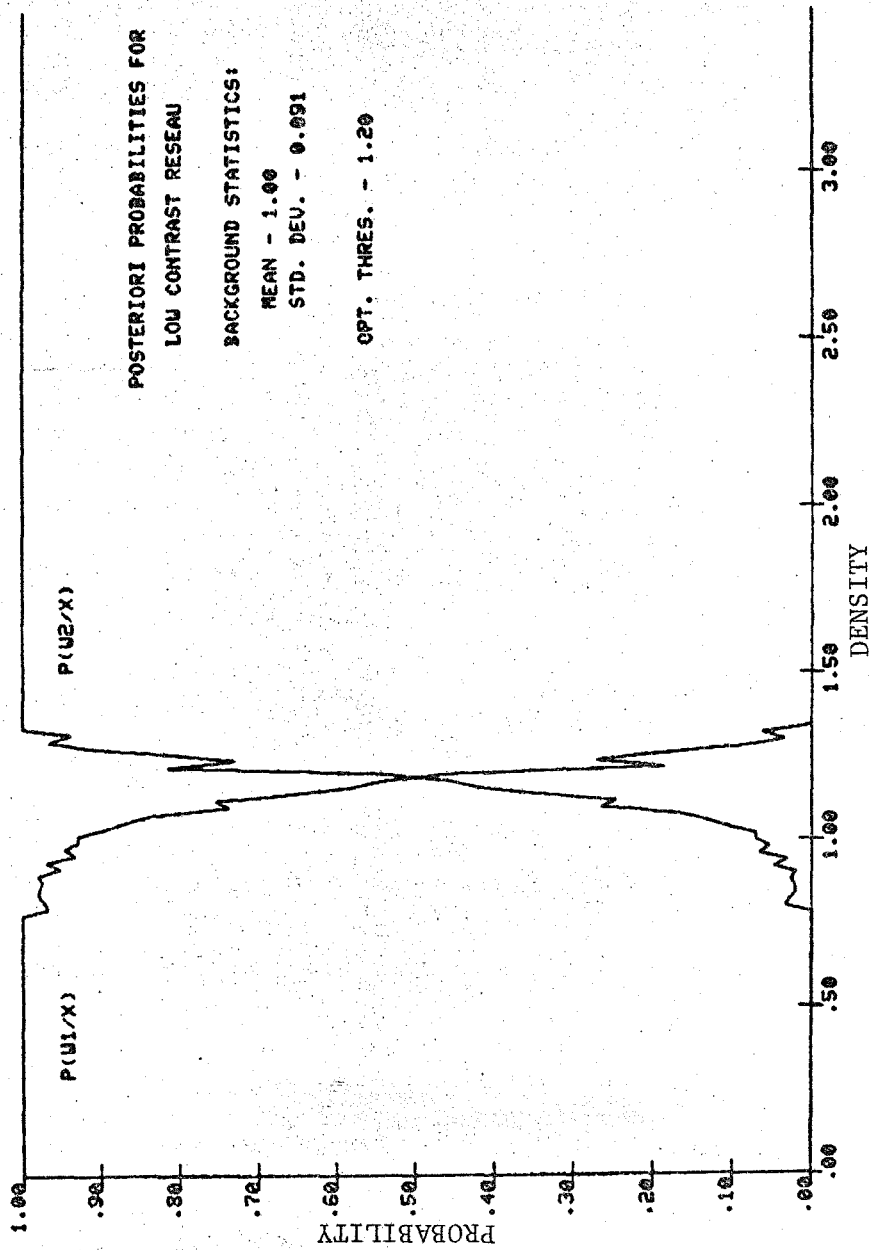
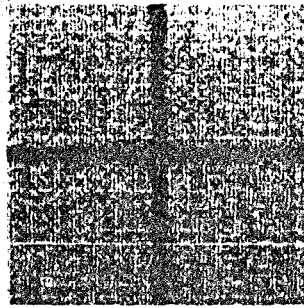
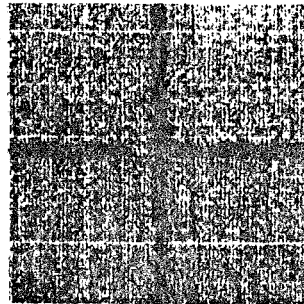


Figure 4.16 A Posteriori Probabilities for Low Contrast Reseau Sample

(a)



(b)



(c)

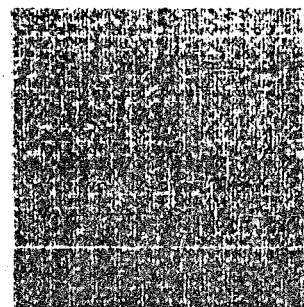


Figure 4.17 Samples of Thresholded Images

- (a) Generated reseau in homogeneous background
- (b) Actual reseau in urban background
- (c) Generated reseau of low contrast

the center measurement. This overall error is a function of the error in the threshold process. However, since several nonlinear transformations will be included in the overall process, the exact relationship between the two errors is difficult to determine. For this reason and the fact that the decision was based on minimum error for the given statistics, the total probability of error was not computed.

4.1.4 Results of Preprocessing Schemes

The Fourier and Walsh-Hadamard transforms, while determined to be unsuitable for filtering purposes in this application, did provide useful information about the sampling process and the image itself. It was determined experimentally that the largest contribution to the frequency spectrum was from the fundamental component. For the case of the 8 micron sampling interval, the amplitudes of the frequency components were down to one-tenth of the fundamental by the sixth harmonic term. Although precautions were taken to satisfy the sampling theorem by using essentially an optical low pass filter, results showed that the images contained only low frequencies since the upper half of the spectrum (harmonics 9 through 16) had negligible power.* This result led to the conclusion that the sampling interval was sufficiently small and the main criterion was then to obtain a sufficient number of samples within a reseau to accurately locate its center.

*The optical low pass filter was accomplished by choosing the sampling window to be slightly larger than the sampling interval. For the 8 micron sampling interval, the sampling window was a circle of a diameter of 8.5 microns.

One reason for not further considering the Fourier and Walsh-Hadamard transforms for filtering applications is the excessive execution time and memory storage for relatively large two-dimensional images. For example, even in the case of the Hadamard transform, a 91×91 element image requires approximately 16K words of memory storage in a 16-bit minicomputer. Also fast algorithms for both transforms execute in minimum time if the measurement space is a positive integer power of two. The restriction could be unnecessarily binding.

Another equally undesirable effect of the filtering process with the Fourier and Hadamard transforms was shared with the regularization scheme. This effect is the smoothing of the reseau image edges and general deformations of the reseau image. Since both processes are linear transformations, the effect of removing film grain noise by eliminating the high frequency information creates the same effect on the reseau image. The modification to the regularization scheme to perform nonlinear filtering greatly improved this situation. Although the result of the nonlinear transformation was a great improvement over the other methods, the associated computation time was great. Since the center detection process still had to be performed and considering the number of reseaux that must be processed in one image, this scheme could not be used either.

The most effective method of preprocessing for this application was found to be the Bayes threshold scheme. Although the gathering of statistics on each picture is time consuming, this process can be performed in parallel with data acquisition. Only a few operations remain once the data is acquired. The second advantage is that density values above the threshold

are unaltered causing no degradation to the major portion of the reseau image.

4.2 Center Location Algorithms

4.2.0 Template Matching

Template matching is a classical picture processing technique used to determine whether or not a previously specified object is contained within a given picture. This involves the systematic scanning across the entire picture of a template or mask used to describe the object while searching for a match. Since exact matches are not likely, a performance index is necessary. One such index is the Euclidean distance between two vectors, i.e.,

$$E(m,n) = \left\{ \sum_i \sum_j [g(i,j) - t(i-m,j-n)]^2 \right\}^{1/2} \quad (4-29)$$

where $g(i,j)$ is the digital picture function being scanned and $t(i,j)$ is the template picture function [10]. The indices (i,j) can only be assigned values such that $(i-m,j-n)$ is in the domain of the template sub-picture. This operation is equivalent to translating the template to position (m,n) and taking the difference of respective picture elements between the template and the picture. Once the correct position of the template is found the performance index $E(m,n)$ becomes small.

If the variation of picture energy, calculated by

$$E_p = \sum_i \sum_j [g(i,j)]^2 \quad (4-30)$$

is small across the template windows, then the square of the Euclidean distance can be reduced to the cross-correlation of the two functions, g and t , and defined by

$$R_{gt}(m,n) = \sum_i \sum_j g(i,j)t(i-m,j-n) \quad (4-31)$$

where (i,j) are bound by the same constraints as for the Euclidean distance metric [10]. The search is now, however, for the maximum value of the correlation function, R_{gt} .

Template matching can be applied to the center detection problem by defining the center of the reseau to be at the center of the template. Once the correct translation values for the template for minimum error are found, the center of the reseau is known.

Although template matching can be used with great success in many applications, several different problems arose in this application. One problem that is common to all applications is how well the template can be made to represent the test data. This was a particularly difficult problem for a reseau image since the image not only varied in amplitude as the background varied, but also varied in shape due to photographic effects and other conditions.

The template chosen was an ideal reseau with a width of seven samples and a density of 1.2. The inexactness of the reseau template led to such phenomena as multiple correct positions and position error due to reseau distortions. When small errors were present, they were also amplified by the fact that center locations could only be found at a minimum resolution step of a sample interval, or 8.5 microns.

For computational purposes, the template window was actually the same size as the picture. The reseau image (+) position was then varied within the template window. This approach was necessary to maintain maximum accuracy since the reseau always extended to the edge of the test picture. The large number of calculations led to the undesirable result of slow processing. Since the picture energy was always the same, cross-correlation was used to increase the speed but the processing time was still excessive. Template matching alone was determined to be insufficient. Comparative results to an alternate method are presented in Table 4.1.

4.2.1 Center of Gravity Technique

Another classical means of defining the center of a picture function is by its centroid [23]. In this method the gray level at each point (i,j) of the picture function is defined as the "mass" at (i,j) . Using this interpretation a centroid or center of gravity can be calculated using the moments of the function. The centroid of a digital picture function $g(i,j)$ is the point (i_c, j_c) where the values of i_c and j_c are given by

$$i_c = \sum_i \sum_j i g(i,j) / \sum_i \sum_j g(i,j) \quad (4-32)$$

and

$$j_c = \sum_i \sum_j j g(i,j) / \sum_i \sum_j g(i,j) \quad (4-33)$$

The strict application of this technique for center location to the pictures of interest is not possible. For pictures consisting of homogeneous

TABLE 4.1 COMPARISON OF CENTER DETECTION SCHEMES

Sample	Direc- tion	Actual Center	Template Matching			Center of Gravity		
			Indicated Position	Error (Microns)	Speed*	Indicated Position	Error (Microns)	Speed
Homogeneous Background	X	46.00	47.00	8.0	12 min	46.00	0	20 sec
	Y	46.00	47.00	8.0	12 min	46.00	0	20 sec
Homogeneous Background	X	45.50	46.00	4.0	12 min	45.50	0	20 sec
	Y	45.50	46.00	4.0	12 min	45.50	0	20 sec
Ross Effect	X	46.00	46.00	0	12 min	46.01	0.08	20 sec
	Y	46.00	47.00	8.0	12 min	46.53	4.24	20 sec
Removed Portions	X	46.00	46.00	0	12 min	46.00	0	20 sec
	Y	46.00	46.00	0	12 min	46.01	0.08	20 sec
Low Contrast	X	46.00	46.00	0	12 min	46.00	0	20 sec
	Y	46.00	46.00	0	12 min	46.00	0	20 sec

* The template was only shifted from the center position of
35 through 55 in each direction

background areas, the result of strict application is reasonable. If, however, the background areas contain other images such as roads and buildings, the operation produces a result that is highly inaccurate since the gray levels within this type of background are of significant "mass" values. Therefore, in order to use this scheme, the region of interest must be restricted to the approximate area of the reseau image.

Several methods for determining the approximate reseau image area were considered. One possible scheme was to employ the technique of template matching on a reseau image as described in the previous section. The deficiencies found to be present in this scheme are insignificant in this application since only an approximate reseau location is necessary. The execution time of this method is excessive, however, especially since further calculations were still necessary. Thus, implementation was not pursued.

A second scheme was devised based on the fact that the reseau images were rotationally fixed. This method again used template matching, but employed a smaller template. The result was an increase in the execution speed.

In this scheme, the template matching operation was used to determine the edges of the reseau. The original digital picture function was first transformed into picture functions of the partial derivatives with respect to the horizontal and vertical directions. The vertical edges were determined by cross-correlating the partial derivative picture with respect to the horizontal with a template of a vertical edge. This template was one sample wide, the length of the picture function, and of unit amplitude. The left edge was located by the largest positive correlation value and

the right edge by the largest negative value. A similar process was performed to determine the edges of the horizontal portion of the reseau image. In order to insure that important portions of the reseau image were not eliminated, the resulting area of interest was increased by one sample point in all directions. A better estimate of the partial derivatives was also obtained by first thresholding the original picture using the Bayes threshold technique.

Having determined the approximate reseau image, the center of gravity of the samples within this region is calculated. Results of this process on various reseau samples are given in Table 4.1.

4.3 Results and Conclusions of the Center Location Process

Comparative results of the two center location schemes are given in Table 4.1 for several phenomena common to reseau images. It should be noted that each entry in the table is the result of a single trial and is not a statistical average of many images possessing a particular property. Although, in general, definitive conclusions can not be drawn from such a small sample space, there are several reasons for terminating the feasibility analysis at this point with the selection of the center of gravity approach.

One important reason for preferring the center of gravity scheme is the reduced execution time required by this method over the template matching approach. Because a typical $8\frac{1}{2} \times 8\frac{1}{2}$ photographic negative contains 529 reseaux, processing time is a key part of the performance measure. Since each image processed requires the same processing time regardless of its properties, additional trials were unnecessary in order to make a

judgment with respect to this part of the performance measure.

The other important portion of the performance index involves measurement accuracy. The best way to obtain a meaningful measure of the average error of each technique is to obtain a large population of actual reseau samples consisting of various background conditions whose centers have been determined manually by analysts and can serve as ground truth. Unfortunately, such data is not available and only artificial samples could be produced.

The generation of artificial data is a very time consuming process. Also since adequate ground truth was not available upon which to establish the validity of the inferences drawn from such data, it was necessary to leave such further analysis to future research. The center of gravity technique was, therefore, selected for incorporation into the proposed system for the following reasons: (1) reduced processing time, (2) indicated trend towards improved accuracy, and (3) the fact that there is always error in the template matching scheme except for rare cases in which the actual center location happens to have integer coordinates.

CHAPTER V

RESULTS AND CONCLUSIONS

5.0 Proposed Correction System

The hardware of the proposed correction system is as described in Chapter II and by Figure 1.1. The optical scanner to be used is the PDS Model 1010A microdensitometer. The software system to be described was developed on a Hewlett Packard 2100A minicomputer system available at the Pattern Analysis and Computer System Laboratory at the University of Virginia. The minicomputer system contained both paper tape and magnetic tape devices along with a magnetic disc system. The actual correction system software required approximately 16K words of memory for execution. Additional memory would be necessary if the correction system was to include an operating system or possibly programs for further analysis of the reseaux center location data.

As stated in Chapter II, the heart of the correction system is the software package. After analysis of various schemes of picture processing, a strategy was developed that took advantage of a particular characteristic of the optical scanner used. This characteristic is the relative slow data acquisition rate.

The overall software scheme for reseaux center determination is described by the flow chart of Figure 5.1. Two major operations are performed. The first operation is the determination of the optimum Bayes threshold value. Using this value, a new image is produced with all

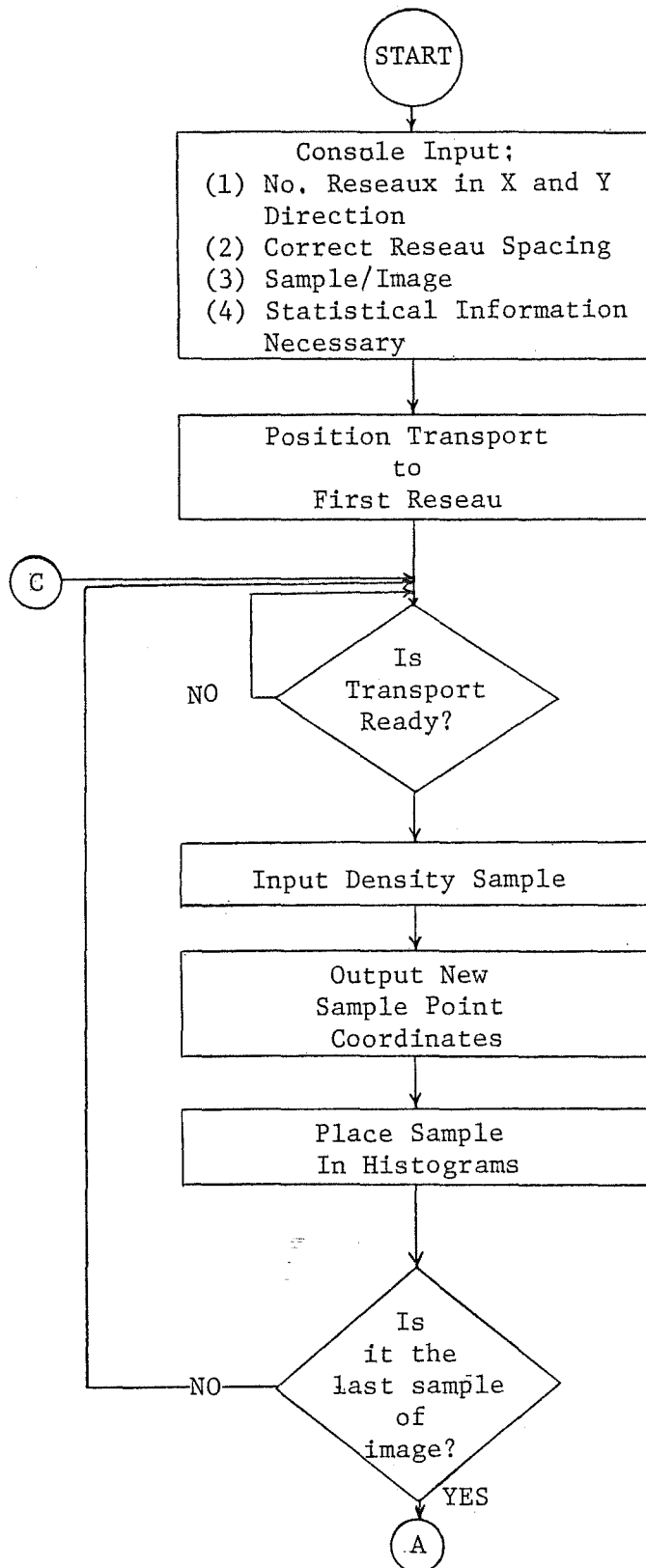


Figure 5.1 Reseaux Correction System Flow Chart

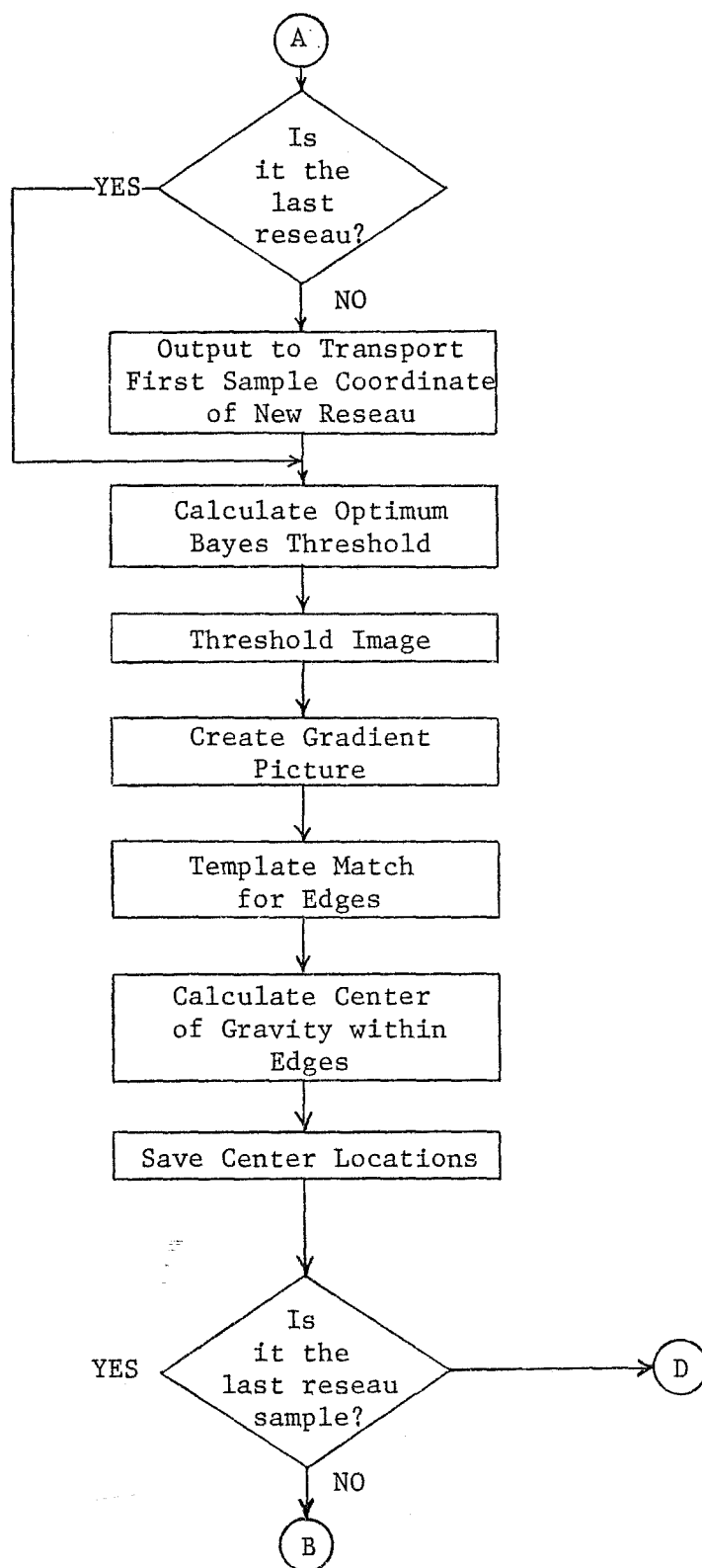


Figure 5.1 Reseaux Correction System Flow Chart
(cont.)

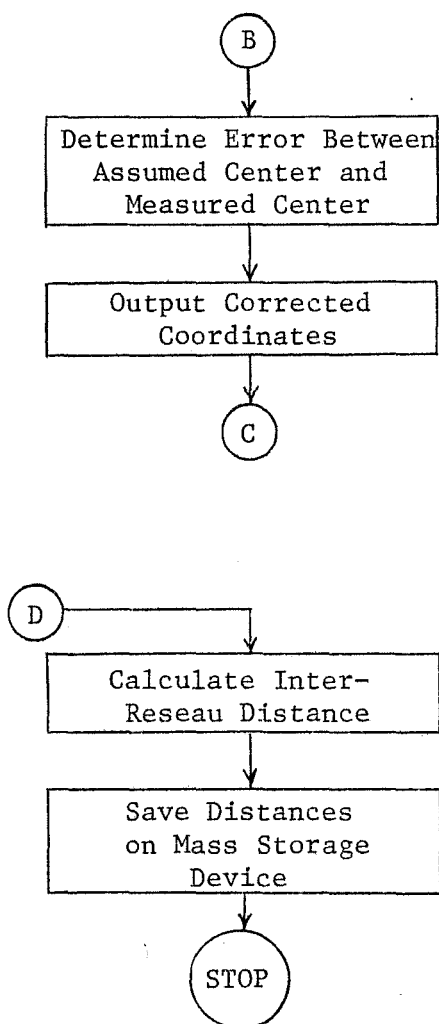


Figure 5.1 Reseaux Correction System Flow Chart
(cont.)

density values below this value set equal the threshold value. One reason for choosing the Bayes preprocessing scheme was that histograms necessary for the scheme could be created in parallel with the data acquisition phase. In order to obtain accurate density measurements from the microdensitometer, the transport of the microdensitometer was run at approximately 1500 microns per second [3,4]. At a sampling interval of eight microns, 5.33 milliseconds was available between samples for the placing of the previous sample into the correct histogram bin. The second operation was to determine the center location from the transformed image. The center of gravity technique was used for this process.

In operating the system, the user will first input through the system console various parameters particular to the photograph being scanned. Such parameters are the number of reseaux in both X and Y directions, correct reseau spacing number of samples per reseau, and any statistical information necessary. The operator will then position the transport over the first sample and the system will automatically sample all reseau in the photograph.

An example of the total processing time required for an entire negative can be estimated from the $8\frac{1}{2} \times 8\frac{1}{2}$ negative from which actual reseau images were taken. The negative contained an array of reseaux arranged in a 23×23 image configuration. The interreseau distance was approximately one millimeter or 1000 microns. At a transport rate of 1500 microns per second, travel alone among the 529 reseaux requires 353 seconds. For a sampling interval of 8 microns and a sampling grid of 91×91 , the data acquisition phase at each reseau requires 45 seconds.

The data acquisition phase for the entire negative is, therefore, 23,805 seconds. Using a center location processing time of 20 seconds per reseau, the total time required to process this negative would be approximately 9.5 hours.

It should be noted that the large portion of the total time required is associated with the movement of the transport,

5.1 Conclusions and Recommendations

A system for the accurate location of the centers of reseaux on aerial photography has been described. The hardware proposed was based largely on a currently available system with the exception of additional computing power required. The software package was developed using state-of-the-art digital picture processing and pattern recognition techniques.

This work consisted of two main parts. The first part was the development of a model which accurately represented the reseau images and of which the exact center location was known. The second part was the development of a center location scheme using these models as input. Both areas suffered from the same problem; that of easy access to the microdensitometer system.

The next phase of the investigation should be the construction of a complete system. The possibility of obtaining or fabricating some sort of calibration grid containing known reseau data in order to test the accuracy of the overall system should be investigated. Investigation into the use of other optical scanners should be conducted in order to decrease the processing time. The effect of this change on the center location scheme should then be studied.

In the area of reseau modelling, additional evidence of various photographic effects should be obtained to create more accurate models of these phenomena. Other effects due to varying background conditions may also appear, creating the need for new or modified models. The center location strategy must then be subjected to the new models and modified as necessary.

BIBLIOGRAPHY

1. A. Rosenfeld and J. L. Pfaltz, "Distance Functions on Digital Pictures," Pattern Recognition, Vol. 1, July, 1968, pp. 33-61.
2. Robert Nathan, "Picture Enhancement for the Moon, Mars, and Man," Pictorial Pattern Recognition, Thompson Book Co., Washington, D.C., 1968, pp. 239-266.
3. Microdensitometer System--Installation, Operation, and Maintenance, Photometric Data Systems Corporation, Webster, New York.
4. Conversation with Samuel Barr, U.S. Army Engineer Topographic Laboratories, Alexandria, Virginia.
5. F. H. James and G. C. Higgins, Fundamentals of Photographic Theory, Morgan and Morgan, Inc., New York, New York, 1960, pp. 266-303.
6. F. M. Brown, H. J. Hall, and J. Kosar, Photographic Systems for Engineers, Society of Photographic Scientist and Engineers, Washington, D.C., 1966, pp. 39-64.
7. C. B. Neblette, Photography Principles and Practice, D. Van Nostrand Co., New York, New York, 1944, pp. 373-413.
8. C. B. Neblette, Photography, Its Materials and Processes, D. Van Nostrand Co., New York, New York, 1952, pp. 309-317.
9. C. E. Kenneth Mees, The Theory of the Photographic Process, The Macmillan Co., New York, New York, 1954, pp. 970-1047.
10. Richard O. Duda and Peter E. Hart, Pattern Classification and Scene Analysis, John Wiley and Sons, New York, New York, 1973.
11. T. G. Newman and H. Diriltlen, "A Nonlinear Transformation for Digital Picture Processing," IEEE Transactions on Computers, September, 1973, pp. 869-873.
12. Murray R. Spiegel, Theory of Problems of Statistics, Schaum's Outline Series, McGraw-Hill Book Company, New York, New York, 1961, pp. 201-216, 345.
13. Frederick Mosteller and Robert Rourke, Study Statistics, Addison-Wesley Publishing Company, Reading, Massachusetts, 1973, pp. 141-209, 317.
14. Karl V. Burry, Statistical Models in Applied Science, John Wiley and Sons, New York, New York, 1975, pp. 194-203.

15. R. W. Hamming, Numerical Methods for Scientists and Engineers, McGraw-Hill Book Company, New York, New York, 1973, pp. 136-143.
16. William T. Cochran, et. al., "What is the Fast Fourier Transform," IEEE Transactions on Audio And Electroacoustics, Vol. AU-15, No. 2, June, 1967, pp. 45-55.
17. N. M. Brenner, "Three Fortran Programs that Perform the Cooley-Tukey Fourier Transform," Lincoln Laboratory, Massachusetts Institute of Technology, Lexington, Massachusetts.
18. James W. Cooley, Peter A. Lewis, and Peter Welch, "Application of the Fast Fourier Transform to Computation of Fourier Integrals, Fourier Series, and Convolution Integrals," IEEE Transactions on Audio and Electroacoustics, Vol. AU-15, No. 2, June, 1967, pp. 79-85.
19. R. J. Polge, et. al., "Theory and Implementation of Fast Fourier and Hadamard Transforms," NTIS, AD762437, pp. 70-72.
20. Nasir Ahmed, K. R. Rao, and A. L. Abdussattar, "BIFORE or Hadamard Transform," IEEE Transactions on Audio and Electroacoustics, Vol. AU-19, No. 3, September, 1971, pp. 225-234.
21. William K. Pratt, Julius Kane, and Harry C. Andrews, "Hadamard Transform Image Coding," Proceedings of the IEEE, Vol. 57, No. 1, January, 1969, pp. 58-68.
22. John L. Shanks, "Computation of the Fast Walsh-Fourier Transform," IEEE Transactions on Computers, Short Note, May, 1969, pp. 457-459.
23. Azriel Rosenfeld and Avinash C. Kak, Digital Picture Processing, Academic Press, New York, New York, 1976, pp. 269-273.
24. "Processor Architectures Utilizing Magnetic Bubble and Semi-Conductor Memories," Final Report NASA Contract No. NSG-1067, NASA, Langley Research Center, Hampton, Virginia, July, 1975, pp. 64-68.
25. J. S. Weszka, R. N. Nagel, and A. Rosenfeld, "A Threshold Selection Technique," IEEE Transactions on Computers, December, 1974, pp. 1322-1326.
26. W. Doyle, "Operations Useful for Similarity-Invariant Pattern Recognition," Journal of Association of Computing Machines, Vol. 9, 1962, pp. 259-267.
27. F. Sakai, M. Nagao, and S. Fujibayashi, "Line Extraction and Pattern Detection in a Photograph," Pattern Recognition, Vol. 1, 1969, pp. 233-248.

APPENDIX

CORRECTION SYSTEM DEVELOPMENT SOFTWARE LISTINGS

The following are listings of various software modules used in the development of the reseau correction system. Each listing contains documentation as to the major function of that module,


```

DO 120 J=43,49
120 IRES(I,J)=480
DO 130 I=43,49
DO 130 J=1,91
130 IRES(I,J)=480
C
IF(ISSW(1))147,127
127 DO 128 I=1,8
128 ISUM(I)=0
DO 129 I=1,5
DO 129 J=1,5
ISUM(1)=ISUM(1)+IW1(I,J)
ISUM(2)=ISUM(2)+IW2(I,J)
ISUM(3)=ISUM(3)+IW3(I,J)
ISUM(4)=ISUM(4)+IW4(I,J)
ISUM(5)=ISUM(5)+IW5(I,J)
ISUM(6)=ISUM(6)+IW6(I,J)
ISUM(7)=ISUM(7)+IW7(I,J)
129 ISUM(8)=ISUM(8)+IW8(I,J)
C
C
C NONLINEAR TRANSFORMATION OF IDEAL RESEAU (89X89)
C
DO 138 I=3,89
DO 138 J=3,89
IA=0
I1=I
J1=J
DY1=0.
DX1=0.
I3=I1-3
J3=J1-3
DO 133 J2=1,2
J12=J1+J2
J32=J3+J2
IJ12=I1+J2
IJ32=I3+J2
DO 132 I2=1,5
DX1=DX1+IRES(I3+I2,J12)-IRES(I3+I2,J32)
132 DY1=DY1+IRES(IJ12,J3+I2)-IRES(IJ32,J3+I2)
DX1=DX1+IRES(I1,J12)-IRES(I1,J32)
133 DY1=DY1+IRES(IJ12,J1)-IRES(IJ32,J1)
IS=0
IF((DX1.EQ.0.).AND.(DY1.EQ.0.)) IS=1
IF((DX1.EQ.0.).AND.(DY1.GT.0.)) IS=3
IF((DX1.EQ.0.).AND.(DY1.LT.0.)) IS=7
IF((DY1.EQ.0.).AND.(DX1.GT.0.)) IS=1
IF((DY1.EQ.0.).AND.(DX1.LT.0.)) IS=5
IF(IS.NE.0) GO TO 137
134 THETA=ATAN(DY1/DX1)
THETA=THETA*(180./3.14159)
IF(ABS(THETA).GE.22.5) GO TO 135

```

```

IS=1
IF(DX1.LT.0.) IS=5
GO TO 137
135 IF(ABS(THETA).GE.67.5) GO TO 136
IF((DX1.GT.0.).AND.(DY1.GT.0.)) IS=2
IF((DX1.GT.0.).AND.(DY1.LT.0.)) IS=8
IF((DX1.LT.0.).AND.(DY1.GT.0.)) IS=4
IF((DX1.LT.0.).AND.(DY1.LT.0.)) IS=6
GO TO 137
136 IS=3
IF(DY1.LT.0.) IS=7
137 DO 131 K=1,5
DO 131 L=1,5
K1=K-3+I1
L1=L-3+J1
IF(IS.EQ.1) IA=IA+IW1(K,L)*IRES(K1,L1)
IF(IS.EQ.2) IA=IA+IW2(K,L)*IRES(K1,L1)
IF(IS.EQ.3) IA=IA+IW3(K,L)*IRES(K1,L1)
IF(IS.EQ.4) IA=IA+IW4(K,L)*IRES(K1,L1)
IF(IS.EQ.5) IA=IA+IW5(K,L)*IRES(K1,L1)
IF(IS.EQ.6) IA=IA+IW6(K,L)*IRES(K1,L1)
IF(IS.EQ.7) IA=IA+IW7(K,L)*IRES(K1,L1)
IF(IS.EQ.8) IA=IA+IW8(K,L)*IRES(K1,L1)
131 CONTINUE
138 IRES(I-2,J-2)=IA/ISUM(IS)
C
C
C SHIFT OPERATION
DO 140 I=1,87
I1=90-I
I2=88-I
DO 140 J=1,87
140 IRES(I1,J+2)=IRES(I2,J)
C FILL
DO 142 I=1,2
I1=89+I
DO 142 J=3,89
IRES(I,J)=IRES(3,J)
142 IRES(I1,J)=IRES(89,J)
DO 144 J=1,2
J1=89+J
DO 144 I=1,91
IRES(I,J)=IRES(I,3)
144 IRES(I,J1)=IRES(I,89)
C
C
C
147 IF(ISSW(0))210,149
149 DO 150 I=1,50
150 A=RAND(0)
DO 200 I=1,91
DO 200 J=1,91

```

```

CALL GAUSS(DEV,AMEAN,V)
IA=INT(V*400.)
200 IRES(I,J)=IRES(I,J)+IA
C
210 WRITE(8)(NAME(I),I=1,20)
DO 300 I=1,91
300 WRITE(8)(IRES(I,J)-J=1,91)
ENDFILE 8
REWIND 8
DO 350 I=1,1000
350 WRITE(1,400)
400 FORMAT(" ")
END
C
C
C
SUBROUTINE GAUSS(SD,AM,V)
A=0.0
DO 50 I=1,12
50 A=A+RAND(0)
V=(A-6.0)*SD+AM
RETURN
END
END$

```

FTN

SOURCE - BKGNS

JAMES HIRAM AYLOR

THIS PROGRAM CALCULATES THE MEAN AND STANDARD DEVIATION
OF THE BACKGROUND AND THE MAXIMUM VALUE OF THE RESEAU
NOT INCLUDING THE CENTER REGION.

IT ALSO PERFORMS THE CHI-SQUARED TEST

SET BIT 0 FOR ACTUAL, BIT 1 FOR DERIVED GAUSSIAN
SET BIT 2 FOR ACTUAL MEAN, ELSE MEAN - 1.00
SET BIT 3 FOR LABELS

PROGRAM BKGNS

DIMENSION IA(91,91),NAME(20)

DIMENSION IM(600),G(600),LABEL(40)

COMMON IC(20)

WRITE(1,10)

10 FORMAT(// "INPUT FILE NO. :")

READ(1,*) M

WRITE(1,15)

15 FORMAT("INPUT WINDOW SIZE:")

READ(1,*) NS

WRITE(1,17)

17 FORMAT("INPUT HISTOGRAM DELTA:")

READ(1,*) DELTA

C NUMBER OF SAMPLES DIMENSION

II=91-NS

NTOI=NS*NS*4

C IP=0

X=0.

Y=0.

K=0

DO 12 I=1,600

IM(I)=0

12 G(I)=0.

IMIN=1200

IMAX=0

WRITE(6,20)

20 FORMAT(1H^)

N1=M-1

C CALL SPFIL(N1)

C GO TO 40

C 30 REWIND 8

```

WRITE(1,35)
35  FORMAT("INPUT ADDITIONAL FILE OR 0")
    READ(1,*)M
    IF(M.EQ.0) GO TO 283
    N1=M-1
    CALL SPFIL(N1)
40  READ(8)(NAME(I),I=1,20)
    WRITE(6,45)(NAME(I),I=1,20)
45  FORMAT(10X,20A2)
    DO 50 I=1,91
50  READ(8)(IA(I,J),J=1,91)
    IP=IP+1
    IF(ISSN(2))62,55.
55  Z=0.0
    DO 60 I=1,NS
    DO 60 J=1,NS
    IZ=IA(I,J)+IA(I+II,J)+IA(I,J+II)+IA(I+II,J+II)
    Z=Z+IZ/400.
60  CONTINUE
    Z=Z/(NS*NS*4)
C
    IZ=Z*400.
    DO 61 I=1,91
    DO 61 J=1,91
61  IA(I,J)=IA(I,J)-IZ+400
C
62  I1=0
    J1=0
    DO 200 L=1,4
    DO 100 I=1,NS
    DO 100 J=1,NS
    K=K+1
    A1=1./(K+1.)
    A2=(K-1.)/K
    DATA=(IA(I+I1,J+J1)/400.)-X
    Y=A2*Y+A1*DATA*DATA
    X=X+A1*DATA
100  CONTINUE
    I1=I1+1
    J1=J1+1
    IF(L.EQ.3) GO TO 200
    IF(L.EQ.1) J1=0
    IF(L.EQ.2) I1=0
200  CONTINUE
C
C
C  CHI-SQUARED TEST
C
C
    DO 270 I=1,NS
    DO 270 J=1,NS
    IF(IA(I,J).GT.IMAX) IMAX=IA(I,J)

```

```

IF(IA(I+II,J).GT.IMAX) IMAX=IA(I+II,J)
IF(IA(I,J+II).GT.IMAX) IMAX=IA(I,J+II)
IF(IA(I+II,J+II).GT.IMAX) IMAX=IA(I+II,J+II)
IF(IA(I,J).LT.IMIN) IMIN=IA(I,J)
IF(IA(I+II,J).LT.IMIN) IMIN=IA(I+II,J)
IF(IA(I,J+II).LT.IMIN) IMIN=IA(I,J+II)
IF(IA(I+II,J+II).LT.IMIN) IMIN=IA(I+II,J+II)
270 CONTINUE
IDEL=DELTA*400.
M1=(IMIN/IDEL)+1
M2=((IMAX/IDEL)+1)+1
DO 280 I=1,NS
DO 280 J=1,NS
DO 280 K=M1,M2
IB=IDEL*(K-1)
IF((IA(I,J).GE.IB).AND.
1(IA(I,J).LT.(IB+IDEL))) IM(K)=IM(K)+1
IF((IA(I,J+II).GE.IB).AND.
1(IA(I,J+II).LT.(IB+IDEL))) IM(K)=IM(K)+1
IF((IA(I+II,J).GE.IB).AND.
1(IA(I+II,J).LT.(IB+IDEL))) IM(K)=IM(K)+1
IF((IA(I+II,J+II).GE.IB).AND.
1(IA(I+II,J+II).LT.(IB+IDEL))) IM(K)=IM(K)+1
280 CONTINUE
GO TO 30
C
C CREATE ACTUAL GAUSSIAN DIST.
C
283 Y=SQRT(Y)
DO 285 I=1,600
A=(I*DELTA)-DELTA/2.
A=EXP(-((A-X)*(A-X))/(2*Y*Y))
285 G(I)=(A*DELTA*NTOT*IP)/(Y*SQRT(6.28318))
C
J=1
A=0.
DO 288 I=1,600
IF(G(I).LE.A) GO TO 288
A=G(I)
J=I
288 CONTINUE
M1=J-((2*Y)/DELTA)
M2=J+((2*Y)/DELTA)+1
C
Z1=0.
Z2=0.
Z=0.
DO 290 I=M1,M2
A=IM(I)
B=G(I)
IF(B.LT.5.0) WRITE(6,282)
282 FORMAT("EXPECTED FREQ. < 5")

```



```

IF(I.GE.1) GO TO 289
A=0.
B=(I*DELTA)-DELTA/2.
B=EXP(-((A-X)*(A-X))/(2*Y*Y))
B=(B*DELTA*NTOT)/(Y*SQRT(6.28318))
289 Z1=Z1+A
Z2=Z2+B
Z=Z+((A-B)*(A-B))/B
290 CONTINUE
WRITE(6,400)X,Y
400 FORMAT(10X,"MEAN - ",F5.3,/,10X,"STD. DEV. - ",F5.3)
M=M2-M1+1
WRITE(6,405)M
405 FORMAT(10X,"NO. POINTS USED - ",I3)
WRITE(6,410)Z
410 FORMAT(10X,"CHI-SQUARED - ",F7.3)
MAX=0
AMAX=0.
DO 415 I=1,600
IF(G(I).GT.AMAX) AMAX=G(I)
IF(IM(I).GT.MAX) MAX=IM(I)
415 CONTINUE
WRITE(6,417)MAX,AMAX
417 FORMAT(10X,"ACTUAL MAX. - ",I3,2X,"GAUSSIAN MAX. - ",F3.1)
Z1=Z1*100./(NTOT*IP)
Z2=Z2*100./(NTOT*IP)
WRITE(6,420)Z1,Z2
420 FORMAT(10X,"% ACTUAL, % GAUSSIAN - ",F5.3," ",F5.3)
WRITE(6,422)NS
422 FORMAT(10X,"SAMPLE WINDOW SIZE - ",I2)
WRITE(6,424)DELTA
424 FORMAT(10X,"HISTOGRAM INCREMENT - ",F6.3///)
REWIND 8
WRITE(6,20)

C
IF(ISSW(0)) 460,462
462 IF(ISSW(1)) 460,500
460 WRITE(6,464)
464 FORMAT("INPUT MAX. HISTOGRAM VALUE:")
READ(6,*)BMAX
CALL WINDO(0,1)
CALL CHRS(27,12,128)
B1=399*DELTA
CALL RANGE(0.,B1,0.,BMAX,0,0)
CALL AXES(0.,0.)
B2=400*DELTA
CALL TICK(0,0.,B2,1.,30,1)
SPY=BMAX/10.
CALL TICK(1,0.,BMAX,SPY,30,1)
IF(ISSW(1))465,470
465 CALL GRAPH(0,400,G)
470 IF(ISSW(0))472,499

```

```
472 DO 475 I=1,600
475 G(I)=IM(I)
    WRITE(6,477)
477 FORMAT("TYPE 1 TO CONTINUE!")
    READ(6,*)IHD
    CALL GRAPH(0,400,G)
499 WRITE(6,510)
510 FORMAT("POSITION THE CURSOR, STRIKE ANY KEY,"
&" AND TYPE A LABEL")
    CALL CURSI(IHD,IXPOS,IYPOS)
    CALL TPLOT(0,IXPOS,IYPOS)
    CALL CHRS(31,128,128)
    READ(6,520)(LABEL(J),J=1,40)
520 FORMAT(40A1)
    WRITE(1,520)(LABEL(J),J=1,40)
    WRITE(6,530)
530 FORMAT("TYPE 1 FOR MORE LABELS, ELSE 0:")
    READ(6,*)IHD
    IF(IHD.EQ.1) GO TO 499
    WRITE(6,477)
    READ(6,*)IHD
500 CONTINUE
    END
    END$
```

```

FTN
C
C
C      SOURCE-F2DS      JIM AYLOR
C
C      THIS PROGRAM PERFORMS FILTERING
C      VIA FFT
C
C      L=0; NO FILTERING AND NO FILT. OUTPUT
C      BIT 0 ON FOR REAL AND IMAG. OUTPUT
C      BIT 1 ON FOR MAGNITUDE PRINT
C      BIT 2 ON FOR PLOTTER
C      BIT 3 ON FOR NO INPUT PLOT
C      BIT 12 ON FOR NO MAPS
C      BIT 13 ON FOR NO INPUT INPUT MAP
C      BIT 15 ON FOR HIGH PASS,OFF FOR LOW PASS
C
C
C      PROGRAM F2D1
C      DIMENSION NN(2),IX(16),A(16)
C      INTEGER IB(91),NAME(20)
C
C      CHANGE FOR NEW ARRAY SIZE
C      DIMENSION DATA(64,32),IDATA(32)
C
C      DATA IX/1H ,1H1,1H2,1H3,1H4,1H5,1H6,1H7,1H8,
&1H9,1HA,1HB,1HC,1HD,1HE,1HF/
C
C      INPUT =8
C
C      CHANGE FOR NEW ARRAY SIZE
C      N=32
3000  FORMAT(20X,"++++++++++++++++++++")
3500  FORMAT(20X,"-",32A1,"+")
      N0=2
C
C      SKIP FILES
C      WRITE(1,17)
17    FORMAT("SKIP ANY FILES?")
      READ(1,*)NFILE
      CALL SPFIL(NFILE)
C
      II=0
      WRITE(1,23)
23    FORMAT("INPUT CUTOFF TERM")
      READ(1,*)L
      N1=L-1
      N2=L+1
      N3=N/2
      IF(ISSW(15))28,29
28    WRITE(6,280)N1
280   FORMAT("TERMS 1 THROUGH ",I2," CUTOFF")

```

```

GO TO 333
29  WRITE(6,290)N2,N3
290  FORMAT("TERMS ",I2," THROUGH ",I2," CUTOFF")
333  NN(1)=N
      NN(2)=N
C
C    INPUT DATA
      READ(INPUT)(NAME(I),I=1,20)
      DO 1 I=1,28
1      READ(INPUT)(IB(J),J=1,91)
      DO 3 I=29,60
      READ(INPUT)(IB(J),J=1,91)
      DO 4 J=29,60
4      DATA(2*(I-28)-1,J-28)=IB(J)/400.
3      CONTINUE
      DO 6 I=61,91
6      READ(INPUT)(IB(J),J=1,91)
      REWIND INPUT
      DO 2 I=2,2*N,2
      DO 2 J=1,N
2      DATA(I,J)=0.
C
C    FIND MAX.
      AMIN=DATA(1,1)
      AMAX=DATA(1,1)
      DO 501 I=1,2*N-1,2
      DO 501 J=1,N
      IF(DATA(I,J)-AMAX)502,502,653
653  AMAX=DATA(I,J)
502  IF(DATA(I,J)-AMIN)503,501,501
503  AMIN=DATA(I,J)
501  CONTINUE
      DELTA=(AMAX-AMIN)/16.
C
C    BIT 13 ON FOR NO INPUT DATA PRINT
C
      IF(ISSW(13))155,211
211  WRITE(6,902)(NAME(I),I=1,20)
902  FORMAT(///"INPUT DATA - ",20A2)
      GO TO 150
C
C    FIND 2D FFT, DIV. BY N*N, PRINT MAG. BEFORE AND AFTER
C
200  CALL FOUR2(DATA,NN,2,-1)
      II=II+1
      DO 5 I=1,2*N
      DO 5 J=1,N
5      DATA(I,J)=DATA(I,J)/(N*N)
C
      IF(ISSW(0))1000,1001
1000 WRITE(6,910)
910  FORMAT(///"REAL PART OF TRANSFORM")

```

```

DO 7 I=1,N-1,2
DO 71 K=1,N,8
71 WRITE(6,101)(DATA(I,J),J=K,K+7)
7 WRITE(6,102)
101 FORMAT(8(F7.5,2X))
102 FORMAT(/)
WRITE(6,104)
104 FORMAT("//IMAG. PART OF TRANSFORM")
DO 105 I=2,2*N,2
DO 106 K=1,N,8
106 WRITE(6,101)(DATA(I,J),J=K,K+7)
105 WRITE(6,102)
C
1001 IF(ISSW(1))1200,1201
1200 WRITE(6,900)
900 FORMAT(///"MAGNITUDE AFTER FILTERING")
DO 30 I=1,2*N-1,2
DO 31 J=1,N,8
DO 32 K=1,8
A(K)=(DATA(I,J+K-1))*2+(DATA(I+1,J+K-1))*2
32 A(K)=SORT(A(K))
31 WRITE(6,101)(A(K),K=1,8)
30 WRITE(6,102)
WRITE(6,40)
C
C SKIP FILTER?
1201 IF(L)201,201,1199
C
C FILTERING
1199 IF(ISSW(15))1198,1197
1198 WRITE(6,1190)
1190 FORMAT("//HIGH PASS FILTERED")
IF(N1.EQ.0) GO TO 1195
DATA(1,1)=0.
DATA(2,1)=0.
IF(N1.EQ.1) GO TO 1195
DO 34 I=3,2*N1
34 DATA(I,1)=0.
DO 9 J=2,N1
DATA(1,J)=0.
9 DATA(2,J)=0.
DO 10 I=1,2*(N1-1)
DO 10 J=1,N1-1
DATA(I+2,J+1)=0.
DATA(2*N+1-I,J)=0.
DATA(I,N+1-J)=0.
10 DATA(2*N+1-I,N+1-J)=0.
GO TO 1195
C
1197 WRITE(6,1191)
1191 FORMAT("//LOW PASS FILTERED")
DO 11 I=2*N2-1,2*N3+2

```

```

DO 11 J=1,N
DATA(I,J)=0.
11 DATA(2*N+3-I,J)=0.
DO 1111 I=1,2*N
DO 1111 J=N2,N3+1
DATA(I,J)=0.
1111 DATA(I,N+2-J)=0.
C
1195 IF(ISSW(1))1202,1203
1202 WRITE(6,901)
901 FORMAT(// "MAGNITUDE AFTER FILTERING")
DO 44 I=1,2*N-1,2
DO 45 J=1,N,8
DO 46 K=1,8
A(K)=(DATA(I,J+K-1))*2+(DATA(I+1,J+K-1))*2
46 A(K)=SQRT(A(K))
45 WRITE(6,101)(A(K),K=1,8)
44 WRITE(6,102)
WRITE(6,400)
C
1203 IF(ISSW(0))1002,1003
1002 WRITE(6,911)
911 FORMAT(// "REAL PART OF FILTERED TRANSFORM")
DO 8 I=1,N-1,2
DO 88 K=1,N,8
88 WRITE(6,101)(DATA(I,J),J=K,K+7)
8 WRITE(6,102)
WRITE(6,912)
912 FORMAT(// "IMAG. PART OF FILTERED TRANSFORM")
DO 913 I=2,2*N,2
DO 914 K=1,N,8
914 WRITE(6,101)(DATA(I,J),J=K,K+7)
913 WRITE(6,102)
1003 CONTINUE
C
C INVERSE FOURIER
CALL FOUR2(DATA,NN,2,1)
AMIN=DATA(1,1)
DO 856 I=1,2*N-1,2
DO 856 J=1,N
IF((DATA(I,J)+10.)-(AMIN+10.))857,856,856
857 AMIN=DATA(I,J)
856 CONTINUE
WRITE(6,903)(NAME(I),I=1,20)
903 FORMAT(// "RESULT - ",20A2)
C
C OUTPUT MAP TO PRINTER
150 IF(ISSW(12))155,151
151 WRITE(6,40)
40 FORMAT(/////////)
WRITE(6,3000)
DO 500 I=1,2*N-1,2

```

```

DO 550 J=1,N
  I1=17
700  I1=I1-1
    IF(I1-1)701,701,702
702  IF((DATA(I,J)+10.)-(DELTA*I1+AMIN+10.))700,701,701
701  IDATA(J)=IX(I1)
550  CONTINUE
    WRITE(6,3500)(IDATA(J),J=1,N)
500  CONTINUE
    WRITE(6,3000)
    WRITE(6,40)
C
C  BIT 2 ON FOR 11TH RECORD ON PLOTTER
155  IF(ISSW(2))160,171
160  IF(I1)156,156,157
156  IF(ISSW(3))171,157
157  WRITE(6,40)
    AB=0.
    DO 161 I=1,N
      IF(AB-DATA(11,I))159,161,161
159  AB=DATA(11,I)
161  CONTINUE
      WRITE(6,163)AB
163  FORMAT("MAX. VALUE = ",F6.3)
      CALL EXEC(1,110B,128,1,1)
      CALL EXEC(1,110B,255,1,1)
      WRITE(1,169)
169  FORMAT("INPUT MAX.:")
      READ(1,*)AB
      PAUSE
      DO 164 I=1,N
        IOUT=(DATA(11,I)/AB)*127.+128.
        DO 166 J1=1,200
          DO 166 J2=1,200
166  CONTINUE
164  CALL EXEC(1,110B,IOUT,1,1)
        DO 167 J1=1,200
          DO 167 J2=1,200
167  CONTINUE
        CALL EXEC(1,110B,128,1,1)
171  CONTINUE
C
    IF(I1)200,200,201
201  CONTINUE
    END
    ENDS

```

FTN

C

C

C

C

SOURCE - H2DS . JIM AYLOR

C

C

THIS PROGRAM PERFORMS FILTERING
VIA THE FHT

C

C

C

L=0; NO FILTERING

C

BIT 1 ON FOR HIGH PASS, OFF FOR LOW PASS

C

BIT ON FOR INPUT MAP

C

BIT 3 ON FOR PLOTTER

C

BIT 12 ON FOR NO MAPS

C

BIT 14 ON FOR PRINTING TRANSFORMS

C

BIT 15 ON FOR PRINTING INPUT DATA

C

C

C

CHANGE IF NEW ARRAY SIZE

DIMENSION XR(6,32),FX(32,32),IDATA(32)

C

DIMENSION IX(16)

INTEGER NAME(20),IB(91)

DATA IX/1H ,1H1,1H2,1H3,1H4,1H5,1H6,1H7,
&1H8,1H9,1HA,1HB,1HC,1HD,1HE,1HF/

C

C

CHANGE IF NEW ARRAY SIZE

N=32

M=6

INPUT = 8

3000

FORMAT(20X,"+++++++")

3500

FORMAT(20X,"+",32A1,"+")

C

WRITE(1,17)

17

FORMAT(/"INPUT SAMPLE NUMBER:")

READ(1,*)NFLE

NFLE=NFLE-1

CALL SPFIL(NFLE)

C

WRITE(1,1)

1

FORMAT("WHAT IS THE CUT-OFF TERM:")

READ(1,*)L

IF(L)40,40,41

40

WRITE(6,2000)

2000

FORMAT("//"NO FILTERING")

GO TO 44

41

IF(ISSW(1))43,42

42

LL=L+1

WRITE(6,2002)

2002

FORMAT("//"LOW PASS FILTER")

WRITE(6,2001)LL,N

2001

FORMAT(///"TERMS ",I3," THROUGH ",I3," CUT OFF")


```

GO TO 44
43  IL=2
    LM=L-1
    WRITE(6,2003)
2003 FORMAT(// "HIGH PASS FILTER")
    WRITE(6,2001) IL,LM
44  CONTINUE
200  FORMAT(8F6.3)
C
    READ(INPUT) (NAME(I), I=1,20)
    DO 2040 I=1,28
2040  READ(INPUT) (IB(J), J=1,91)
    DO 2010 I=29,60
    READ(INPUT) (IB(J), J=1,91)
    DO 2020 J=29,60
2020  FXY(I-28, J-28) = IB(J) / 400.
2010  CONTINUE
    DO 2030 I=61,91
2030  READ(INPUT) (IB(J), J=1,91)
    REWIND INPUT
C
C    PRINT INPUT
C    BIT 15 ON FOR PRINTING
    IF(ISSW(15)) 25,15
25  WRITE(6,201)
201  FORMAT(// "INPUT DATA MAP")
    DO 20 I=1,N
    K=1
    DO 21 K=1,N,16
    WRITE(6,300) (FXY(I,J), J=K,K+15)
21  K=K+16
20  WRITE(6,301)
300  FORMAT(16F4.2)
301  FORMAT(/)
C
C    FIND MAX.
15  AMAX=FXY(1,1)
    AMIN=FXY(1,1)
    DO 1000 I=1,N
    DO 1000 J=1,N
    IF(FXY(I,J)-AMAX) 1002,1002,1001
1001  AMAX=FXY(I,J)
1002  IF(FXY(I,J)-AMIN) 1003,1000,1000
1003  AMIN=FXY(I,J)
1000  CONTINUE
    DELTA=(AMAX-AMIN)/16.
C
    II=-1
    IF(ISSW(2)) 1313,3
1313  WRITE(6,900)
900  FORMAT(// "INPUT DATA MAP")
    GO TO 1500

```

```

C
C      HADAMARD TRANSFORM
3      II=II+1
        DO 7 J=1,N
        DO 5 I=1,N
5      XR(I,I)=FXY(I,J)
        CALL HAD(XR,N,M)
        DO 6 I=1,N
6      FXY(I,J)=XR(M,I)
7      CONTINUE
        DO 10 I=1,N
        DO 8 J=1,N
8      XR(I,J)=FXY(I,J)
        CALL HAD(XR,N,M)
        DO 9 J=1,N
9      FXY(I,J)=XR(M,J)
10     CONTINUE
C
C      DIV. INVERSE BY N*N BUT NOT FORWARD
        IF(II)60,60,61
60     DO 70 I=1,N
        DO 70 J=1,N
70     FXY(I,J)=FXY(I,J)/(N*N)
C
C      PRINT TRANS FORM VALUES
C      BIT 14 ON FOR PRINTING
C
61     IF(ISSW(14))50,51
50     IF(II)53,53,56
53     WRITE(6,960)
960    FORMAT("//HADAMARD TRANSFORM OF INPUT")
        GO TO 55
56     WRITE(6,970)
970    FORMAT("//FILTERED INPUT DATA")
250    FORMAT(///)
55     DO 11 I=1,N
        K=1
        DO 111 K1=1,N,16
        WRITE(6,302)(FXY(I,J),J=K,K+15)
111    K=K+16
11     WRITE(6,301)
302    FORMAT(16F4.3)
C
C      PASS 1 OR PASS 2?
51     IF(II)52,52,30
C
C      PASS 1
C      FILTER
52     IF(L)1600,1600,90
C
C      BIT 1 ON FOR HIGH PASS
90     IF(ISSW(1))92,93

```

```

93      DO 95 I=L+1,N
        DO 95 J=1,N
95      FXY(I,J)=0.
        DO 955 J=L+1,N
        DO 955 I=1,N
955     FXY(I,J)=0.
        WRITE(6,901)
        GO TO 3

C
C      HIGH PASS
92      A=FX(1,1)
        DO 96 I=1,L-1
        DO 96 J=1,L-1
96      FXY(I,J)=0.
        FXY(1,1)=A
        WRITE(6,901)
        GO TO 3

C
C      PRINT RESULTING MAP
30      WRITE(6,980)
980     FORMAT(///"RESULTING MAP")

C
C      PRINT MAP
1500    IF(ISSW(12))1515,1514
1514    WRITE(6,901)
901     FORMAT(////)
        WRITE(6,3000)
        DO 1501 I=1,N
        DO 1502 J=1,N
        I1=17
1503    I1=I1-1
        IF(I1-1)1504,1504,1505
1505    IF((FX(I,J)+10.)-(DELTA*I1+AKIN+10.))1503,1504,1504
1504    IDATA(J)=IX(I1)
1502    CONTINUE
        WRITE(6,3500)(IDATA(J),J=1,N)
1501    CONTINUE
        WRITE(6,3000)
        WRITE(6,901)

C
C      BIT 3 ON FOR PLOTTER
C
1515    IF(ISSW(3))2500,2501
2500    WRITE(6,901)
        WRITE(6,2502)FX(8,1)
2502    FORMAT("FIRST VALUE = ",F6.3)
        CALL EXEC(1,110B,128,1,1)
        CALL EXEC(1,110B,255,1,2)
        WRITE(6,2506)
2506    FORMAT("INPUT MAX.:")
        READ(1,*)AB
        PAUSE

```

```

DO 2600 I=1,N
  IOUT=(FXY(8,I)/AB)*127.+128.
DO 2700 J1=1,200
DO 2700 J2=1,200
2700 CONTINUE
2600 CALL EXEC(1,110B,IOUT,1,1)
2501 CONTINUE
C
  IF(II+1)3,3,1600
1600 CONTINUE
END

```

```

C
C
C HADAMARD TRANSFORM SUBROUTINE
C

```

```

SUBROUTINE HAD(XR,N,M)
DIMENSION XR(M,N)
NH=N/2
DO 120 L=1,M-1
  LP=L+1
  LM=L-1
  NY=0
  NZ=2*LM
  NZI=2*NZ
  NZN=N/NZI
  DO 110 I=1,NZN
    NX=NY+1
    NY=NY+NZ
    JS=(I-1)*NZI
    JD=JS+NZI+1
    DO 100 J=NX,NY
      JS=JS+1
      J2=J+NH
      XR(LP,JS)=XR(L,J)+XR(L,J2)
      JD=JD-1
      XR(LP,JD)=XR(L,J)-XR(L,J2)
100 CONTINUE
110 CONTINUE
120 CONTINUE
RETURN
END
END$

```

FTN

C
C
C
C
C
C
C
C
C
C
C

SOURCE - BAYIS JIM AYLOR 2/6/77

THIS PROGRAM GENERATES THE BAYES THRESHOLD FOR
THE TWO CLASS CASE OF SIGNAL AND NOISE

BIT 0 ON FOR CHANGE OF RESEAU WIDTH

BIT 1 ON FOR CHANGE OF DELTA

BIT 2 ON FOR NO PRINT OUT

BIT 15 ON FOR LABELING

PROGRAM STAT

DIMENSION IDATA(91,91),NAME(20)

DIMENSION IK(300),IM(300)

DIMENSION PXW1(300),PXW2(300),PW1X(300),PW2X(300)

DIMENSION LABEL(40),IC(20)

COMMON IC

WRITE(1,10)

10 FORMAT("INPUT SAMPLE NUMBER:")

READ(1,*)N

N=N-1

CALL SPFIL(N)

C
C

ASSUME 9 WIDE

IS=9

IF(ISSW(0))16,19

16 WRITE(1,12)

12 FORMAT("INPUT WIDTH OF RESEAU(ODD NUMBER):")

READ(1,*)IS

19 N=((IS*91*2)-IS*IS)

W1=(8281.-N)/8281.

W2=N/8281.

IS1=IS/2

C
C

ASSUME DELTA=0.02

DELTA=0.02

IF(ISSW(1))25,28

25 WRITE(1,26)

26 FORMAT("INPUT DELTA:")

READ(1,*)DELTA

28 IDEL=DELTA*400.

C

DO 37 I=1,300

IK(I)=0

IM(I)=0

PXW1(I)=0.

PXW2(I)=0.

PW1X(I)=1.0

37 PW2X(I)=0.0

C

```

      READ(8)(NAME(I),I=1,20)
      WRITE(6,38)(NAME(I),I=1,20)
38    FORMAT(20A2/)
      WRITE(6,40)IS,DELTA
40    FORMAT("RESEAU WIDTH = ",I2/"DELTA DENSITY = "F5.3//)
C
      DO 50 I=1,91
50    READ(8)(IDATA(I,J),J=1,91)
      REWIND 8
C
      IMIN=1200
      IMAX=0
      DO 60 I=1,91
      DO 60 J=1,91
      IF(IDATA(I,J).LE.IMAX) GO TO 55
      IMAX=IDATA(I,J)
55    IF(IDATA(I,J).GE.IMIN) GO TO 60
      IMIN=IDATA(I,J)
60    CONTINUE
      M1=(IMIN/IDEL)+1
      M2=((IMAX/IDEL)+1)+1
C
C  FILL THE DENSITY BINS
      IJ1=46-(IS1+1)
      IJ3=46+IS1
      DO 230 I=1,IJ1
      DO 230 J=1,IJ1
      DO 230 K=M1,M2
      IA=IDEL*(K-1)
      IF((IDATA(I,J).GE.IA).AND.
&(IDATA(I,J).LT.(IA+IDEL))) IM(K)=IM(K)+1
      IF((IDATA(I,J+IJ2).GE.IA).AND.
&(IDATA(I,J+IJ2).LT.(IA+IDEL))) IM(K)=IM(K)+1
      IF((IDATA(I+IJ2,J).GE.IA).AND.
&(IDATA(I+IJ2,J).LT.(IA+IDEL))) IM(K)=IM(K)+1
230  IF((IDATA(I+IJ2,J+IJ2).GE.IA).AND.
&(IDATA(I+IJ2,J+IJ2).LT.(IA+IDEL))) IM(K)=IM(K)+1
      DO 235 I=M1,M2
235  IK(I)=IM(I)
      DO 240 I=IJ1+1,IJ2
      DO 240 J=1,IJ1
      DO 240 K=M1,M2
      IA=IDEL*(K-1)
      IF((IDATA(I,J).GE.IA).AND.
&(IDATA(I,J).LT.(IA+IDEL))) IK(K)=IK(K)+1
240  IF((IDATA(I,J+IJ2).GE.IA).AND.
&(IDATA(I,J+IJ2).LT.(IA+IDEL))) IK(K)=IK(K)+1
      DO 245 I=1,IJ1
      DO 245 J=IJ1+1,IJ2
      DO 245 K=M1,M2
      IA=IDEL*(K-1)
      IF((IDATA(I,J).GE.IA).AND.

```

```

&(IDATA(I,J).LT.(IA+IDEL))) IK(K)=IK(K)+1
245 IF((IDATA(I+IJ2,J).GE.IA).AND.
&(IDATA(I+IJ2,J).LT.(IA+IDEL))) IK(K)=IK(K)+1
DO 250 I=IJ1+1,IJ2
DO 250 J=IJ1+1,IJ2
DO 250 K=M1,M2
IA=IDEL*(K-1)
250 IF((IDATA(I,J).GE.IA).AND.
&(IDATA(I,J).LT.(IA+IDEL))) IK(K)=IK(K)+1
C
L1=0
K1=0
DO 270 I=M1,M2
K1=K1+IK(I)
270 L1=L1+IM(I)
X1=K1
X2=L1
DO 300 I=M1,M2
A1=IK(I)/X1
PXW1(I)=IM(I)/X2
PXW2(I)=(A1-PXW1(I)*W1)/W2
PW1X(I)=W1*PXW1(I)
PW2X(I)=W2*PXW2(I)
IF(A1.EQ.0.) GO TO 290
PW1X(I)=PW1X(I)/A1
PW2X(I)=PW2X(I)/A1
GO TO 300
290 PW1X(I)=PW1X(I-1)
PW2X(I)=PW2X(I-1)
300 CONTINUE
WRITE(6,350)W1,W2
350 FORMAT("P(W1) = ",F7.4,/, "P(W2) = ",F7.4//)
IF(ISSW(2))415,360
360 WRITE(6,370)
370 FORMAT("DENSITY",8X,"P(X)",9X,"P(X/W1)",2X,"P(X/W2)",
&4X,"P(W1/X)",4X,"P(W2/X)"/)
B1=0.
B2=0.
B3=0.
DO 400 I=M1,M2
A=DELTA*(I-1)
A1=IK(I)/X1
B1=B1+A1
B2=B2+PXW1(I)
B3=B3+PXW2(I)
400 WRITE(6,410)A,IK(I),A1,IM(I),PXW1(I),PXW2(I),PW1X(I),PW2X(I)
410 FORMAT(F4.2,3X,I4,2X,F6.4,3X,
&I4,2X,F6.4,3X,F6.4,5X,F6.4,5X,F6.4)
415 WRITE(6,420)K1,L1
420 FORMAT("//SAMPLE SPACES: ",5X,"W2 - ",I5,5X,"W1 - ",I5/)
WRITE(6,430)B1,B2,B3
430 FORMAT("FOR ALL X:"/"P(X) = ",F6.4/"P(X/W1) = ",F6.4/

```

```

      &"P(X/W2) = ",F6.4)
      WRITE(6,500)
500   FORMAT(//)
C
      DO 505 I=M1,M2
      IF(PW1X(I).LT.0.5) GO TO 507
505   CONTINUE
507   A=DELTA*(I-1)
      WRITE(6,508)A
508   FORMAT("BAYES OPTIMUM THRESHOLD = ",F4.2//)
C
      DO 510 I=1,M1-1
      PW1X(I)=PW1X(M1)
510   PW2X(I)=PW2X(M1)
      DO 515 I=M2+1,300
      PW1X(I)=PW1X(M2)
515   PW2X(I)=PW2X(M2)
      AMAX1=0.
      AMAX2=0.
      DO 520 I=M1,M2
      IF(PXW1(I).GT.AMAX1) AMAX1=PXW1(I)
      IF(PXW2(I).GT.AMAX2) AMAX2=PXW2(I)
520   CONTINUE
      WRITE(6,540)AMAX1,AMAX2
540   FORMAT("MAX P(X/W1) = ",F6.4/"MAX P(X/W2) = ",F6.4///)
      WRITE(1,550)
      WRITE(6,545)
545   FORMAT(1H^)
550   FORMAT("INPUT MAX. PROB:")
      READ(1,*)BMAX
C
C PLOT RESULTS
      CALL WINDO(0,1)
C
C
      AMAX=3.50
C
C
      IBIN=AMAX/DELTA
      DO 700 I=1,2
      CALL CHRS(27,12,128)
      B1=AMAX-DELTA
      GO TO (600,620),I
600   CALL RANGE(0.,B1,0.,BMAX,0,0)
      CALL AXES(0.,0.)
      CALL TICK(0,0.,B1,0.5,30,1)
      SPY=BMAX/10.
      CALL TICK(1,0.,BMAX,SPY,30,1)
      CALL GRAPH(0,IBIN,PXW1)
      CALL GRAPH(0,IBIN,PXW2)
      GO TO 650
620   CALL RANGE(0.,B1,0.,1.0,0,0)

```



```
CALL AXES(0.,0.)
CALL TICK(0,0.,B1,0.5,30,1)
CALL TICK(1,0.,1.0,.1,30,1)
CALL GRAPH(0,IBIN,PW1X)
CALL GRAPH(0,IBIN,PW2X)
650 IF(ISSW(15))670,680
670 WRITE(6,672)
672 FORMAT("SET CURSOR, STRIKE A KEY, AND INPUT A LABEL:")
CALL CURSI(IHD,IXPOS,IYPOS)
CALL TPLLOT(0,IXPOS,IYPOS)
CALL CHRS(31,128,128)
READ(6,675)(LABEL(J),J=1,40)
675 FORMAT(40A1)
WRITE(1,675)(LABEL(J),J=1,40)
WRITE(6,677)
677 FORMAT("TYPE 1 FOR MORE LABELS, ELSE 0:")
READ(6,*)IHD
IF(IHD.EQ.1) GO TO 670
680 WRITE(6,690)
690 FORMAT("TYPE 1 TO CONTINUE:")
READ(6,*)IHD
IF(IHD.NE.1) GO TO 680
700 CONTINUE
END
END$
```

FTN

C

C

C

C

C

C

C

C

C

C

SOURCE - TPMS JIM AYLR

THIS PROGRAM PERFORMS TEMPLATE MATCHING ON
THE RESEAU IMAGES ASSUMING A PULSE-LIKE IMAGE
AS A TEMPLATE

BIT 0 ON FOR INTERMEDIATE BEST MATCH

PROGRAM TPMT

DIMENSION IA(91,91),NAME(20),RSLT(5),IXR(5),IYR(5)

INTEGER RSTT,RSTP,CSTT,CSTP

C

C IMAGE STEP SIZE

IS=480

C

REWIND 8

WRITE(1,10)

10 FORMAT(/"INPUT TAPE FILE NUMBER:")

READ(1,*)N

N=N-1

WRITE(1,20)

20 FORMAT("INPUT IMAGE WIDTH(ODD NUMBER):")

READ(1,*)IW

IW1=IW/2

WRITE(1,24)

24 FORMAT("INPUT START/STOP ROW VALUES FOR TEMPLATE:")

READ(1,*)RSTT,RSTP

WRITE(1,26)

26 FORMAT("INPUT START/STOP COLUMN VALUES FOR TEMPLATE:")

READ(1,*)CSTT,CSTP

C

CALL SPFIL(N)

C

READ(8)(NAME(I),I=1,20)

WRITE(6,30)(NAME(I),I=1,20)

WRITE(6,28)RSTT,RSTP

28 FORMAT("ROWS SCANNED BY TEMPLATE:",3X,I2," - ",I2)

WRITE(6,29)CSTT,CSTP

29 FORMAT("COLUMNS SCANNED BY TEMPLATE:",3X,I2," - ",I2//)

30

FORMAT(20A2/)

DO 32 I=1,91

32

READ(8)(IA(I,J),J=1,91)

REWIND 8

C

C CREATES ERROR FOR ZERO IMAGE

C ASSUMES ABSOLUTE VALUE ERROR

SUM=0.

DO 35 I=1,5

RSLT(I)=1E20

IXR(I)=0

```

35     IYR(I)=0
C
      DO 40 I=1,91
      DO 40 J=1,91
40     SUM=SUM+IA(I,J)
      SUMS=SUM
C
      DO 100 I=RSTT,RSTP
      DO 100 J=CSTT,CSTP
      SUM=SUMS
      IND=J-IW1-1
      DO 60 I1=1,91
      DO 60 J1=1,IW
      IX=IA(I1,IND+J1)
      Y=IABS(IS-IX)
60     SUM=SUM-IX+Y
      IND=I-IW1-1
      DO 70 J1=1,J-IW1-1
      DO 70 I1=1,IW
      IX=IA(I1+IND,J1)
      Y=IABS(IS-IX)
70     SUM=SUM-IX+Y
      DO 72 J1=J+IW1+1,91
      DO 72 I1=1,IW
      IX=IA(I1+IND,J1)
      Y=IABS(IS-IX)
72     SUM=SUM-IX+Y
C
C  INSERT RESULT
      DO 90 I1=1,5
      IF(SUM.GT.RSLT(I1)) GO TO 90
      IF(I1.EQ.5) GO TO 80
      I2=5-I1
      DO 75 J1=1,I2
      J2=6-J1
      J3=5-J1
      RSLT(J2)=RSLT(J3)
      IXR(J2)=IXR(J3)
75     IYR(J2)=IYR(J3)
80     RSLT(I1)=SUM
      IXR(I1)=I
      IYR(I1)=J
      GO TO 95
90     CONTINUE
C
95     IF(ISSW(0))97,100
97     WRITE(6,160)IXR(1),IYR(1),RSLT(1)
      WRITE(6,98)I,J,SUM
98     FORMAT("I = ",I2," J = ",I2," SUM = ",F12.0/)
100    CONTINUE
C
C  PRINT RESULTS

```

```
DO 150 I=1,5
150  WRITE(6,160)IXR(I),IYR(I),RSLT(I)
160  FORMAT("X = ",I2," Y = ",I2," ERROR = ",F12.1)
      WRITE(6,170)
170  FORMAT(////)
      STOP
      END
      ENDS
```

FTN

C

C

SOURCE -- EDGES

JIM AYLOR

C

C

THIS PROGRAM PERFORMS THE CENTER OF GRAVITY
TECHNIQUE WITH OR WITHOUT THRESHOLDING

C

C

BIT 0 ON TO REMOVE ABOVE THRESHOLD

C

PROGRAM EDGE

DIMENSION NAME(20),IA(91,91)

DIMENSION CP(2),CM(2),RP(2),RM(2),IRM(2),IRP(2)

DIMENSION ICM(2),ICP(2)

C

N=91

C

WRITE(1,10)

10

FORMAT("INPUT FILE NUMBER:")

READ(1,*)IF

IF=IF-1

CALL SPFIL(IF)

C

READ(8)(NAME(I),I=1,20)

WRITE(1,15)(NAME(I),I=1,20)

15

FORMAT(20A2//)

DO 20 I=1,N

20

READ(8)(IA(I,J),J=1,N)

REWIND 8

C

IF(ISSW(0))30,60

30

WRITE(1,40)

40

FORMAT("INPUT THRESHOLD VALUE:")

READ(1,*)THRES

ITH=400.*THRES

DO 50 I=1,N

DO 50 J=1,N

IF(IA(I,J).LT.ITH) IA(I,J)=ITH

50

CONTINUE

60

CONTINUE

C

C TEMPLATE MATCHING -- COLUMNS

DO 130 J=1,N-2

SX=0.

DO 100 I=1,N-2

SX1=IA(I,J+2)+2*IA(I+1,J+2)+IA(I+2,J+2)

SX2=IA(I,J)+2*IA(I+1,J)+IA(I+2,J)

100

SX=SX+(SX1-SX2)

I1=J

DO 125 I=1,2

IF(SX)115,130,120

115

IF(SX.GE.CM(I)) GO TO 125

X1=CM(I)

```

      CM(I)=SX
      SX=X1
      I2=ICM(I)
      ICM(I)=I1
      I1=I2
      GO TO 125
120  IF(SX.LE.CP(I)) GO TO 125
      X1=CP(I)
      CP(I)=SX
      SX=X1
      I2=ICP(I)
      ICP(I)=I1
      I1=I2
125  CONTINUE
130  CONTINUE
C
C  TEMPLATE MATCHING - ROWS
      DO 230 I=1,N-2
      SY=0.
      DO 200 J=1,N-2
      SY1=IA(I+2,J)+2*IA(I+2,J+1)+IA(I+2,J+2)
      SY2=IA(I,J)+2*IA(I,J+1)+IA(I,J+2)
200  SY=SY+(SY1-SY2)
      I1=I
      DO 225 J=1,2
      IF(SY)215,230,220
215  IF(SY.GE.RM(J)) GO TO 225
      X1=RM(J)
      RM(J)=SY
      SY=X1
      I2=IRM(J)
      IRM(J)=I1
      I1=I2
      GO TO 225
220  IF(SY.LE.RP(J)) GO TO 225
      RP(J)=SY
      SY=X1
      I2=IRP(J)
      IRP(J)=I1
      I1=I2
225  CONTINUE
230  CONTINUE
C
C  CORRECT LAG NUMBER
      DO 240 I=1,2
      ICP(I)=ICP(I)+1
      ICM(I)=ICM(I)+1
      IRM(I)=IRM(I)+1
      IRP(I)=IRP(I)+1
240  CONTINUE
C
C  PRINT RESULTS

```

```

WRITE(6,250)(NAME(I),I=1,20)
250 FORMAT(20A2//)
WRITE(6,255)THRES
255 FORMAT("BAYES THRESHOLD = ",F5.2/)
WRITE(6,260)
260 FORMAT("COLUMN TEMPLATE")
WRITE(6,280)(CP(I),I=1,2),(CM(I),I=1,2)
280 FORMAT(2(F8.3,3X),3X,2(F8.3,3X))
WRITE(6,290)(ICP(I),I=1,2),(ICM(I),I=1,2)
290 FORMAT(2X,2(I2,9X),3X,2(I2,9X))
WRITE(6,300)
300 FORMAT("/"ROW TEMPLATE")
WRITE(6,280)(RP(I),I=1,2),(RM(I),I=1,2)
WRITE(6,290)(IRP(I),I=1,2),(IRM(I),I=1,2)
C
C CALCULATE CGS
J1=ICP(1)-1
J2=ICM(1)+1
I1=IRP(1)-1
I2=IRM(1)+1
SUMX=0.
SUMY=0.
C
SUM=0.
DO 310 I=1,N
DO 310 J=J1,J2
A=IA(I,J)/400.
SUM=SUM+A
SUMY=SUMY+J*A
310 CONTINUE
SUMY=SUMY/SUM
C
SUM=0.
DO 320 I=I1,I2
DO 320 J=1,N
A=IA(I,J)/400.
SUM=SUM+A
SUMX=SUMX+I*A
320 CONTINUE
SUMX=SUMX/SUM
C
WRITE(6,350)
350 FORMAT(///// "CENTER DETECTION BY MOMENTS(CG):")
WRITE(6,400)SUMX,SUMY
400 FORMAT(// "X(ROW) = ",F6.2/"Y(COLUMN) = ",F6.2///)
STOP
END
END$

```