Proceedings of the Third Annual

Symposium on Mathematical Pattern Recognition & Image Analysis

(NASA-CR-171925) PROCEEDINGS OF THE THIRD ANNUAL SYMPOSIUM ON MATHEMATICAL PATTERN RECOGNITION AND IMAGE ANALYSIS Final Report, 16 Jul. 1984 - 15 Jul. 1985 (Texas A&M Univ.) 535 p CSCL 12A G3/64

N86-33087

Unclas 44626

June 10-11, 1985
Texas A&M University
College Station, Texas

FINAL REPORT
"MATHEMATICAL PATTERN RECOGNITION
AND IMAGE ANALYSIS PROGRAM"

Contract NAS 9-16664

Period: July 16, 1984 - July 15, 1985

Prepared for:

Earth Resources Research Division
NASA/Johnson Space Center
Houston, Texas 77058

by

L. F. Guseman, Jr.
Principal Investigator and MPRIA Program Coordinator
Department of Mathematics
Texas A&M University
College Station, Texas 77843

TABLE OF CONTENTS

Page

Preface - L. F. Guseman, Jr. iii

Papers Presented by Principal Investigators:

The Use of Multivariate Spline Methods in Classification
L. F. Guseman, Jr. and L. L. Schumaker. 1

Methods of Normal Mixture Analysis Applied to Remote Sensing
Charles Peters. 33

Quantile Data Analysis Methods for Image Data Analysis and Edge Detection
Emanuel Parzen. 65

Classification in a Spatially Correlated Environment
C. N. Morris, D. V. Hinkley, and W. Johnston. 95

Estimating Parameters in a Finite Mixture of Probability Densities
R. P. Heydorn 129

Experiences with Examining Large Multivariate Data Sets with Graphical Nonparametric Methods
David W. Scott. 141

Analysis of Subpixel Registration
David Lavine, Eric C. Olson, Barbara A. Lambird, Carlos A. Berenstein, and Laveen N. Kanal. 149

Recovery of Surface Shape from Multiple Images
Grahame B. Smith. 217

Hypothesis Integration in Image Understanding Systems
Vincent Shang-Shoug Hwang, Larry S. Davis, and Takashi Matsuyama 239

Investigation of Critical Issues in Rectification and Registration of Satellite Scanner Imagery
Fidel C. Paderes, Jr. and Edward M. Mikhail 339

Relating Ground Scenes to Spatial Variation in Remotely Sensed Images
Curtis E. Woodcock and Alan H. Strahler 393

Page

Influence of Ground Control Ppoint Selection on Landsat MSS and TM Rectification Accuracy: Whole Scene vs. Portions of a Scene
David D. Dow 451

Smooth Multidimensional Interpolation
W. Tobler and S. Kennedy 481

An Optimal Frequency Domain Textural Edge Detection Filter
J. Keith Townsend, K. Sam Shanmugan, and Victor S. Frost . . . 497

PREFACE

This volume comprises the Proceedings of the Third Annual Symposium on Mathematical Pattern Recognition and Image Analysis (MPRIA) held June 10-11, 1985, at Texas A&M University, College Station, Texas.

The Symposium was initiated with a brief Program Overview presented by Drs. Diane Wickland, NASA Headquarters, and R. P. Heydorn, NASA/JSC.

The thirteen papers of the Proceedings reflect the results of various research efforts initiated during FY 1983 as part of NASA's Remote Sensing Research Program. Two of the papers present results from research efforts carried out by the following NASA principal investigators:

R. P. Heydorn - NASA/Johnson Space Center

David D. Dow - National Space Technology Laboratories

Results from an additional NASA research effort carried out at JPL appear in the report (available from the authors):

> Scene Segmentation," Final Report, NASA Fundamental Research Program (1982-1984), Jet Propulsion Laboratory, California Institute of Technology, Pasadena, California, 91109, March, 1985.

The remaining papers present third-year results from the eleven research efforts initiated July 16, 1982, under Contract NAS 9-16664 and carried out by the following principal investigators:

L. Schumaker/L. F. Guseman, Jr. - Texas A&M University

H. P. Decell, Jr./B. C. Peters, Jr. - University of Houston

E. Parzen/W. B. Smith - Texas A&M Universty

Carl Morris - University of Texas at Austin

L. Kanal - LNK Corporation

Grahame Smith - SRI International

L. S. Davis/A. Rosenfeld - University of Maryland

E. M. Mikhail - Purdue University

A. H. Strahler - Hunter College

W. Tobler - University of California at Santa Barbara

K. S. Shanmugan - University of Kansas

In an attempt to group presentations of a similar nature, the Symposium was divided into two MATH/STAT sessions and two PATTERN RECOGNITION sessions.

The papers appear in the Proceedings in the order in which they were presented at the Symposium. An agenda and a list of attendees who registered for the Symposium are included in the Appendix.

L. F. Guseman, Jr.
Principal Investigator and
MPRIA Program Coordinator
COntract NAS 9-16664

THE USE OF MULTIVARIATE SPLINE METHODS IN CLASSIFICATION

by

L. F. Guseman, Jr. and L. L. Schumaker
Center for Approximation Theory
Department of Mathematics
Texas A&M University
College Station, Texas 77843

Abstract

This report is a continuation of earlier papers prepared for the 1983 and 1984 NASA MPRIA Symposia Proceedings. The earlier reports dealt with theoretical aspects of the use of spline functions in the construction of classification algorithms. In this report we synthesize our earlier works into a specific algorithm and discuss the results of applying this algorithm to several test examples. The method involves tensor-product spline fits to histograms obtained from training data, followed by numerical determination of Bayes classification regions. Numerical estimates for the probabilities of missclassification are also calculated for each example.

§1. Introduction.

This paper is concerned with the use of spline functions as a tool in statistical pattern classification algorithms. A theoretical approach to Bayes classification based on spline functions was discussed in two earlier NASA symposium proceedings -- see [13,14]. Our aim here is to present the results of several numerical experiments using software based on the theoretical results of [13,14].

The paper is divided into 4 sections. In Section 2 we briefly review the Bayes classification procedure. In Section 3 we outline the algorithm which we are using. Some numerical results are presented in Section 4.

§2. The Bayes Classification Procedure.

Suppose that some group Π of objects can be divided into NC classes which we will denote by $\Pi_1, \Pi_2, \ldots, \Pi_{NC}$. Now suppose that we are trying to decide which class a given randomly selected object belongs to on the basis of d measurements which have been taken on the object. In particular, suppose X is a mapping from $\Pi = \Pi_1 \cup \ldots \cup \Pi_{NC}$ into R^d such that if $w \in \Pi$, then $X(w) = (x_1, \ldots, x_d)$ is the vector of measurements taken on w. Finally, suppose that for each $i = 1, \ldots, NC$, we know the a priori probability α_i that an object will fall in class Π_i and that we also know the conditional density function P_i associated with measurments taken from the i-th class.

Given this stochastic framework, the Bayes optimal classifier is defined as follows:

> Assign an element w to the i-th class Π_i if and only if its measurement vector X(w) belongs to the set R_i,

where $R_1,\ldots,R_{NC}$ are the Bayes decision regions defined by

$$(2.1) \qquad R_i = \{x \in R^d : \alpha_i P_i(x) \geq \alpha_j P_j(x) \text{ for all } j \neq i\} .$$

The numerical problem of identifying the Bayes decision regions is equivalent to finding the boundaries of the sets R_i. These in turn are defined by the equations $\alpha_i P_i(x) - \alpha_j P_j(x) = 0$ for i, j = 1,...,NC.

There are several well-known ways of measuring the quality of the Bayes classification scheme described above. One convenient way is to compute the probability of misclassification (PMC) (cf. [1,2]) denoted below by G, and defined by

$$(2.2) \quad G = 1 - \int_{R^d} \max_i [\alpha_i P_i(x)]dx = 1 - \sum_{i=1}^{NC} \alpha_i \int_{R_i} P_i(x)dx .$$

In general, the evaluation of the PMC G is a difficult problem since it involves integration over irregularly-shaped regions in d-space.

To apply the Bayes classification procedure in a practical setting, the following steps need to be carried out:

1) estimate NC = number of classes,

2) estimate the a priori probabilities $\alpha_1,\ldots,\alpha_{NC}$,

3) estimate the density functions $P_1,\ldots,P_{NC}$,

4) estimate the decision regions $R_1,\ldots,R_{NC}$,

5) estimate the value G of the PMC.

In this paper we shall discuss our experience with steps 3) - 5), assuming that steps 1) and 2) have already been performed. Following [13, 14] we handle step 3) by using training data to construct a histogram associated with each density P_i, after which we construct a tensor-product spline fit s_i to this histogram based on volume matching. Step 4) is carried out by computing the approximate Bayes regions

$$(2.3) \qquad R_i^* = \{x \in R^2 : \alpha_i s_i(x) \geq \alpha_j s_j(x), \text{ all } j \neq i\}, \quad i = 1,..,NC.$$

When the equality $\alpha_i s_i(x) = \alpha_j s_j(x)$ holds, we put x in the set R_i^* provided i is the least integer j for which $\alpha_i P_i(x) = \alpha_j P_j(x)$. The boundaries of the decision regions are contour lines defined by the equations $\delta_{ij}(x) = \alpha_i P_i(x) - \alpha_j P_j(x) = 0$. In practice we compute only polygonal approximations R_i^{**} to the regions R_i^*.

Given the approximate Bayes regions $R_1^{**},\ldots,R_{NC}^{**}$, we can now compute an estimate G* for the PMC G defined in (2.2) as follows:

$$(2.4) \qquad G^* = 1 - \sum_{i=1}^{NC} \alpha_i \int_{R_i^{**}} s_i(x)dx \ .$$

These integrals cannot be computed exactly, but using the fact that is is possible to integrate tensor-product splines exactly over rectangular sets, they can be computed to within arbitrary accuracy (cf. [14]). We shall denote our approximation to G* by G**.

§3. The algorithm.

In this section we summarize the steps in the numerical algorithm outlined in the previous section. The notation here follows [13,14].

ALGORITHM:

A. (Perform the density fits)

1. Choose a rectangle H which contains most of the volume of the densities $P_1,\ldots,P_{NC}$.
2. For each i = 1,NC
   a. Choose the number of bins nbxi and nbyi in the x and y-directions, respectively.
   b. Choose the bin edges in the x and y directions to subdivide H into nbxi × nbyi equal-sized bins.
   c. Choose the number npi of samples to be drawn from the ith population to be used as training data.
   d. Draw npi samples from the ith population.
   e. Construct a histogram based on this data using the above bins
   f. Using the volume matching method of [13,14] with knots located at the bin edges, construct a quadratic tensor-product spline s_i approximating the density P_i.

B. (Compute the Bayes regions)

1. Choose a rectangular grid of points $K = \{t_{ij}: 1 \leq i \leq ngi, 1 \leq j \leq ngj\}$ on H.

2. For each $1 \leq k \leq NC$

a. For each i = 1,...,ngi and j = 1,...,ngj

Compute the values of $z_{ij}^k = \alpha_k s_k(t_{ij})$

Compute $w_{ij}^k = \max\{z_{ij}^\ell , \ell \neq i\}$

Compute $u_{ij}^k = z_{ij}^k - w_{ij}^k$

b. Use this grid of u-values to construct the contours defining R_k^{**} by the method of [14].

C. (Compute the approximate PMC value G**)

1. For k = 1,NC

Compute the approximate integral I_k of s_k over R_k^{**}

2. Form $G^{**} = 1 - (I_1 + \ldots + I_{NC})$.

Discussion: The choice of the number of bins and the number of samples to be used in step A2 of the algorithm has a major effect on the nature of the spline fit s_i to the density P_i. Our experience suggests choosing the bin-width to be about one standard deviation.

Step A2f amounts to finding the LU-decomposition of a square matrix of size nbxi followed by nbxi back substitutions (and a similar amount of work involving a matrix of size nbyi). This is highly efficient (cf. the discussion in [14]).

The construction of the contours in step B2b is accomplished by Algorithm 5.1 of [14]. Here we have elected to eliminate step 7 of that algorithm and have simply taken the polygonal boundary defined by the triangle edges. Since we have highly efficient algorithms for evaluating splines on grids, we can afford to use a fairly fine grid and the result is a set of visually smooth boundary curves for the decision regions.

If desired, this algorithm can be supplemented with a step B3 in which contours defining R_i^{**} are removed when the total volume of the spline u inside the given contour is less than some predetermined cutoff parameter ε. We call this process "clutter removal".

§4. Test results.

In this section we present the results of applying the algorithm of Section 3 to three test examples. For each example we give all relevant input parameters and the computed PMC values, with and without clutter removal. Each example is accompanied by a series of figures including

-- a perspective view of pmax = $\max\{P_1,\ldots,P_{NC}\}$

-- a perspective view of xmax = $\max\{s_1,\ldots,s_{NC}\}$

-- a plot of the decision regions based on the use of the true densities $P_1,\ldots,P_{NC}\}$

-- a plot of the approximate decision regions $R_1^{**},\ldots,R_{NC}^{**}$ computed using the spline density fits

-- A similar plot using clutter removal with ε = .01

The construction of the contours in step B2b is accomplished by Algorithm 5.1 of [14]. Here we have elected to eliminate step 7 of that algorithm and have simply taken the polygonal boundary defined by the triangle edges. Since we have highly efficient algorithms for evaluating splines on grids, we can afford to use a fairly fine grid and the result is a set of visually smooth boundary curves for the decision regions.

If desired, this algorithm can be supplemented with a step B3 in which contours defining R_i^{**} are removed when the total volume of the spline u inside the given contour is less than some predetermined cutoff parameter ε. We call this process "clutter removal".

§4. Test results.

In this section we present the results of applying the algorithm of Section 3 to three test examples. For each example we give all relevant input parameters and the computed PMC values, with and without clutter removal. Each example is accompanied by a series of figures including

-- a perspective view of pmax = $\max\{P_1,\ldots,P_{NC}\}$

-- a perspective view of xmax = $\max\{s_1,\ldots,s_{NC}\}$

-- a plot of the decision regions based on the use of the true densities $P_1,\ldots,P_{NC}\}$

-- a plot of the approximate decision regions $R_1^{**},\ldots,R_{NC}^{**}$ computed using the spline density fits

-- A similar plot using clutter removal with ε = .01

EXAMPLE 1:

Setup

NC = 2 classes

P_1 = normal density with mean (0,0) and covariance matrix I

P_2 = normal density with mean (2,0) and covariance matrix I

A-priori probabilities $\alpha_1 = \alpha_2 = .5$

Data

10,000 random points from each population

Histogram

Equally spaced bins of width 1 on the rectangle H = [-3,5]x[-3,3].
Total number of bins = 48

Spline Fit

Using quadratic splines with knots at bin centers
Total number of coefficients = 48

Computed PMC

Without clutter removal = 0.1526072
With clutter removal = 0.1527698

```
May 11 12:06 1985  h211 Page 1

 2
 8   6
-3.000000   -2.000000   -1.000000   +0.000000E+00   +1.000000
+2.000000   +3.000000   +4.000000   +5.000000
-3.000000   -2.000000   -1.000000   +0.000000E+00   +1.000000
+2.000000   +3.000000
    3.    24.    60.    68.    27.     2.
   34.   187.   471.   477.   186.    30.
   65.   462.  1220.  1223.   457.    86.
   80.   458.  1176.  1147.   440.    75.
   28.   180.   426.   424.   192.    32.
    7.    24.    68.    74.    30.     3.
    0.     5.     4.     2.     3.     1.
    0.     0.     0.     0.     0.     0.
 8   6
-3.000000   -2.000000   -1.000000   +0.000000E+00   +1.000000
+2.000000   +3.000000   +4.000000   +5.000000
-3.000000   -2.000000   -1.000000   +0.000000E+00   +1.000000
+2.000000   +3.000000
    0.     0.     0.     0.     0.     0.
    1.     3.     6.     6.     0.     1.
    4.    27.    70.    73.    27.     5.
   27.   175.   468.   452.   169.    29.
   75.   441.  1167.  1193.   475.    86.
   75.   466.  1105.  1144.   493.    77.
   31.   177.   484.   495.   183.    36.
    0.    32.    67.    75.    29.     6.
```

TABLE 1. Data for Example 1.

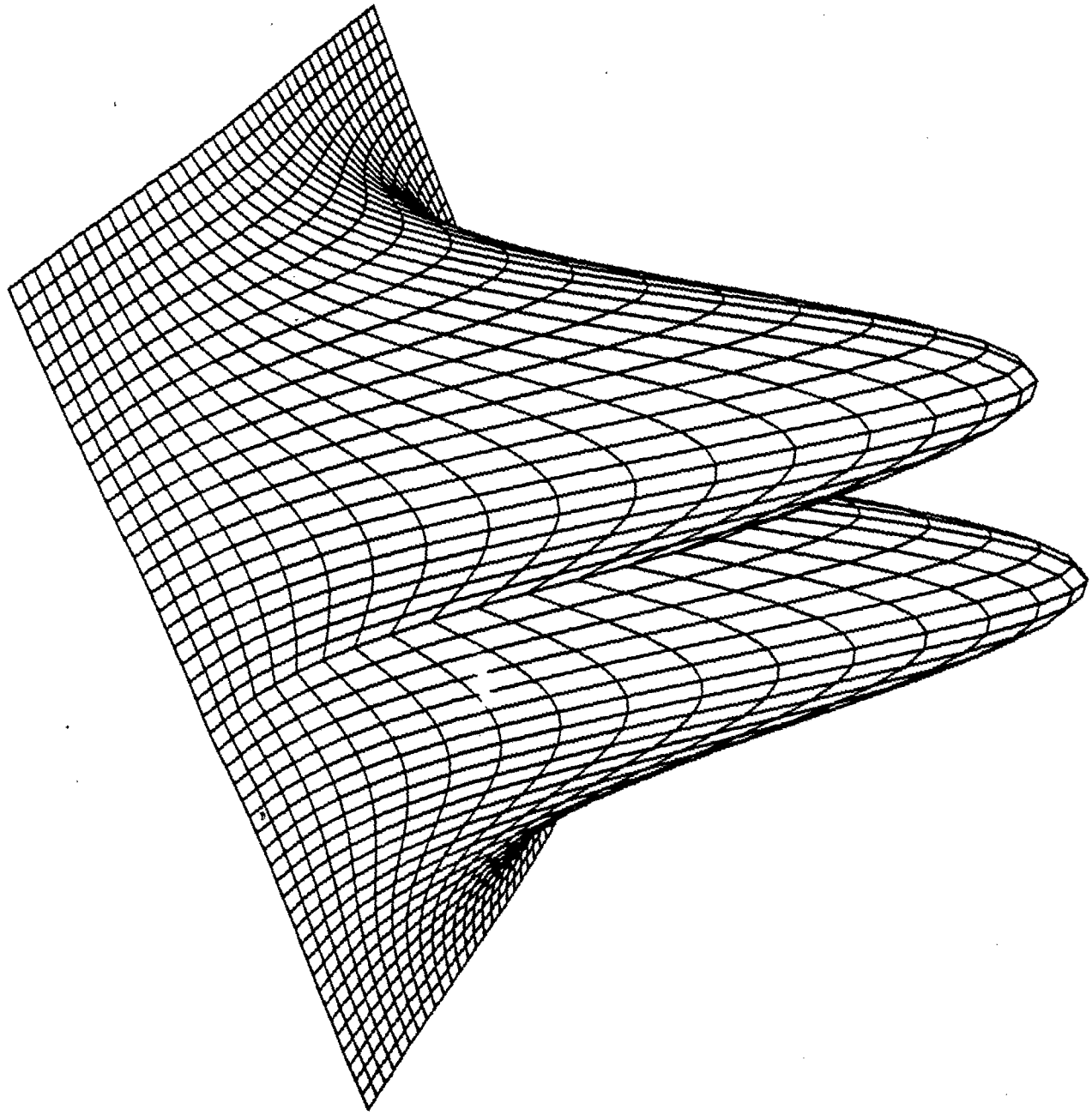


Fig. 1. The true densities of Example 1.

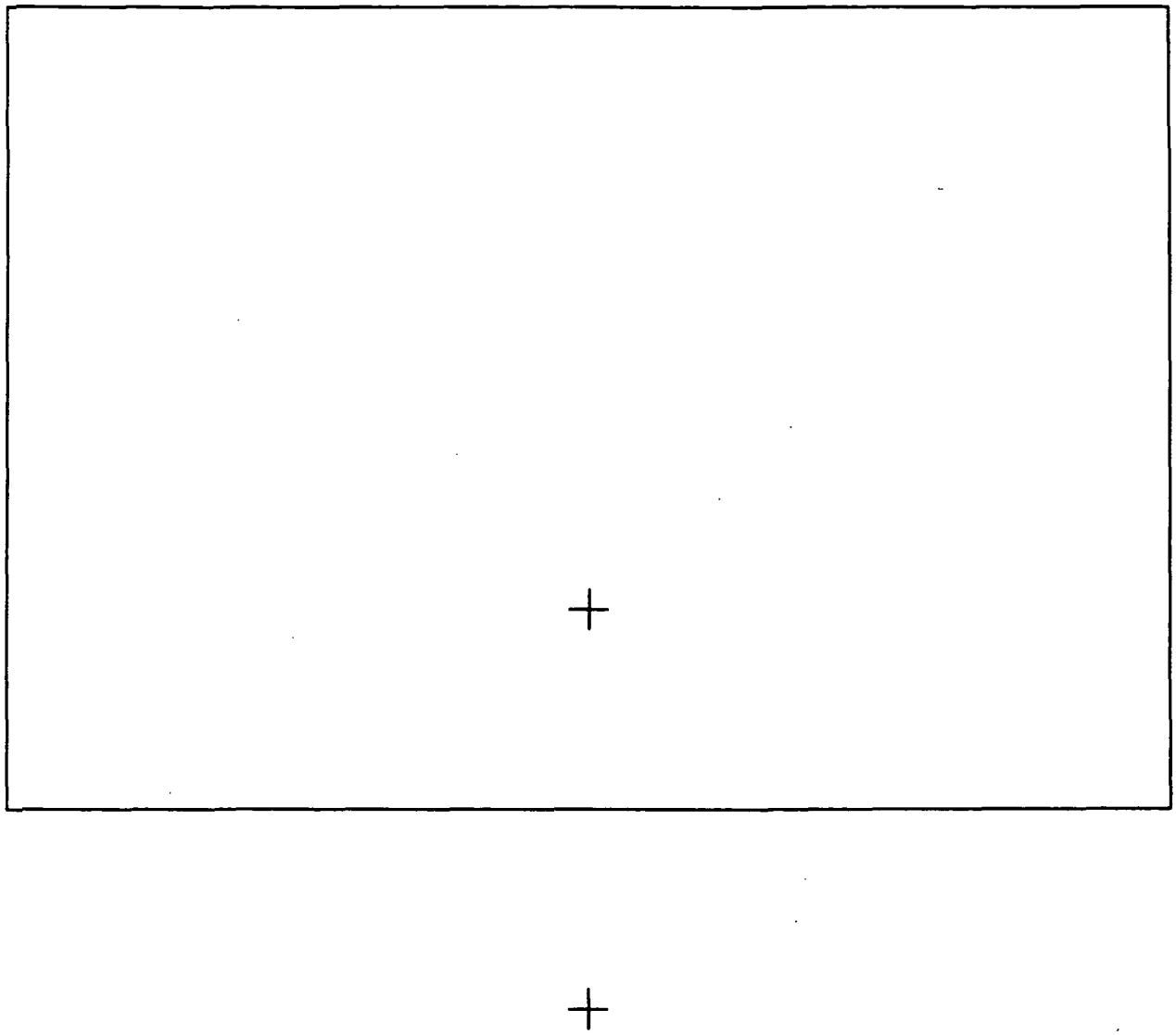


Fig. 2. The True decision regions for Example 1.

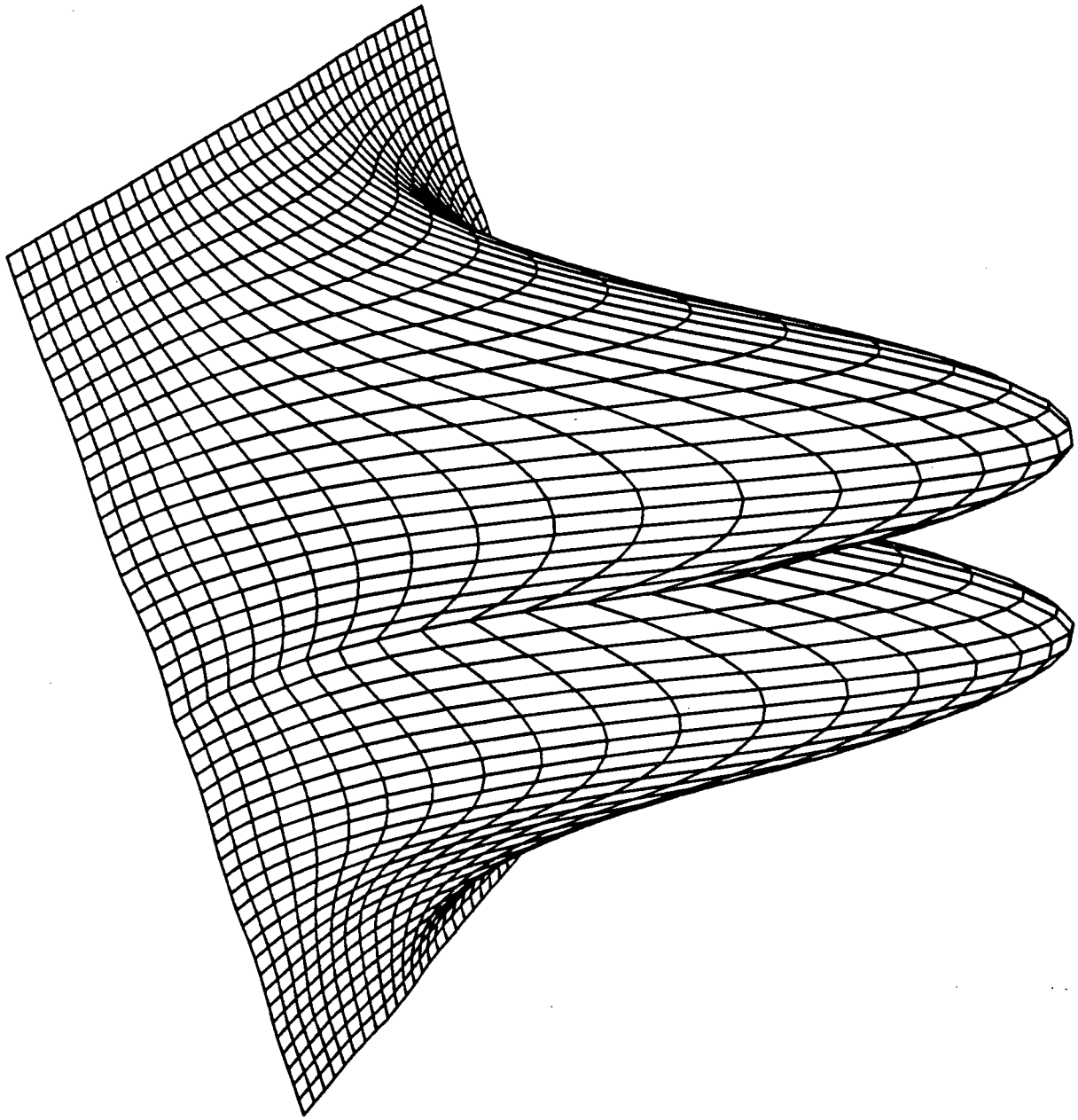


Fig. 2. The spline fits to the densities of Example 1.

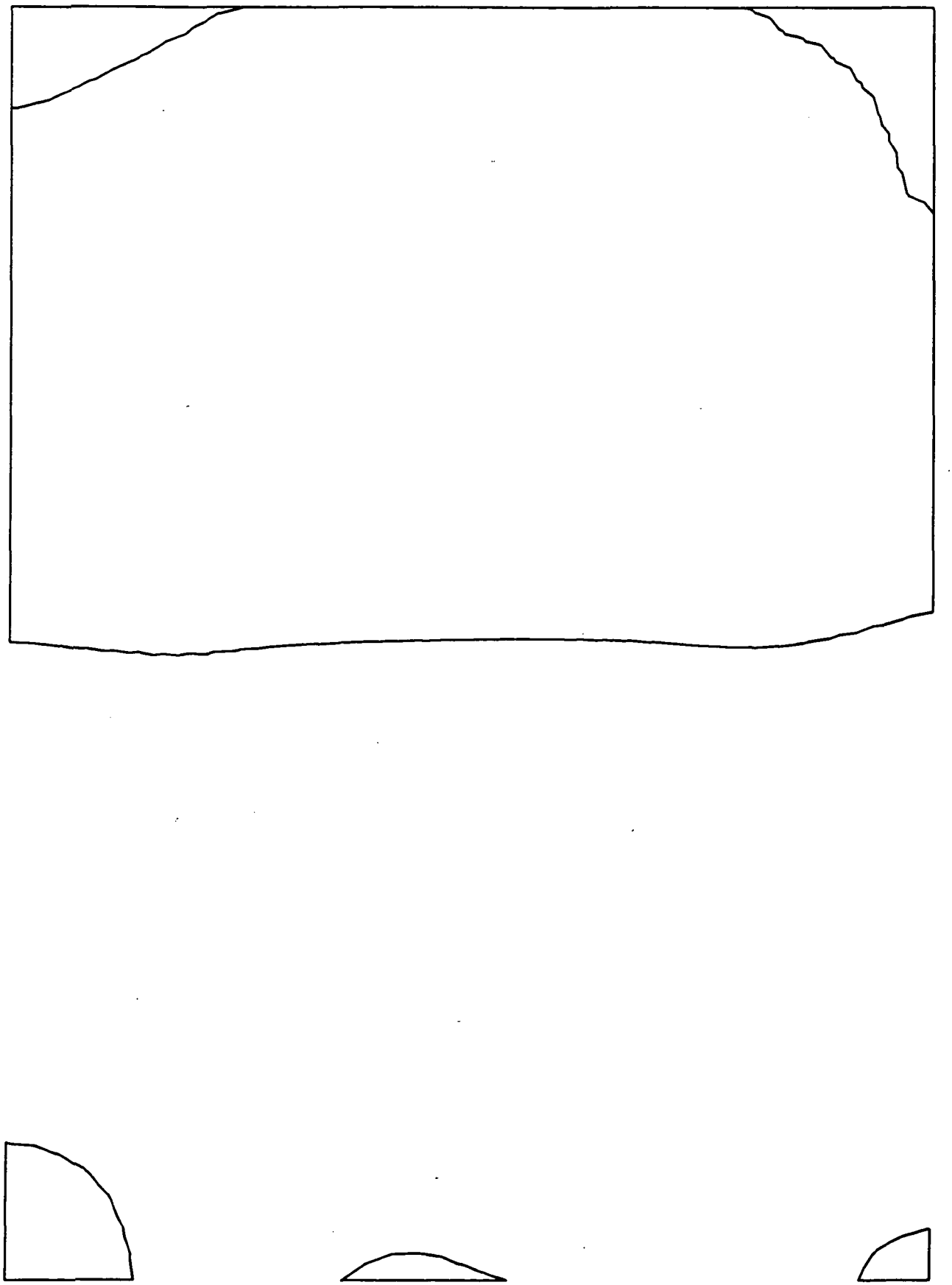


Fig. 4. The estimated decision regions for Example 1.

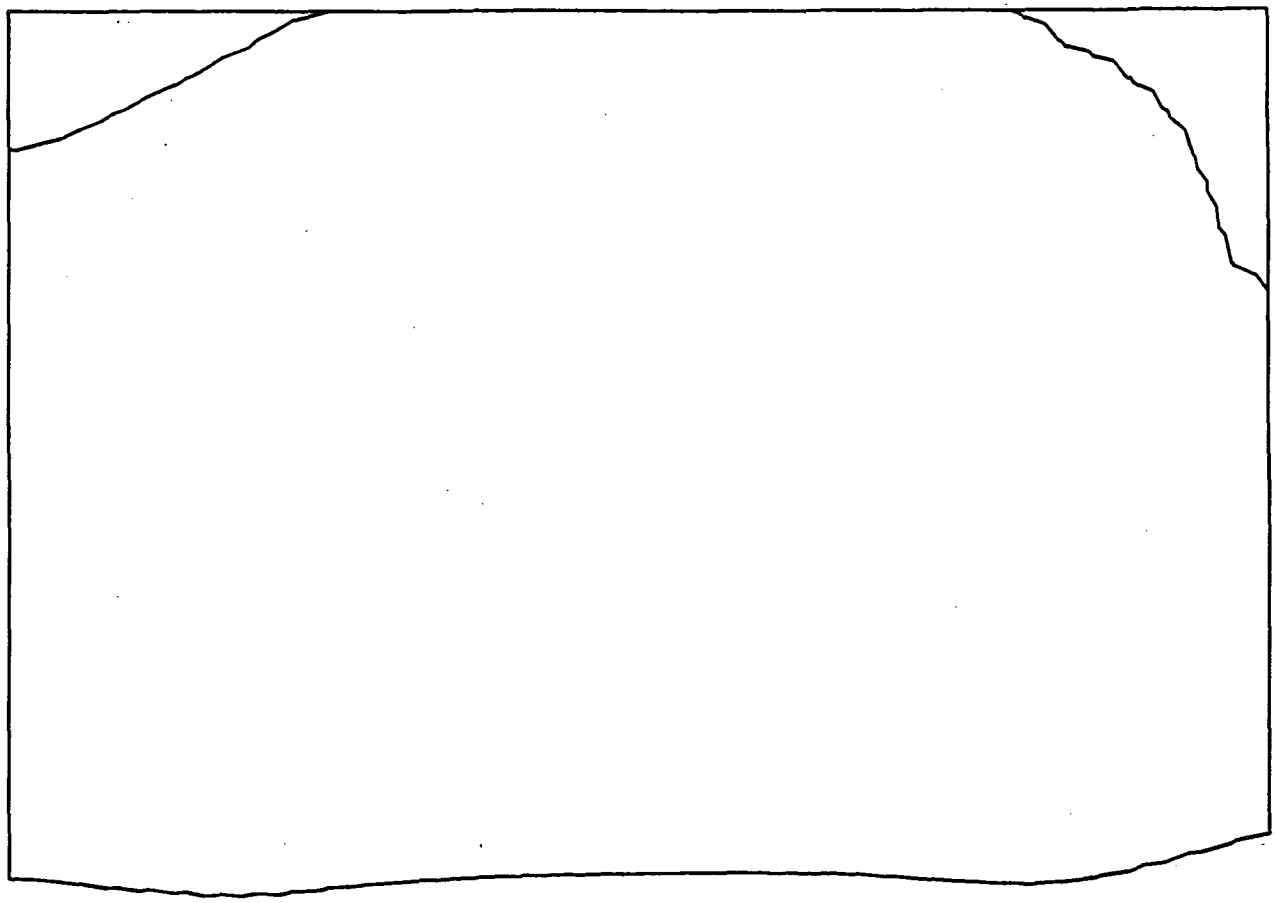


Fig. 5. The estimated decision regions for Example 1 (clutter removed).

EXAMPLE 2:

Setup

NC = 2 classes

P_1 = normal density with mean (0,0) and covariance matrix .5I

P_2 = normal density with mean (2,0) and covariance matrix I

A-priori probabilities $\alpha_1 = \alpha_2 = .5$

Data

25,000 random points from both populations

Histogram

P_1: Equally spaced bins of width 2/3 on the rectangle H = [-3,5]x[-3,3].
Total number of bins = 108

P_2: Equally space bins of width 1 on the rectangle H.
Total number of bins = 48

Spline Fit

Using quadratic splines with knots at bin centers
Total number of coefficients = 48 and 108, respectively.

Computed PMC

Without clutter removal = 0.1128998
With clutter removal = 0.1128956

```
Apr 22 20:41 1985  hgroms3 Page 1

 2
 8   6
-3.000000    -2.000000    -1.000000    +0.000000E+00    +1.000000
+2.000000    +3.000000    +4.000000    +5.000000
-3.000000    -2.000000    -1.000000    +0.000000E+00    +1.000000
+2.000000    +3.000000
   11.    66.   152.   175.    64.    12.
   75.   450.  1179.  1131.   447.    65.
  179.  1163.  2959.  2999.  1172.   213.
  193.  1163.  2850.  2902.  1172.   187.
   76.   452.  1134.  1153.   468.    81.
   11.    75.   165.   187.    75.     9.
    0.     8.     6.     9.     6.     3.
    0.     0.     0.     0.     0.     0.
12   9
-3.000000    -2.333333    -1.666667    -1.000000E+00    -3.333333E-01
+3.333334E-01    +1.000000    +1.666667    +2.333333    +3.000000
+3.666667    +4.333333    +5.000000
-3.000000    -2.333333    -1.666667    -1.000000E+00    -3.333333E-01
+3.333334E-01    +1.000000    +1.666667    +2.333333    +3.000000
    0.     0.     0.     0.     0.     0.     0.     0.     0.
    0.     0.     0.     0.     0.     0.     0.     0.     0.
    0.     0.     0.     0.     0.     0.     0.     0.     0.
    0.     0.     1.     0.     7.     2.     2.     0.     0.
    1.     2.     9.    69.    76.    59.    17.     0.     0.
    2.    22.    95.   410.   676.   407.   122.     8.     0.
    3.    44.   424.  1428.  2198.  1410.   453.    50.     2.
    4.    81.   610.  2217.  3332.  2212.   606.    87.     6.
    4.    52.   418.  1386.  2174.  1409.   427.    61.     3.
    0.    15.   110.   417.   583.   396.   128.    15.     3.
    0.     4.    15.    51.    83.    61.    13.     3.     0.
    0.     0.     2.     3.     2.     4.     3.     0.     0.
```

TABLE 2. The data for Example 2.

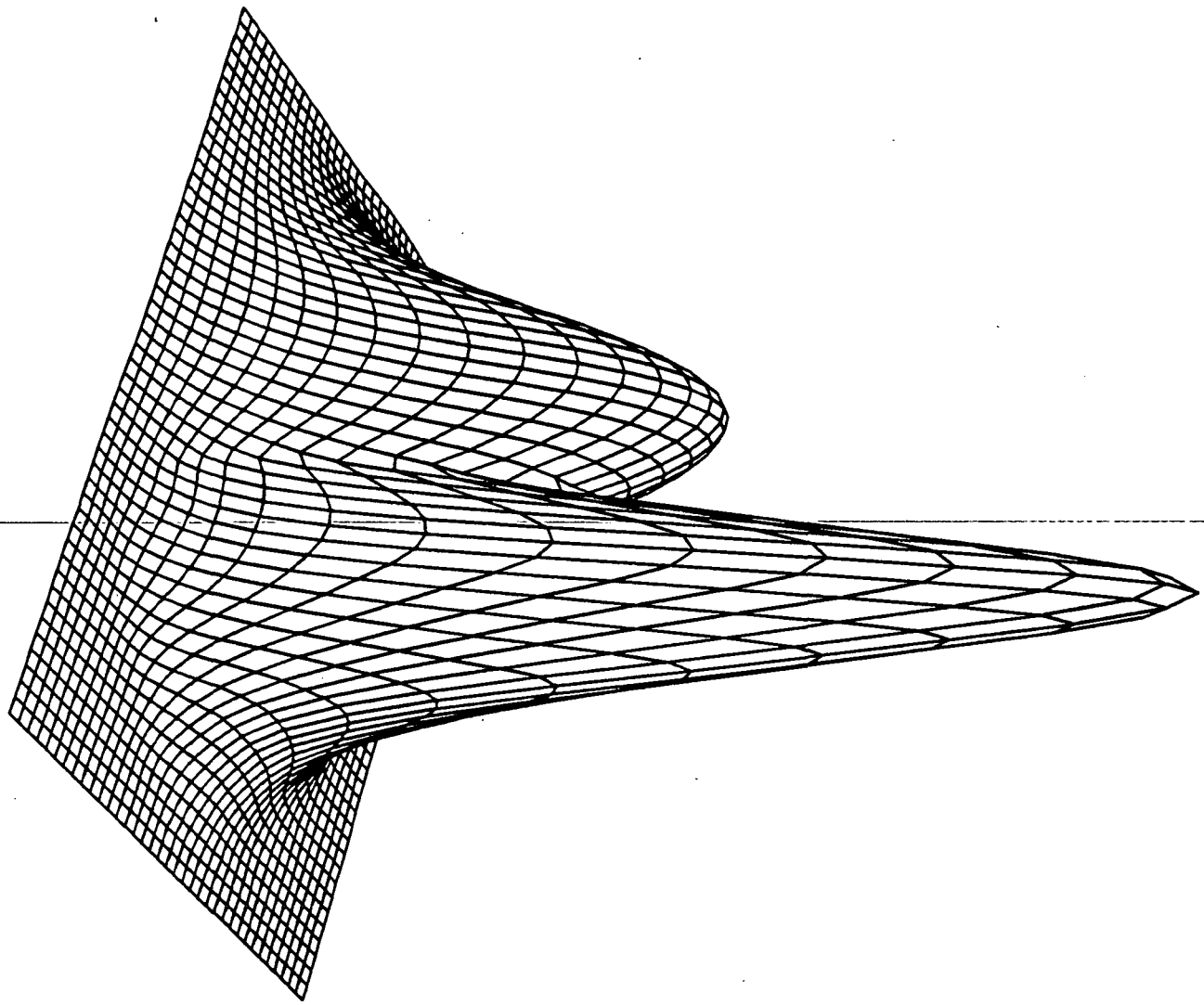


Fig. 6. The true densities for Example 2.

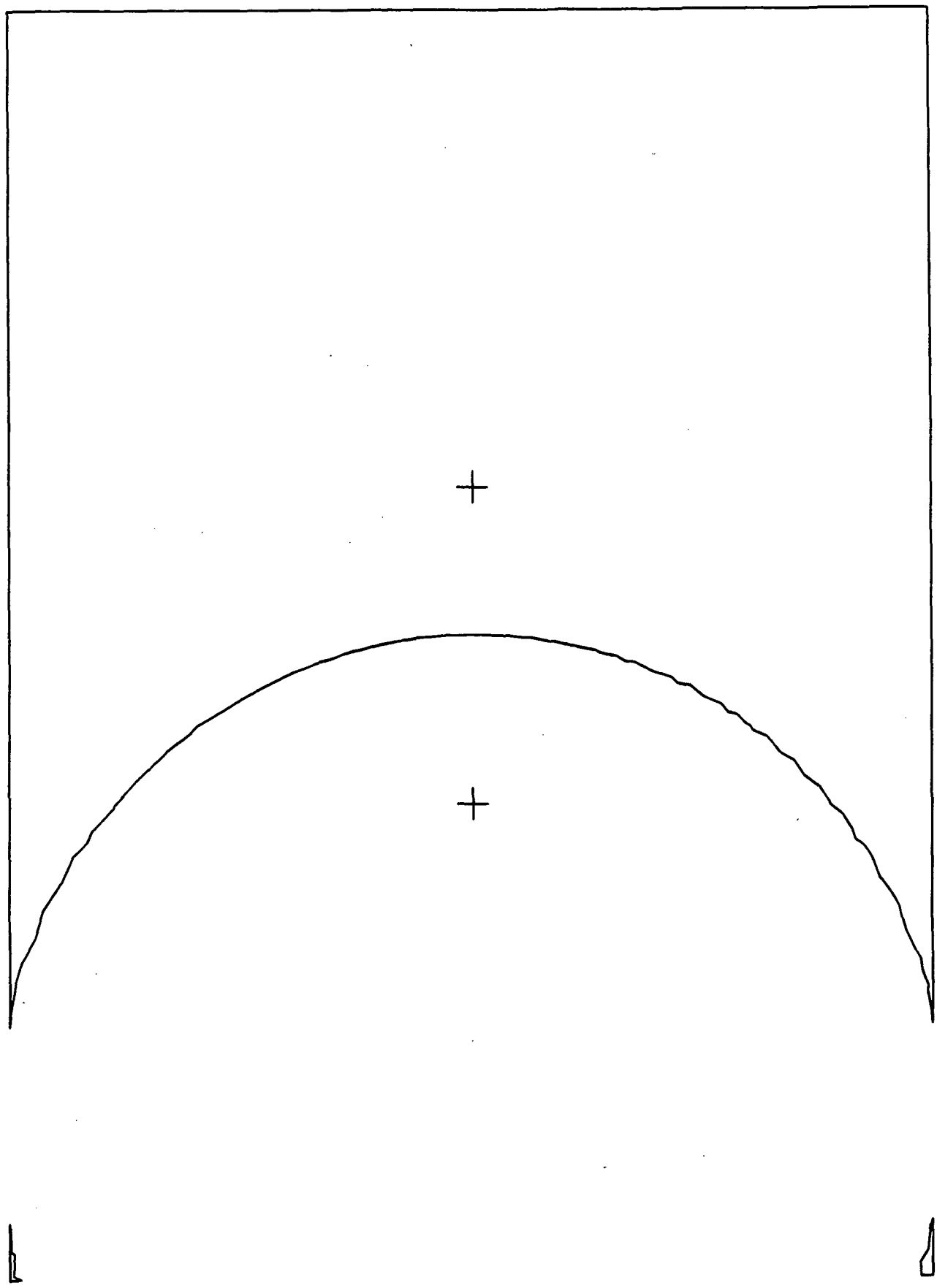


Fig. 7. The true decision regions for Example 2.

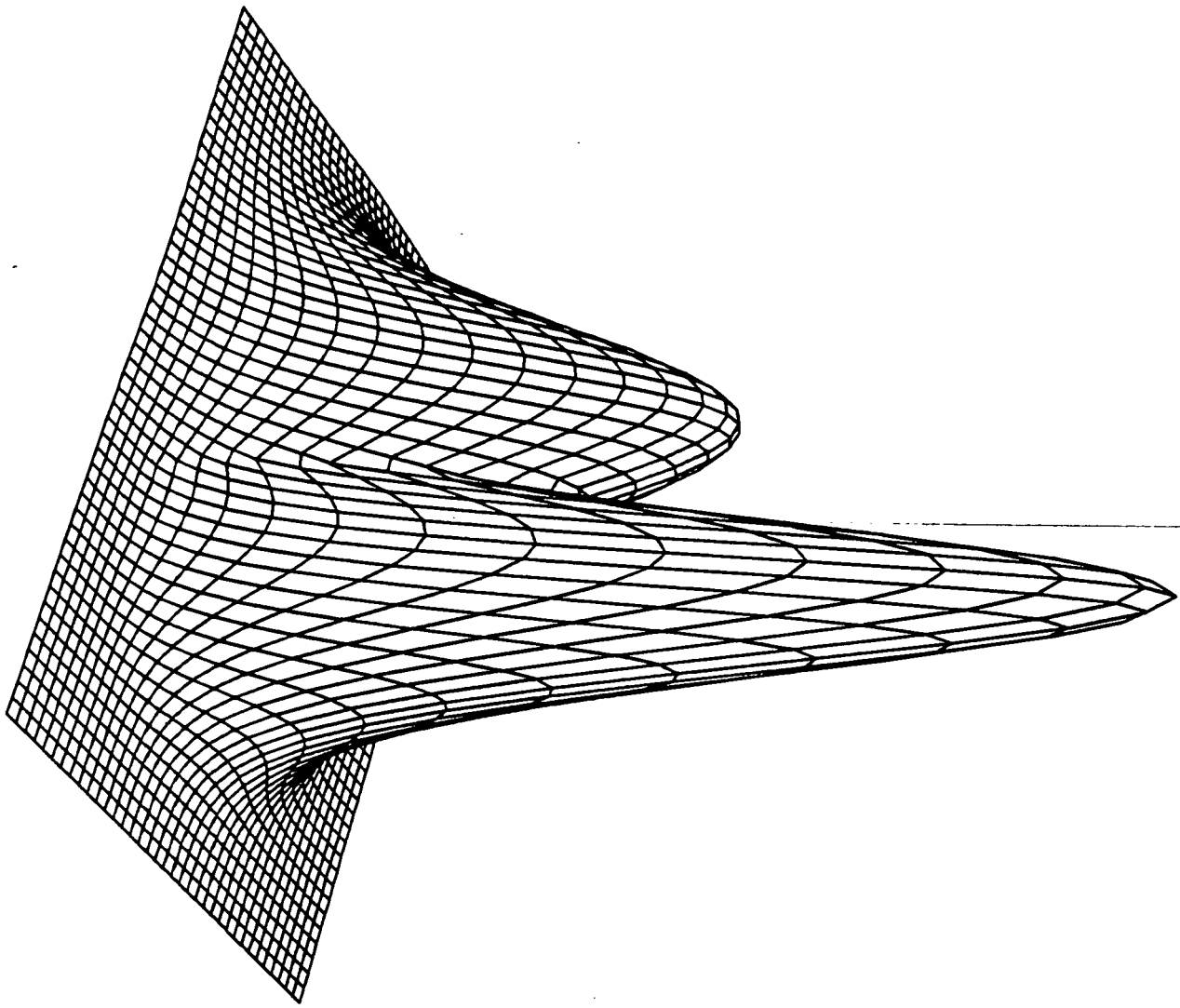


Fig. 8. The spline estimates for the densities of Example 2.

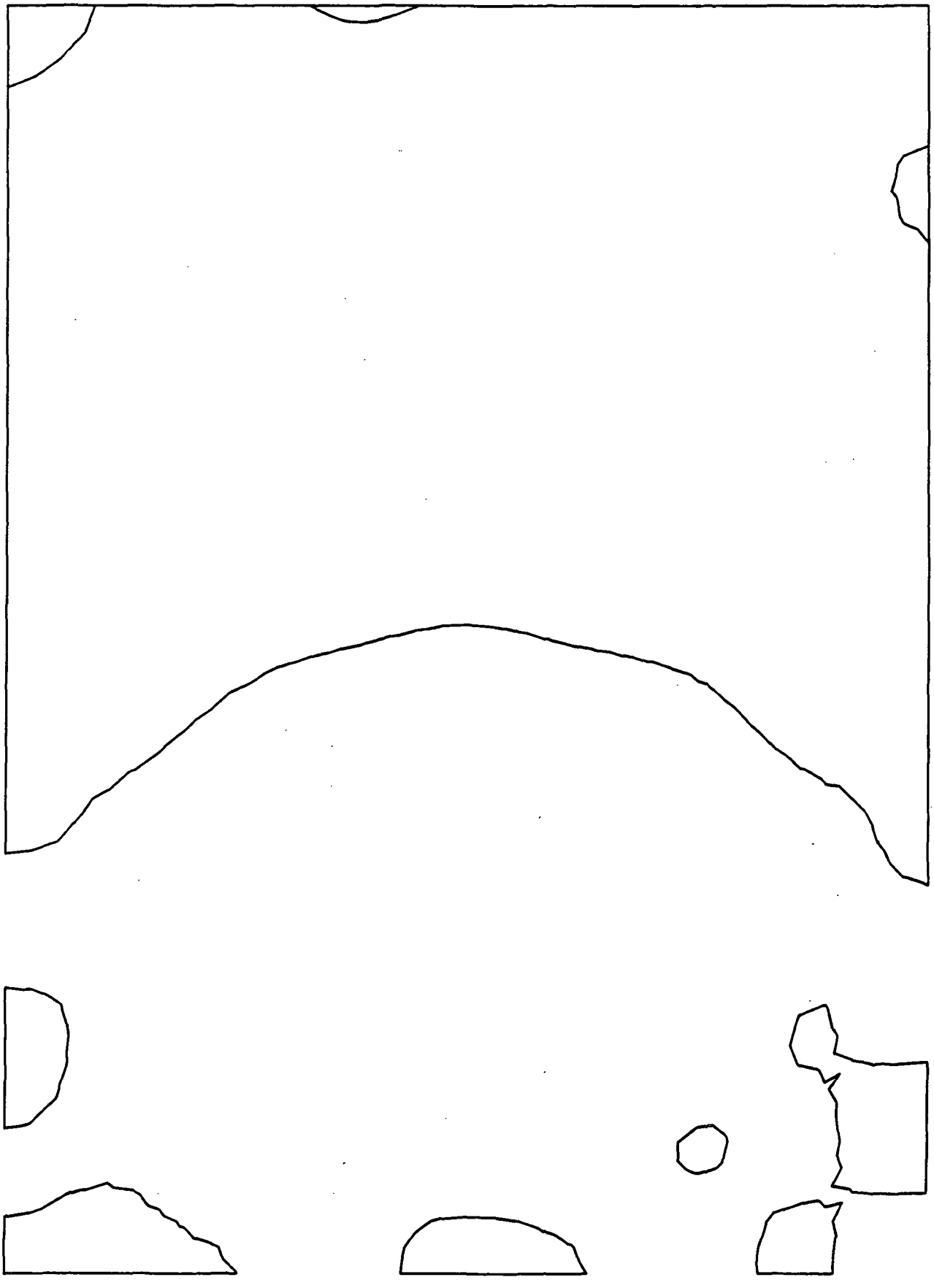


Fig. 9. The estimated decision regions for Example 2.

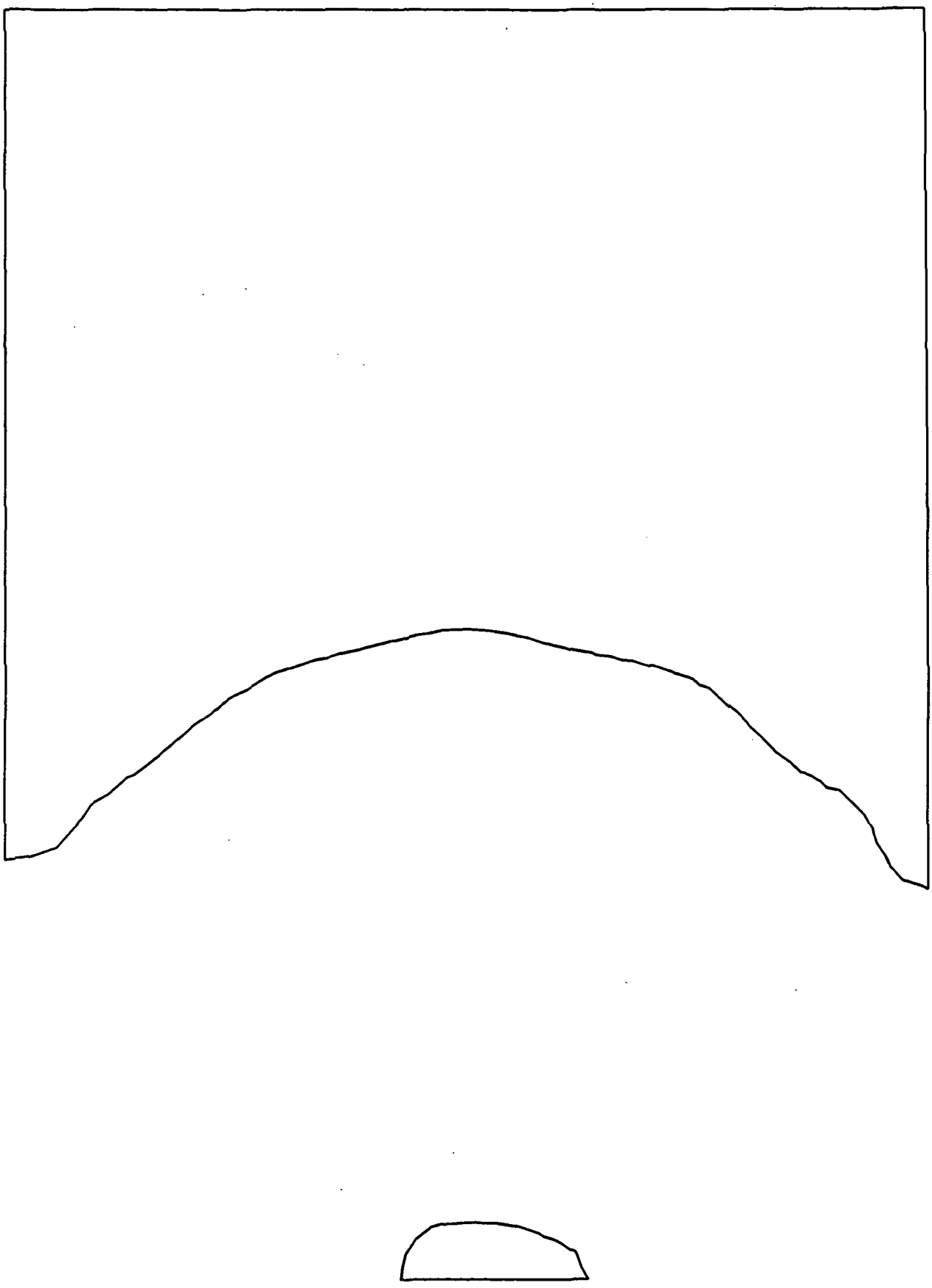


Fig. 10. The estimated decision regions for Example 2 (clutter removed).

EXAMPLE 3

Setup

NC = 3 classes

P_1 = normal density with mean (0,-1) and covariance matrix .5I

P_2 = normal density with mean (0,1) and covariance matrix .5I

P_3 = normal densit with mean (3,0) and covariance matrix I

A-priori probabilities $\alpha_1 = \alpha_2 = \alpha_3 = 1/3$

Data

15,000 random points from each population

Histogram

P_1 and P_2: Equally spaced bins of width 2/3 on the rectangle H = [-3,5]x[-3,3]. Total number of bins = 108

P_3: Equally spaced bins of width 1 on the rectangle H. Total number of bins = 48.

Spline Fit

Using quadratic splines with knots at bin centers
Total number of coefficients = 48 and 108, respectively

Computed PMC

Without clutter removal = 0.08552417
With clutter removal = 0.08537598

```
Jun  3 18:47 1985  hgrams Page 1

 3
 12   9
-3.000000   -2.333333   -1.666667   -1.000000E+00   -3.333333E-01
+3.333334E-01   +1.000000   +1.666667   +2.333333   +3.000000
+3.666667   +4.333333   +5.000000
-3.000000   -2.333333   -1.666667   -1.000000E+00   -3.333333E-01
+3.333334E-01   +1.000000   +1.666667   +2.333333   +3.000000
    0.     0.     2.     1.     0.     0.     0.     0.     0.
    2.    16.    43.    47.    16.     3.     0.     0.     0.
   24.   141.   346.   336.   136.    26.     1.     1.     0.
   96.   517.  1231.  1187.   523.   103.     4.     1.     0.
  146.   763.  1839.  1801.   784.   158.    15.     0.     0.
   98.   514.  1130.  1168.   500.   107.    11.     0.     0.
   30.   145.   321.   294.   154.    32.     2.     1.     0.
    5.    19.    36.    50.    24.     3.     0.     0.     0.
    0.     1.     1.     2.     3.     0.     0.     0.     0.
    0.     0.     0.     0.     0.     0.     0.     0.     0.
    0.     0.     0.     0.     0.     0.     0.     0.     0.
    0.     0.     0.     0.     0.     0.     0.     0.     0.
 12   9
-3.000000   -2.333333   -1.666667   -1.000000E+00   -3.333333E-01
+3.333334E-01   +1.000000   +1.666667   +2.333333   +3.000000
+3.666667   +4.333333   +5.000000
-3.000000   -2.333333   -1.666667   -1.000000E+00   -3.333333E-01
+3.333334E-01   +1.000000   +1.666667   +2.333333   +3.000000
    0.     0.     0.     1.     1.     0.     6.     1.     1.
    0.     0.     0.     7.    25.    36.    43.    18.     4.
    0.     0.     2.    18.   145.   331.   337.   153.    21.
    0.     0.     9.   111.   563.  1161.  1186.   476.   110.
    0.     0.    15.   141.   808.  1716.  1810.   812.   153.
    0.     0.     9.    99.   517.  1173.  1124.   516.    87.
    0.     0.     1.    23.   156.   360.   361.   146.    32.
    0.     0.     0.     5.    17.    38.    56.    19.     5.
    0.     0.     0.     0.     2.     0.     4.     0.     1.
    0.     0.     0.     0.     0.     0.     0.     0.     0.
    0.     0.     0.     0.     0.     0.     0.     0.     0.
    0.     0.     0.     0.     0.     0.     0.     0.     0.
 8   6
-3.000000   -2.000000   -1.000000   +0.000000E+00   +1.000000
+2.000000   +3.000000   +4.000000   +5.000000
-3.000000   -2.000000   -1.000000   +0.000000E+00   +1.000000
+2.000000   +3.000000
    0.     0.     0.     0.     0.     0.
    0.     0.     1.     0.     0.     0.
    0.     5.    10.     3.     5.     1.
    6.    39.   107.   115.    45.     5.
   43.   261.   690.   738.   279.    44.
   94.   684.  1842.  1768.   685.   133.
  107.   676.  1698.  1768.   674.   117.
   41.   271.   653.   668.   294.    41.
```

TABLE 3. The data for Example 3.

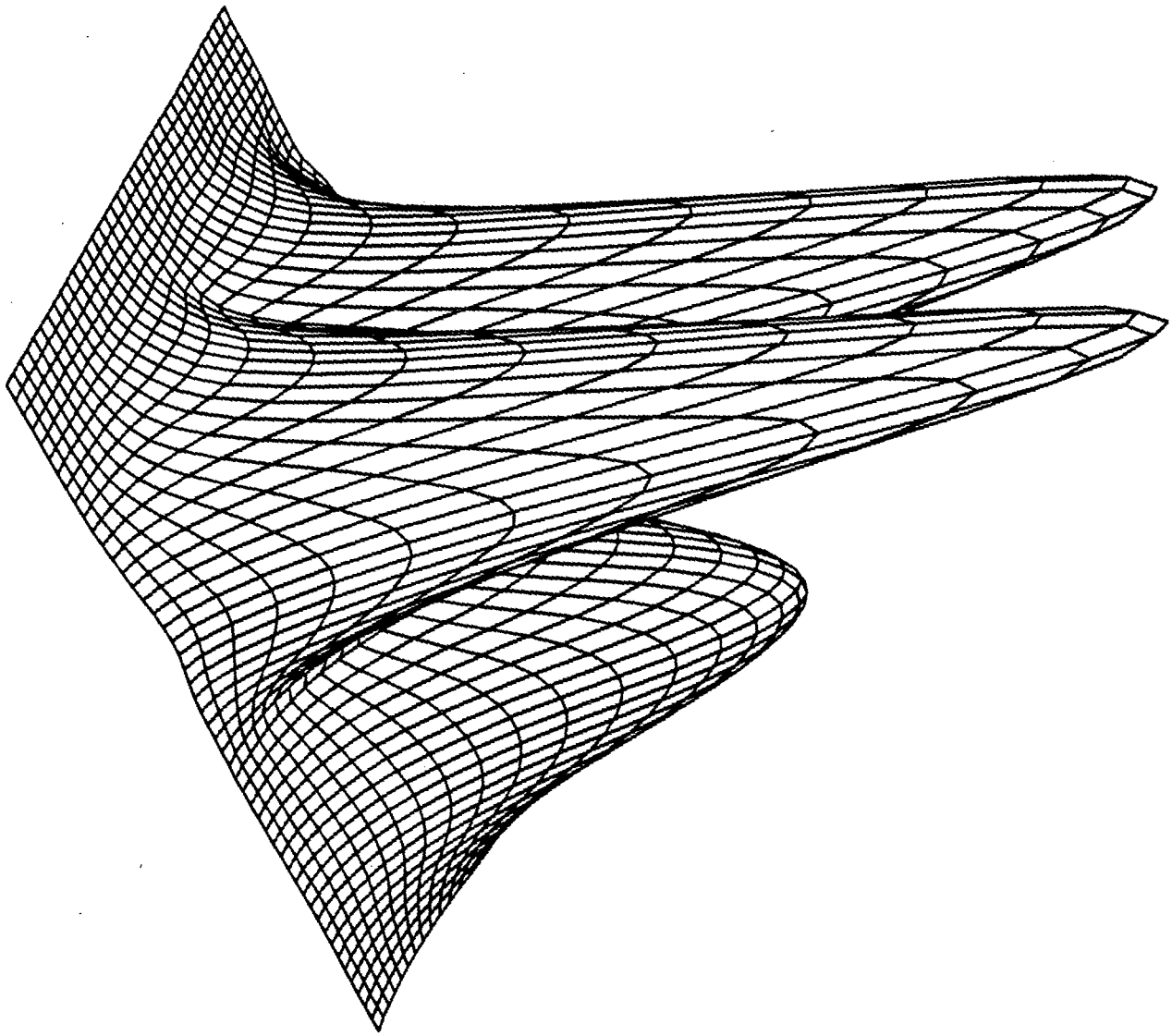


Fig. 11. The true densities for Example 3.

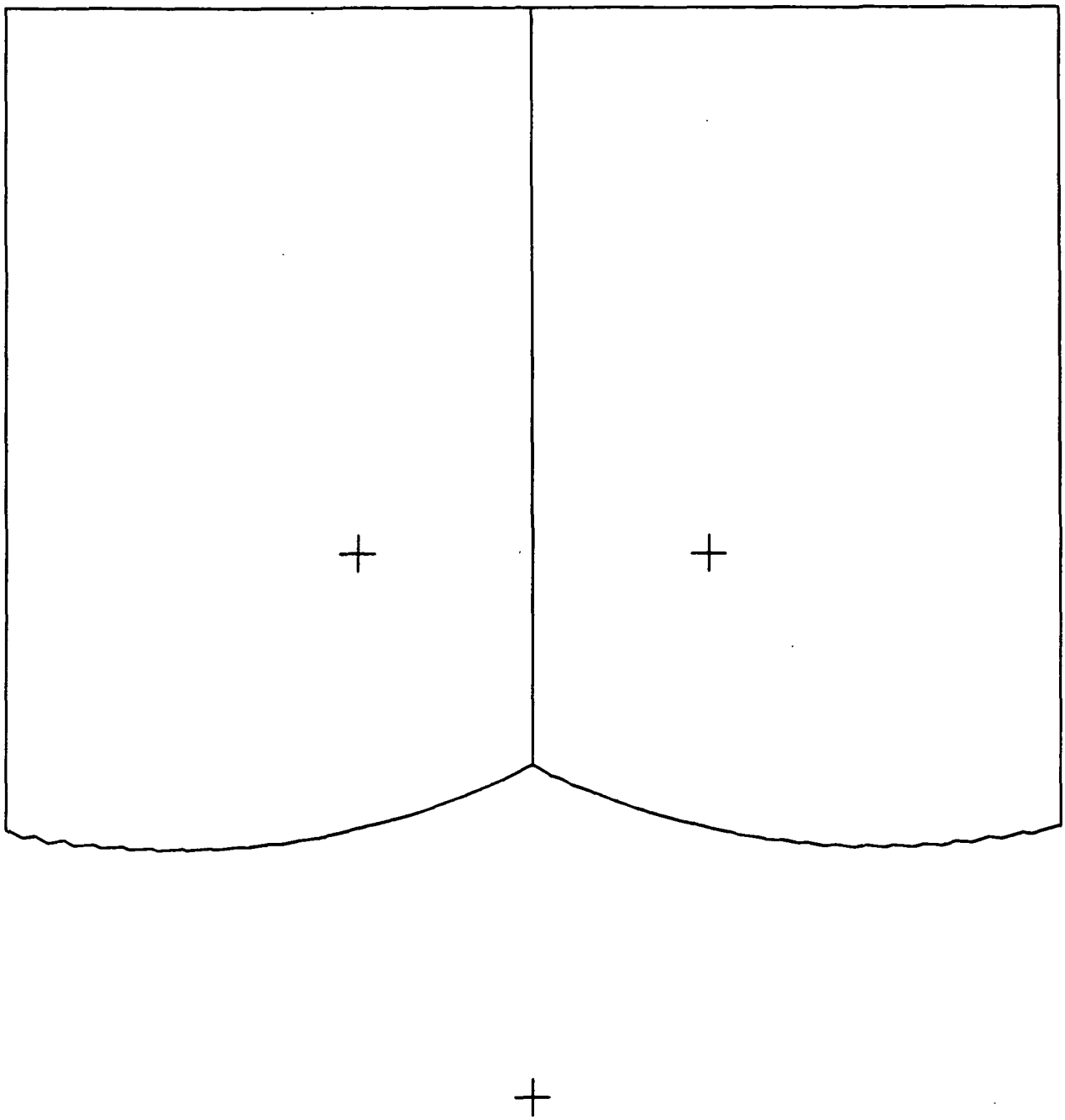


Fig. 12. The true decision regions for Example 3.

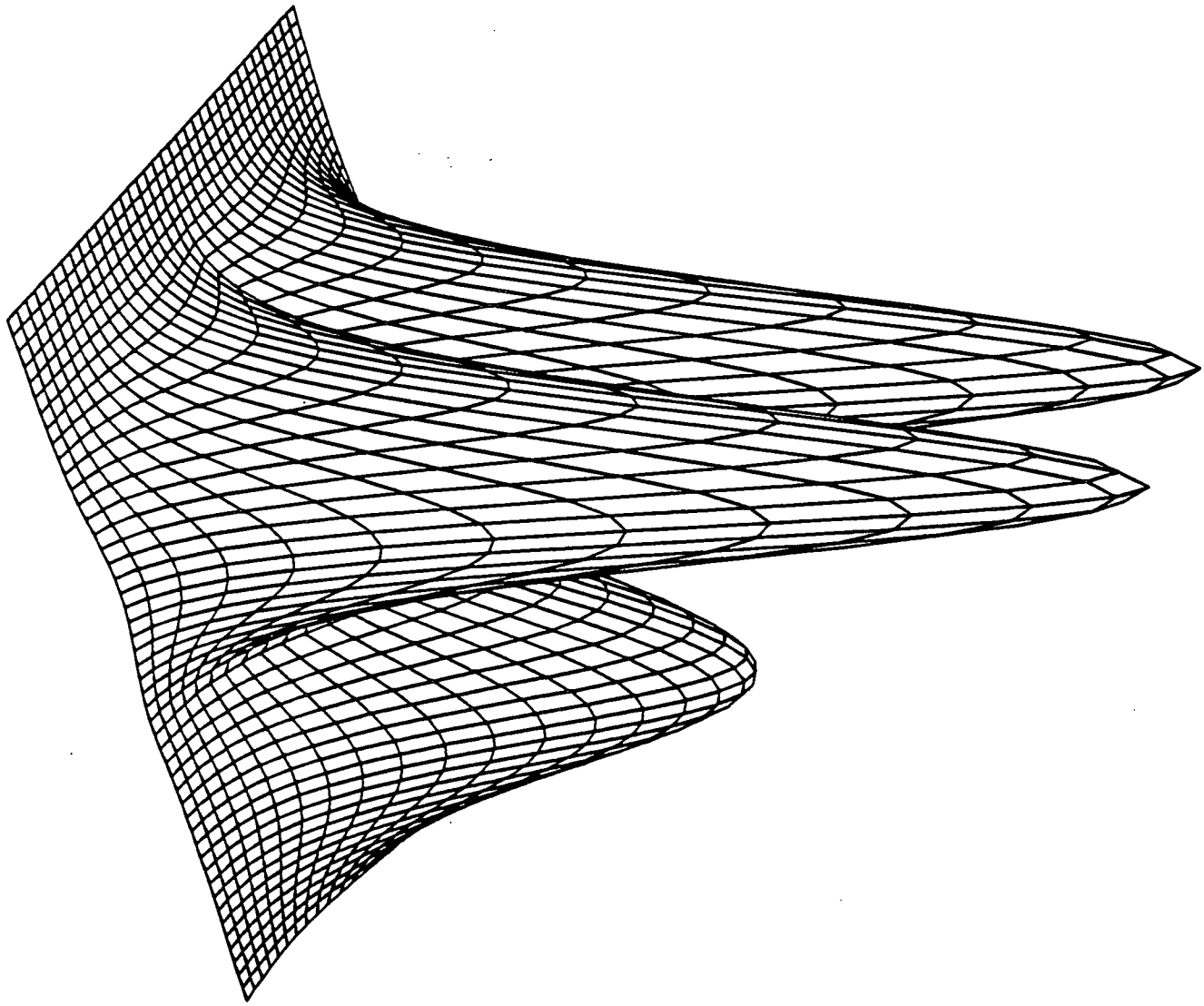


Fig. 13. The spline fit to the densities of Example 3.

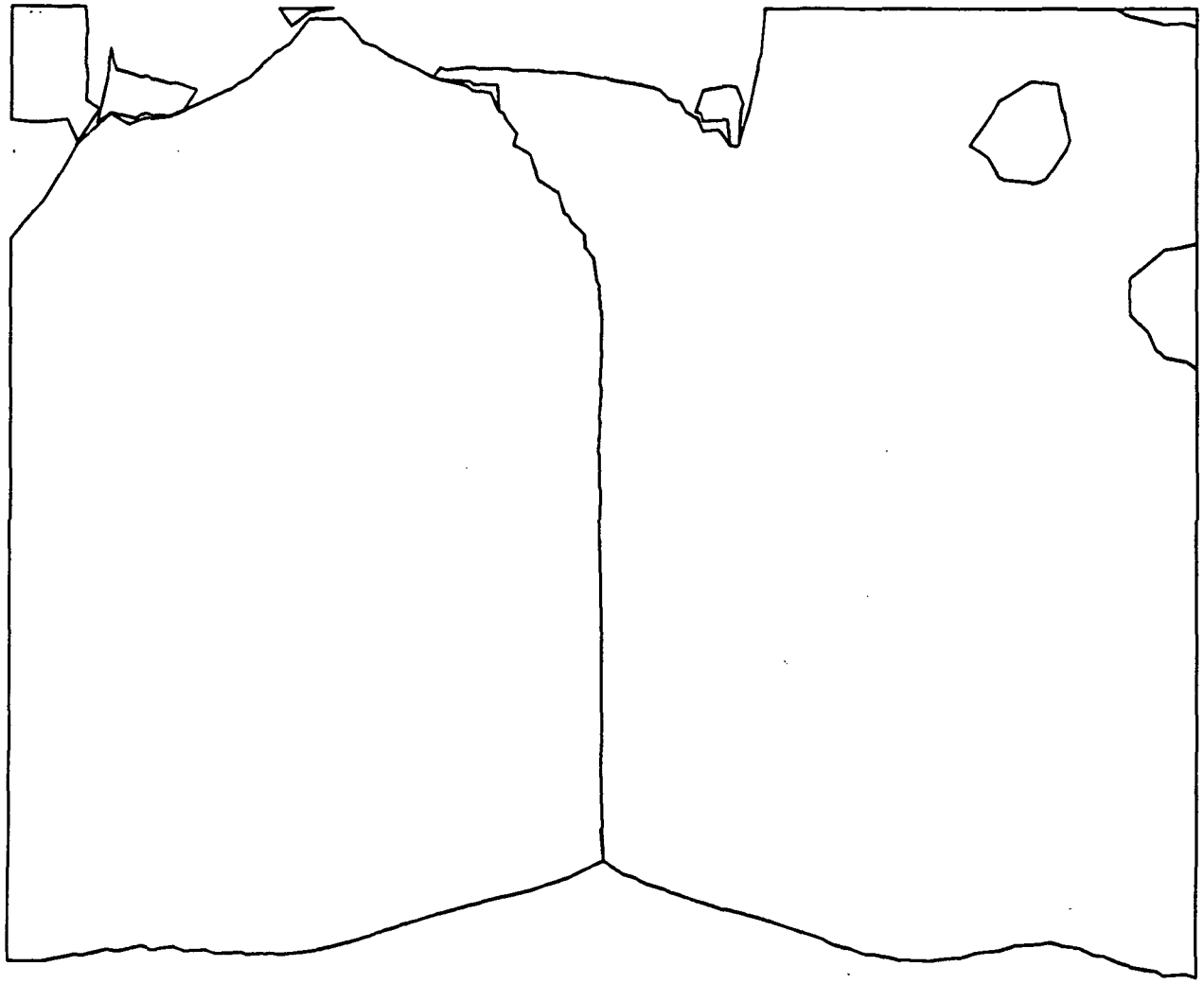


Fig. 14. The estimated decision regions for Example 3.

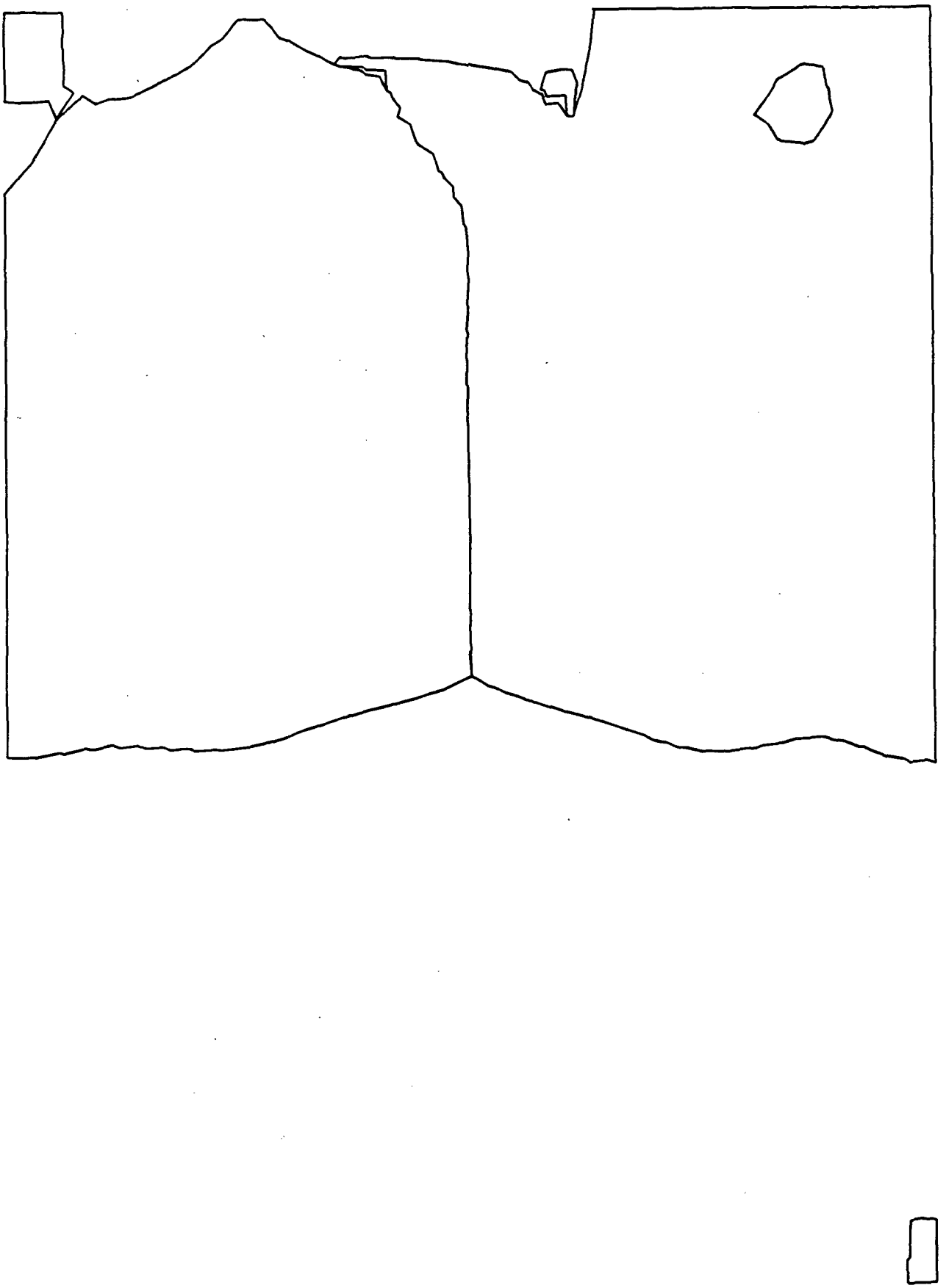


Fig. 15. The estimated decision regions for Example 3 (clutter removed).

REFERENCES

1. Anderson, T. W., An Introduction to Multivariate Statistical Analysis, Wiley, New York, 1958.

2. Andrews, H. C., Introduction to Mathematical Techniques in Pattern Recognition, Wiley-Interscience, New York, 1972.

3. Bedau, K. D., Darstellung und Fortschreibung von Einkommenschichtungen unter Verwendung von Spline-Funktionen, Vierteljahrshefte zur Wirtschaftsforschung (1969), 406-425.

4. Bennett, J. W., Estimation of multivariate probability density functions using B-splines, Ph.D. dissertation, Rice Univ., 1974.

5. Bennett, J. W., de Figueiredo, R. J. P. and Thompson, J. R., Classification by means of B-spline potential functions with application to remote sensing, Rice Report, 1974.

6. Boneva, L., Kendall, D., and Stefanov, I., Spline transformations: Three new diagnostic aids for the statistical data-analyst, J. Royal Stat. Soc. B, 33 (1971), 1-77.

7. de Boor, C., Appendix to Splines and histograms by I. J. Schoenberg, in Spline Functions and Approximation Theory, A. Meir and A. Sharma, eds., ISNM 21, Birkhauser, Basel, 1973, 329-358.

8. de Boor, C., A Practical Guide to Splines, Springer-Verlag, New York, 1978.

9. de Boor, C., Efficient computer manipulation of tensor products, ACM Trans. Math. Software 5 (1979), 173-182.

10. de Boor, C. and Schumaker, L. L., On calculating with B-splines, II. Integration, in Numerische Methoden der Approximations Theorie, Band 3, ISNM Vol. 30, Birkhauser, 1976, 123-146.

11. Cacoullos, T., Estimation of a multivariate density, Annals Inst. Stat. Math. 18 (1966), 179-189.

12. Guseman, L. F., Jr., Peters, C., and Walker, H. F., On minimizing the probability of misclassification for linear feature selection, The Annals of Statist. 3 (1975), 661-668.

13. Guseman, L. F., Jr. and Schumaker, L. L., Spline Classification Methods, Proceedings of the NASA Symposium on Math. Pattern Recog. & Image Anal., L. F. Guseman, Jr., ed., 1983.

14. Guseman, L.F., Jr., and L.L. Schumaker, Multivariate spline methods in surface fitting, in Proceedings of the 2nd NASA Symposium on Math. Pattern Recog. & Image Anal., L.F. Guseman, Jr., ed., 1984, 137-163.

15. Guseman, L. F., Jr., and Walker, H. F., On minimizing the probability of misclassification for linear feature selection: A computational procedure, in The Search for Oil, D. B. Owens, ed., Marcel Dekker, New York, 1975, 61-81.

16. Lennington, R. K. and Rassbach, M. E., CLASSY - An adaptive maximum likelihood clustering algorithm, Report LEC 12145, Lockheed Electronics, 1978.

17. Lyche, T. and Schumaker, L. L., Finding the zeros of spline functions, manuscript.

18. Marsaglia, G., One sided approximations by linear combinations of functions, in Approximation Theory, A. Talbot, ed. Academic Press, New York, 1970, 233-242.

19. Schoenberg, I. J., Splines and histograms, in Spline functions and Approximation Theory, A. Meir and A. Sharma, eds., ISNM 21, Birhauser, Basel, 1972, 277-327.

20. Schumaker, L. L., Spline Functions: Basic Theory, Wiley-Interscience, New York, 1981.

21. Schumaker, L. L. , Fitting surfaces to scattered data, in Approximation Theory II, C. K. Chui, L. L. Schumaker and G. G. Lorentz, eds., Academic Press, N. Y., 203-268.

22. Scott, D. W., Multivariate density estimation and remote sensing, Proceedings of the NASA Symposium on Math. Pattern Recog. & Image Anal., L. F. Guseman, Jr., ed., 1983, 77-92.

23. Silverman, B. W., Estimation for univariate and bivariate data, in Interpreting Multivariate Data, V. Barnett, ed., Wiley, N.Y., 1981, 37-53.

24. Tapia, R. A. and Thompson, J.R., Nonparametric Probability Density Estimation, John Hopkins Univ. Press, Baltimore, 1978.

25. Wolfe, J. H., Pattern clustering by multivariate mixture analysis, Multivariate Behavorial Research 5(1970), 329-350.

26. Yamazaki, H., Nonparametric and parametric estimation of wave statistics and spectra, Ph.D. dissertation, Texas A&M Univ., 1984.

METHODS OF NORMAL MIXTURE ANALYSIS APPLIED TO REMOTE SENSING

BY

Charles Peters
Department of Mathematics
University of Houston

ABSTRACT

This paper concerns parametric mixture models appropriate for data presented in homogeneous blocks of varying sizes from several unidentified source populations. For most applications, the data elements within each block are dependent. Models are proposed for multivariate normal data incorporating two types of dependence, exchangeability of elements within blocks, and a Markov structure for blocks. The consequences of assuming exchangeability, when in fact the Markov structure holds, are explored. Computational problems for each model are considered, and results of a simple test of the exchangeability hypothesis for LANDSAT data are presented.

A Bayesian, or penalized maximum likelihood, approach to the problem of estimating the parameters of a mixture of multivariate normal distributions is proposed. The Bayesian formulation eliminates the problem of singularities in the likelihood function and results in an attractive EM-like procedure. Although the question of consistency is not settled, it is suggested that the proposed method has certain advantages over both the constrained and unconstrained maximum likelihood procedures.

Introduction

The mixture density estimation problem considered in this section may be described as follows. A sample of N independent observations $\Theta_1,\ldots,\Theta_N$ is given, each observation Θ_i consisting of a positive integer n_i (block size) and a $p \times n_i$ matrix

$$X_i = (X_{i1}|\ldots|X_{in_i})$$

whose columns $X_{ij} \in \mathbb{R}^p$ are the basic experimental measurements. Each observation Θ_i comes from one of k populations $\Pi_1,\ldots,\Pi_k$, where k is known but the population of origin of each observation is unknown. Let $q_\ell > 0$ denote the probability that an observation comes from Π_ℓ.

Although the data blocks X_i are independent, the basic measurements X_{ij} within each block are possibly dependent. For applications in remote sensing of agricultural resources, the parameters of primary interest are q_ℓ and $E[n_i|\Pi\ell]$, the mean block size for the ℓth population, where each block is a set of multispectral measurements from a single agricultural field belonging to a single crop class Π_ℓ. The product $q_\ell E[n_i|\Pi_\ell]$ is related to the acreage in the sampling region covered by the class Π_ℓ. The procedures suggested herein are automatic procedures capable of handling large sample sizes N as well as large dimensionality p, with human intervention restricted mainly to a posterior description of classes. It should be possible to modify these procedures, along the lines indicated by Walker [17], to provide for the inclusion of a relatively small number of labelled samples, whose class origins are known, and perhaps to improve upon the estimates of the parameters derived from the labelled samples at

a relatively small additional cost.

Let the observations be generically denoted by $\Theta = (n, X)$ and let $f(n, x \mid \Pi_\ell)$ be the density function of Θ, given that Θ comes from Π_ℓ. Let $f(x \mid n, \Pi_\ell)$ be the density function of X, given n and given that Θ comes from Π_ℓ, and let $f(n \mid \Pi_\ell)$ be the density of n given population Π_ℓ. The mixture density for Θ is

$$(1.1) \qquad f(n, x) = \sum_{\ell=1}^{k} q_\ell f(n, x \mid \Pi_\ell)$$

$$= \sum_{\ell=1}^{k} q_\ell f(n \mid \Pi_\ell) f(x \mid n, \Pi_\ell).$$

and the log likelihood for the sample is

$$(1.2) \qquad L = \sum_{i=1}^{N} \log \sum_{\ell=1}^{k} q_\ell f(n_i \mid \Pi_\ell) f(x_i \mid n_i, \Pi_\ell).$$

We shall assume particular parametric forms for $f(n \mid \Pi_\ell)$ and $f(x \mid n, \Pi_\ell)$ which are simple enough that they are estimable from (1.2). In particular, we shall consider multivariate normal forms for $f(x \mid n, \Pi_\ell)$ which incorporate either exchangeability of observations within blocks or a first order autoregressive covariance structure. The consequences of the exchangeability hypothesis are presented in some detail, and the possibility of approximating the autoregressive form by exchangeability is considered. Finally, we present the results of a simple test of exchangeability for LANDSAT data.

Two Covariance Hypotheses

Throughout the remainder of this paper it will be assumed that $f(x \mid n, \Pi_\ell)$ is a pxn-variate normal density function. To simplify notation, let $Y = (Y_1 | \ldots | Y_n)$ be a random p x n matrix having density $f(x \mid n, \Pi_\ell)$. We assume that the column process $Y_1, \ldots, Y_n$ of Y is stationary with unknown mean $\mu_{n\ell}$ and covariance function $\Gamma_{n\ell}(h) = \mathrm{cov}(Y_j, Y_{j+h})$. Next to independence, the simplest assumption about $\Gamma_{n\ell}(h)$ is the exchangeability hypothesis that Y and YW have the same distribution for each n x n permutation matrix W (to denote this we write $Y \overset{d}{=} YW$). In terms of $\Gamma_{n\ell}$, the exchangeability hypothesis can be formally expressed as

$$E : \Gamma_{n\ell}(h) = \begin{cases} \Sigma_{n\ell} & \text{if } h \neq 0 \\ \psi_{n\ell} + \Sigma_{n\ell} & \text{if } h = 0 \end{cases}$$

for some (unspecified) symmetric p x p matrices ψ_n and $\Sigma_{n\ell}$ satisfying the conditions that $\psi_{n\ell}$ and $\psi_{n\ell} + n\Sigma_{n\ell}$ are positive definite.

Experiments in image texture generation [13] and studies of spatial correlation in LANDSAT images [5] suggest that the correlation of data elements as a function of spatial separation might be modeled as an autoregressive process of low order. Accordingly, as an alternative to (E), we are led to consider the hypothesis (M) that $\Gamma_{n\ell}(h)$ has a first order autoregressive, or Markov, structure.

$$M : \Gamma_{n\ell}(h) = \Omega_{n\ell}^{\frac{1}{2}} A_{n\ell}^{|h|} \Omega_{n\ell}^{\frac{1}{2}},$$

for some unspecified positive definite $p \times p$ matrix $\Omega_{n\ell}$ and symmetric $p \times p$ matrix A with spectral radius less than one.

The theorems stated below exhibit some consequences of the exchangeability hypothesis which are of importance in computation and in testing the hypothesis. J_n denotes the vector $(1, 1, \ldots, 1)^T_{1 \times n}$, while I_n denotes the $n \times n$ identity matrix. Λ'_n denotes the group of $n \times n$ orthogonal matrices W such that $WJ_n = J_n$.

Theorem 1: If Y is a normally distributed $p \times n$ matrix whose distribution satisfies (E) then $YW \overset{d}{=} Y$ for each member of Λ'_n. If P is an $n \times (n-1)$ matrix satisfying $P^TP = I_{n-1}$ and $P^TJ_n = 0$, then $Z = YP$ has columns $Z_1, \ldots, Z_{n-1}$ which are independently distributed as $N_p(0, \psi_{n\ell})$. The statistics $\bar{Y} = \frac{1}{n} \sum_{i=1}^{n} Y_i$ and $S = \sum_{i=1}^{n} (Y_i - \bar{Y})(Y_i - \bar{Y})$ are independent, $\bar{Y}$ is normal $N_p(\mu_{n\ell}, \Sigma_{n\ell} + \frac{1}{n}\psi_{n\ell})$, and S has the Wishart distribution $W_p(n-1, \psi_{n\ell})$.

As a corollary of Theorem 1, if $n > p + 2$ and (E) is true, then the distribution of

$$F = \frac{n - p - 2}{p} \, Z_1^T \left(\sum_{j=2}^{n-1} Z_i \, Z_j^T \right) Z_1$$

is central $F_{p, n-p-2}$. This observation is used as a simple test of (E) described in a later section. It is interesting to note that the distribution of F does not depend essentially on the normality of Y. Using results of A.P. Dawid [7] it can be shown that if Y is any random $p \times n$ matrix such that $YW \overset{d}{=} Y$ for each $W \in \Lambda'_n$, and $\sum_{j=2}^{n-1} Z_i Z_j^T$ is

almost surely positive definite, where Z is defined in Theorem 1, then F has the $F_{p,\ n-p-2}$ distribution. Therefore the test based on F is a distribution free test for the invariance of the distribution of Y under right multiplication by elements of Λ_n'.

By writing out the density of Y under (E) it is easy to see that $(\bar{Y},\ S)$ is sufficient for the family of all normal distributions satisfying exchangeability. Under very mild restrictions the sufficiency of $(\bar{Y},\ S)$ implies (E). Thus, unless (E) holds for all source populations Π_ℓ, some loss of estimation accuracy in the parameters of primary interest (q_ℓ and $E[n_i \mid \Pi_\ell]$) in the mixture model is to be expected when the data within blocks is condensed to block means and scatters.

Theorem 2: Let $\mathcal{F}$ be a family of normal distributions of a $p \times n$ matrix Y and suppose that some member of $\mathcal{F}$ satisfies (E). If $(\bar{Y},\ S)$ is sufficient for $\mathcal{F}$, then (E) holds for each member of $\mathcal{F}$.

Approximating the Markov Structure by Exchangeability

Even if the Markov assumption is more appropriate for applications, the computations involved in estimating the mixture parameters are very much simpler if exchangeability is assumed. In this section we will show that approximating the Markov form by exchangeability leads to certain conclusions about the dependence on n of the covariance parameters $\psi_{n\ell}$ and $\Sigma_{n\ell}$ of (E).

Let $f(y)$ be the normal density of a $p \times n$ matrix Y whose columns satisfy the Markov assumption with mean μ and convariance function

$\Gamma(h) = \Omega^{\frac{1}{2}} A |h|_{\Omega}^{\frac{1}{2}}$. Let $\hat{f}(y)$ be a normal density satisfying (E) with column mean $\hat{\mu}$ and covariance function

$$\hat{\Gamma}(h) = \begin{cases} \hat{\Sigma} & h \neq 0 \\ \hat{\Sigma} + \hat{\alpha} & h = 0 \end{cases},$$

The degree to which $\hat{f}$ approximates f is measured by the relative entropy

$$H(\hat{f}, f) = \int_{\mathbb{R}^{pn}} f(y) \log \frac{f(y)}{\hat{f}(y)} \, dy \quad .$$

The relationship between this criterion and the L_1 distance, which might be considered more meaningful, is not very clear. The sharpest relationship we have been able to find is given in the next theorem. A corollary of the theorem is that if $H(\hat{f}_j, f) \to 0$ then $\int_{\mathbb{R}^{pn}} |\hat{f}_j - f| \to 0$, a result proved by Geman [11].

Theorem 3: Let $\hat{f}$ and f be arbitrary density functions on $\mathbb{R}^m$. For each $\epsilon > 0$,

$$\frac{1}{2} \int_{\mathbb{R}^m} |\hat{f}(y) - f(y)| \, dy \leq \epsilon + \frac{\epsilon}{\epsilon - \log(1 + \epsilon)} H(\hat{f}, f) \, .$$

It is straightforward to show that if expectations are taken with respect to the true density f, then

(3.1) $$E(\overline{Y}) = \mu,$$

$$\operatorname{cov}(\overline{Y}) = \frac{1}{n} \Omega^{\frac{1}{2}} B \, \Omega^{\frac{1}{2}} ,$$

and $\quad E(S) = n\Omega - \Omega^{\frac{1}{2}} B\, \Omega^{\frac{1}{2}}$,

where $\quad B = (I - A)^{-1}(I + A) - \frac{2}{n}(I - A)^{-2}A(I - A^n)$.

The log-likelihood for the density $\hat{f}$ is

$$\log \hat{f}(y) = -\frac{n-1}{2} \log|\hat{\psi}| - \frac{1}{2} \log|\hat{\psi} + n\hat{\Sigma}|$$

$$- \frac{1}{2} \operatorname{tr}\hat{\psi}^{-1}S - \frac{n}{2} \operatorname{tr}(\hat{\psi} + n\hat{\Sigma})^{-1}(\bar{Y} - \hat{\mu})(\bar{Y} - \hat{\mu})^T$$

The parameters which maximize the expectation, with respect to f, of $\log \hat{f}(y)$ are

$$\hat{\mu} = E(\bar{Y})$$

$$\hat{\psi} = \frac{1}{n-1} E(S)$$

$$\hat{\Sigma} = \operatorname{cov}(\bar{Y}) - \frac{1}{n(n-1)} E(S).$$

Combining these equations with equations (3.1), and replacing $\hat{\Sigma}$ by the new parameter $\hat{R} = \hat{\psi} + n\hat{\Sigma} = n \operatorname{cov}(\bar{Y})$ we have

Theorem 4: $H(\hat{f}, f)$ is minimized when

$$\hat{\mu} = \mu$$

$$\hat{\psi} = \frac{n}{n-1}\Omega - \frac{1}{n-1}\Omega^{\frac{1}{2}} B\, \Omega^{\frac{1}{2}}$$

$$\hat{R} = \Omega^{\frac{1}{2}} B\, \Omega^{\frac{1}{2}} \quad ,$$

where $B = (I - A)^{-1}(I + A) - \frac{2}{n} A (I - A)^{-2}(I - A^n)$.

Although it is not obvious, these parameters satisfy the required constraints; that is, $\hat{\psi}$ and $\hat{R}$ are positive definite. As $n \to \infty$, $\hat{R}$ and $\hat{\psi}$ tend to constants. This implies that $\hat{\Sigma}$ is $O(\frac{1}{n})$ for large n. We will make use of this observation in the next section.

The maximum value of $E[\log \hat{f}(Y)]$ is

$$- \frac{n - 1}{2} \log|\hat{\psi}| - \frac{1}{2} \log|\hat{R}| - \frac{np}{2} ,$$

where $\hat{\psi}$ and $\hat{R}$ are given in Theorem 4.

For large values of n this is approximately

$$- \frac{n}{2} \log|\Omega| - \frac{1}{2} \log|(I - A)^{-1}(I + A)| - \frac{np}{2} .$$

Since

$$E[\log f(Y)] = - \frac{n}{2} \log|\Omega| - \frac{n - 1}{2} \log|I - A^2| - \frac{np}{2} .$$

we have the following expression, for large values of n, for the minimum entropy:

$$H(\hat{f}, f) \approx - \frac{n}{2} \log|I - A^2| .$$

Estimating the Mixture Parameters

The most successful method for estimating the parameters in a mixture of distributions from a single exponential family is maximum likelihood [16]. When the component distributions of the mixture are parametrized

in the right way, the EM procedure has a very natural and easily implemented formulation [16], [9]. For density functions $f(x \mid n, \Pi_\ell)$ corresponding to the Markov assumption the likelihood equations for the mixture parameters are extremely complicated, and there is no obvious alternative to using a standard optimization procedure to maximize the likelihood function. There are difficulties involved in obtaining exact maximum likelihood estimates with a sample sequence from a single autoregressive series (see [10, p.329] and [2]), and it is reasonable to think that these problems will be compounded in the mixture setting proposed, resulting in multiple solutions, slow convergence, etc. In general, the situation when $f(x \mid n, \Pi_\ell)$ satisfies the exchangeability condition is not much better; however, the special case wherein $\Sigma_{n\ell} = \frac{1}{n}\Sigma_\ell$ and $\psi_{n\ell} = \psi_\ell$, and Σ_ℓ and ψ_ℓ are independent of n, is amenable to solution by the EM procedure. For large values of n these assumptions are consistent with the remarks at the end of the last section, if the Markov assumption holds with parameters independent of n.

Let each $f(x \mid n, \Pi_\ell)$ have the form (E) with mean $\mu_{n\ell} = \mu_\ell$ and covariance parameters $\psi_{n\ell} = \psi_\ell$, $\Sigma_{n\ell} = \frac{1}{n}\Sigma_\ell$. Define $R_\ell = \psi_\ell + \Sigma_\ell$. Then $\frac{1}{n} R_\ell$ is the covariance matrix of the column-mean $\bar{X}$ of an observed block of measurements given that the observation comes from Π_ℓ and given the block size n. Suppose the density $f(n \mid \Pi_\ell)$ is from an exponential family

$$f(n \mid \Pi_\ell) = C(\lambda_\ell)h(n)e^{F(\lambda_\ell)t(n)} \qquad n = 1, 2, \ldots$$

where the parameter λ_ℓ is the expected value of $t(n)$ under $f(n \mid \Pi_\ell)$,

[4]. From (1.1) and (1.2) the derivative of the log-likelihood with respect to λ_ℓ is

$$(4.1) \qquad \frac{\partial L}{\partial \lambda_\ell} = \sum_{i=1}^{N} \frac{q_\ell f(n_i, X_i|\Pi_\ell)}{f(n_i, X_i)} \left[\frac{C'(\lambda_\ell)}{C(\lambda_\ell)} + F'(\lambda_\ell) t(n_i) \right] .$$

By differentiating the equation

$$\sum_{n} C(\lambda_\ell) h(n) e^{F(\lambda_\ell) t(n)} = 1$$

with respect to λ_ℓ, one sees that

$$\frac{C'(\lambda_\ell)}{C(\lambda_\ell)} = - F'(\lambda_\ell) \lambda_\ell$$

(see [4]). Hence $\frac{\partial L}{\partial \lambda_\ell} = 0$ if and only if

$$(4.2) \qquad \lambda_\ell = \sum_{i=1}^{N} \frac{f(n_i, X_i|\Pi_\ell)}{f(n_i, X_i)} t(n_i) \Big/ \sum_{i=1}^{N} \frac{f(n_i, X_i|\Pi_\ell)}{f(n_i, X_i)} .$$

Similarly, by considering $\frac{\partial L}{\partial q_\ell}$, one sees that for a maximum of L we must have

$$(4.3) \qquad q_\ell = \frac{1}{N} \sum_{i=1}^{N} \frac{q_\ell f(n_i, X_i|\Pi_\ell)}{f(n_i, X_i)} .$$

Now let $\overline{X}_i$ and S_i be the mean and scatter of the columns of X_i. Then

$$\frac{\partial}{\partial \mu_\ell} f(n_i, \bar{X}_i | \Pi_\ell) = f(n_i, \bar{X}_i | \Pi_\ell) \left[-\frac{n_i}{2} R_\ell^{-1} (\bar{X}_i - \mu_\ell) \right]$$

$$\frac{\partial}{\partial \psi_\ell} f(n_i, \bar{X}_i | \Pi_\ell) = f(n_i, \bar{X}_i | \Pi_\ell) \left[-\frac{n_i - 1}{2} \psi_\ell^{-1} + \frac{1}{2} \psi_\ell^{-1} S_i \psi_\ell^{-1} \right]$$

$$\frac{\partial}{\partial R_\ell} f(n_i, \bar{X}_i | \Pi_\ell) = f(n_i, \bar{X}_i | \Pi_\ell) \left[-\frac{1}{2} R_\ell^{-1} + \frac{n}{2} R_\ell^{-1} (\bar{X}_i - \mu_\ell) (\bar{X}_i - \mu_\ell)^T R_\ell^{-1} \right] .$$

From these equations it follows that the derivatives of L with respect to μ_ℓ, ψ_ℓ and R_ℓ all vanish when

$$(4.4) \quad \mu_\ell = \sum_{i=1}^{N} n_i \frac{f(n_i, \bar{X}_i | \Pi_\ell)}{f(n_i, \bar{X}_i)} \bar{X}_i \Big/ \sum_{i=1}^{N} n_i \frac{f(n_i, \bar{X}_i | \Pi_\ell)}{f(n_i, \bar{X}_i)} ,$$

$$(4.5) \quad \psi_\ell = \sum_{i=1}^{N} \frac{f(n_i, \bar{X}_i | \Pi_\ell)}{f(n_i, \bar{X}_i)} S_i \Big/ \sum_{i=1}^{N} (n_i - 1) \frac{f(n_i, \bar{X}_i | \Pi_\ell)}{f(n_i, \bar{X}_i)} ,$$

$$(4.6) \quad R_\ell = \sum_{i=1}^{N} \frac{f(n_i, \bar{X}_i | \Pi_\ell)}{f(n_i, \bar{X}_i)} n_i (\bar{X}_i - \mu_\ell)(\bar{X}_i - \mu_\ell)^T \Big/ \sum_{i=1}^{n} \frac{f(n_i, \bar{X}_i | \Pi_\ell)}{f(n_i, \bar{X}_i)} .$$

The iterative procedure suggested by equations (4.2)-(4.6), namely, evaluating the right hand sides with the estimates $\lambda_\ell^{(j)}$, $q_\ell^{(j)}$, $\mu_\ell^{(j)}$, $\psi_\ell^{(j)}$, $R_\ell^{(j)}$ at the jth step, to obtain the estimates $q_\ell^{(j+1)}$, $\mu_\ell^{(j+1)}$, $\psi_\ell^{(j+1)}$, $R^{(j+1)}$, at the (j+1)st step, can be shown to be a slightly modified EM procedure (see [16], and [9]).

Testing the Exchangeability Hypothesis

Standard testing procedures for the two covariance hypotheses considered would require large block sizes n_i and a large sample of observations segregated as to block size and type. The remarks at the end of the second section concerning the distribution of the statistic F under the hypothesis (E) suggest a test which is much easier to implement. For the ith block of measurements X_i, let $Z_i = (Z_{i1}|\ldots|Z_{i,\,n_i-1}) = X_iP_i$, where P_i is a $n_i \times (n_i - 1)$ matrix satisfying the conditions given in Theorem 1. Let

$$F_i = \frac{n_i - p - 2}{p} Z_{i1}^T \left(\sum_{j=2}^{n_i - 1} Z_{ij}Z_{ij}^T \right)^{-1} Z_{i1}$$

If (E) holds for all classes then each F_i is distributed as $F_{p,\,n_i-p-2}$. Thus the number of observed blocks for which F_i falls in some given quantile range of its distribution can be tabulated and compared to its expected value. Table 1 shows these comparisons for 216 quasi-fields of LANDSAT agricultural data from LACIE segment 1645 and 57 quasi-fields from LACIE segment 1633. The quasi-fields are those found by an automatic image segmentation program (AMOEBA) and may not be representative of real agricultural fields. The given χ^2 goodness of fit statistics are significant at levels between 10% and 20%. The hypothesis (E) appears to be rather weakly disconfirmed for this data.

TABLE 1 - Disbribution of F-Ratios

Segment 1645 - 216 Fields

Percentiles	0 - 5%	5 - 10%	10 - 90%	90 - 95%	95 - 100%
Number	18	14	163	9	12
Frequency	8.2%	6.5%	75.5%	4.2%	5.6%

$\chi^2 = 6.72$

Segment 1633 - 57 Fields

Percentiles	0 - 5%	5 - 10%	10 - 90%	90 - 95%	95 - 100%
Number	6	1	44	4	2
Frequency	10.5%	1.3%	77.7%	7.0%	3.5%

$\chi^2 = 5.45$

BAYESIAN ESTIMATION OF MIXTURE PARAMETERS

Let $X_1, \cdots, X_n$ be a random sample from a finite mixture density

$$f(x|\theta) = \sum_{i=1}^{m} q_i f_i(x|\theta_i),$$

where the component densities are d-dimensional multivariate normal and the mixing propostions q_i satisfy $q_i \geq 0$, $\sum_{i=1}^{m} q_i = 1$. We let $\theta_i = (\mu_i, \Sigma_i)$ denote the mean and covariance of the $i^{\underline{th}}$ component density and let θ denote the aggregate of all the parameters involved in the mixture density, including $q = (q_1, \cdots, q_m)$. We assume throughout that m is known. It will be convenient to consider also the precision matrix $\tau_i = \Sigma_i^{-1}$, and we sometimes let $\theta_i = (\mu_i, \tau_i)$.

Maximum lieklihood is the method of estimating the parameters θ which has recently attracted the most interest, [16]. According to this method, the estimate $\hat{\theta} = \hat{\theta}(X_1, \cdots, X_n)$ is the parameter value which maximizes the log likelihood function

$$\ell(\theta) = \sum_{i=1}^{n} \log f(X_i(\theta).$$

Unfortunately, as simple examples show, the function $\ell(\theta)$ is unbounded, and one must consider local maximizers of $\ell(\theta)$ or else modify $\ell(\theta)$ in some way so as to produce a global maximizer. Hathaway [12] took the second approach in proposing a constrained maximum likelihood estimator. For mixtures of univariate normal densities, he developed an effective computational procedure for finding a maximum of $\ell(\theta)$ subject to the constraints

$$\sigma_i \geq c\,\sigma_{i+1} \qquad i = 1, \cdots, m,$$

where σ_i is the $i^{\underline{th}}$ standard deviation, $\sigma_{m+1} = \sigma_1$, and $c > 0$ is a constant, chosen by the user. He also proved that $\ell(\theta)$ has a global maximizer, subject to the above constraints, and that the global maximizer is a strongly consistent estimator, as long as the true parameter satisfies the given constraints. Redner [15], mentions a penalized likelihood function of the form

$$\ell(\theta) - \lambda \sum_{i=1}^{m} ||\tau_i||^k,$$

where λ, $k > 0$ and $||\tau_i||$ is a norm on symmetric $d \times d$ matrices.

Bayes solutions for common loss functions, such as quadratic loss, appear to be computationally infeasible [8]. For example, assuming that the mixing propostions are the only unknown parameters, and using the Dirichlet prior distribution given in the next section, there is an explicit formula for the Bayes solution with quadratic loss. However, it contains m^n terms and is not useful except for very small sample sizes. The method proposed in the next section utilizes a prior density $g(\theta)$ of a certain form on the parameter θ and takes as the estimator the mode of the posterior density

$$g(\theta|X_1,\cdots,X_n) = \frac{[\prod_{j=1}^{n} f(X_j|\theta)]g(\theta)}{\int_{\theta}[\prod_{j=1}^{n} f(X_j|\vartheta)]g(\theta)d\theta}.$$

Equivalently, the estimator maximizes the penalized log likelihood function,

$$\ell_1(\theta) = \ell(\theta) + \log g(\theta).$$

Such a procedure can be justified in Bayesian theory as being the limit as $\epsilon \to 0$ of Bayes solutions $\hat{\theta}_\epsilon$ corresponding to 0-1 loss functions

$$L_\epsilon(\theta,\hat{\theta}) = \begin{cases} 0 & \text{if } ||\theta-\hat{\theta}|| < \epsilon \\ 1 & \text{if } ||\theta-\hat{\theta}|| \geq \epsilon \end{cases}.$$

It will be seen that $\ell_1(\theta)$ is similar to, but is more elaborate than the penalized likelihood function suggested by Redner.

THE PRIOR DISTRIBUTION

Recall that $q = (q_1, \cdots, q_m)$ is the vector of mixing propostions and that $\theta_i = (\mu_i,\tau_i)$ is the pair consisting of the mean vector and precision matrix of the $i^{\underline{th}}$ component normal density.

Assumption 1 : $q, \theta_1, \cdots, \theta_m$ are mutually independent.

Assumption 2 : q has a Dirichlet distribution with hyperparameters $\lambda_1, \cdots,\lambda_m$, all > 0. The prior density of q is

$$f_0(q) = \frac{\Gamma(\lambda_1+ \cdots + \lambda_m)}{\Gamma(\lambda_1) \cdots \Gamma(\lambda_m)} q_1^{\lambda_1-1} \cdots q_{m-1}^{\lambda_{m-1}-1} q_m^{\lambda_m-1} .$$

Assumption 3 : Given τ_i, the prior distribution of μ_i is d-variate normal $N_d(\alpha_i, c_i\tau_i)$ with mean $\alpha_i \in R^d$ and precision matrix $c_i\tau_i$, where $c_i > 0$ is a hyperparameter. The prior distribution of τ_i is Wishart with $v_i > d-1$ degrees of freedom and expected value $v_i h_i^{-1}$, where h_i is a positive definite matrix. Thus the joint prior density of

$\theta_i = (\mu_i, \tau_i)$ is

$$f_i(\mu_i,\tau_i) = c_i^{d/2}|h_i|^{\frac{v_i}{2}}|\tau_i|^{\frac{v_i-d}{2}} \times$$

$$\times \exp\{-\frac{c_i}{2}(\mu_i-\alpha_i)^T\tau_i(\mu_i-\alpha_i) - \frac{1}{2}\mathrm{tr}h_i\tau_i\} .$$

The prior distributions given in Assumptions 2 and 3 are the standard conjugate priors for multinomial probabilities and the parameters of the normal-Wishart distribution of the sample mean and covariance, [1]. Their use here is for mathematical convenience, rather than because of any prior conviction as to their suitability. However, it is apparent that the large number of hyperparameters involved (λ_i, v_i, c_i, α_i, h_i) allows a great deal of flexibility in applications.

The penalized likelihood function corresponding to this prior is

$$\ell_1(\Theta) = \sum_{j=1}^{n} \log f(X_j|\Theta) + \sum_{i=1}^{m} \lambda_i \log q_i$$

$$+ \frac{1}{2}\sum_{i=1}^{m}(v_i-d)\log|\tau_i| - \frac{1}{2}\sum_{i=1}^{m} c_i(\mu_i-\alpha_i)^T\tau_i(\mu_i-\alpha_i)$$

$$- \frac{1}{2}\sum_{i=1}^{m} \mathrm{tr}h_i\tau_i .$$

Here, we have eliminated terms which depend neither on the parameters, nor on the samples and, for convenience, have also replaced λ_i in the original definition of $f_0(q)$ by $\lambda_i + 1$.

GLOBAL AND LOCAL MAXIMA OF $\ell_1(\theta)$

The prior density of θ given in the preceding section is unbounded, as is $\ell_1(\theta)$, unless the hyperparameters satisfy $\lambda_i \geq 0$, $v_i \geq d$. Therefore, these restrictions will be assumed for the remainder of this paper. The ordinary likelihood function can be obtained by allowing $\lambda_i = 0$, $v_i = d$, $c_i = 0$, $h_i = 0$ for each i. This corresponds to a posterior distribution derived from an improper, noninformative prior.

Choices of the hyperparameters which guarantee a global maximizer of $\ell_1(\theta)$ are given in the following theorem.

THEOREM 5. If $v_k > d$ and h_k is positive definite for each k, then $\ell_1(\theta)$ has a maximum.

PROOF: Since $\lambda_i \geq 0$,

$$\ell_1(\theta) \leq \sum_{j=1}^{n} \log \max_i f_i(x_j|\theta_i) + \frac{1}{2}\sum_{i=1}^{m}(v_i-d)\log|\tau_i| - \frac{1}{2}\sum_{i=1}^{m} \operatorname{tr} h_i\tau_i$$

$$= \frac{1}{2}\{\sum_{j=1}^{n} \max_i [\log|\tau_i| - (x_j-\mu_i)^T\tau_i(x_j-\mu_i)] + \sum_{i=1}^{m}[(v_i-d)\log|\tau_i| - \operatorname{tr} h_i\tau_i]\}$$

For each i, let $C_i(\theta) = \{x \in R^d | \log|\tau_i| - (x-\mu_i)^T\tau_i(x-\mu_i) \geq \log|\tau_k| - (x-\mu_k)^T\tau_k(x-\mu_k)$ for each $k\}$, let $\phi_i(\theta)$ be the number of samples in $C_i(\theta)$, and let

$$s_i(\theta) = \sum_{x_j \in c_i(\theta)} (x_j-\mu_i)^T \tau_i (x_j-\mu_i).$$

Then

$$\ell_1(\theta) \le \frac{1}{2} \sum_{i=1}^{m} [A_i(\theta)\log|\tau_i| - \mathrm{tr}B_i(\theta)\tau_i]$$

where $A_i(\theta) = v_i-d+\phi_i(\theta)$ and $B_i(\theta) = h_i + s_i(\theta)$.

$$\ell_1(\theta) \le \frac{1}{2} \sum_{|\tau_i|\le 1} [(v_i-d)\log|\tau_i| - \mathrm{tr}h_i\tau_i]$$

$$+ \frac{1}{2} \sum_{|\tau_i|>1} [(v_i+d+n)\log|\tau_i| - \mathrm{tr}h_i\tau_i]$$

Let $\eta(\tau_i)$ and $\rho(\tau_i)$ denote the largest and smallest eigenvalues of τ_i respectively. If $\rho(\tau_k) \to \infty$ or $\eta(\tau_k) \to 0$ for some k, then the term corresponding to τ_k in the inequality above tends to $-\infty$ while the other terms are bounded. Therefore, there is an $r > 0$ such that

$\sup_{\theta} \ell_1(\theta) = \sup_{\theta \in \Theta r} \ell_1(\theta) < \infty$, where

$$\Theta_r = \{\theta \mid \frac{1}{r} \le \eta(\tau_k) \le \rho(\tau_k) \le r \text{ for each } k\}.$$

Represent Θ_r as $Q \times \bar{\psi}_1 \times \cdots \times \bar{\psi}_m$, where $Q = \{q \in R^m | q_i \ge 0$ for each i and $\sum_{i=1}^{m} q_i = 1\}$, and $\psi_i = \{(\mu_i,\tau_i) \mid \frac{1}{r} \le \eta(\tau_i), \rho(\tau_i) \le r\}$. Let $\bar{\psi}_i$ be the one point compactification of ψ_i, so that $\theta_i \in \psi_i$ tends to ∞ if and only if $||\mu_i|| \to \infty$. If $\theta_i \to \infty$, then $f_i(x_j|\theta_i) \to 0$ for all j; thus,

by allowing $-\infty$ as a value, $\ell_1(\theta)$ can be extended continuously to $\overline{\Theta}_r = Q \times \overline{\psi}_1 \times \cdots \times \overline{\psi}_m$, and has a maximum on that set, say at $\overline{\theta}$.

Suppose $\overline{\theta}$ is a point at infinity; i.e., that $\overline{\mu}_k = \infty$ for some k. Then $c_k = 0$, because otherwise $\ell_1(\overline{\theta}) = -\infty$. $\ell_1(\overline{\theta})$ is obviously not decreased by replacing $\overline{\mu}_k$ by any finite value. Therefore, $\ell_1(\theta)$ is maximized by a point in Θ_r. QED.

Unfortunately, as with other penalized likelihood functions the circumstances under which a consistent global maximizer of $\ell_1(\theta)$ exists are not known. Even if one exists there is no procedure for finding the global maximizer. Therefore, we must consider local maximizers. The necessary conditions for a local maximizer of $\ell_1(\theta)$ are, for $i = 1, \cdots, m$:

$$q_i = \frac{\sum_{j=1}^{n} \frac{q_i f_i(x_j|\theta_i)}{f(x_j|\theta)} + \lambda_i}{n + \lambda}, \tag{7.1}$$

where $\lambda = \sum_{i=1}^{m} \lambda_i$,

$$\mu_i = \frac{c_i \alpha_i + \sum_{j=1}^{n} \frac{q_i f_i(x_j|\theta_i)}{f(x_j|\theta)} x_j}{c_i + \sum_{j=1}^{n} \frac{q_i f_i(x_j|\theta_i)}{f(x_j|\theta)}}, \tag{7.2}$$

$$(7.3) \qquad \Sigma_i = \frac{h_i + c_i(\mu_i-\alpha_i)(\mu_i-\alpha_i)^T + \sum_{j=1}^{n} \frac{q_i f_i(x_j|\theta_i)}{f(x_j|\theta)}(x_j-\mu_i)(x_j-\mu_i)^T}{v_i-d + \sum_{j=1}^{n} \frac{q_i f_i(x_j|\theta_i)}{f(x_j|\theta)}}$$

These equations are the basis for an EM-like iteration procedure defined by evaluating the right hand sides with the current values of the parameters to obtain updated values of the parameters. Each of the updated parameters is a convex combination of some prior estimate and the EM update for ordinary maximum likelihood estimation. Interestingly, the updated q_i is a convex combination of the EM update and the prior mode $\frac{\lambda_i}{\lambda}$ of q_i, whereas the updated Σ_i is a convex combination of the EM update and the prior conditional mean

$$\frac{h_i+c_i(\mu_i-\alpha_i)(\mu_i-\alpha_i)^T}{v_i-d}$$

of Σ_i given μ_i, not the prior mode. Obviously, the larger the sample size, the greater will be the weight given to the EM updates and the less given to the prior estimates. When the update equation (7.3) for Σ_i is evaluated using the just updated value of μ_i in the products $(x_i-\mu_i)(x_i-\mu_i)^T$ and $(\mu_i-\alpha_i)(\mu_i-\alpha_i)^T$ this successive substitutions procedure is equivalent to the modified EM procedure suggested by Dempster, Laird, and Rubin [9] for finding posterior modes. Hereafter, we shall refer to this procedure as the generalized EM procedure (GEM). The general convergence properties of the GEM procedure follow from

[16 Theorem 4.1], more specifically, starting from any point $\theta^{(0)}$ in parameter space, the sequence $\{\theta^{(k)}\}_{k=0}^{\infty}$ produced by the GEM procedure converges to a nonempty, connected, compact subset of parameter space on which the penalized likelihood $\ell_1(\theta)$ is constant, and on which the equations (7.1)-(7.3) are satisfied.

The next theorem assures that the GEM procedure converges to a consistent local maximizer of $\ell_1(\theta)$, given a good enough starting value.

THEOREM . If the true parameter $\overline{\theta}$ is in the interior of the parameter set, then there is a neighborhood N of $\overline{\theta}$ such that with probability 1, if n is sufficiently large there is a unique solution $\hat{\theta}$ of (7.1)-(7.3) in N and $\hat{\theta} \to \overline{\theta}$ as $n \to \infty$. Furthermore, with probability 1, for large n the GEM procedure converges to $\hat{\theta}$ if the starting point is near enough to $\hat{\theta}$.

PROOF. The existence and uniqueness of a consistent local maximizer is a consequence of a consistency theorem due to Chanda [6], (see also Peters and Walker [14]). A simple modification of the proof of that theorem shows that the Hessian $d^2\ell_1(\theta)$ is negative definite at $\theta = \hat{\theta}$ for large n. Therefore, $\ell_1(\theta)$ is strictly concave in a neighborhood of $\hat{\theta}$. The local convergence of the GEM procedure to $\hat{\theta}$ now follows from the consistency theorem and Lemmas 1 and 2 of [15].

OVERMODELED MIXTURES

For mixture problems in which the number of normal components is not precisely known, the present model is not appropriate from a Bayesian point of view. However, it is possible that the penalized likelihood

function exhibits better numerical and statistical properties in this situation than the ordinary likelihood function. To illustrate, suppose that the model contains m normal components, but the true density is a mixture of $k < m$ normal components. Thus,

$$f(x|\bar{\theta}_{(k)}) = \sum_{i=1}^{k} \bar{q}_i f_i(x|\bar{\theta}_i) \qquad (\bar{q}_i > 0)$$

is the true density, and

$$f(x|\theta_{(m)}) = \sum_{i=1}^{m} q_i f_i(x|\theta_i)$$

is the model. Let the hyperparameters for the model satisfy $\lambda_i = 0$, $v_i > d$, $c_i > 0$, $\alpha_i \in R^d$, and h_i positive definite for $i = 1, \cdots, m$. By Theorem 6, there is a consistent solution $\bar{\theta}_{(k)} = (q_1, \cdots, q_k, \hat{\theta}_1, \cdots, \hat{\theta}_k)$ of equations (7.1)-(7.3) for the k component mixture. Let $\hat{q}_i = 0$, $\hat{\mu}_i = \alpha_i$, $\hat{\Sigma}_i = h_i/(v_i-d)$ for $i = k+1, \cdots, m$, and let $\hat{\theta}_{(m)} = (\hat{q}_1, \cdots, \hat{q}_m, \hat{\theta}_1, \cdots, \hat{\theta}_m)$. Clearly $\hat{\theta}_{(m)}$ is a solution of (7.1)-(7.3) for the m component mixture which is consistent in the sense that $f(x|\hat{\theta}_{(n)}) \to f(x|\hat{\theta}_{(k)})$ as $n \to \infty$. In contrast, it is not known if there is a consistent solution of the ordinary likelihood equations in this situation.

REMARKS AND CONCLUSIONS

The remarks at the end of the preceding section suggest that in cases where the number m of normal components is unknown, but a reasonable upper bound can be assumed, one should take $\lambda_i = 0$, $v_i > d$, $c_i > 0$, h_i positive definite. Otherwise, the choice of the hyperparameters may be guided by prior guesses at location and dispersion of the mixture

parameters. For example

$$E(q_i) = \frac{\lambda_i+1}{\lambda+m}$$

$$\text{cov}(q_i,q_k) = -\frac{(\lambda_i+1)(\lambda_k+1)}{(\lambda+m)^2(\lambda+2)}$$

$$\text{var}(q_i) = \frac{(\lambda_i+1)(\lambda-\lambda_i+m-1)}{(\lambda+m)^2(\lambda+m+1)}$$

can be used to aid in choosing the λ_i, while the equation

$$E(\Sigma_i) = c_i \text{ var}(\mu_i)$$

(provided $v_i > d+1$) can aid in choosing c_i.

The procedures outlined herein may be especially useful in applications such as crop inventories from satellite data. There, spectral measurements may be sampled from a large ground area (segment) which is itself chosen from a large number of possibilities. The normal mixture model has often been used for the distribution of spectral responses from particular segments. Thus the parameters $(q, \theta_1, \cdots, \theta_m)$ can be considered characteristic of segments, while the prior distribution of these parameters can reflect their variability among the possible choices of segments. Since there are "ground truth" segments available in which each pixel has a known class identity, it is possible that the hyperparameters of the prior distribution could be estimated from the ground truth segments.

Further research into the numerical and statistical properties of the GEM procedure is planned. The properties to be studied include the

consistency of the global maximizer, the behavior of the GEM procedure for overmodeled mixtures, and the sensitivity of the procedure to starting values, for various choices of the hyperparameters.

Appendix

Proofs of the Theorems

Proof of Theorem 1: The covariance of Y can be written as $\psi_{n\ell} \otimes I_n + \Sigma_{n\ell} \otimes J_n J_n^T$, where $\otimes$ denotes the kronecker product. For $W \in \Lambda_n'$, $YW = I_p \otimes W^T(Y)$ has covariance $(I_p \otimes W^T)(\psi_{n\ell} \otimes I_n + \Sigma_{n\ell} \otimes J_n J_n^T)(I_p \otimes W)$ $= \psi_{n\ell} \otimes I_n + \Sigma_{n\ell} \otimes J_n J_n^T$. The mean of YW is $\mu_{n\ell} J_n^T W = \mu_{n\ell} J_n^T$. Therefore, $YW \overset{d}{=} Y$. By a similar argument, if $P^T J_n = 0$, $P^T P = I_{n-1}$ and $Z = YP$, then $E(Z) = 0$ and $\text{cov}(Z) = (I_p \otimes P^T)(\psi_{n\ell} \otimes I_n + \Sigma_{n\ell} \otimes J_n J_n^T)(I_p \otimes P) = \psi_{n\ell} \otimes I_{n-1}$. Therefore the columns of Z are independently distributed as $N_p(0, \psi_{n\ell})$. To prove the last assertion let

$$Q = (n^{-1} J_n \mid P)_{n \times n}$$

where P has the same properties as above. In block form, the covariance of $YQ = (\bar{Y} \mid Z)$ is

$$\left(\begin{array}{c|c} \frac{1}{n}\psi_{n\ell} + \Sigma_{n\ell} & 0 \\ \hline 0 & \psi_{n\ell} \otimes I_{n-1} \end{array} \right)$$

Therefore, $\bar{Y}$ and Z are independent and $\bar{Y} \sim N_p(\mu_n, \frac{1}{n}\psi_{n\ell} + \Sigma_{n\ell})$. Moreover, $S = ZZ^T$ and by the first part of the theorem $S \sim W_p(n-1, \psi_{n\ell})$.

Proof of Theorem 2: Let f_0 be a density function in F satisfying the hypothesis (E). Define

$$h_f(y) = f(y) \,/\, f_0(y)$$

for $f \in F$. By a version of the Neyman-Fisher theorem (Theorem 6.1 of [3]), if $(\bar{Y}, S)$ is sufficient,

$$h_f(y) = g_f(\bar{y}, S)$$

almost everywhere, where g_f is a Borel measureable function on the space of $(\bar{Y}, S)$. For a given $f \in F$ and $W \in \Lambda_n'$, the set

$$U = \{y \mid h_f(y) \neq h_f(yW)\}$$

is an open set contained in $B_1 \cup B_2$, where

$$B_1 = \{y \mid h_f(y) \neq g_f(\bar{y}, S)\} ,$$

and

$$B_2 = B_1 W^T = \{y \mid h_f(yW) \neq g_f(\bar{y}, S)\} .$$

By Theorem 1, the pr. measure λ_0 corresponding to f_0 is invariant under Λ_n'. Since $\lambda_0(B_1) = 0$ if follows that $\lambda_0(B_2) = 0$ also, and hence, $\lambda_0(U) = 0$. Therefore U is empty and h_f is an invariant function. This implies that each $f \in F$ is invariant under Λ_n' and must satisfy (E).

Proof of Theorem 3: The function

$$g(\epsilon) = \frac{\epsilon}{\epsilon - \log(1 + \epsilon)}$$

is positive and strictly decreasing on $(0, \infty)$. Thus, if $\frac{\hat{f}}{f} - 1 \geq \epsilon$ we have

$$\frac{\hat{f}}{f} - 1 \le g(\epsilon)\left[\frac{\hat{f}}{f} - 1 - \log\frac{\hat{f}}{f}\right].$$

Therefore,

$$\frac{1}{2}\int_{\mathbb{R}^m} |\hat{f} - f| = \int_{\hat{f}>f} (\hat{f} - f)$$

$$= \int_{0 < \frac{\hat{f}}{f} - 1 \le \epsilon} (\frac{\hat{f}}{f} - 1)f + \int_{\frac{\hat{f}}{f} - 1 > \epsilon} (\frac{\hat{f}}{f} - 1)f$$

$$\le \epsilon + g(\epsilon)\int_{\mathbb{R}^m} [\frac{\hat{f}}{f} - 1 - \log\frac{\hat{f}}{f}]\, f$$

$$= \epsilon + g(\epsilon)\int_{\mathbb{R}^m} f \log(\frac{f}{\hat{f}})$$

$$= \epsilon + g(\epsilon)H(\hat{f}, f) .$$

References

1. Aitchison, J. and Dunsmore, I. R., Statistical Prediction Analysis, (Cambridge University Press, 1975).

2. Anderson, T. W. and Mentz, R.P., On the structure of the likelihood function for autoregressive and moving average models, J. Time Ser. Anal. 1 (1980) 83-94.

3. Bahadur, R. R., Sufficiency and Statistical decision functions, Ann. Math. Statist. 25 (1954) 423-463.

4. Barndorff-Nielsen, O., Information and Exponential Families in Statistical Theory (John Wiley and Sons, New York, 1978.)

5. Campbell, J. B., Spatial correlation effects upon accuracy of supervised classification of land cover, Photogrammetric Eng. and Rem. Sens. 47 (1981) 335-363.

6. Chanda, K. C., A note on the consistency and maxima of the roots of the likelihood equations, Biometrika, 41 (1954), 56-61.

7. Dawid, A. P., Spherical matrix distributions and a multivariate model, J. Roy. Statist. Soc., Ser. B, 39 (1977) 254-261.

8. Day, N. E., Estimating the components of a mixture of normal distributions, Biometrika, 56 (1969), 463-474.

9. Dempster, A. P. et al, Maximum likelihood from incomplete date via the EM algorithm, J. Roy. Statist. Soc. B, 39 (1977), 1-38.

10. Fuller, W. A., Introduction to Statistical Time Series (John Wiley and Sons, New York, 1976.)

11. Geman, S., Sieves for nonparametric estimation of densities and regressions, Reports in Pattern Analysis No. 99 (1981), Div. of Appl. Math., Brown University.

12. Hathaway, R., Contrained Maximum Likelihood Estimation for a Mixture of m Univariate Normal Distributions, Statistics, University of South Carolina, March 1983.

13. McCormick, B., Time series models for texture synthesis, Int. J. Comput. Inform. Sci. 3 (1974) 329-343.

14. Peters, B. C. and Walker, H. F., An iterative procedure for obtaining maximum likelihood estimates of the parameters for a mixture of normal distributions, SIAM J. Appl. Math., 35 (1978), 362-378.

15. Redner, R. A., Maximum Likelihood Estimation for Mixture Models, NASA/JSC Technical Report, Sept. 1980.

16. Redner, R. A. and Walker, H. F., Mixture densities, maximum likelihood and the EM algorithm, SIAM Review, 26 (1984), 195-239.

17. Walker, H. F., The numerical evaluation of the mle for the parameters for a mixture of normal distributions from a partially identified sample. Report 54, Department of Mathematics, University of Houston, 1976.

QUANTILE DATA ANALYSIS METHODS FOR IMAGE DATA ANALYSIS AND EDGE DETECTION

Emanuel Parzen
Statistics Department
Texas A&M University

ABSTRACT

A quantile data analysis approach to some problems of image data analysis is outlined. The approach is illustrated on (1) two simulated pixel vectors representing reflectance spectra of a mineral measured in 32 bands in the wavelength range 1.2 μm to 1.4 μm, and (2) a simulated two dimensional 6 by 6 grid of pixels, each with one spectral band measurement. The goal is to determine statistical properties which can be used to classify pixels and determine edges in pixel scenes separating pixels with different statistical properties. Quantile data analytic techniques illustrated are identification quantile functions, identification quantile plot, comparison quantile function, and IQQ (identification-quantile-quantile) plots.

0. INTRODUCTION

Image data is acquired by remote sensing of the earth's surface from spacecraft and aircraft. Image data consists of enormous amounts of multidimensional data; its analysis, interpretation, and classification requires development of new data analytic algorithms and methods. The difficulties inherent in the analysis of multi-dimensional data is often called the "curse of dimensionality." The dimensionality of image data is increasing as measurements at higher spatial resolution and narrower spectral bands are made possible by new technology for sensors and instruments which is rapidly developing [see Goetz et al (1985)].

Our approach to image data analysis seeks to replace parametric statistical methods based on approximate normal distributions with nonparametric statistical methods based on suitably defined ranks and quantile functions. An important theoretical problem which this research program has investigated is the effect of dependence on linear rank statistics and quantile functions. Dependence is modelled by a stationary time series. The theoretical results are described in the Ph.D. thesis of A. Harpaz (1985). This paper outlines the ideas of the quantile data analysis approach to image data analysis in order to stimulate interest in them by the broad image processing scientific community.

Section 1 defines the mathematical problem of data analysis of the field of pixel vectors which represents an image.

Section 2 defines the edge detection approach to pixel classification. Section 3 outlines the concepts involved in quantile data analysis of a pixel vector. Section 4 outlines the concepts involved in comparing pixel vectors in order to test the homogeneity of groups of pixel vectors.

1. IMAGE DATA ANALYSIS

Consider measurements taken by spaceborne or airborne sensors on a specified date at a specified site on the earth's surface. A site is divided into thousands of surface elements called pixels (picture elements). On each pixel the visible and solar reflected portions of the electromagnetic energy spectrum are measured by sensors which provide spectral measurements in a number, denoted L, of spectral bands. The number L of spectral bands has as typical values 4, 7, 32, 128, 224.

Sensors such as the Landsat Multispectral Scanner (MSS) and Landsat Thematic Mapper (TM) are optomechanical systems which use discrete detectors to convert the reflectd solar photons from each pixel in the scene into a sensible electronic signal. The detector elements are placed behind filters that pass broad portions of the spectrum. MSS has 4 sets of filters and detectors to measure 4 spectral bands; TM measures 7 spectral bands. Imaging spectrometry can measure images in hundreds of spectral bands simultaneously.

Each spectral measurement is typically an integer from 0 to 255 representing 256 possible intensity levels.

We use the following notation for measurements made by sensors; denote by

$$Y(\lambda_j, x_1, x_2)$$

measurement of reflected energy in the spectral band indexed by a wavelength λ_j from the pixel with coordinates x_1, x_2.

A pixel with coordinates (x_1, x_2) is represented by an L vector

$$\vec{Y}(x_1, x_2) = \begin{bmatrix} Y(\lambda_1, x_1, x_2) \\ \vdots \\ Y(\lambda_L, x_1, x_2) \end{bmatrix}$$

whose components are the intensities of reflected energy in the spectral bands.

Associated with each pixel is a "ground truth" which could be: type of crops, trees, water, type of mineral, type of vegetation, etc.

The ground truth of a pixel at (x_1, x_2) is denoted $\theta(x_1, x_2)$ and is regarded as a value of a discrete parameter θ which indexes the different classifications of ground truth which the investigator is discriminating.

The general problem of image data analysis: Form an estimator $\hat{\theta}(x_1, x_2)$ of the ground truth field from the image field $\vec{Y}(x_1, x_2)$

A decision theoretic statistical approach to this problem can be described formally as follows: assume a probability

model $Y|\theta$ for the distribution of Y given θ. The estimator $\hat{\theta}$ is the conditional probability distribution of θ given Y, denoted $\theta|Y$.

An alternative to the decision theoretic approach, which we adopt, is an exploratory data analysis or nonparametric data modeling approach. To illustrate this approach we consider in this paper two simulated data sets (called class 1 and 2) representing respectively reflectance spectra of a mineral assumed to be measured over 32 bands in the range of wavelengths 1.2 μm to 2.4 μm. Our simulated numbers were adapted from rough approximations to the spectral waveforms in Goetz et al (1985) of alunite and kaolinite which we call class 1 and class 2.

From class 1 we assume we have a (simulated) pixel vector (whose components represent spectral intensities in successive bands):

82,82,80,82,80,80,70,60,66,54,70,74,74,72,60,70,
68,66,60,58,56,54,54,50,40,32,40,58,58,44,52,40.

From class 2 we assume we have a (simulated) pixel vector:

88,86,88,84,80,70,80,90,92,92,92,92,92,90,90,90,
88,90,90,90,90,88,86,84,80,70,56,70,70,64,62,60.

Plots of these pixel vectors are given in Figures 2 and 3 respectively in a new dimension-less format introduced in our research program called the identification quantile plot (described in section 3).

We refer to the above data sets as the 32 channel case. If we average over disjoint sets of 4 bands to obtain measurements in only 8 bands, then two spectral classes are represented by the following pixel vectors which we call the 8 channel case:

Class 1 82,72,64,69,63,53,43,48

Class 2 87,80,92,90,90,87,69,64

In the sequel we analyze each pixel vector as a data set and compare the data sets to determine features which can be used to discriminate between the two classes.

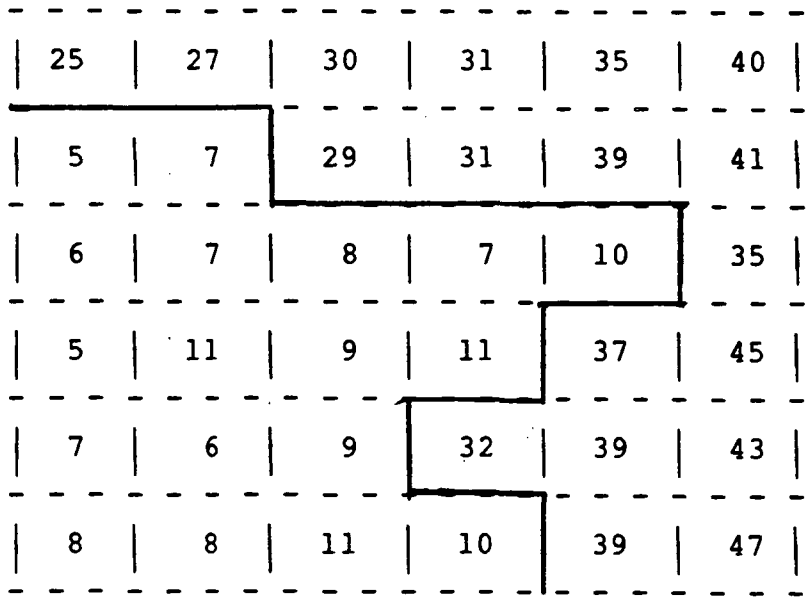
2. EDGE DETECTION APPROACH TO PIXEL CLASSIFICATION

The problem of edge detection plays a central role in the image data analysis problem; it is to determine edges which separate pixels into contiguous groups having the same classification of ground truth. An edge is defined to be a boundary imagined to be drawn as a separation between pixels which do not have the same ground truth classification. After one determines edges on the basis of statistical (data analytic) considerations one has the problem of determining (estimating) the classification (ground truth) of each contiguous group of pixels (which have been identified as having the same ground truth).

The literature of pattern recognition and image analysis contains a wide variety of algorithms for extracting edges from noisy images. Methods of edge extraction are classified in two

types: gradient or statistical. Suk and Hon (1984) provide a bibliography of representative gradient and statistical approaches to edge detection.

To illustrate our quantile data analysis approaches to edge detection we consider in this paper an example given by Suk and Hon (1984) of a simulated two dimensional 6 by 6 grid of pixels with each pixel represented by one spectral band measurement:



The edge drawn in the interior of the grid as a solid line was determined by Suk and Hong (1984) using the algorithms that they give in their paper.

Quantile data analysis can be regarded as an approach to statistical data analysis in which the first step is ranking the data. The concepts introduced theoretically in the next section

are introduced at this point by an example which shows how they are applied.

Quantile data analysis provides a systematic way of determining a threshold value which can be used to divide the pixels in a grid by an edge which separates values below the threshold from values above the threshold. Consider the data set formed from the pixel intensities in the above 6 by 6 grid. One determines that (1) there are K=21 values in the data set, (2) the values in increasing order [denoted symbolically by V(1)<...<V(K)] are

5,6,7,8,9,10,11,25,27,29,30,31,32,35,37,39,40,41,43,45,47.

These values occur in the data set with the following respective multiplicities (number of repetitions)

2,2,4,3,2,2,3,1,1,1,1,2,1,2,1,3,1,1,1,1,1.

The empirical probabilities, empirical distribution function, and empirical identification quantile function of the data set are as follows (these concepts are defined in the next section):

Index J	Value V(J)	Empirical Probability P[V(J)]	Cumulative Probability F[V(J)]	Midrank U(J)	Identification Quantile QI(U(J))
1	5	.056	.056	.028	-.290
2	6	.056	.111	.083	-.272
3	7	.111	.222	.167	-.255
4	8	.083	.306	.264	-.237
5	9	.056	.361	.333	-.219
6	10	.056	.417	.389	-.202
7	11	.083	.500	.458	-.184
8	25	.028	.528	.514	.061
9	27	.028	.556	.542	.097
10	29	.028	.583	.567	.132
11	30	.028	.611	.597	.149
12	31	.056	.667	.639	.167
13	32	.028	.694	.681	.184
14	35	.056	.750	.722	.237
15	37	.028	.778	.764	.272
16	39	.083	.861	.819	.307
17	40	.028	.889	.875	.325
18	41	.028	.917	.903	.342
19	43	.028	.944	.931	.347
20	45	.028	.972	.958	.413
21	47	.028	1.000	.986	.448

Summary statistics are: mean MVY=21.9, median MQY=21.5; standard deviation DSY=14.6, quartile deviation DQY=57; lower and upper quartiles [$Q\tilde{}(.25)$ and $Q\tilde{}(.75)$] equal 7.896 and 36.33 respectively. The measure of tail behavior are:

$$Q\tilde{}I(.028) = -.290, \text{ supershort left tail;}$$

$$Q\tilde{}I(.986) = .448, \text{ short right tail.}$$

Supershort tails are an indication of the possibility of bimodality. The big gap in $Q\tilde{}I(u)$ from a value of -.184 to a value of .061 is used to locate the values $V(K^*)$ = 11 and

$V(K^*+1) = 25$ which separate the values into two clusters. The edge in the pixel scene is drawn to separate the values in the two clusters. The edge drawn in this example by this criterion is the same as the edge drawn by Suk and Hong (1984) using their algorithms.

3. QUANTILE DATA ANALYSIS OF A PIXEL VECTOR

The L components of a vector $\vec{Y}(x_1,x_2)$ of spectral measurements are denoted $Y_1,\ldots,Y_L$. From the components of a pixel we form a data set for which one computes the empirical probability distribution

$$\tilde{F}(y) = \text{fraction of data set} \leq y, \qquad -\infty<y<\infty$$

and the empirical quantile function

$$\tilde{Q}(u) = \tilde{F}^{-1}(u) = \inf\{y: \tilde{F}(y) \geq u\}, \quad 0\leq u\leq 1 \ .$$

The empirical quantile function can be regarded as a rearrangement in increasing order of the values in the data set of the values $Y_1,\ldots,Y_L$ whose order statistics are denoted by $Y(1;L)<\ldots<Y(L;L)$. One can show that

$$\tilde{Q}(u) = Y(j;L) \text{ for } (j-1)/L < u \leq j/L \ .$$

Statisticians have studied the statistical properties of $\tilde{F}(y)$ and $\tilde{Q}(u)$ mainly under the assumption that $Y_1,\ldots,Y_L$ are a random sample (independent random variables which are identically distributed as a random variable Y).

To apply quantile function and nonparametric test methods to image data requires fundamental research to extend the theory

from random samples to data sequences of Y values which are dependent. Our approach is to model dependence by the model of a stationary time series, which assumes that $Cov[Y_j, Y_k]$ is a function only of $|j-k|$, denoted $R(j-k)$.

The theory of stationary time series imagines an infinite sequence of random variables Y, and defines a sequence of autocorrelation coefficients

$$\rho(v) = R(v)/R(0)$$

The spectral density $f(\omega)$, $0 \leq \omega \leq 1$, is defined to be the Fourier transform of the autocorrelation function:

$$f(\omega) = \sum_{v=-\infty}^{\infty} \exp(-2\pi i v \omega)\, \rho(v) \quad , \quad 0 \leq \omega \leq 1 \quad .$$

The variable ω represents frequency; $f(\omega)$ is a measure of the proportion of the variance of Y values which can be assigned to hidden sine waves of frequency ω in the sequence of Y values. The value of the spectral density function at zero frequency $\omega=0$ plays a central role in statistical inference, especially in assessing the effect of dependence on the probability distribution of estimators of means and tests for comparing two samples.

An empirical quantile function $Q(u)$ can be formed for any set of data. Our interpretation of an empirical quantile function is guided by initially regarding it as an estimator of the properties of a hypothetical random variable Y of which the data batch of Y values is a random sample.

The true distribution function F(y) and true quantile function Q(u) of Y are denoted

$$F(y) = \text{PROB}[Y \leq y], \qquad -\infty < y < \infty \quad ;$$

$$Q(u) = F^{-1}(u), \qquad 0 \leq u \leq 1 \quad .$$

Mean MY and variance VARY of Y can be expressed in terms of Q(u):

$$MY = E[Y] = \int_0^1 Q(u)\, du$$

$$VARY = VAR[Y] = \int_0^1 \{Q(u) - MY\}^2\, du$$

Standard deviation of Y is denoted $DSY = \{VARY\}^{1/2}$.

Alternative measure of location is the median MQY = Q(.5). An alternative measure of scale can be defined when Q(u) is continuous with quantile density function q(u) = Q'(u);

quantile deviation DQY = Q'(.5) = q(.5).

An approximator of the quantile deviation which we use in practice and denote by the same symbol (but a different name) is

$$\text{quartile deviation } DQY = \{Q.75)-Q(.25)\}/(.75-.25)$$

$$= 2\{Q(.75) - Q(.25)\} \quad .$$

To classify the type or shape of the distribution we form a normalized version which is independent of location and scale

parameters by normalizing Q(u) to have, at u=.5, value 0 and approximate slope 1. The identification quantile function is denoted QI(u) or QIY(u) and defined by

$$QI(u) = \{Q(u) - MQ\}/DQ, \quad QIY(u) = \{Q(u) - MQY\}/DQY.$$

Identification quantile function truncated plot: The identification quantile version $Q\tilde{}IY(u)$ of the empirical quantile function $Q\tilde{}(u)$ of the data set is plotted truncated at ± 1 in order to present the plot on a standardized scale. On the same graph one plots the identification quantile functions of the uniform and normal distributions. The values of $Q\tilde{}IY(u)$ for u near 0 and 1 provide quick indicators of the type of distribution that fits the data. Intervals used to discriminate various types of probability distributions are as follows:

$Q\tilde{}IY(0) < -1$	long tail	$Q\tilde{}IY(1) > 1$
$-1 < Q\tilde{}IY(0) < -.5$	medium tail	$.5 < Q\tilde{}IY(1) < 1$
$-.5 < Q\tilde{}IY(0) < 0$	short and supershort tail	$0 < Q\tilde{}IY(1) < .5$

Figure 2 illustrates the format of an identification quantile function; one always plots theoretical identification quantile functions of a uniform distribution [the line from (0,-.5) to (1,.5)] and a normal distribution [the curve which coincides with the line for u near 0.5].

A goal of our research program is to extend these concepts to discrete quantile functions since empirical quantile functions are discrete. Let K be the number of discrete values in the data set (number of points of discontinuity of the discrete quantile function). Denote these distinct values by $V(1)<...<V(K)$. The important concept of midranks $U(1)<...<U(K)$ of a discrete quantile function is defined by

$$U(j) = \{FV(j-1) + FV(j)\}/2 \quad , \; j=1,...,K.$$

where we define $FV(0) = 0$, $FV(j) = F(V(j))$.

The continuous version QC(u) of a discrete quanitle function Q(u) is defined by

$$QC(U(j)) = V(j) \quad , \; j=1,...,K.$$

At u=0 and u=1 we define QC(u) to equal respectively natural minimum and natural maximum when they are available; otherwise we define their values to be the sample minimum and sample maximum:

$$QC(0) = V(1), \quad QC(1) = V(K)$$

At other values u, QC(u) is defined by linear interpolation between its values at $0,U(1),...,U(K),1$.

The median MQ and quartile deviation DQ of a discrete quantile function are defined by

$$MQ = QC(.5), \; DQ = 2\,\{QC(.75) - QC(.25)\} \quad .$$

The identification quantile function of a discrete quantile function is defined by

$$QI(u) = \{QC(u) - MQ\}/DQ \ .$$

Identification quantile plot of data: A dimensionless graph of a vector of measurements (representing spectral intensities in successive wavelength bands) is obtained using the identification quantile transformed values $\{Y_j - MQY\}/DQY$ instead of the original values Y_j. A grid of lines $y=0, \pm.5, \pm1$ are plotted on the same graph to visually indicate the range (maximum and minimum values) of the identification quantile transformed values.

Example: The concepts have now been defined to illustrate the foregoing diagnostic tools of the quantile approach to data analysis.

The 32 channel pixel vector from class 1 (given in section 1) has mean 62, median 60, standard deviation 13.9, quartile deviation 39. Figure 2 is a plot of the time series not in its original units but in dimensionless units, using the identification quantile plot.

The 32 channel pixel vector from class 2 has mean 82.3, median 88, standard deviation 10.9, quartile deviation 35. Its identification quantile plot is in Figure 3.

To use identification quantile functions to determine the tail behavior of the distribution it is not necessary to plot it

but only to examine their values for u near 0 and 1. For the data sets of pixel vectors we obtain

U	Q~I(u) Class 1	Q~I(u) Class 2
.01	-.72	-.91
.05	-.58	-.84
.10	-.51	-.73
.25	-.15	-.44
.75	.35	.06
.90	.55	.11
.95	.56	.11
.99	.56	.11

A pixel vector can be classified into class 1 or class 2 using features of the different behavior of the identification quantile function for the two classes. The value .11 for class 2 is interpreted as a supershort distribution which is explained by the constancy of the spectral waveform from class 2 which shows up in the quantile function as a clustering of values.

We next identify the relations between the components of the pixel vector regarded as a time series. We model the dimensionless time series denoted YI(t) plotted in the identification quantile plot. Both the samples (classes 1 and 2) are identified by our time series model identification programs as fitted by an AR(1), autoregressive scheme of order 1. For class 1, the model is

$$YI(t) = .77\ YI(t-1) + e(t)$$

where e(t) denotes a residual time series which is white noise. It should be noted that e(t) denotes a different white noise

process in each model in which it appears. For class 2, the model is

$$YI(t) = .82\ YI(t-1) + e(t)$$

The goal of the time series estimation phase is to estimate the value of the spectral density of the two time series at zero frequency. For these two models the value is approximately the same, and approximately equals 6. One can interpret this value as the factor to be used as a correction for dependence when computing the variance of estimators of location (such as the mean) or estimators of difference of location of two samples (such as the Wilcoxon test). The spectral density values can be used to answer the question of how much additional information is obtained by measuring the electromagnetic spectrum in more but narrower bands.

4. QUANTILE COMPARISONS OF PIXEL VECTORS

To detect edges in a scene a statistical approach is to detect contiguous groups of pixels that can be considered as clusters of pixels with the same statistical properties. Thus a major problem in the statistical approach to edge detection is how to compare two pixel vectors $\vec{Y}(x_1,x_2)$ and $\vec{Y}(x'_1,x'_2)$ corresponding to geographic locations (x_1,x_2) and (x'_1,x'_2) respectively. From the L components of $\vec{Y}(x_1,x_2)$ one can form a data set $Y_1,\ldots,Y_L$. From the L components of $\vec{Y}(x'_1,x'_2)$ one can form a data set $Y'_1,\ldots,Y'_L$. The pixel vectors can be compared by testing the equality of distributions of the two data sets.

Conventional statistical techniques for comparing two sets can be formulated in the language of relating a variable Y to another variable X. If one pools (combines) all the data sets to be compared, one imagines the pooled data set to be a sample of a variable Y whose empirical distribution is denoted FY. The variable X attached to a data value represents the population (pixel location) to which it belongs. The empirical conditional distributions of Y given X=1 (denoted FY:X=1) is the distribution computed from $Y_1,\ldots,Y_L$. The empirical conditional distribution of Y given X=2 (denoted FY:X=2) is the distribution computed from $Y'_1,\ldots,Y'_L$.

Tests for the equality of the distributions of the two samples can be formulated as comparing the unconditional empirical distribution FY with the conditional empirical distribution of Y given X=1. Our approach is to define a comparison quantile function D(u;FY,FY:X) and a comparison quantile density function d(u;FY,FY:X) as follows. Let V(1)<...<V(K) be the ordered distinct values in the pooled sample. Let PY(V(J)) be the empirical probability that Y=V(J), and let PY:X(V(J)) be the conditional empirical probability that Y=V(J) in the sample represented by the value of X. Define

$$FY(V(J)) = PY(V(1))+\ldots+PY(V(J)),$$

$$U(J) = 0.5\{FY(V(J)) + FY(V(J-1))\} .$$

Recall from section 3 that U(1)<...<U(K) are called the midranks of the pooled sample; they play a central role in statistical

methods based on ranks rather than values. The concepts have been introduced to define

$$d(u;FY,FY:X) = PY:X(V(J))/PY(V(J)),\ FY(J-1))<u<FY(v(J))$$

$$D(u;FY,FY:X) = \int_o^u d(t;FY,FY:X)\ dt$$

To test equality of the distributions FY and FY:X one tests for the equality of $D(u;FY,FY:X)$ and $D_o(u)=u$.

Example: To test the equality of the 32 channel class 1 and class 2 pixel vectors in section 1, we plot in Figure 4 the comparison quantile function D(u) [where for convenience we write D(u) for D(u;FY,FY:X)] which compares the distribution of the class 1 sample with the pooled sample. The graph can be used to judge qualitatively the difference between D(u) and $D_o(u)=u$ [whose graph is the 45^o line].

To judge quantitatively the significance of the difference between D(u) and $D_o(u)=u$ many test statistics are available; they can be regarded as having as components test statistics of the form, called linear rank statistics,

$$\int_o^1 J(u)\ dD(u)$$

for suitable choices of score function J(u).

A test statistic which is always among those used is the Wilcoxon statistic, with score function J(u)=u-0.5. It can be writtten in an equivalent form

$$W = \int_0^1 \{D(u) - u\}\, du \quad .$$

In words, W is the area between D(u) and D_0=u.

To compute W in practice one introduces statistical methods based on ranks and the rank transform denoted theoretically UY = FY(Y). Statistical methods derived from the normal distribution are based on the conditional distribution (given values of X) of the values V(1)<...<V(K) of Y. Rank methods are based on the conditional distribution (given values of X) of the midranks U(1)<...<U(K). In particular the Wilcoxon statistic for comparing two samples can be expressed as conditional means of midranks given that X=1:

$$W = E[UY:X=1] - E[UY] = E[UY:X=1] - 0.5$$

We compute W by

$$W = \sum_{J=1}^{K} UY(J)\, PY:X=1(V(J)) - 0.5 \quad .$$

To test the significance of W computed from a random sample of size n one would treat W as approximately N(0,1/12n), normal with mean 0 and variance 1/12n. If the sample consists of dependent random variables (rather than independent) the variance of W must be adjusted to account for the dependence. Harpaz (1985) shows how to calculate the variance of linear rank statistics when the dependence structure is that of a stationary time series. The factor by which the variance increases (or

decreases) can be expressed in terms of the values at zero frequency of the spectral density of rank transformed time series.

This paper has defined various quantile data analytic graphic techniques for visually testing for patterns in data: identification quantile functions (Fig. 1), identification quantile plots (Fig. 2 and 3), and comparison quantile function plots (Fig. 4). Another new graphical display we propose are identification quantile-quantile (IQQ) plots. To compare two samples, or to compare a sample with a theoretical distribution, their respective quantile functions $Q_1(u)$ and $Q_2(u)$ can be compared by plotting the points $(Q_1I(u), Q_2I(u))$. We call this plot an IQQ plot, in contrast to a QQ plot which is a graph of $(Q_1(u), Q_2(u))$. One interprets this plot by visually detecing how well it is fit by a straight line. To help a visual identification of a straight line fit to the IQQ plot one adds to the graph a grid of lines $x=0,\pm.5,\pm1$ and $y=0,\pm.5,\pm1$.

The IQQ plot of the two 32 channel pixel vectors is given in Figure 5. Its deviation from a 45^{o} line indicates that the two classes have different types of distributions.

BIBLIOGRAPHY

Goetz, A.F.H.; Vane, Gregg; Solomon, Jerry E.; Rock, B.N. (1985). "Imaging Spectrometry for Earth Remote Sensing." Science, 228. 1148-1153.

Harpaz, A. Stationary Time Series, Quantile Functions, Nonparametric Inference, and Rank Transform Spectrum. Ph.D. Thesis, Department of Statistics, Texas A&M University.

Suk, Minsoo and Hong, Soonho. (1984). "An Edge Extraction Technique for Noisy Images." Computer Vision Graphics and Image Processing 25, 25-45.

Captions for Figures

Figure 1. Identification quantile functions are graphed truncated at $\pm$ 1. The uniform distribution appears as a line from (0,-.5) to (1,.5). The normal distribution appears as the curve which coincides with the line in the neighborhood of u=0.5 because the functions have been normalized.

Figure 2, 3. Identification quantile plot of a vector or time series plots dimensionless values formed by subtracting median from original value, and dividing the result by twice the interquartile range. The pixel vectors plotted represent simulated mineral spectral reflectance data given in Section 1.

Figure 4. Comparison quantile function (defined in section 4) tests for the equality of distribution of the two samples formed from the class 1 and 2 pixel vectors defined in section 1.

Figure 5. Identification quantile-quantile plot for comparing the equality of distribution of the class 1 and 2 pixel vectors.

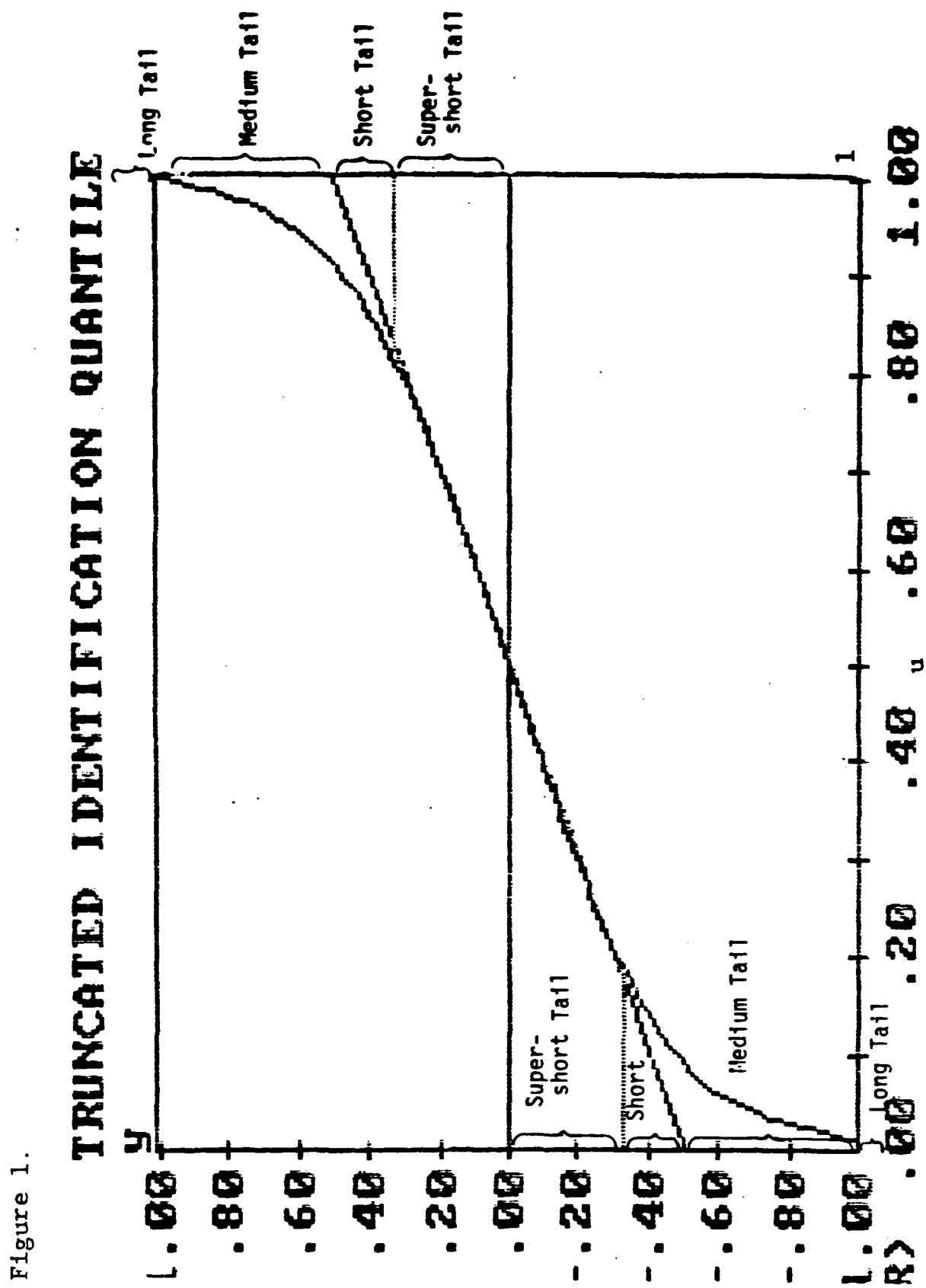

TRUNCATED IDENTIFICATION QUANTILE
y
Long Tail
Medium Tail
Short Tail
Super-
short Tail
1.00
.80
.60
.40
.20
.00
-.20
-.40
-.60
-.80
-1.00
.00
.20
.40
.60
.80
1.00
u
Super-
short Tail
Short
Medium Tail
Long Tail

Figure 1.

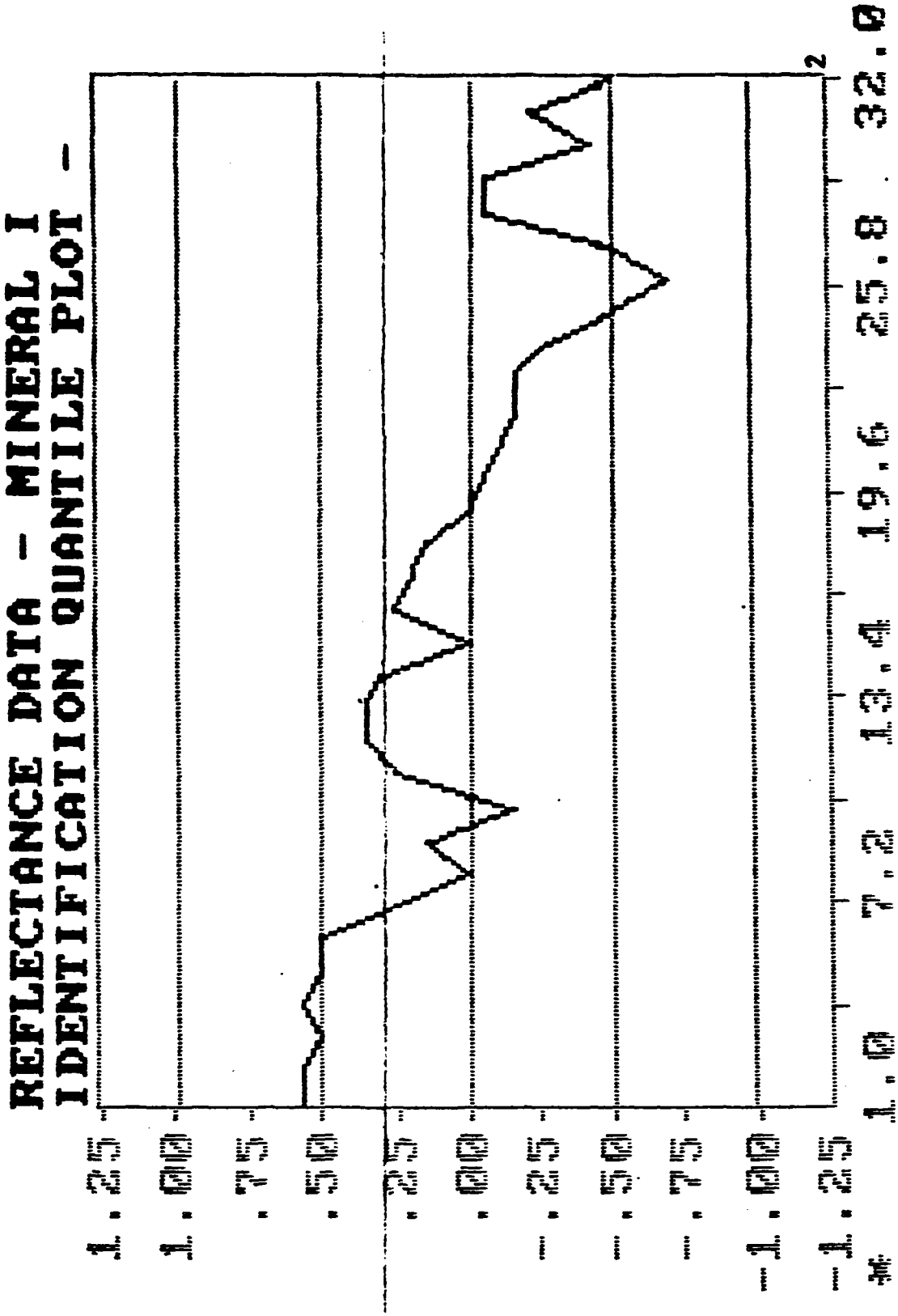

REFLECTANCE DATA - MINERAL I
IDENTIFICATION QUANTILE PLOT -
1.250
1.000
.750
.500
.250
.000
-.250
-.500
-.750
-1.000
-1.250
-3.0
-2.5
-2.0
-1.5
-1.0
-.5
0.0
.5
1.0
1.5
2.0
2.5
3.0
2

Figure 2.

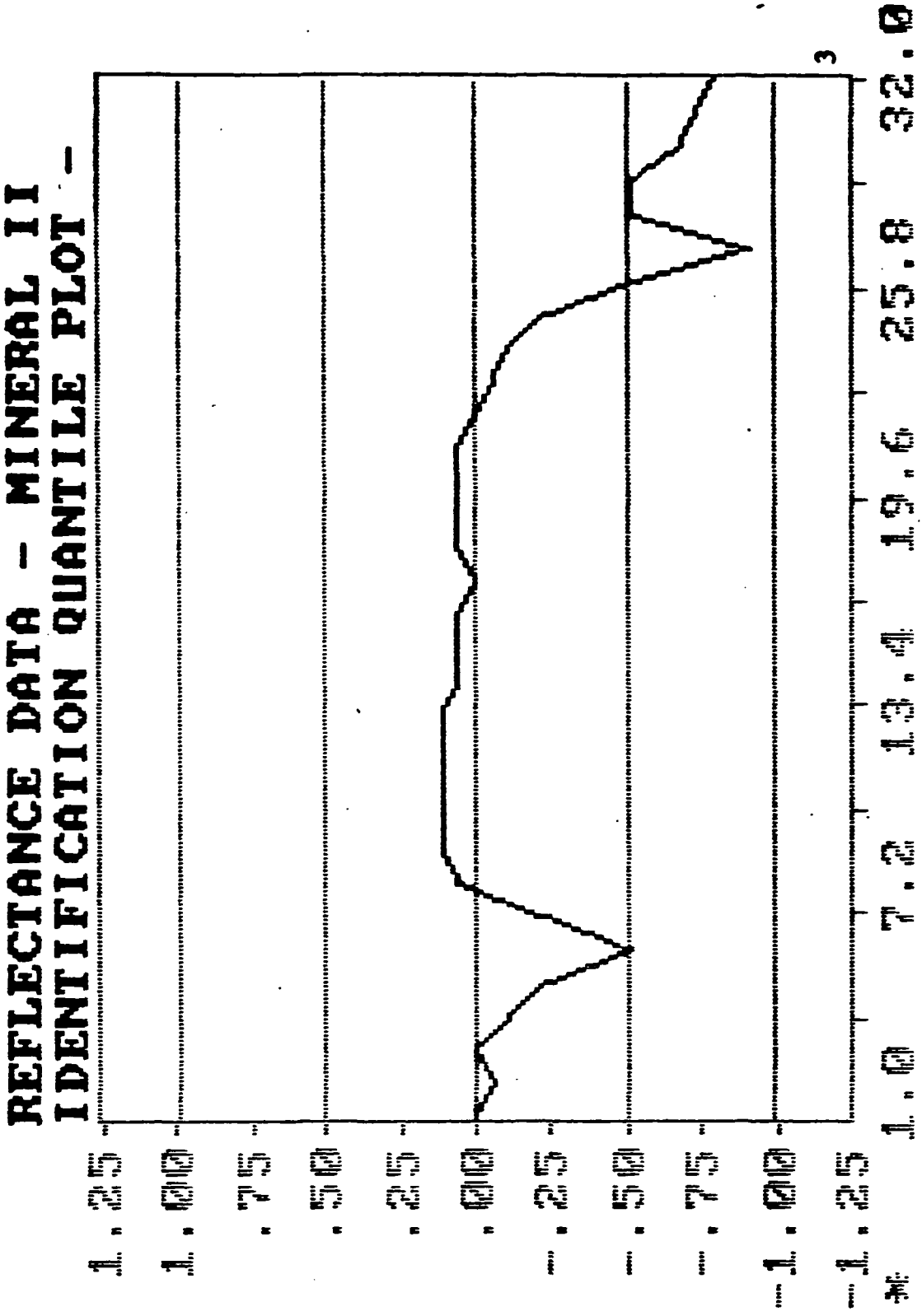

REFLECTANCE DATA - MINERAL II
IDENTIFICATION QUANTILE PLOT -
1.250
1.000
.750
.500
.250
.000
-.250
-.500
-.750
-1.000
-1.250
1.0
7.2
13.4
19.6
25.8
32.0
3

Figure 3.

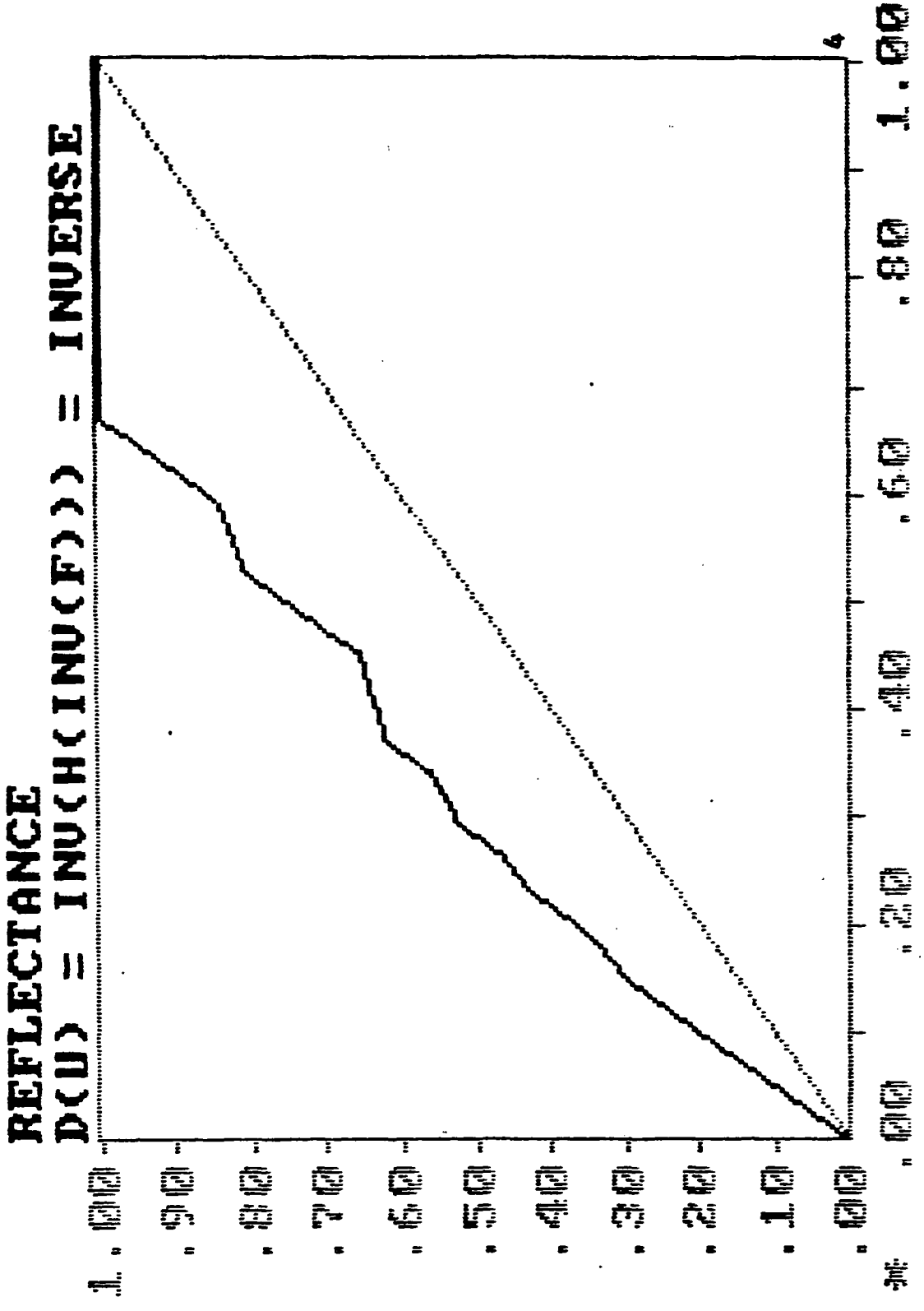

REFLECTANCE
D(U) = INU(H(INU(F))) = INVERSE
1.000
.900
.800
.700
.600
.500
.400
.300
.200
.100
.000
.000
.100
.200
.300
.400
.500
.600
.700
.800
.900
1.000
U

Figure 4.

C-2

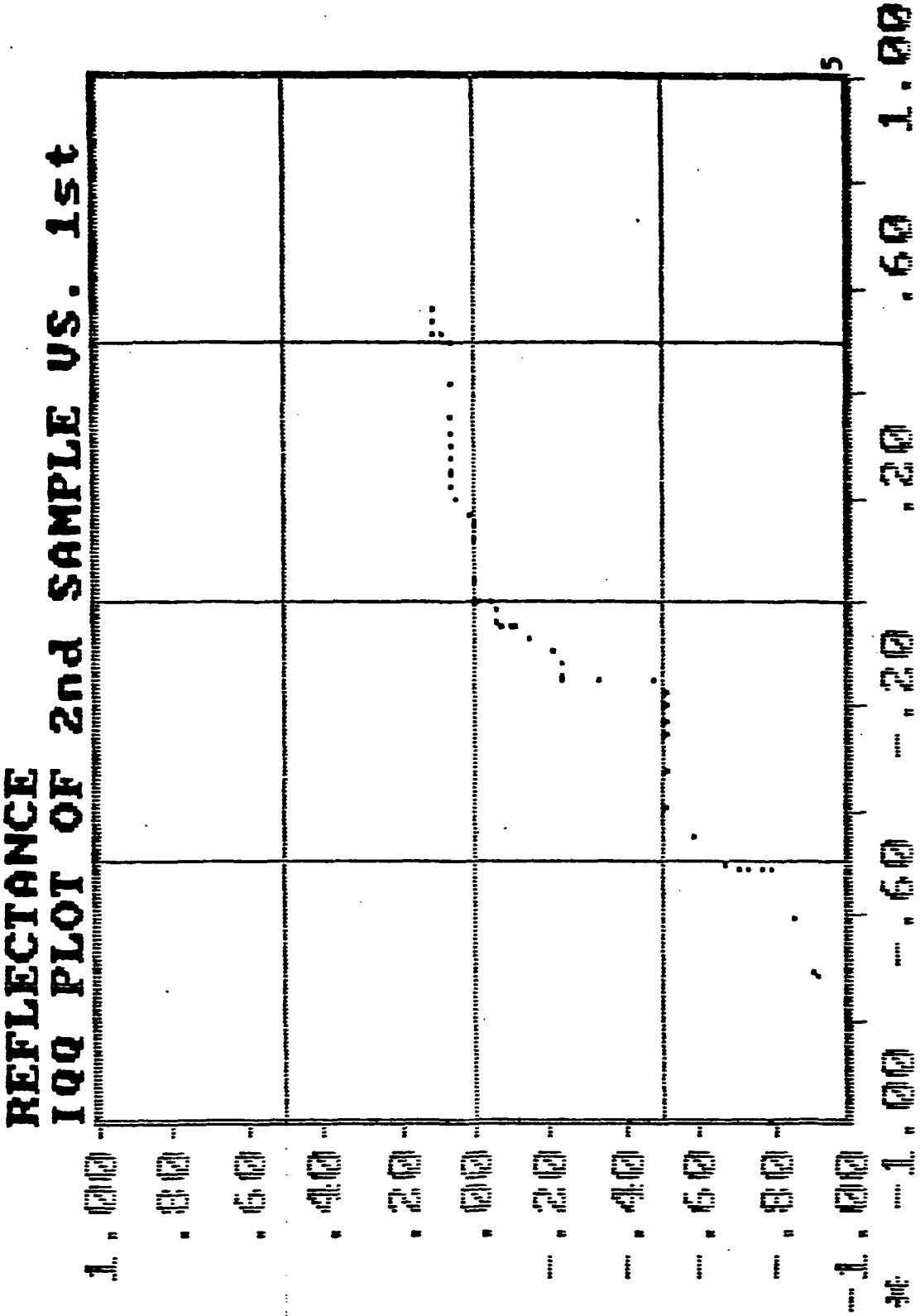

REFLECTANCE
IQQ PLOT OF 2nd SAMPLE US. 1st
1.00
.80
.60
.40
.20
.00
-.20
-.40
-.60
-.80
-1.00
-1.00
-.60
-.20
.20
.60
1.00
S

Figure 5.

Classification in a Spatially Correlated Environment

C. N. Morris, D. V. Hinkley, W. Johnston *

Technical Report No. 20
July 1985.

To Be Published In:
Proceedings of the Third Annual Symposium on
Mathematical Pattern Recognition and Image Analysis,
Texas A&M, June, 1985.

* This investigation was conducted pursuant to Subcontract Agreement No. L20076 to the Texas A&M Research Foundation, College Station, Texas. Funds were provided by the National Aeronautics and Space Administration for a project "Mathematical Pattern Recognition and Image Analysis"

Abstract

Motivated by the LANDSAT problem of inferring crop or geological types at the pixel level by automatic means, we discuss the general empirical Bayes approach to the estimation of n attributes $\theta = (\theta_1, \ldots, \theta_n)$ in a spatial setting, assuming availability of observed data $\mathbf{y} = (y_1, \ldots, y_n)$ made on them. Within the general empirical Bayes paradigm, a spatial logistic estimator is developed for the special case of binary attributes and independent, normal, homoskedastic data. This estimator is relatively simple to compute and provides a logistic estimate at each pixel of the probability $\mathrm{P}(\theta_t = 1 \mid \text{data})$ *without* assuming knowledge of θ ("ground truth") in the region of interest. The rule is shown to perform reasonably well in relation to the "ideal" discriminant rule, which could only be computed with full knowledge of the attribute θ. We conclude with a discussion of technical extensions that could be developed for wider applicability via the empirical Bayes approach.

1. Introduction

Multi-channel satellite image data, available as LANDSAT imagery, are recorded as a multivariate time series (four or more channels, multiple fly-overs) in two spatial dimensions, specifically on a rectangular lattice of points called pixels. A polychotomous attribute, such as crop type, is to be estimated at each pixel from the image data, whose aggregate frequency properties are assumed known in relation to the attribute. The set of attributes forms an attribute map. The regularity may be characterised by spatial correlations. The estimation problem is then one of attribute classification, with spatial correlation among the attribute values.

In an earlier paper (Hill, Hinkley, Kostal, Morris, 1984), various suggestions were made concerning the use of parametric empirical Bayes modeling in this classification problem. Much of the notation and many of the ideas of that earlier paper will be used here. That paper also contains a bibliography of related empirical Bayes literature and the use of Markov random fields as distributions needed for this work.

The attribute at pixel i will be denoted by θ_i, which is polychotomous, i.e., taking on one of $m \geq 2$ values, with $i = (j,k)$ running over a rectangular lattice $j = 1,...,J; k = 1,...,K$. Measurement data y_i are reduced forms of imagery data, e.g. Badhwar numbers, which have a joint frequency distribution $f(\mathbf{y} \mid \theta)$ conditional on the underlying attribute map parameters θ. The empirical Bayes perspective of the problem also adds a family of joint prior distributions Π_α on $\theta, \alpha \in A$ for the attributes. These distributions are chosen to incorporate varying degrees of correlation, this being adaptable to a particular application through the free parameter α.

With this description of the problem, our goal is to estimate posterior probabilities $P_\alpha(\theta_i \mid \mathbf{y})$ for each pixel, either for direct use in global inventory of attributes, or in classification, such as map construction. We focus attention here on estimates of the posterior probabilities approximated by a logistic form, with predictor variables determined

by the image data in neighborhoods of the pixel of interest. After reviewing earlier work in Section 2, this logistic procedure is described in a spatial setting for binary θ's in Section 3. Section 4 illustrates performance of the new procedure on some trial data sets, revealing good performance relative to "ideal" spatially-based classifiers. Desirable future generalizations of this approach are outlined in Section 5.

2. Review of Previous Theory

The objective is to estimate the attribute map $\theta = \{\theta_{jk} : j = 1,...,J, k = 1,...,K\}$ given the image data. For convenience, we specialize immediately to binary attributes.

A. Distributions for Observed Data.

The simple potentially useful distribution for observed data y_{jk} in pixel (j,k) involves binary θ_{jk}'s, with the univariate y_{jk}'s conditionally independent and $\sim \mathrm{N}(\mu_t, \sigma^2 \mid \theta_{jk} = t)$. The parameters $\{\mu_t\}$ and σ^2 are taken as known, since they are assumed to have been estimated precisely from training set data. In fact much of the theory does not depend on normality, but only on conditional independence of the y_{jk} with density $f_t(y)$ given $\theta_{jk} = t$, $t = 0$ or 1. Then the likelihood function of θ depends on the image data only through the "discriminants"

$$\mathrm{U} = \log\left\{\frac{f_1(y)}{f_0(y)}\right\}, \tag{2.1}$$

$$\mathrm{lik}(\theta \mid \mathbf{y}) = \exp\left(\sum \theta_{jk} u_{jk}\right); \tag{2.2}$$

in the homoskedastic normal case above,

$$u_{jk} = \left(\frac{\mu_1 - \mu_0}{\sigma}\right)\left(\frac{y_{jk} - \frac{1}{2}(\mu_0 + \mu_1)}{\sigma}\right). \tag{2.3}$$

Because of (2.3), and because μ_0, μ_1, σ are known, preliminary location and scale changes of the data permit us to take $\overline{\mu} \equiv \frac{(\mu_0+\mu_1)}{2}$ to be zero and $\sigma = 1$ without essential

loss of generality, and be left only with the parameter

$$\delta \equiv \frac{(\mu_1 - \mu_0)}{\sigma}. \tag{2.4}$$

Thus (2.3) reduces to $u_{jk} = \delta y_{jk}$ and μ_1 and μ_0 are replaced by $\frac{\delta}{2}$ and $-\frac{\delta}{2}$ respectively.

B. Distributions for the Unobserved Parameters.

The spatial structure evidenced in blocks (fields) of common attribute values has been approximated through Markov models for the θ_{jk}'s. The simplest instance of this involves a line transect on the lattice, e.g. the j^{th} row of pixels $(j,1),...,(j,K)$, on which the first-order Markov model is

$$\mathrm{P}(\theta_{j,k+1} \mid \theta_{j,k} = t) = p_t = 1 - q_t, \; t = 0 \quad \text{or} \quad 1. \tag{2.4}$$

The parameters $\alpha = (p_0, p_1)$ characterize the lengths of blocks of common attributes.

We discussed in (Hill et. al., 1984) that the posterior log odds ratio on the j^{th} horizontal transect is approximately of moving average form

$$\lambda_k(\mathbf{y}) \equiv \log\left\{\frac{\mathrm{P}(\theta_{jk} = 1 \mid \mathbf{y})}{\mathrm{P}(\theta_{jk} = 0 \mid \mathbf{y})}\right\} \doteq \log\left(\frac{\pi_1}{\pi_0}\right) + x_{k0} + \sum\nolimits_{i=1}^{r} \gamma_i x_{ki} \tag{2.5}$$

for r large, $x_{ki} \equiv (y_{j,k+i} + y_{j,k-i})/2$, the average of pixel readings i units from the k^{th} pixel, and with $\pi_1 \equiv \mathrm{P}(\theta_{jk} = 1) \equiv 1 - \pi_0$. The approximation is most accurate if the discriminatory power between the two cases f_1 and f_0 is small. As the discrimination increases, the logistic form is less appropriate for these posterior probabilities, but then the probability of correct classification improves greatly so that the need for an optimal classifier is not as great.

A few comments are in order. First, even in this simple first order Markov case, the exact Bayes approach gives a complicated joint posterior for θ, whose maximization or minimization is non-trivial, and for which efficient (likelihood) estimation of $\alpha = (p_1, p_0)$ is difficult. Second, the form of $\lambda_k(\mathbf{y})$ in (2.5) is adaptable to priors more general than

the first-order Markov distribution and can be demonstrated to hold in low discrimination cases for more general prior distributions $\pi_\alpha(\theta)$, including two dimensional situations. We discuss this further in subsequent sections.

Moving from the transect to the full lattice, the natural generalization of the Markov prior distribution (2.4) is the Gibbs distribution (Section 4 of Hill et. al., 1984) in which θ_{jk} depends on surrounding θ's only through attributes in neighboring pixels. For example, the isotropic first-order model $\pi_\alpha(\theta)$ would give, with $\alpha = (\beta_0, \beta_1)$,

$$(2.6) \qquad \log\left\{\frac{P(\theta_{j,k}=1 \mid \text{ other } \theta's)}{P(\theta_{j,k}=0 \mid \text{ other } \theta's)}\right\} = \beta_0 + \beta_1(\theta_{j-1,k} + \theta_{j+1,k} + \theta_{j,k-1} + \theta_{j,k+1}).$$

Such models can be integrated with the likelihood function (2.2) to give a manageable joint posterior for θ provided β_0 and β_1 are known. With this provision, a time-consuming relaxation-annealing algorithm (Geman and Geman, 1984) is available to calculate the posterior mode of θ given $\mathbf{x}$.

There are very real attractions to the Gibbs distribution. But these attractions are offset by difficulties, even in the binary case which we have been discussing. First, the marginal likelihood for parameters $\alpha = (\beta_0, \beta_1)$ seems quite intractible. Second, we want more than the posterior mode for θ, we also want to know $P(\theta_i \mid \mathbf{x})$. Third, the iterative algorithm can be very time consuming in large problems.

3. Spatial Logistic Classification

We turn now to the main result, the development of an *automatic* spatial statistical method for estimating the probabilities of a dichotomous attribute at each pixel *that does not utilize training attribute data* from the target site. This last feature is most significant. For example, in applications to LANDSAT data, automatic methods (i.e. methods not utilizing a human "analyst") commonly assume a sample of ground truth attributes θ in the target site in order to provide an appropriate prediction formula for the unobserved attributes in that site. We do not make that requirement. Instead, we estimate

the distribution of target site characteristics using only the remotely sensed data y, and knowledge of the likelihood function $f(\mathbf{y} \mid \theta)$, which can be obtained from training data in a non-target site.

It is important to realize that the attribute characteristics in the target site may differ widely from those of the training data site for which attribute data are readily available. In such cases serious errors will result from a standard discriminant approach, i.e. one that assumes the prediction relation between θ and y in the training site is the same as that in the target site. For example, in predicting crop types, the relative proportions of crop types and field sizes in a particular site may vary markedly from the corresponding parameters in the target site, and these parameters will affect vitally the predictions of θ from y. Thus, the target-site attributes θ must be determined from information in the target site. We are saved, however, if the likelihood function $f(\mathbf{y} \mid \theta)$ is the same in the training and target sites, for then the crop proportion and field size parameters can be estimated from the available data y, without direct observation of θ.

Numerous simplifying assumptions are made in this report relative to the complications presented by LANDSAT data. For example, independence of the $\{y_i\}$'s conditional on a fixed ground truth attribute, is assumed. We allow no split pixels. We concentrate mainly on the binary case. Border effects are ignored. We do not assume multivariate data or data from multiple satellite fly-overs. We justify making these simplifying assumptions here in order to concentrate on one fundamental advance needed for some LANDSAT applications, i.e. the unavailablility of target site attribute training data, and because the assumptions made here should be appropriate for less complicated situations, e.g. for black and white image processing and restoration. Even so, the results that follow could apply directly to certain summary functions of LANDSAT data, despite some model failures.

Apart from the particular results developed here, we also note that the empirical Bayes viewpoint in general provides useful insights into the more complicated situations described. For example, the empirical Bayes model makes clear that one proper use of

training data from sites other than the target site is to determine optimal pixel-level data reductions, i.e. the likelihood ratio statistic. In the LANDSAT case, the Badhwar numbers, which summarize data from multiple fly-overs, as well as the "greenness" and "brightness" functions of multidimensional spectral data are examples of efficient reductions to which our methods might apply directly. On another level, the empirical Bayes model allows the conclusion that the bulk of the correlation in the target area measurement $\{y_i\}$ observations may be due to correlation introduced from the ground truth $\{\theta_i\}$ process. If significant correlation remains in the conditional distribution of $\mathbf{y}$ given θ, perhaps caused by cloud cover and other effects, then in principle this correlation can be modeled within the empirical Bayes framework and used to obtain alternative results for correlated likelihoods.

3.1 Models for data and parameters.

As in Section 2, we assume that at the pixel $i = (k,l)$ in the lattice we make the observation y_i such that

$$y_i \overset{\text{ind}}{\sim} \mathrm{N}(\delta(\theta_i - 0.5), 1),\ i = 1, \ldots, n. \tag{3.1}$$

This distribution is conditional on the attribute vector $\theta = (\theta_1, \ldots, \theta_n)$ of binary values θ_i = 0 or 1. Increasing values of the known parameter $\delta \geq 0$ will yield greater discrimination power. We also assume a spatially isotropic (invariant under translations and rotations) distribution for the vector θ with $\pi_1 \equiv \mathrm{P}(\theta_i = 1) = 1 - \pi_0$ and auto-covariance function $\phi_t = \mathrm{Cov}(\theta_{k,l}, \theta_{k,l+t}) = \mathrm{Cov}(\theta_{k,l}, \theta_{k+t,l})$, which depends on t, but not on k, l. The corresponding correlations ρ_t then satisfy

$$\rho_t = \mathrm{Corr}(\theta_{k,l}, \theta_{k,l+t}) = \phi_t/(\pi_0\pi_1). \tag{3.2}$$

3.2 Logistic form

Because the distribution of θ_j given θ_i depends only on the physical distance between pixels i and j, then as $\delta \to 0$ it follows that in this lattice case, as previously in the transect case (2.5), that the logistic approximation holds for pixel i. Define

$$p_i \equiv \mathrm{P}(\theta_i = 1 \mid \mathbf{y}) \equiv 1 - q_i,\ \lambda_i(\mathbf{y}) \equiv \log(p_i/q_i). \tag{3.3}$$

Then

$$\lambda_i(\mathbf{y}) \doteq \log\left(\frac{\pi_1}{\pi_0}\right) + \sum_{t=0}^{r} \gamma_t \bar{x}_{it} \tag{3.4}$$

with $\bar{x}_{it}$ the average of measurements in the t^{th} "ring" away from pixel i. Neighbors of pixels at a fixed distance away from pixel i are called "rings", denoted R_0, R_1, R_2, etc with $R_0 = R_{i0}$ being the zeroth ring (the pixel itself), $R_1 = R_{i1}$ the four nearest points, and so on, as in Figure 3.1.

5	4	3	4	5
4	2	1	2	4
3	1	0	1	3
4	2	1	2	4
5	4	3	4	5

Figure 3.1

Location of pixels comprising rings $R_0, \ldots, R_5$

relative to pixel i at center.

Formula (3.4) defines $\bar{x}_{i0} = y_i$, $\bar{x}_{i1}$ = ring 1 average for pixel $i = (k, l)$, so

$$\bar{x}_{i1} = (y_{k,l+1} + y_{k-1,l} + y_{k,l-1} + y_{k+1,l})/4$$

for $t = 1$, and so on, following Figure 3.1. (We ignore here, for convenience, the question of how to modify these definitions at the borders of the region.) These averages depend

symmetrically on the ring elements because of the isotropic assumption. If δ is not small, however, the posterior probability (3.3), (3.4) will not depend on the data in a linear way, and in such cases the ring averages then are not completely adequate for use in the approximation. Nevertheless, we continue to use the ring averages and the logistic approximation for moderate δ for simplicity and because discrimination will be accurate for large δ even for this non-optimal logistic classifier; see also Switzer (1980).

Suppose momentarily that the values $\{\theta_i\}$ are known and available to compute discriminant probabilities for predicting θ_i in linear logistic form from the observed $\bar{x}_{i0} = y_i$, $\bar{x}_{i1}, \ldots, \bar{x}_{ir}$, $i = 1,2,\ldots,n$. Here r is the number of rings used; in Figure 3.1 and for the applications of Section 4, we take $r = 5$. Let $\bar{\theta} \equiv \sum \theta_i/n$. The discriminant function $\lambda_i(\mathbf{y}, \theta)$ such that

$$\text{(3.5)} \qquad \mathrm{P}(\theta_i = 1 \mid \mathbf{y}) = \frac{1}{1 + \exp(-\lambda_i(\mathbf{y}, \theta))}$$

is computed by

$$\text{(3.6)}, \qquad \lambda_i(\mathbf{y}, \theta) = \log\left(\frac{\bar{\theta}}{1-\bar{\theta}}\right) + \frac{1}{RSS(\theta)} \sum_{t=0}^{r} b_t(\theta)(\bar{x}_{it} - \bar{m}_t(\theta)),$$

with the quantities $RSS(\theta), b_t(\theta)$ and $\bar{m}_t(\theta)$ defined in (3.9) through (3.11). See Morris and Rolph (1981, pp 206 and 88-89) for this development of discriminant estimation.

The quantities $\bar{\theta}, b_t(\theta), \bar{m}_t(\theta)$ and $RSS(\theta)$ in (3.6) can be estimated as follows. Define the $n \times (r+1)$ data matrix to be

$$\text{(3.7)} \qquad \mathbf{X} = \begin{pmatrix} y_1 - \bar{y} & \bar{x}_{1,1} - \bar{x}_1 & \ldots & \bar{x}_{1,r} - \bar{x}_r \\ \vdots & \vdots & \ddots & \vdots \\ y_n - \bar{y} & \bar{x}_{n,1} - \bar{x}_1 & \ldots & \bar{x}_{n,r} - \bar{x}_r \end{pmatrix}$$

with $\bar{y}, \bar{x}_1, \ldots, \bar{x}_r$ the averages of $y_i, \bar{x}_{i1}, \ldots, \bar{x}_{ir}$ for rings $R_0, R_1, \ldots, R_r$, so that the columns of $\mathbf{X}$ add to zero (in a large area, we will have approximately $\bar{y} = \bar{x}_1 = \ldots = \bar{x}_r$, the errors

occuring because of border effects). Then, letting $\mathbf{b}(\theta)$ be the vector $(b_0(\theta),\ldots,b_r(\theta))'$, and

$$\mathbf{C}(\theta) \equiv \mathbf{X}'\theta/n, \tag{3.8}$$

we have the expressions

$$\mathbf{b}(\theta) = \mathbf{S}\mathbf{C}(\theta), \quad \mathbf{S} \equiv n(\mathbf{X}'\mathbf{X})^{-1}, \tag{3.9}$$

and

$$RSS(\theta) = \bar{\theta}(1-\bar{\theta}) - \mathbf{C}'(\theta)\mathbf{S}\mathbf{C}(\theta). \tag{3.10}$$

The quantities $\bar{m}_t = \bar{m}_t(\theta)$, in (3.6) are the unweighted averages of the $\bar{x}_{t,0}$ and $\bar{x}_{t,1}$, respectively of the t^{th} ring averages for pixels with $\theta_i = 0$ and $\theta_i = 1$, i.e.

$$\bar{m}_t(\theta) = \frac{1}{2}\frac{\sum \theta_i \bar{x}_{it}}{\sum \theta_i} + \frac{1}{2}\frac{\sum (1-\theta_i)\bar{x}_{it}}{\sum (1-\theta_i)}, \quad t = 0,\ldots,r. \tag{3.11}$$

After some algebra, the $r+1$ vector $\bar{\mathbf{m}}(\theta)$ of elements (3.11) can be re-written as

$$\bar{\mathbf{m}}(\theta) = \bar{\mathbf{x}} + \frac{1-2\bar{\theta}}{2\bar{\theta}(1-\bar{\theta})}\mathbf{C}(\theta), \tag{3.12}$$

with $\bar{\mathbf{x}}$ the vector $(\bar{y},\bar{x}_1,\ldots,\bar{x}_r)'$.

We see from (3.9), (3.10) and (3.12) that we do not need to know all the attribute values $\{\theta_i\}$ to compute (3.6), but only the $r+2$ linear combinations $\bar{\theta}, \mathbf{C}(\theta)$, and, of course, the quantities $\mathbf{X}, \mathbf{S}, \bar{\mathbf{x}}$ which are directly available from data in the target site.

Note that $\mathrm{E}(\bar{y}) = \delta(\bar{\theta} - 0.5)$ from (3.1) and hence that $\bar{\theta}$ has unbiased estimate

$$\hat{\pi} = \frac{1}{2} + \bar{y}/\delta. \tag{3.13}$$

This notation is used because $\hat{\pi}$ is also an unbiased estimate of $\pi \equiv \mathrm{P}(\theta_i = 1)$.

Define the sample autocovariances of elements in ring 0 with those in ring t by

$$c_t \equiv \frac{1}{n}\sum y_i(\bar{x}_{it} - \bar{x}_t), \quad t = 0,1,\ldots,r. \tag{3.14}$$

From (3.1), write $y_i = \delta(\theta_i - 0.5) + z_i,\ z_i \sim \mathrm{N}(0,1)$. Then from (3.14),

$$\begin{aligned} c_t - \delta c_t(\theta) &= \frac{1}{n}\sum\nolimits_{i=1}^{n}(y_i - \delta\theta_i)(\overline{x}_{it} - \overline{x}_t) \\ &= \frac{1}{n}\sum\nolimits_{i=1}^{n} z_i\Big\{(\overline{z}_{it} - \overline{z}_t) + \delta(\overline{\theta}_{it} - \overline{\theta}_t)\Big\} \end{aligned}$$

with $\overline{z}_{it}$ and $\overline{\theta}_{it}$ indicating ring t averages for pixel i. For $t \geq 1$, this has expectation, given θ, equal to $-r_t/n^2$ with r_t the number of pixels in ring t. Thus, a nearly unbiased estimate of $c_t(\theta)$ is

$$c_t/\delta, \quad t = 1, \ldots, r; \tag{3.15}$$

this could be made exactly unbiased if r_t/n^2 were added. For $t = 0$, we have, given θ,

$$\mathrm{E}\Big\{c_0(\theta)\Big\} = \frac{1}{n}\mathrm{E}\Big\{\sum \theta_i(y_i - \overline{y})\Big\} = \delta\frac{1}{n}\sum \theta_i(\theta_i - \overline{\theta}) = \delta\overline{\theta}(1 - \overline{\theta}).$$

Thus, a nearly unbiased estimate of $c_0(\theta)$ is $\delta\hat{\pi}(1 - \hat{\pi})$. (Actually, these estimates of $c_t(\theta)$ are "empirical Bayes unbiased", which means they have the same expectation as the random quantities they estimate.) We now state the the estimation results formally.

Main Result: Empirical Bayes Logistic Spatial Estimator. The discriminant function $\lambda_i(\mathbf{y}, \theta)$ in (3.6), which yields probability

$$\mathrm{P}(\theta_i = 1 \mid \mathbf{y}) \equiv \frac{1}{1 + \exp\{-\lambda_i(\mathbf{y}, \theta)\}},$$

may be estimated under the distributional assumption (3.1) and the isotropic assumption for θ by

$$\hat{\lambda}_i(\mathbf{y}) = \log\{\hat{\pi}/(1 - \hat{\pi})\} + \sum\nolimits_{t=0}^{r} \mathrm{b}_t(\overline{x}_{it} - \mathrm{M}_t) \tag{3.16}$$

with

$$\mathrm{b} = (\mathrm{b}_0, \mathrm{b}_1, \ldots, \mathrm{b}_r)' = \mathbf{SK}/\mathrm{W} \quad , \tag{3.17}$$

where

$$\mathbf{K} \equiv (\delta\hat{\pi}(1-\hat{\pi}), \frac{c_1}{\delta}, \ldots, \frac{c_r}{\delta})' \tag{3.18}$$

estimates $\mathbf{C}(\theta)$ using elements defined in (3.13) and (3.14), $\mathbf{S}$ is given by (3.9), and

$$\mathrm{W} \equiv \hat{\pi}(1-\hat{\pi}) - \mathbf{K}'\mathbf{S}\mathbf{K} \tag{3.19}$$

estimates $RSS(\theta)$ in (3.10). In practice we will force $W \geq 0.05\hat{\pi}(1-\hat{\pi})$ in order to be sure that the resulting estimate of $RSS(\theta)$ cannot be negative, or an unstable value close to 0. The quantities $M_0, \ldots, M_r$ are nearly unbiased estimates of $\overline{m}_0(\theta), \ldots, \overline{m}_r(\theta)$, being defined by

$$M_t \equiv \overline{x}_t - \left(\frac{\overline{y}}{\delta^2}\right)\left(\frac{c_t}{\hat{\pi}(1-\hat{\pi})}\right) \quad \text{for} \quad t \geq 1 \quad \text{and} \quad M_0 \equiv 0. \tag{3.20}$$

Because of the remarks following (3.7) we have $M_t \doteq \overline{y}[1 - c_t/\delta^2\hat{\pi}(1-\hat{\pi})]$ for $t \geq 1$.

Formula (3.16) estimates the discriminant function without knowledge of θ, but by using the target area average $\overline{y}$ and the autocorrelations $c_1, \ldots, c_r$. These same statistics also can be used to estimate the characteristics

$$\alpha = (\pi, \phi_1, \ldots, \phi_r)' \tag{3.21}$$

of the attribute (θ) process, assuming isotropy with $\pi = \mathrm{P}(\theta_i = 1)$, and ϕ_t the covariance between attributes θ_i and θ_j with θ_j in the t^{th} ring for θ_i. Thus we assume the main characteristics of the binary attribute process θ are summarized by the spatial covariance $\{\phi_t\}$ or spatial correlation $\{\rho_t\}$, $\rho_t = \phi_t/(\pi(1-\pi))$ and the probability π. In an application to binary crop-type estimation, π represents the proportion of pixels assigned to a particular crop type, and the spatial correlations $\{\rho_t\}$ characterize the field sizes. These same parameters can be chosen to govern an isotropic Markov random field (MRF) of order r, and hence the methods developed here compete with empirical Bayes procedures that assume isotropic MRF distributions for the attribute process. Because the rule (3.16) estimates

functions of θ rather than the parameters α of a particular distribution on θ, however, the rule appears to be valid for a wider class of attribute distributions than isotropic MRFs.

4. Behavior of the Binary Logistic Spatial Estimator in Several Test Cases.

The rule given by the main result (3.16), which is termed an "empirical Bayes logistic spatial estimator" (EB-LSE), will be compared with several other rules:

(a) the "ideal" logistic spatial estimator (I-LSE) (3.6) which assumes that the attributes θ are known in order to calculate $\lambda_i(\mathbf{y}, \theta)$;

(b) an "ideal" logistic non-spatial estimator (I-LNSE), which uses the pixel level information y_i only, approximately the estimator (3.6) when $r = 0$:

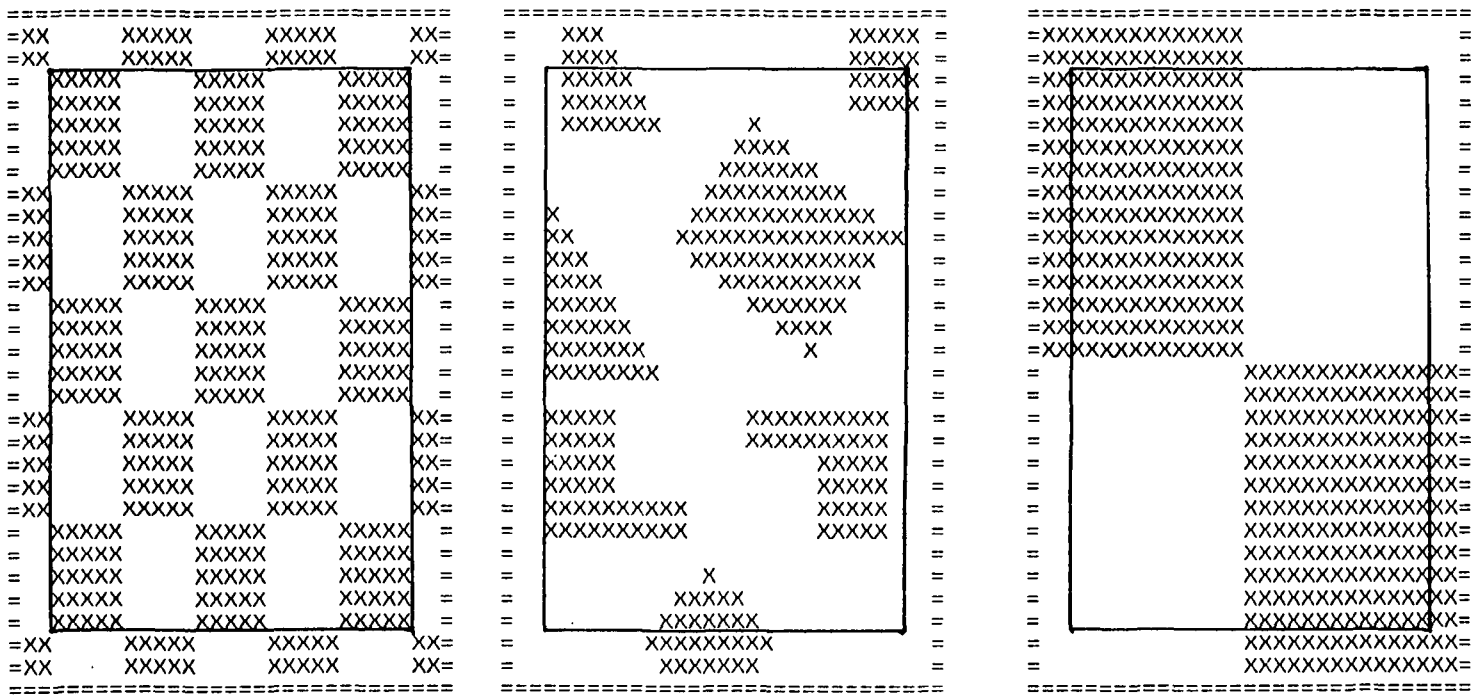
$$\text{(4.1)} \qquad \mathrm{P}(\theta_i = 1 \mid y_i) = \frac{\overline{\theta}\exp(\delta y_i)}{1 - \overline{\theta} + \overline{\theta}\exp(\delta y_i)};$$

and,

(c) an empirical Bayes logistic non-spatial estimator (EB-LNSE), which is (4.1) but replacing $\overline{\theta}$ by $\hat{\pi} = 0.5 + \overline{y}/\delta$, as in (3.13). This is the EB-LSE rule (3.16) for $r = 0$, except that $s_y^2 = \sum(y_i - \overline{y})^2/n$ is replaced by an estimate of its expectation $1 + \delta^2\hat{\pi}(1 - \hat{\pi})$ when necessary.

The four estimators will be compared in nine different environments, with all combinations of $\delta = 1.0, 1.5, 2.0$ and three different ground truth maps with $n = 625$ pixels in a 25 by 25 grid. In each case the grid is extended to a 29 by 29 ($n = 841$) grid in the most obvious manner, in order to provide a border of width two pixels for using neighborhood data with rings $R_0, \ldots, R_5$, as in Figure 3.1. These three θ patterns, labeled "checkerboard" (CKBD), "two by two" (2BY2), and "miscellaneous" (MISC), are shown in Figure 4.1. We chose 2BY2 to exhibit strong spatial correlation in relation to CKBD, and MISC to exhibit non-patterned shapes.

We use several different measures of performance for each rule, assuming the rule assigns the value $p_i = \mathrm{P}(\theta = 1 \mid \mathbf{y})$ to pixel i, $i = 1, \ldots, n$, $n = 625$. They are:



Checkerboard (CKBD) Miscellaneous (MISC) Two-by-Two (2BY2)

Figure 4.1

Three test maps for θ of size 25 by 25 with 29 by 29 backgrounds;

X means $\theta_i = 1$, blank means $\theta_i = 0$.

(a) the percentage of classification errors (%ERR), counting a classification as incorrect if $p_i \geq 1/2$ and $\theta_i = 0$ or $p_i < 1/2$ and $\theta_i = 1$ (it never happened that $p_i = 1/2$ exactly)

$$\%\text{ERR} = 50 - 50\frac{1}{n}\sum \text{sign}\Big[(\theta_i - 0.5)(p_i - 0.5)\Big]; \tag{4.2}$$

(b) the mean absolute error

$$\text{MAE} = \frac{1}{n}\sum \mid p_i - \theta_i \mid; \tag{4.3}$$

(c) the mean squared error

$$\text{MSE} = \frac{1}{n}\sum (p_i - \theta_i)^2; \tag{4.4}$$

and,

(d) the information measure

$$\text{INFO} = -\frac{1}{n}\sum \Big\{\theta_i \log(p_i) + (1 - \theta_i)\log(1 - p_i)\Big\}. \tag{4.5}$$

All four measures are always non-negative, and all are zero if $p_i = \theta_i$ for all i (in the INFO case $p_i = \theta_i$ can occur only in the limit). Small values of each measure are desirable, and rules with generally small values are to be preferred.

All data examples in Tables 4.1 and 4.2 involve one simulation (841 data points) according to $y_i \sim \text{N}(\delta(\theta_i - 0.5), 1)$, with values of $z_i = y_i - \delta(\theta_i - 0.5)$ re-used in all nine examples, so that only δ is changed with the cases. Thus, results are random, but this technique of re-using the z_i values aids by reducing the variability for comparative purposes. The "*Theoretical*" values in parentheses, e.g. (30.8%)in Table 4.1 for $\delta = 1.0$, CKBD, %ERR, are the exact error fractions for the ideal non-spatial estimator I-LNSE computed from the normal distribution in repeated sampling. Comparing these values with %ERR for I-LNSE provides some calibration of these particular data sets to the long run. In this case I-LNSE error rates are slightly larger than expected. The efficiency ("*Efficiency of EB-LSE*") values in Table 4.1 illustrate, on a proportional basis, how close

Table 4.1

Overall error proportions and mean absolute errors for various rules.

	%ERR			MAE		
	CKBD	MISC	2BY2	CKBD	MISC	2BY2
$\delta = 1.0$						
$I - LSE$	20.2	15.5	5.4	.30	.24	.08
$EB - LSE$	24.0	16.8	8.6	.27	.25	.11
$I - LNSE$	32.0	32.0	31.0	.40	.39	.40
(Theoretical)	(30.8)	(29.1)	(30.9)			
$EB - LNSE$	33.0	31.0	32.0	.40	.39	.40
Efficiency of $EB - LSE$	.68	.92	.88	1.30	.93	.91
$\delta = 1.5$						
$I - LSE$	13.1	10.7	2.6	.20	.15	.03
$EB - LSE$	15.2	10.1	3.4	.18	.15	.04
$I - LNSE$	24.0	23.0	23.0	.32	.31	.31
(Theoretical)	(22.7)	(21.6)	(22.7)			
$EB - LNSE$	23.0	22.0	24.0	.32	.31	.31
Efficiency of $EB - LSE$	.81	1.05	.96	1.17	1.00	.96
$\delta = 2.0$						
$I - LSE$	7.8	6.7	0.6	.13	.10	.01
$EB - LSE$	7.4	6.9	1.4	.11	.09	.02
$I - LNSE$	18.0	16.0	18.0	.23	.23	.23
(Theoretical)	(15.9)	(15.2)	(15.9)			
$EB - LNSE$	18.0	17.0	18.0	.23	.23	.23
Efficiency of $EB - LSE$	1.04	.98	.95	1.20	1.08	.95

the EB-LSE measure comes to the I-LSE measure relative to the I-LNSE measure; e.g. for $\delta = 1.0$, CKBD, MSE: *efficiency* = (24.0 - 32)/(20.2 - 32) = 0.68. The EB-LSE proportions for %ERR average 92% efficiency in the nine examples. However, the efficiency drops to as little as 68% in the case with lowest discrimination, i.e. $\delta = 1.0$, CKBD. Of course I-LSE is an impossible-to-meet standard among logistic rules in the long run because: (a) it utilizes the unknown values θ; and (b) it is biased favorably because it uses the true values of θ to predict themselves. The relatively strong performance of the empirical Bayes logistic spatial estimator is very encouraging in these examples.

In terms of the mean absolute error metric, MAE of Table 4.1, EB-LSE performs even better, about as well as I-LSE, averaged over all nine cases. However, the MAE measure is deficient as a measure because it rewards pushing all probability estimates p_i away from $\frac{1}{2}$ and closer to 0 or 1, even if such extreme values are not justified or believed. The EB-LSE rule has a slight defect in this direction and thereby prospers with respect to MAE.

Table 4.2 shows the mean squared errors (MSE) and the information metrics (INFO) for the four estimators in the nine situations. The two measures, unlike MAE, share the property that they reward reporting that p_i which is believed to be the best estimate of $P(\theta_i = 1)$. As with %ERR, in terms of MSE, EB-LSE has average efficiencies of 92% of I-LSE, relative to the ideal non-spatial method. Again, the efficiency varies in direct relation to the discrimination parameter δ, with only 63% efficiency provided when $\delta = 1.0$ in the checkerboard case.

The results for the INFO metric in Table 4.2 parallel those of MSE, with EB-LSE averaging 90.2% efficiency, and the exceptional case again occuring for $\delta = 1.0$, CKBD, where only 50% of the I-LSE efficiency is attained by EB-LSE.

There is, as acknowledged, variability in these results. To check this, the intermediate case $\delta = 1.5$, MISC, was repeated 10 times. In these ten cases %ERR for EB-LSE ranged between 8.3% and 12.2%, with mean 10.0%, making the case considered earlier with %ERR = 10.0% quite central. Figure 4.2 graphs these two extreme %ERR cases for EB-LSE with

Table 4.2

Mean squared errors and information measure for various estimates.

	MSE			INFO		
	CKBD	MISC	2BY2	CKBD	MISC	2BY2
$\delta = 1.0$						
$I - LSE$	.142	.117	.041	.439	.368	.139
$EB - LSE$	.165	.126	.062	.512	.392	.207
$I - LNSE$	.203	.196	.201	.587	.573	.585
$EB - LNSE$	.203	.197	.201	.524	.575	.586
$Efficiency$ $of EB - LSE$	.63	.89	.87	.50	.88	.85
$\delta = 1.5$						
$I - LSE$	.094	.074	.019	.304	.239	.064
$EB - LSE$	.099	.075	.026	.319	.244	.089
$I - LNSE$	.160	.156	.159	.478	.472	.478
$EB - LNSE$	.160	.156	.159	.479	.472	.479
$Efficiency$ $of EB - LSE$	.92	.99	.95	.92	.98	.94
$\delta = 2.0$						
$I - LSE$	.060	.046	.008	.203	.152	.034
$EB - LSE$	.056	.045	.012	.191	.151	.044
$I - LNSE$	.117	.115	.117	.361	.360	.365
$EB - LNSE$	.117	.115	.117	.361	.360	.365
$Efficiency$ $of EB - LSE$	1.06	1.01	.97	1.07	1.01	.97

$\delta = 1.5$, MISC, alongside the case considered earlier in Tables 4.1 - 4.2.

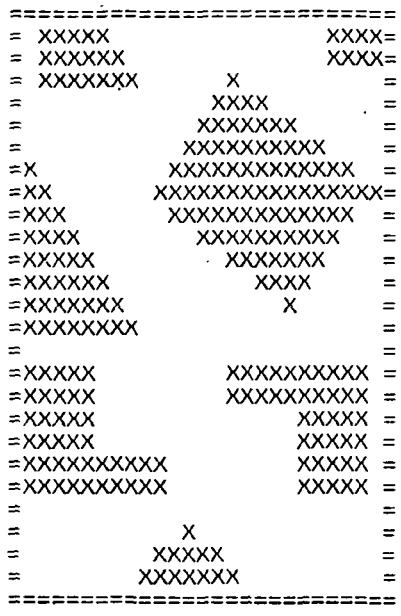
The errors for various estimates in the cases $\delta = 1.0$, CKBD and $\delta = 1.5$, MISC are shown pictorially in Figures 4.3 and 4.4. Assignments for the two logistic rules EB-LSE and I-LSE are made according to $p_i \geq \frac{1}{2}$ or $p_i < \frac{1}{2}$, with resulting %ERR error rates of 10.1% for %EB-LSE and 22.9% for I-LNSE. The spatial rule not only improves on the non-spatial rule, but the greatest improvements occur in the interior of the contiguous regions. This phenomenon of maximal improvement in interiors of regions occurs with the other test cases too, as can be seen from the graphs of EB-LSE performances in Figures 4.5, 4.6 and 4.7, and aids in locating the central masses of large shapes accurately.

The actual error rates for EB-LSE appear in Table 4.3 as a function of the number of nearest neighbors that are of the same type as the center pixel. Thus the possible number of agreements range from 0 to 8, but with CKBD and 2BY2 it is always 4 (at a corner), 5 (on a border), or 8 (for an interior point). Other possibilities occur for MISC, but 4, 5, 6, 7 or 8 agreeing neighbors predominate (otherwise, MISC has 20 pixels with 3 agreeing neighbors, 6 with 2 agreeing neighbors and 1 with 1 agreeing neighbor), and so only those results for N $\geq$ 4 agreeing neighbors are reported in Table 4.3. The only noticeable difference between I-LSE and EB-LSE occurs for CKBD with N = 5, i.e. on edges. In this case I-LSE makes noticeable improvements on EB-LSE for $\delta \leq 1.5$.

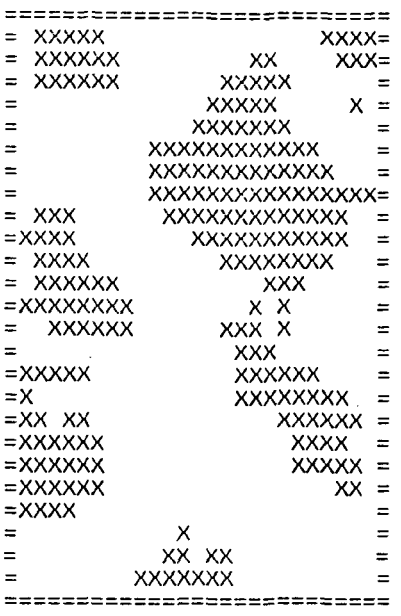
When exactly four of the eight neighbors agree, the value of spatial information diminishes to the point that a spatial rule for these pixels performs about as well as the non-spatial rule EB-NSE (because the neighboring pixels provide noise but no information). More complicated procedures than considered here, ones designed to be sensitive to straight edges, could outperform spatial estimators at such boundary and corner pixels.

Table 4.4 shows the regression coefficients for both EB-LSE and I-LSE for the nine cases, but normalized by the number of pixels in each ring. Instead of displaying $b_t(\theta)$ from (3.9) or B_t from (3.17), we display

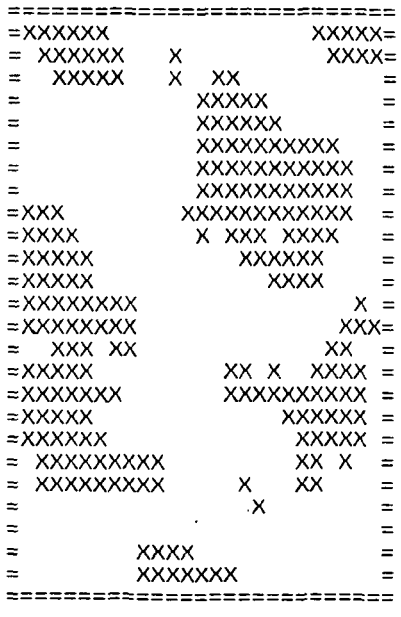
$$b_t^* = b_t(\theta)/r_t \quad \text{or} \quad b_t^* = B_t/r_t \tag{4.6}$$



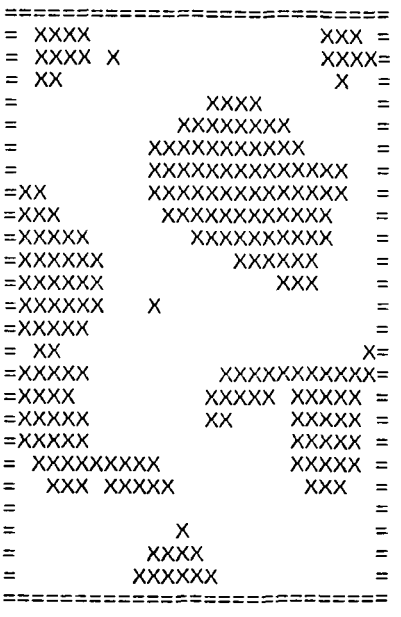
True values, %ERR = 0



Case used, %ERR = 10.1%



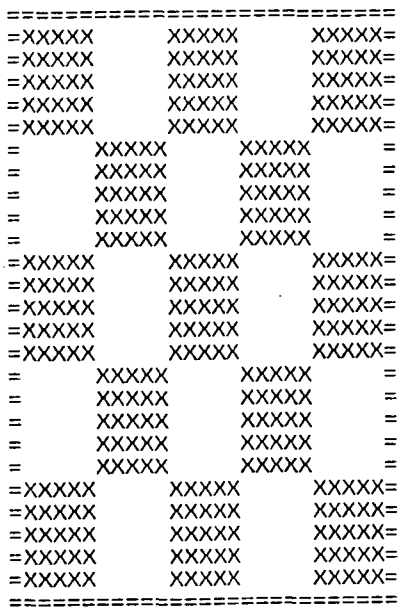
Worst case, %ERR = 8.3%



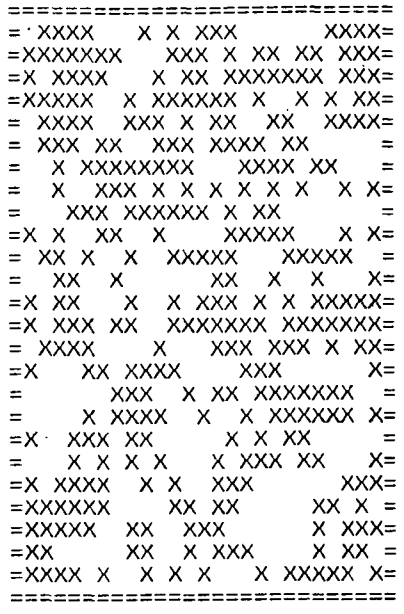
Best case, %ERR = 12.2%

Figure 4.2

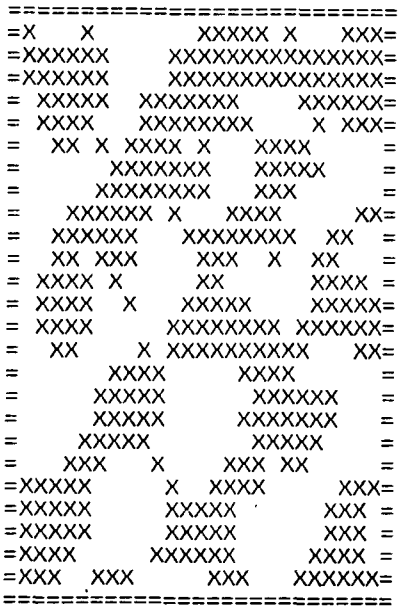
Worst and best cases for EB-LSE in 10 runs of example: $\delta = 1.5$, MISC.



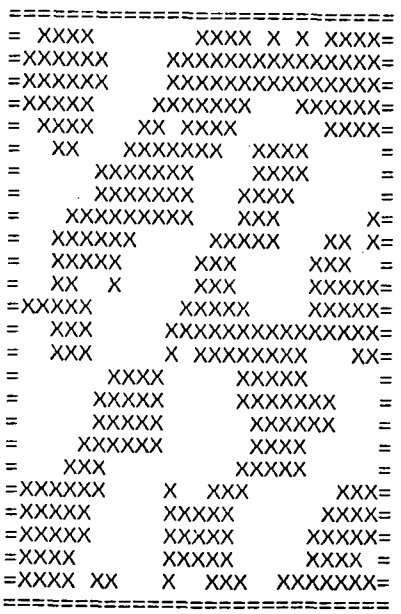
True values, %ERR = 0



I-NLSE, %ERR = 32.0%



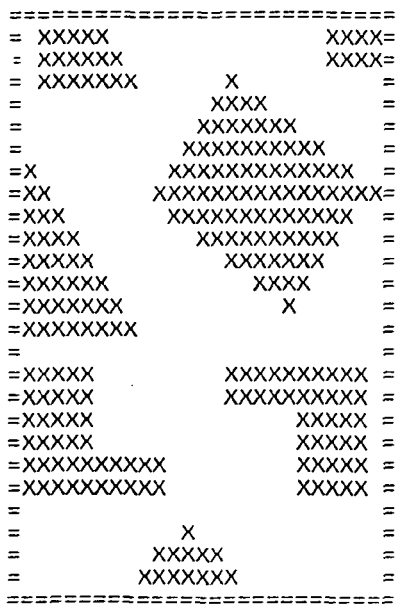
EB-LSE, %ERR = 24.2%



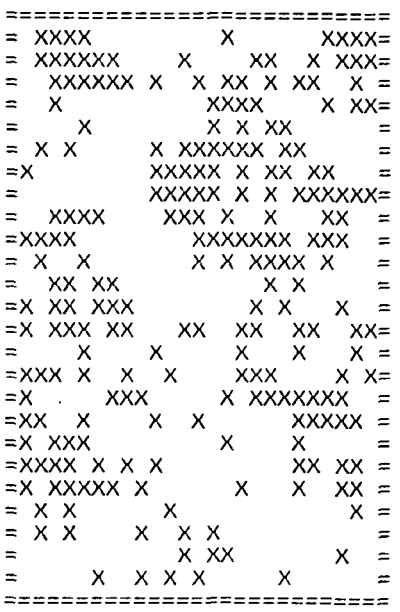
I-LSE, %ERR = 20.2%

Figure 4.3

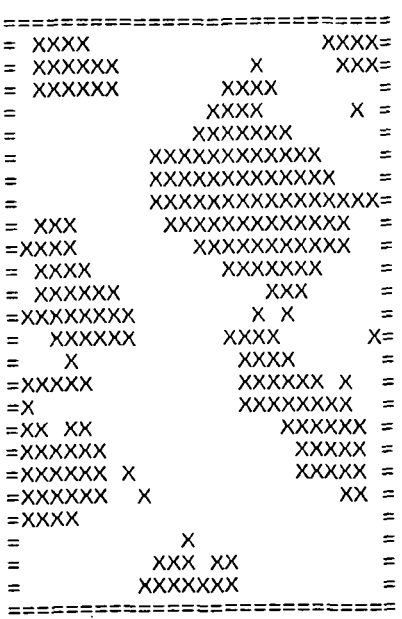
True values and assignments made by three rules. Case: $\delta = 1.0$, CKBD.



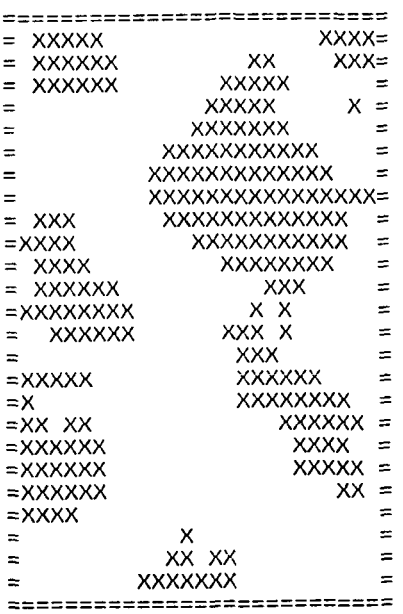
True values, %ERR = 0



I-NLSE, %ERR = 23.0%



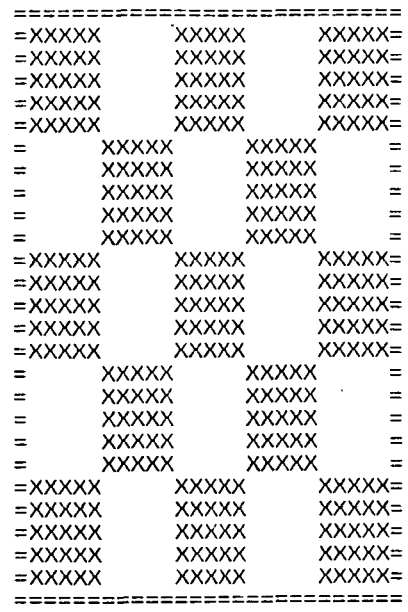
EB-LSE, %ERR = 10.1%



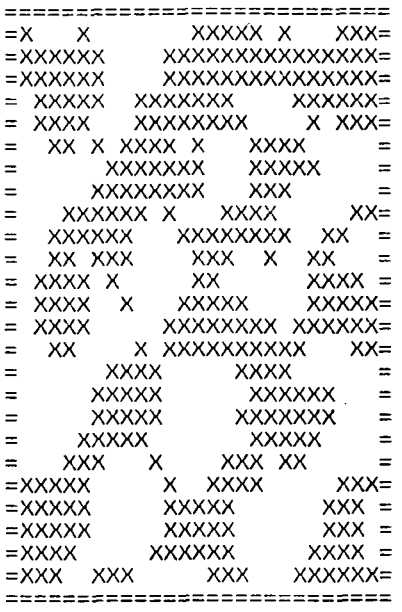
I-LSE, %ERR = 10.7%

Figure 4.4

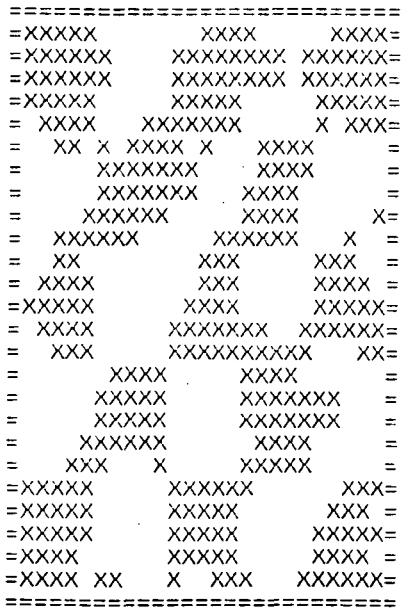
True values and assignments made by three rules. Case: $\delta = 1.5$, MISC.



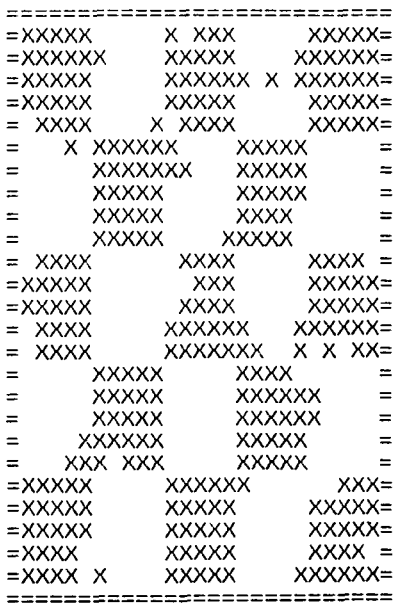
True values, %ERR = 0



$\delta = 1.0$, %ERR = 24.0%



$\delta = 1.5$, %ERR = 15.2%

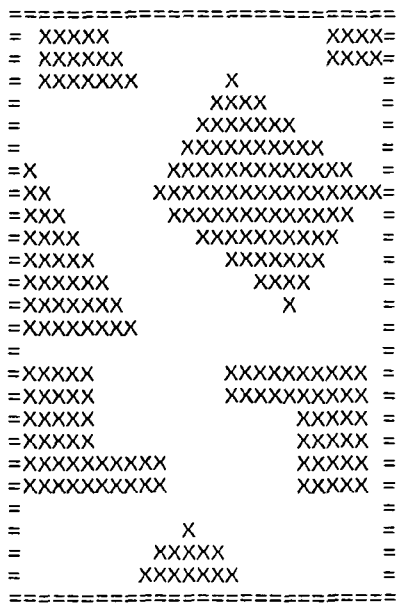


$\delta = 2.0$, %ERR = 7.4%

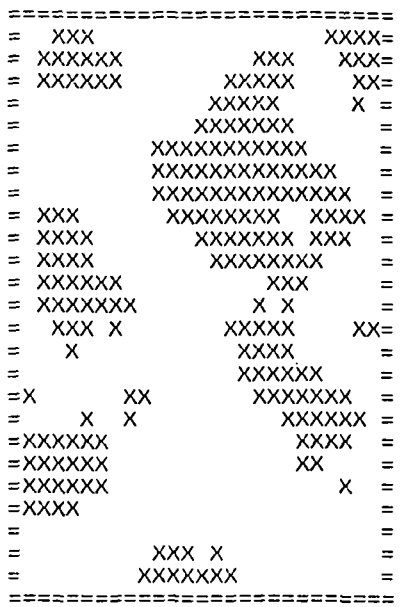
Figure 4.5

True values and predictions by EB logistic spatial estimator (EB-LSE).

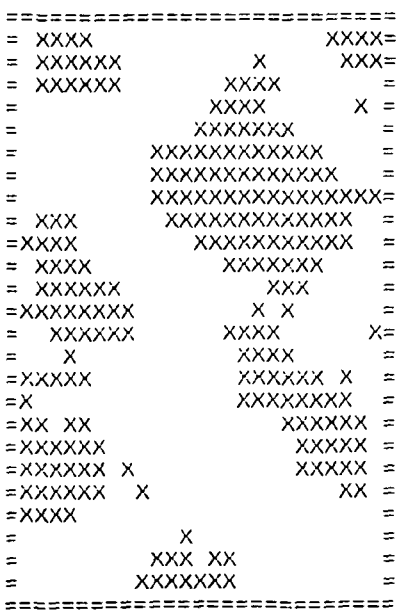
Checkerboard case, $\delta = 1.0, 1.5, 2.0$



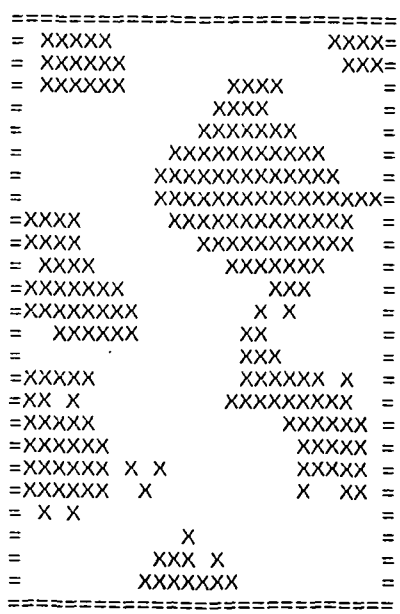
True values, %ERR = 0



$\delta = 1.0$, %ERR = 16.8%



$\delta = 1.5$, %ERR = 10.1%

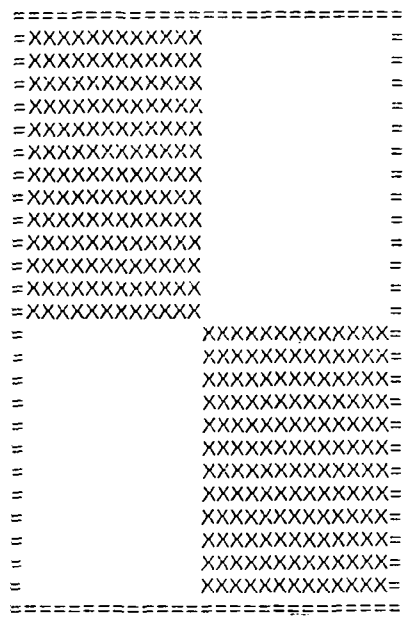


$\delta = 2.0$, %ERR = 6.9%

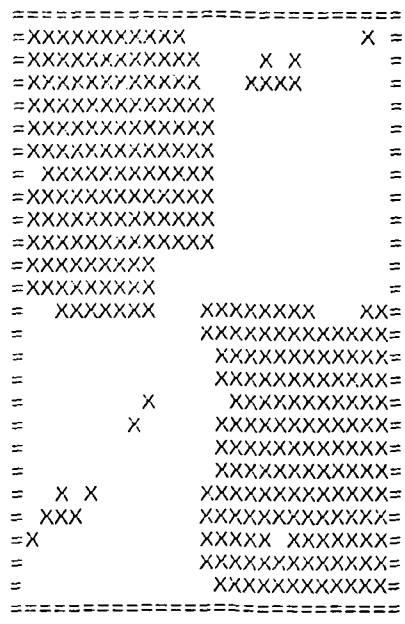
Figure 4.6

True values and predictions by EB logistic spatial estimator (EB-LSE).

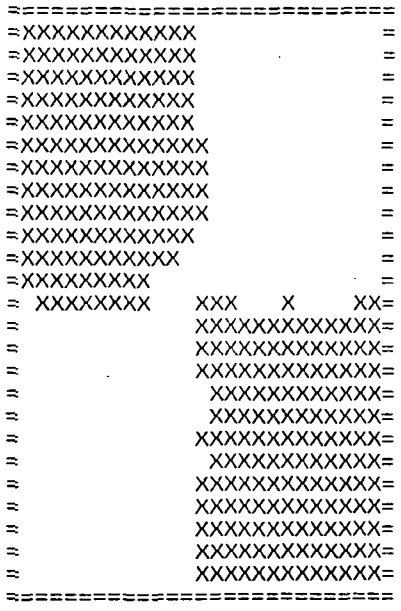
Miscellaneous case, $\delta = 1.0, 1.5, 2.0$



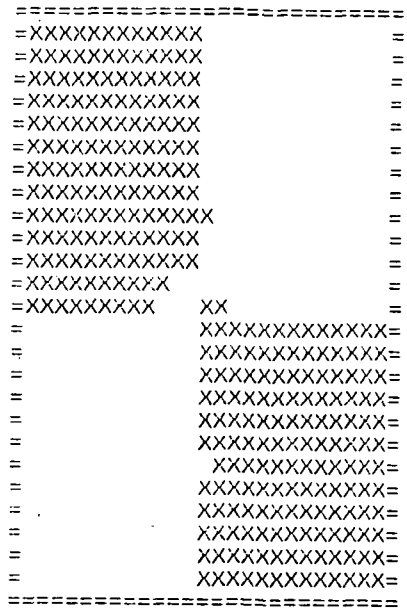
True Values, %ERR = 0



$\delta = 1.0$, %ERR = 8.6%



$\delta = 1.5$, %ERR = 3.4%



$\delta = 2.0$, %ERR = 1.4%

Figure 4.7

True values and predictions by EB logistic spatial estimator (EB-LSE).

Two-by-Two case, $\delta = 1.0, 1.5, 2.0$

Table 4.3
How error percentages for EB-LSE depend on the number of agreeing neighbors.
Entries are percentages, N = number of pixels ($\theta_i = 0$ and 1 combined)
with given number of agreeing neighbors.

		CKBD			*MISC*			*2BY2*	
No.Agreeing	*(N)*	*I − LSE*	*EB − LSE*	*(N)*	*I − LSE*	*EB − LSE*	*(N)*	*I − LSE*	*EB − LSE*
$\delta = 1.0$									
4	(100)	35	33	(28)	21	32	(4)	50	50
5	(300)	24	32	(146)	32	31	(92)	30	33
6	(0)	–	–	(54)	7	7	(0)	–	–
7	(0)	–	–	(79)	8	5	(0)	–	–
8	(225)	9	9	(291)	6	8	(529)	1	4
Summary	(625)	20.2	24.0	(578)	15.5	16.8	(625)	5.4	8.6
$\delta = 1.5$									
4	(100)	28	28	(28)	21	14	(4)	50	50
5	(300)	14	19	(146)	22	20	(92)	13	18
6	(0)	–	–	(54)	7	7	(0)	–	–
7	(0)	–	–	(79)	3	4	(0)	–	–
8	(225)	5	4	(291)	4	3	(529)	0	0
Summary	(625)	13.1	15.2	(598)	10.7	10.1	(625)	2.6	3.4
$\delta = 2.0$									
4	(100)	18	20	(28)	14	14	(4)	25	50
5	(300)	8	8	(146)	14	16	(92)	2	7
6	(0)	–	–	(54)	6	6	(0)	–	–
7	(0)	–	–	(79)	1	1	(0)	–	–
8	(225)	4	1	(291)	2	1	(529)	0	0
Summary	(625)	7.8	7.4	(598)	6.7	6.9	(625)	0.6	1.4

Table 4.4
Values of the normalized regression coefficients in the nine cases.
"Normalized" means b_t^* below is $b_t^* = b_t(\theta)/r_t$ for I-LSE or $b_t^* = \mathrm{B}_t/r_t$ for EB,
$r_t \equiv$ number of pixels in ring t. See text for explanation.

δ	θ	$\bar{\theta}$	$\hat{\pi}$		b_0^*	b_1^*	b_2^*	b_3^*	b_4^*	b_5^*
1.0	*CKBD*	.520	.543	*EB*	.997	.631	.690	.136	−.090	−.141
				IDEAL	.885	.413	.180	.141	.034	−.012
1.5	*CKBD*	.520	.535	*EB*	1.503	.702	.555	.182	−.072	−.099
				IDEAL	1.337	.524	.134	.147	−.014	−.057
2.0	*CKBD*	.520	.532	*EB*	2.013	.748	.418	.207	−.087	−.081
				IDEAL	1.818	.614	.062	.142	−.074	−.115
1.0	*MISC*	.386	.409	*EB*	1.021	.424	.447	.216	.052	−.078
				IDEAL	.897	.425	.209	.253	.132	−.016
1.5	*MISC*	.386	.401	*EB*	1.558	.504	.396	.255	.059	−.085
				IDEAL	1.359	.548	.214	.276	.077	−.105
	Sim.		*EB*	*Avg.*	1.54	.61	.43	.33	.05	−.07
			EB	*S.D.*	(.11)	(.20)	(.14)	(.15)	(.08)	(.13)
			IDEAL	*Avg.*	1.42	.62	.30	.28	.05	−.07
			IDEAL	*S.D.*	(.15)	(.08)	(.04)	(.04)	(.03)	(.05)
2.0	*MISC*	.386	.397	*EB*	2.113	.544	.338	.263	.036	−.102
				IDEAL	1.849	.641	.201	.282	.012	−.186
1.0	*2BY2*	.499	.522	*EB*	.990	.904	.996	.617	.075	−.020
				IDEAL	1.073	.719	.383	.685	.368	.262
1.5	*2BY2*	.499	.515	*EB*	1.486	1.139	.973	.819	.063	−.074
				IDEAL	1.557	.950	.376	.870	.338	.183
2.0	*2BY2*	.499	.511	*EB*	1.983	1.337	.914	.991	.003	−.160
				IDEAL	2.028	1.146	.320	1.017	.264	.069

with r_t = number of pixels in ring t, $r_0 = 1, r_1 = r_2 = r_3 = 4, r_4 = 8, r_5 = 4$. Thus $b_t(\theta)$ or B_t multipled by the ring average $\overline{x}_{it}$ is just b_t^* multiplied by the ring sum $r_t\overline{x}_{it}$. We expect, on *a priori* grounds, that b_0^*, b_1^*, b_2^*, ... would be monotone decreasing because data from more remote rings usually should receive less weight. (This would not hold in periodically patterned situations, however, like the checkerboard.)

The rules EB-LSE and I-LSE nearly follow this monotone pattern, except, curiously, $b_2^* < b_3^*$ frequently for the ideal rule, with large differences in the 2BY2 case. This is an unexpected phenomenon, and seems to be peculiar to the particular $\{z_i\}$ values used (recall that the same simulated data $\{z_i\}$ were used in all nine cases.)

Only in the case $\delta = 1.5$, MISC, were the data simulated further, with 10 repeats. The means (*Avg.*) and standard deviations (*S.D.*) of the EB-LSE and I-LSE regression coefficients for that case are reported in the middle of Table 4.4. Clearly, the main case considered for EB-LSE produced regression coefficients quite central to the 10 cases, with the corresponding main case for I-LSE being less central, but not extreme. The tendancy toward a monotone decreasing pattern is obvious for EB-LSE, and usually for I-LSE. However, the problem of $b_2^* < b_3^*$ occured for I-LSE in four of the ten cases.

Several features deserve comment:

a) The coefficient of y_i, b_0^*, tends to be close to δ. There is theoretical justification for this.

b) The EB-LSE coefficients b_t^* are consistently larger than the I-LSE coefficients, especially in the low discrimination cases like $\delta = 1.0$, checkerboard. This pushes the EB-LSE probability estimates too far toward zero or one, away from 1/2. We discussed this property of EB-LSE before, in relation to its performance with respect to mean absolute error. This effect continues, but only slightly, in the only repeated case $\delta = 1.5$, MISC. A correction, perhaps simply applying a constant multiple to the B_i values, would likely improve EB-LSE significantly for the MSE and INFO measures, but would not affect the %ERR measure.

c) Clearly b_4^* and b_5^* add little to the precision of these rules, and use of $r = 3$ rings would have been nearly as effective. This issue of where to truncate the regression vector, i.e. how to chose r, deserves further investigation.

5. Summary.

We have seen that empirical Bayes theory can help in a spatial analysis by clarifying the separate roles that must be played by training data and data taken from the target site. Training data can be used to determine the likelihood function, while the target area data are required to learn about the distribution of the parameters in the target site. These ideas are implemented for a binary spatial setting by (3.16), an estimator seen to work quite well relative to "ideal" procedures that utilize the true target site values θ. The key point is that, with the structure assumed, one need not have direct access to any true values θ from the target site. This is very useful if the target site is inaccessible or quite costly to observe, as might occur in some LANDSAT applications.

Of course much more can be done, some things fairly straightforwardly, and others less so. The straightforward tasks include further tests on new data sets, and comparisons of the estimator (3.16) with other methods for spatial classification, as follows.

(A) In comparison with the method of Geman and Geman (1984), by how much does (3.16) method dominate the Geman annealing method with respect to computing time (the annealing algorithm is very slow)? How does (3.16) compare in terms of %ERR for estimating the best map, that is the most likely θ value (which is the Geman and Geman objective)? When the characteristics α of the θ process must be estimated, how do the rules compare?

(B) Some fine tuning of the method (3.16) is needed. The coefficients b_j^* as defined for Table 4.4 (coefficients of ring-sums) probably should be adjusted so that their magnitude decreases as j increases, to reflect the property that the influence of rings should diminish with their distance from the central pixel. What is the appropriate value of r,

the number of rings required? How can we correct for the tendency of the coefficients B_t^* in (3.17) to overestimate the coefficients $b_t(\theta)/RSS(\theta)$ in (3.6) by a systematic factor, at least for small δ?

(C) The method needs to be checked with real data. Even though the assumptions are violated, the method (3.16) may work in LANDSAT applications. For example, similar assumptions were used successfully by Owen (1984) with LANDSAT data.

Other extensions are needed for applications like crop type estimation from LANDSAT data. They specifically include:

(A) **The polytomous case.** Extensions are needed for more than two crop types.

(B) **Heteroskdastic data,** or non-normal distributions.

(C) **Multivariate data.** E.g. several spectral bandwidths. The empirical Bayes viewpoint has emphasized, however, that the proper reduction of multivariate data may be determined from training data alone.

(D) **Time dependent data.** This would be important in some applications. The preceding remarks from (C), about data reduction, may apply here.

(E) **Edge effects.** Can the method be extended to be more sensitive to the possibility that there frequently will be straight line borders?

(F) **Dependent observations.** Cloud cover and weather effects, for example, would cause correlation among neighboring spectral measurements even if the crop-type remains constant. How can the EB-LSE method (3.16), derived for independent data, be modified to account for known correlation patterns?

(G) **Split pixels.** What can be done if more than one kind of true value, e.g. crop type, exists in a pixel? By computing estimates of the fractions of each kind of crop type, that the method discussed already offers some advantage for split pixels.

The polytomous case seems most urgent. The problem can be approached as an empirical Bayes problem in the same manner as for the binary case. The main difficulties arise, however, in proposing appropriate estimates for the parameters α of the θ process.

The same difficulty of finding good estimates of α arises in cases (B), and (F), although the general theory for known α seems straighforward. Case (E) provides a challenge dealt with earlier in (Geman and Geman, 1984) for a slightly different context.

References

Geman, D. and Geman, S., (1984), "Stochastic relaxation, Gibbs distribution, and Bayesian restoration of images", **IEEE Transactions on Pattern Analysis and Machine Intelligence**, Vol. PAMI-6, No. 6, p 721-741 (November).

Hill, J., Hinkley, D., Kostal, H. and Morris, C., (1984), "Spatial Estimation from Remotely Sensed Data Via Empirical Bayes Models", **Proceedings of the NASA Symposium on Mathematical Pattern Recognition and Image Analysis**, Houston, Texas, June 1984, pp. 115-136, L. F. Guseman, Jr. (ed.), Department of Mathematics, Texas A&M.

Morris, C. and Rolph, J., (1981), **Introduction to Data Analysis and Statistical Inference**, Prentice Hall, 1981.

Owen, A., (1984), "A Neighborhood-based classifier for LANDSAT data", **Canadian Journal of Statistics**, Vol. 12, No. 3, pp. 191-200.

Switzer, P., (1980), "Extensions of Linear Discriminant Analysis for Statistical Classification of Remotely Sensed Satellite Imagery", **Mathematical Geology**, Vol. 12, No. 4, pp. 367-376.

ESTIMATING PARAMETERS IN A FINITE MIXTURE OF PROBABILITY DENSITIES

R. P. Heydorn

ABSTRACT

The problem of estimating parameters in finite mixture of probability densities is formulated as a continuous mixture estimation problem. Writing the finite mixture as $h = \int f_\theta dG(\theta)$, where G changes only at a finite number of points, it is shown that it is possible to construct a sequence of probability density functions (g_n) whose cumulative distribution functions (G_n) converge weakly to G. It is proposed that this sequence be constructed using a linear programming approach.

1. INTRODUCTION

Let x be a vector in $\mathbb{R}^N$ and θ another vector in $\mathbb{R}^K$, where $\mathbb{R}^N$ and $\mathbb{R}^K$ are real product spaces over the real numbers of dimension N and K respectively. In remote sensing, x represents the measurement values obtained from a remotely positioned sensor (e.g., from a satellite) for some given point on the Earth and θ is a vector that can be uniquely associated with the class of materials at that point. The x-values are the observables but θ, the variable of interest, is not observable.

To illustrate this x, Θ relationship in terms of a remote sensing problem, imagine that a set of x-measurements are obtained from an agricultural area containing fields of corn, soybeans, and pasture. A possible probability model would be

$$h(x) = \sum_{j=1}^{M} \Pr(\Theta = \theta_j) \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}(x-\theta_j)^2}$$

where h is probability density function (called a mixture density) and it is a linear combination of normal density functions. A normal density is assumed to statistically represent the x-measurements from each one of M possible crop classes. In this model Θ is a random variable that can take on the possible class mean values θ_j, $j = 1, 2, \bullet\bullet\bullet, M$. It is seen that Θ is indeed the variable of interest since it describes the class means, and therefore it provides a complete statistical description of the x-measurements from a given class. Moreover, by the fact that positive probability is assigned to only M possible values of Θ, we can determine the number of classes. If the assumption about this representation of h is correct, then from the identifiability (a concept that will be presented formally below) of normal mixtures, there is only one possible choice for M and, θ_j, $j = 1, 2, \bullet\bullet\bullet, M$. Specifically, for this example given h and the model, it should be possible to determine that M=3, the values of three crop means θ_1, θ_2, and θ_3, and the values of their proportions $\Pr(\Theta = \theta_j)$, $j = 1, 2, 3$. If the additional fact is known that the mean of corn is always less than the mean of soybeans and that the mean of soybeans is always less than the mean of pasture, then it would be possible to assign these crop labels to the means and proportions. Even though x-values

can not be uniquely associated with θ-values, it is possible to compute a likelihood or a posterior probability of this association from the mixture model. If the means can be assigned crop labels then the mixture model can be used to infer a classification for each pixel.

A general formulation of a mixture density that is similar to the one given for mixtures of distributions by Teicher [1] is as follows: Let $F = \{f_\theta : \theta \in \mathbb{R}^K\}$ be a family of probability density functions and let G be a distribution function on $\mathbb{R}^K$. For the given G, define the mixture density

$$h = \int f_\theta dG(\theta) \tag{1}$$

The family F defines a mapping, (say $\tilde{F}$), from the set of all G-distributions (say G), to the set of all induced h-densities (say H). If $\tilde{F} : G \to H$ is one-to-one and onto, then it can be said that H is identifiable. In the case of the finite mixture, the measure induced by G assigns positive probability to only a finite number of θ-values. For this case

$$h(x) = \sum_{j=1}^{M} \Pr(\Theta = \theta_j) f_{\theta_j}(x) \tag{2}$$

As reported in two prior papers, previous work concentrated on the case where θ is a translation parameter. In the first paper, Heydorn and Basu [2], h was assumed to be known, and an approach based on a theorem of Caratheodory (relating to the trigonometric moment problem as discussed in Grenander and Szego [3]) was used to determine the number of translation parameters and their values. In the second paper, Heydorn and Martin [4], h was estimated, and an integral equation formulation was used to find a probability density on θ-values.

This paper also assumes that h is not given but must be estimated; however, unlike the second paper, this paper offers a more general approach in which θ is not restricted to be a translation parameter. In common with the second paper, the idea of estimating a probability density on θ-values as a means of deducing the number of parameters (i.e., the value of M) as well as their values is again pursued.

2. CONSTRUCTION OF ESTIMATORS

Given a finite mixture h on $\mathbb{R}^1$ that follows the model

$$h(x) = \sum_{j=1}^{M} \Pr(\Theta = \theta_j)\, f(x - \theta_j) \tag{3}$$

a method was proposed in Heydorn and Martin [4] in which h is first smoothed with some function t to produce $h_t \triangleq h*t$ ("*" denoting convolution). The function h_t can then be represented as a continuous model of the form

$$h_t(x) = \int f(x-\theta)\, g_t(\theta)\, d\theta \tag{4}$$

By choosing the support of t to be small, the integral equation in (4) is a good approximate representation for the finite mixture in (3) since g_t will have M modes with the modes occurring at the θ_j-values.

For cases where θ is not necessarily a translation parameter and h follows the more general finite mixture model of equation (2), an integral equation representation is still possible. It will be shown that this representation can take the form

$$h(x) = \int f(x,\theta)\, g_n(\theta)\, d\theta + \varepsilon_n(\theta)$$

where $||\varepsilon_n|| \to 0$ as $n \to \infty$ ($||\cdot||$ being the supremum norm). In this case (g_n) is a sequence of probability density functions whose cumlative distribution functions, G_n, converge (weakly) to $G \varepsilon \mathcal{G}$ (c.f. discussion related to equation (1)).

The approach used for estimating G given h is as follows:

1) First define g_n as

$$g_n(\theta) = \sum_{k=1}^{K} \alpha_k\, B_k(\theta),\ c \le \theta \le d$$

where (B_k) is a sequence of normalized (i.e., $\int B_k(\theta)d\theta = 1$) B-splines placed at equally spaced knots in $[c,d]$ and where $\alpha_k \geq 0$, $\sum \alpha_k = 1$. This sequence of g_n-functions will induce the sequence (h_n) where

$$h_n(x) = \int f(x,\theta)\, g_n(\theta)d\theta$$

2) Assuming that $f(\bullet,\bullet)$ is continuous on $(-\infty,\infty)$ x $[c,d]$, since h is a finite mixture, h is uniformly continuous on any closed interval, say $[a,b]$. Let $a = x_1 < x_2 < \bullet\bullet\bullet < x_n = b$ be a partition of $[a,b]$ and define the histogram of h to be

$$\tilde{h}(x) = \begin{cases} \dfrac{1}{x_j - x_{j-1}} \displaystyle\int_{x_{j-1}}^{x_j} h(t)dt, & x \in (x_{j-1}, x_j],\ j = 2, 3, \bullet\bullet\bullet, n \\ 0, & x \notin [a,b] \end{cases}$$

Since h must vanish at $-\infty$ and $+\infty$, the constants a, b can be chosen so that for a given $\varepsilon > 0$, $0 \leq h(x) < \varepsilon$ holds for $x \notin [a,b]$. Also since h is uniformly continuous, we can construct the partition of $[a,b]$ so that

$$\sup_{x \in (x_{j-1}, x_j]} |h(x) - \tilde{h}(x)| < \varepsilon$$

for any j.

3) For $j = 1, 2, \bullet\bullet\bullet, n+1$ let

$$S_j(x) = \begin{cases} 1, & x \in (x_{j-1}, x_j] \\ 0, & x \notin (x_{j-1}, x_j] \end{cases}$$

where $x_0 \triangleq -\infty$ and $x_{n+1} \triangleq +\infty$. It follows that

$$\sup_x |h(x) - h_n(x)| \le \sum_{j=1}^{n+1} \sup_x |h(x) - \tilde{h}(x)| S_j(x)$$

$$+ \sum_{j=1}^{n+1} \sup_x |\tilde{h}(x) - \tilde{h}(x_j)| S_j(x)$$

$$+ \sum_{j=1}^{n+1} |\tilde{h}(x_j) - h_n(x_j)| S_j(x)$$

$$+ \sum_{j=1}^{n+1} \sup_x |h_n(x_j) - h_n(x)| S_j(x)$$

The first sum on the right is less than ε from step 2, and the second sum is zero from the definition of the histogram. Consider the last sum. From the definition of h_n in step 1

$$|h_n(x_j) - h_n(x)| \le \sum_{k=1}^{K} \alpha_k \int |f(x_j,\theta) - f(x,\theta)| B_k(\theta) d\theta$$

Since $f(\bullet,\bullet)$ is continuous on $[a,b]x[c,d]$ the family $\{f_\theta : \theta \in [c,d]\}$ is uniformly equicontinuous; therefore, it is possible to refine the partition of step 2 so that $|f(x_j, \theta) - f(x, \theta)| < \varepsilon$ for any $(x, \theta) \in [x_{j-1}, x_j] \times [c, d]$, $j = 1, 2, \bullet\bullet\bullet, n + 1$. Hence for all j

$$|h_n(x_j) - h_n(x)| \le \varepsilon \sum_{k=1}^{K} \alpha_k = \varepsilon$$

Following through these steps, therefore, it can be seen that

$$\sup_x |h(x) - h_n(x)| \le 2\varepsilon$$

provided we select the spline coefficients α_k, with the constraints $\Sigma\alpha_k = 1$, $\alpha_k \ge 0$ for $k = 1, 2, \bullet\bullet\bullet, k$, so that h_n coincides with $\tilde{h}$ at the partition points.

If the histogram is only matched to within ε at the points x_j, $j = 1, 2, \bullet\bullet\bullet, n$, then

$$\sup_x \mid h(x) - h_n(x) \mid \leq 3\ \varepsilon$$

Normally h is not given and therefore must be estimated. In the above formulation this means that rather than computing the histogram $\tilde{h}$ it must be estimated. Given a sufficiently large sample size this can be done (see e.g., Tapia and Thompson [6]) so that the above construction steps will still produce a sequence (h_n) converging to h.

It is proposed that linear programming be used to solve for G_n. The linear programming formulation is:

Minimize

$$\Delta_1 + \Delta_2 + \bullet\bullet\bullet + \Delta_n$$

Subject to, for $j = 1, 2, \bullet\bullet\bullet, n$, $k = 1, 2, \bullet\bullet\bullet, K$,

$$-\Delta_j \leq \hat{h}\ (x_j\ ) - \int_c^d f(x_j, \theta)\ g_n(\theta)d\theta \leq \Delta_j$$

$$\Delta_j \geq 0,\ \alpha_k \geq 0, \sum \alpha_k = 1$$

Guseman and Schumaker [6] and Narula and Wellington [7] have used a similar linear programming formulation for other problems.

3. CONVERGENCE OF THE ESTIMATOR

Given the above approach (steps 1-3) it now can be shown that the cumulative distribution function (c.d.f) G_n related to the density g_n will converge to the true c.d.f, G, weakly. That is, if $G_n(\theta) = \int_c^{\theta} g_n(y)\, dy$, then,

$$\left| \int_c^d q\, dG_n - \int_c^d q\, dG \right| \to 0, \quad (n \to \infty), \text{ for all } q \in C[c,d],$$

where $C[c,d]$ is the set of all continuous functions on $[c,d]$.

Theorem: Let H be identifiable and (g_n) a sequence of probability densities. Define:

$$h_n(x) = \int_c^d f(x, \theta)\, g_n(\theta)\, d\theta.$$

If $||h - h_n|| \to 0$, $(n \to \infty)$, then $G_n \to G$ weakly.

The proof of the theorem follows easily from the following lemma of Blum and Susarla [8]. In their lemma the family of kernels in the mixture is parameterized on x (not θ) i.e., let $D = \{f(x, \bullet) : x \in \mathbb{R}\}$.

Lemma: If $D \subset C[c,d]$, then H is identifiable if and only if D generates $C[c,d]$ in supremum norm.

Proof of the Theorem: Pick $q \in C[c,d]$, $\varepsilon > 0$. From the lemma then exists a sequence (ξ_k, x_k), $k = 1, 2, \bullet\bullet\bullet, K$ so that (denoting $f(x_k, \bullet)$ by f_{x_k})

$$\left|\left| q - \sum_{k=1}^{K} \xi_k f_{x_k} \right|\right| < \varepsilon$$

For any function $s \in C[c,d]$ let

$$\ell(s) = \int_c^d s \, dG$$

$$\ell_n(s) = \int_c^d s \, dG_n$$

Since G, G_n are c.d.f's, the Riesz Representation theorem has that (denoting variation by V(•))

$$||\ell|| = V(G) = 1$$

and

$$||\ell_n|| = V(G_n) = 1$$

Thus

$$|(\ell-\ell_n)\ (q)| \leq |\ (\ell-\ell_n)\ (q-\sum_k \xi_k\ f_{x_k})|$$

$$+|\ (\ell-\ell_n)\ (\sum_k \xi_k\ f_{x_k})|$$

or

$$|(\ell-\ell_n)\ .(q)| \leq ||\ \ell-\ell_n\ ||\ ||q-\sum_k \xi_k\ f_{x_k}||$$

$$+\ |\sum_k \xi_k\ (h(x_k) - h_n\ (x_k))|$$

$$\leq 2\ \epsilon + ||\ h - h_n\ ||\sum_k |\xi_k|$$

Therefore

$$\lim_n |(\ell-\ell_n)\ (q)| \leq 2\ \epsilon$$

which implies

$$\int q dG_n \to \int q dG$$

for any $q \in C\ [c,d]$ this completes the proof.

4. CONCLUDING REMARKS

In this paper, which is the third in a series of papers on mixtures, the idea of studying finite mixtures from a continuous mixture point of view has continued. That is, the finite mixture is approximated with a continuous mixture and the resulting mixing function (denoted by g_t or g_n) is estimated. This mixing function gives an estimate of the number (M) of components in the mixture as well as estimates of the θ-parameter values. There are still a number of numerical and statistical estimation problems to be studied in relation to this approach; however, from the few numerical studies that were done (in the second paper) it would appear that the ideas can produce reasonable answers, and the graph of the mixing function is more informative to the eye than is the mixture itself.

There may be some mathematical problems, however. By approximating a finite mixture with a continuous mixture one could possibly loose some uniqueness. It is well known, for example, that a finite mixture of normals in which the means and variances are allowed to vary is an identifiable mixture, (c.f. Teicher [9] or Yakowitz and Spragins [10]). However, the same is not true of the continuous mixture of normals, as pointed out by Teicher [11]. If, however, we hold either the means or the variances fixed, while letting the other parameter vary, then the continuous mixture is identifiable. The extent to which this is a limiting factor in this approach to studying finite mixtures needs to be studied.

5. REFERENCES

1. Teicher, H.: Identifiability of Mixtures. Annals of Mathematical Statistics, vol. 32, 1961, pp. 224-248.

2. Heydorn, R. P.; and Basu, R.: Estimating Location Parameters in a Mixture Model. Proceedings of the NASA Symposium on Mathematical Pattern Recognition and Image Analysis, June 1983, pp. 55-76.

3. Grenander, U.; and Szego G.: Toeplitz Forms and Their Applications. Berkeley, University of California Press, 1958.

4. Heydorn, R. P.; and Martin, M. V.: Estimating Location Parameters in a Mixture. Proceedings of the NASA Symposium on Mathematical Pattern Recognition and Image Analysis, June 1984.

5. Tapia, R. A.; and Thompson, J. R.: Nonparametric Probability Density Estimation. John Hopkins University Press, 1978.

6. Guseman, L. F.; and Schumaker, L.: Spline Classification Methods. Proceedings of the NASA Symposium on Mathematical Pattern Recognition and Image Analysis, June 1983.

7. Narula, S. C.; and Wellington, J. F.: Interior Analysis for the Minimum Sum of Absolute Errors Regression. Technometrics vol. 27, no. 2, May 1985, pp. 181-188.

8. Blum, J. R.; and Susarla, V.: Estimation of a Mixing Distribution Function. Annals of Probability, vol 5, no. 2, 1977, pp. 200-209.

9. Teicher H.: Identifiability of Finite Mixtures. Annals of Mathematical Statistics, vol. 34, 1963, pp. 1265-1269.

10. Yakowitz, S. J.; and Spragins, J. D.: On the Identifiability of Finite Mixtures. Annals of Mathematical Statistics, vol. 39, no. 1, 1968, pp. 209-214.

11. Teicher, H.: On the Mixture of Distributions. Annals of Mathematical Statistics, vol. 31, 1960, pp. 55-73.

Experiences with Examining Large Multivariate Data Sets with Graphical Nonparametric Methods

David W. Scott

Rice University

ABSTRACT

In this paper we review our work over the past three years, indicate our current thinking, and point to work generated for those wanting to pursue these ideas.

1. Introduction

For our setting, the purpose of statistics is to extract information from data. One auxiliary goal may be to maximize the information extracted, for example, design efficient estimators. NASA data presents special challenges and hence opportunities. First, the data are often high dimensional. Second, the data sets may be extremely large. Third, the data are expected to be non-Gaussian, that is, second-order information such as correlation is not sufficient. As a remark, we note that any one of these features makes good data analysis very difficult. Some present and many future NASA projects will routinely have to handle all three features. If we accept present technology and methodology, we are simply "losing" information, perhaps critical to mission success.

These ideas are echoed in a recent article by Goetz et al. (7), who discuss imaging spectrometry, which is the simultaneous acquisition of images in as many as 224 narrow contiguous spectral bands. The authors write:

> Just as imaging spectrometry requires new technology for instruments and detectors, effective utilization of the data requires development of new analytic approaches and techniques. Bellman's 'curse of dimensionality' is fulfilled...

The authors rather curiously predict that deterministic methods will be superior to statistical methods. In any case, it is clear that the new technology raise many interesting questions such as the tradeoff between higher spatial resolution and narrow spectral bands.

Statisticians' proper role in NASA is varied but extensive: design (with physicistics, MD's, engineers), data collection (with engineers and CS's), data analysis, data presentation (with managers, artists), program evaluation, among others. In data analysis, relevant research activities include estimation, filtering, optimization, algorithm construction. The planning activities in the design role are critical, such as determining whether a proposed system will generate data giving the desired information (can we predict who gets severe motion sickness or which spectral bands should be included in a satellite?) and is the system optimal (Landsat's 4-channel sensor contains essentially 2-dimensional information, wasting 50% of the bandwidth)?

In the following, I will briefly indicate our work and progress in the areas of data analysis and presentation of very large non-Gaussian data sets with 3,4, and more variables. We have not included any graphs, since these are contained in referenced articles. Particular topics include dimension reduction of non-Gaussian data sets, graphical representation of structure in data sets with more than 2 variables, efficient algorithms for multivariate density estimation, automatic calibration of density estimates, and tests of our ideas on real data sets. We note we are only beginning to have the computer power required to try new techniques for properly analyzing "difficult" data. For example, it has been estimated that real-time computer animation will require the power of 1000 supercomputers!

2. Large High-Dimensional Data

Scientists have attempted to cope with high dimensional data for several decades. When such data follow elliptical patterns, statisticians have developed extremely powerful, fast, and

efficient algorithms for all aspects of data analysis and presentation. For data not following such "nice" patterns, we are in much worse shape. As an example, Weaver et al. (35) analyze the series of 500 small and large earthquakes preceding the large eruption of Mount St. Helens. These data are 5 dimensional: 3 spatial coordinates, time, and quake magnitude. The authors attempt to display these data graphically. Two "side" views are constructed. But it is clear that some information may be hidden in the true higher-dimensional space. We are attempting to devise methods to reveal such structure, if it exists.

Pictures of large data sets often are misleading. This is illustrated in Scott (19) in a scatter diagram of 412,000 pixels. The eye focuses on the edge of the data cloud where relatively little information lies and cannot dissect structure in the middle of the data cloud. Thus relying solely on graphs for non-Gaussian data is not likely to be sufficient for the new data analysis.

John Tukey has been a leading proponent of the new exploratory data analysis (32,33). With Paul Tukey (34), he has given us a wealth of different ideas for graphing multivariate data. Many may not withstood the test of time, but it is likely that many will do so. Many of his examples deal with Anderson and Fisher's Iris data, which is 4-dimensional. In addition, examples from 3 and 4 body particle physics experiments are presented, which are 4 and 7 dimensional, respectively. We will mention these data sets later.

In Scott (13,14,15,16,17,18,19,20), Scott and Thompson (28), Scott et al. (23,24), and Scott and Jee (25), we have discussed and illustrated the variety of ways available (including our new proposals) for displaying multivariate data. Tukey has emphasized scatter diagrams and variants. We prefer to estimate and display density curves, such as the histogram and the new improved histograms. While estimation cannot be ignored, it is the representation of high-dimensional histograms that is exciting and full of new possibilities for finding non-Gaussian structure. Many examples are given in the references (see in particular reference 18, which contains color prints).

3. Efficient Density Estimation

Computationally, the most efficient density estimator is the classical histogram. The histogram is in the class of nonparametric density estimators, which provide reasonable estimates for a large class of smooth sampling densities. The statistical properties of the histogram were provided by Scott (12). From this work it is clear that the statistical efficiency of the histogram, particularly the multivariate histogram, is inferior to other methods such as kernel algorithms (which, unfortunately, are not computationally effective). Thus the histogram is only useful as a preliminary tool with univariate data.

Recently, the second most computationally efficient estimator, the frequency polygon, was analyzed by Scott (19). It was shown that the frequency polygon possesses the same statistical efficiency of the kernel algorithms. This is quite remarkable and the frequency polygon is quite useful for univariate and bivariate data. Bin edge effects limit its usefulness for 3 and 4 dimensional data.

Thus Scott (14) introduced a modification of the histogram to obviate the bin edge effects while retaining statistical efficiency. This object, the averaged shifted histogram, has been

demonstrated in Scott (15) and will be formally analyzed in Scott (20). This estimator may even prove useful with 5 or 6 dimensional data!

4. Projections of Non-Gaussian Data

Last year, Rod Jee (25) presented a movie illustrating the capabilities of the density estimation approach in an interactive computer graphics workstation environment. The data were collected by Bob MacDonald over forests in Minnesota. In this work, Rod first saw the relationship among projection methods, information content, density estimation, and feature spaces. This led him into an investigation of projection methods called projection pursuit. Rod has just completed his thesis (10), and we now briefly discuss those results.

It is common to orthogonally project very high-dimensional data prior to analysis. This is the result of a common occurrence with such data: the data cloud is nearly singular in the full space. Thus projections to ease the associated numerical problems are usually sought. There are three projection choices. First, one may choose classical principal components. This is fast, but not robust. We also note that principal components uses only second-order correlation information and will not usually be satisfactory with non-Gaussian data. The second type of projection is a "guided" or model-driven (often nonlinear) projection. For example, Badhwar (1,2) constructed agronomic models to project 24-dimensional Landsat data (multiple acquisitions) into 3 dimensions. This type of projection is usually very effective and often the best, but it requires a great deal of research, work, and luck and is not generalizable to other data types. The third type is "exploratory." Here, we are interested in finding projections in which the data are maximally "clumped." This technique was made popular by Friedman and Tukey (5), who named their particular algorithm "projection pursuit."

Recently, Huber (9) has completed a lengthy treatise on the theoretical foundations of projection pursuit. Huber shows that Friedman and Tukey's optimization criterion function is essentially

$$\int f(x)^2 dx \ ,$$

which clearly is larger for "bumpier" projected densities that for smoother densities, after correcting for scale. Huber notes that other more classical information criteria may be considered, such as Fisher Information or Shannon Entropy.

Rod shows that none of these criteria use any second-order information, which emphasizes the difference of the projection pursuit and principal components methods. By some clever choices of simulations, Rod finds that Fisher Information and Shannon Entropy do not prefer the same projection subspaces, and that Fisher Information seems to provide more pleasing pictures. Friedman and Tukey (F-T) illustrated the projection of the 7-D 4-body particle physics data into 2 dimensions. Rod found the optimal Fisher Information 2-dimensional subspace and it differs remarkably from the F-T subspace. Fisher Information has many local optima, and the F-T is one of those. When applied to Bob MacDonald's 7-D Minnesota data, Fisher Information is quite similar to principal components, although clearly superior.

5. Choosing Smoothing Parameters

Many diverse algorithms for non-Gaussian data rely upon choice of a smoothing parameter, for example, the bin width of a histogram or the size of a neighborhood for projection pursuit. We have found several interesting results in this area. The first was the discovery by Terrell and Scott (31) of intrinsic upper bounds on these smoothing parameters. For the histogram,

$$number\ of\ bins \geq (2n)^{1/3} .$$

In fact, Rod Jee used similar rules in his Fisher Information projection pursuit algorithm. Other algorithms require subjective choice of this parameter. Similar results have been found for frequency polygon, kernel, and averaged shifted histogram estimators.

A more ambitious goal would be to estimate nearly optimal smoothing parameters directly from the data. Such estimates are called "cross-validation" estimates. Wahba had some early results here, and current work is due to Rudemo, Bowman, Stone, and Hall; for a survey, see (21). We have analyzed the small (finite) sample properties of these algorithms and have been led to construct new algorithms as a result (21,27). Many of the algorithms with good theoretical properties are surprisingly noisy with small samples (10,000 points?).

6. Future Directions

In spite of the gratifying progress in the 5 areas, we still have only begun to understand all of the theoretical and practical issues as they relate to NASA data, particularly the new high-dimensional sensor data. We expect to find "true" multi-dimensional features that will lead to unusual classification and detection algorithms. Such information cannot be extracted by classical statistical methods or "new" deterministic algorithms. Many of the issues remaining deal with efficiency and optimization problems that we still don't fully understand. Effective implementation in rapidly changing computer environments is also challenging. Our research goal remains the same: to extract the maximum useful information from data, both analytically and graphically, in an efficient, effective, and pleasing manner.

7. Acknowledgments

I would like to thank Dick Heydorn, George Terrell, and Jim Thompson for their many contributions towards this work. This research was supported by NASA/JSC, ONR, and ARO under grants NCC 9-10, N00014-85-C0100, and DAAG-29-82-K0014, respectively.

8. References

(1) Badhwar, G.G. (1980), Crop emergence data determination from spectral data, *Photogram. Eng. Remote Sens.*, 46:369-377.

(2) Badhwar, G.G., Carnes, J.G. and Austin, W.W. (1982), Use of Landsat-derived temporal profiles for corn-soybean feature extraction and classification, *Remote Sensing of*

Environment 12:57-59.

(3) Carr, D.B. and Nicholson, W.L. (1984), Graphical tools for analysis of multiple two- and three-dimensional scatterplots, *Proceedings of the Fifth Annual National Computer Graphics Association Conference*, Volume II, pp. 743-752.

(4) Friedman, J.H., McDonald, J.A., and Stuetzle, W. (1982), An introduction to real time graphical techniques for analyzing data, *Proceedings 3rd Annual Conference of National Computer Graphics Association.*

(5) Friedman, J.H. and Tukey, J.W. (1974), A projection pursuit algorithm for exploratory data analysis, *IEEE Trans. Comp.*, C-23:881-890.

(6) Fwu, C., Tapia, R.A., and Thompson, J.R. (1981), The nonparametric estimation of probability densities in ballistics research, *Proceedings of the Twenty-Sixth Conference on the Design of Experiments in Army Research Development and Testing*, pp. 309-326.

(7) Goetz, A.F.H., Vane, G., Solomon, J.E., and Rock, B.N. (1985), Imaging spectrometry for earth remote sensing, *Science*, 228:1147-1153.

(8) Heydorn, R.P., and Basu, R. (1983), Estimating proportions of materials using mixture models, in Guseman, L.F. (ed.), *Proceedings of the NASA/MPRIA Workshop: Math/Stat.*

(9) Huber, P.J. (1983), Projection pursuit, Technical report MSRI 009-83, Berkeley.

(10) Jee, R. (1985), A study of projection pursuit methods, unpublished doctoral dissertation, Rice University, Houston, Texas.

(11) Nicholson, W.L. and Littlefield, R.J. (1983), Interactive color graphics for multivariate data, *Comp Sci and Statistics: Proc 14th Symposium on the Interface*, K.W. Heiner, R.S. Sacher, and J.W. Wilkinson, eds., Springer-Verlag, New York.

(12) Scott, D.W. (1979), On optimal and data-based histograms, *Biometrika*, 66:605-610.

(13) Scott, D.W. (1982), Review of some results in bivariate density estimation, *Proceedings of the NASA Workshop on Density Estimation and Function Smoothing*, pp. 165-194.

(14) Scott, D.W. (1983), Nonparametric probability density estimation for data analysis in several dimensions, *Proceedings of the Twenty-Eighth Conference on the Design of Experiments in Army Research Development and Testing*, pp. 387-397.

(15) Scott, D.W. (1983), Multivariable density estimation and remote sensing, *Proceedings of the Symposium on Mathematical Pattern Recognition and Image Analysis*, pp. 77-92.

(16) Scott, D.W. (1984), Multivariate density function representation, *Proceedings of the Fifth Annual National Computer Graphics Association Conference*, Volume II, pp. 794-801.

(17) Scott, D.W. (1985), Classification using multivariate nonparametric density estimation, *Proceedings of the Sixth Annual National Computer Graphics Association Conference*, Volume III, 715-718.

(18) Scott, D.W. (1985), Data analysis in 3 and 4 dimensions with nonparametric density estimation, in *Statistical Image Processing*, E.J. Wegman and D. DePriest, Eds., Marcel Dekker, New York, in press.

(19) Scott, D.W. (1985), Frequency polygons: theory and application, *J. American Statistical Association*, 80:348-354.

(20) Scott, D.W. (1985), Averaged shifted histograms: effective nonparametric estimators in several dimensions, to appear in *The Annals of Statistics*.

(21) Scott, D.W. (1985), Choosing smoothing parameters for density estimators, To appear in *Proceedings of the 17th Symposium on the Interface of Computer Science and Statistics*, North-Holland.

(22) Scott, D.W. (1985), A note on choice of bivariate histogram bin shape, Rice Technical Report No. 85-311-3.

(23) Scott, D.W., Gorry, G.A., Hoffmann, R.G., Barboriak, J.J., and Gotto, A.M. (1980), A new approach for evaluating risk factors in coronary artery disease: a study of lipid concentrations and severity of disease in 1847 males, *Circulation* 62:477-484.

(24) Scott, D.W., Gotto, A.M., Cole, J.S., and Gorry, G.A. (1978), Plasma lipids as collateral risk factors in coronary artery disease: a study of 371 males with chest pain, *Journal of Chronic Diseases* 31:337-345.

(25) Scott, D.W. and Jee, R. (1984), Nonparametric analysis of Minnesota spruce and aspen tree data and Landsat data, *Proceedings of the Second Annual Symposium on Mathematical Pattern Recognition and Image Analysis*, NASA publication, pp. 27-49.

(26) Scott, D.W. and Sheather, S.J. (1985), Kernel density estimation with binned data, *Communications in Statistics*, in press.

(27) Scott, D.W. and Terrell, G.R. (1985), Consistent data-based kernel density estimation, Working paper.

(28) Scott, D.W. and Thompson, J.R. (1983), Probability density estimation in higher dimensions, *Proceedings of the 15th Symposium on the Interface of Computer Science and Statistics*, J.E. Gentle, Ed., North-Holland, Amsterdam, pp. 173-179.

(29) Silverman, B.W. (1981), Density estimation for univariate and bivariate data, in Barnett, V. (ed.), *Interpreting Multivariate Data*, New York: John Wiley & Sons.

(30) Tapia, R.A., and Thompson, J.R. (1978), *Nonparametric Probability Density Estimation*, Johns Hopkins Univ. Press, Baltimore.

(31) Terrell, G.R. and Scott, D.W. (1985), Oversmoothed nonparametric density estimates, *J. American Statistical Association*, 80:209-214.

(32) Tukey, J.W. (1977), *Exploratory Data Analysis*, Reading, MA: Addison-Wesley.

(33) Tukey, J.W. (1984), Data analysis: history and prospects, in *Statistics: An Approach*, H.A. David and H.T. David, eds., pp. 183-202, Iowa State University Press, Ames.

(34) Tukey, P.A. and Tukey, J.W. (1981), Graphical display of data sets in 3 or more dimensions, in Barnett, V. (ed.), *Interpreting Multivariate Data*, New York: John Wiley & Sons.

(35) Weaver, C.S., Zollweg, J.E., and Malone, S.D. (1983), Deep earthquakes beneath Mount St. Helens: Evidence for magnetic gas transport?, 221:1391-1394.

ANALYSIS OF SUBPIXEL REGISTRATION

BY

David Lavine*

Eric C. Olson*

Barbara A. Lambird*

Carlos A. Berenstein+*

Laveen N. Kanal+*

*L.N.K. Corporation
+University of Maryland

TABLE OF CONTENTS

Introduction

Digital Straight Line Segment Parameter Estimation

Probabilistic Edge Analysis

Edge Detection

Digital Moments

Theoretical Intercept Estimation Errors

Experimentation Using the Moments

Summary and Conclusions

Appendix A

Section 1.

INTRODUCTION

This is the third report on our research aimed at understanding and obtaining analytical, quantitative results on subpixel accuracy in image registration. This research was motivated by the observation that while subpixel accuracy is very important in many practical applications of image matching, and while many claims concerning the degree of accuracy achieved in an application have appeared, analyses have been limited and a theoretical basis for understanding subpixel accuracy was lacking.

Our study, represented by this report and two previous reports [Lavine et al, 1983; Berenstein et al, 1984], has attempted to lay foundations for such a theoretical basis. These foundations have taken two primary directions: geometric models for subpixel accuracy in edge detection; and the matching of image composed of random fields.

Our previous reports on the analysis of subpixel accuracy focused heavily on the determination of the location of a real world straight edge based on a detection of its digitization in an image. Analytical results were obtained for the attainable accuracy in the estimation of the edge position. One limitation of the analysis was the assumption that the correct digitization of the edge could be determined. We made several attempts to address this problem in the previous work. Those attempts led to several approaches which were more flexible and accurate but still suffered from difficulties in the estimation of average grey levels for regions abutting the edge. Then the paper by Tabatabai and Mitchell [1984] appeared, and led us to think of new ways to simplify the estimation problem.

The relationship between the work of Tabatabai and Mitchell and our

previous work was unclear at first, but the computational simplicity of their work together with the accuracy of our results made a study of this relationship desirable. Our examination of the two approaches led to extensions of the Tabatabai-Mitchell approach which should be useful in applications to LANDSAT image registration. The relationship between the approaches also has suggested the possibility of a whole range of algorithms bridging the gap between the two approaches, in which one trades off accuracy for computational simplicity. This report describes our investigations in these directions (Sections 5 and 7).

A second research direction pursued in the present study was the resolution of a conjecture on an asymptotic expression for the number of digital lines of specified length. In our previous study of the accuracy of line position estimation given a digitization of a line, we developed general methods of error analysis and performed more detailed analysis for digital segments of a fixed length, which was chosen to be ten pixels. For the development of a more flexible theory of error analysis, we sought an asymptotic expression for the number of digital lines of any length. A conjecture for such an asymptotic expression, developed in our previous study, is proved in the current report (Section 3).

The overall direction of our study in the three phases of this study have been directed to the analysis of methods for achieving subpixel accuracy in image registration with emphasis on the use of subpixel accuracy in edge detection. Though many approaches to the problem have appeared, analysis has been limited and a general theory of subpixel accuracy is lacking. Our study has attempted to lay foundations for such a theory. These foundations have taken two primary directions, the matching of images composed of random fields and geometric models for subpixel accuracy in edge detection.

Appendix A of this report titled Subpixel Translation-Registration of Random Fields continues our work on the analysis of correlation based techniques for matching images composed of random fields and presents results of computer simulations which confirm the theoretical results. This represents one of the first systematic performance evaluations of the maximum correlation method of image registration and of a known effective variant based on maximizing a least squares quadric surface locally approximating the (discrete) correlation - statistic near its (discrete) maximum. Section 8 presents a summary and conclusion of our work.

Section 2.

DIGITAL STRAIGHT LINE SEGMENT PARAMENTER ESTIMATION

Estimation of the location parameters of a real world edge giving rise to an image edge is discussed in this section. We start with a summary of those parts of [Do-Sm] which are useful for subpixel registration. Their basic result is a determination of all lines whose digitization is a specified chain code. In a later section, we use this set of lines to derive error bounds on registration accuracy.

Several line digitization procedures are commonly used in graphics and image processing. Given a line segment in the upper right hand quadrant of the plane, with slope and y-intercept both between 0 and 1 and strictly less than 1, we define its digitization as follows: To each intersection (a,b) between the line and a line x=a, a an integer, we associate the pixel with lower left hand corner $(a, \lfloor b \rfloor)$ (see Figure 1). The chain code of the sequence of pixels with lower left hand coordinates $(0,b_0)$, $(1,b_1)$, ..., (N,b_N) is the sequence $c_1,\ldots,c_N$ where

$$c_i = \begin{cases} 0 \text{ if } \lfloor b_i \rfloor = b_{i-1} \\ 1 \text{ otherwise} \end{cases}$$

The restrictions on the sope and y-intercept of the lines under consideration are made for simplicity of presentation. By symmetry the results can be extended to remove these conditions.

To determine the lines with specified chain code, it is useful to have a parametrization of the set of all chain codes of digital line segments resulting from digitizing the class of lines specified above. In [Do-Sm] the following parametrization is given. A digital line segment chain code $(c_1,\ldots,c_N)$ is given by a quadruple of integers (N,p,q,s).

N is the length of the chain code, i.e., the number of 0's and 1's. We note that not every string of 0's and 1's is generated by a line segment. For a characterization of those that are, see [W-R].

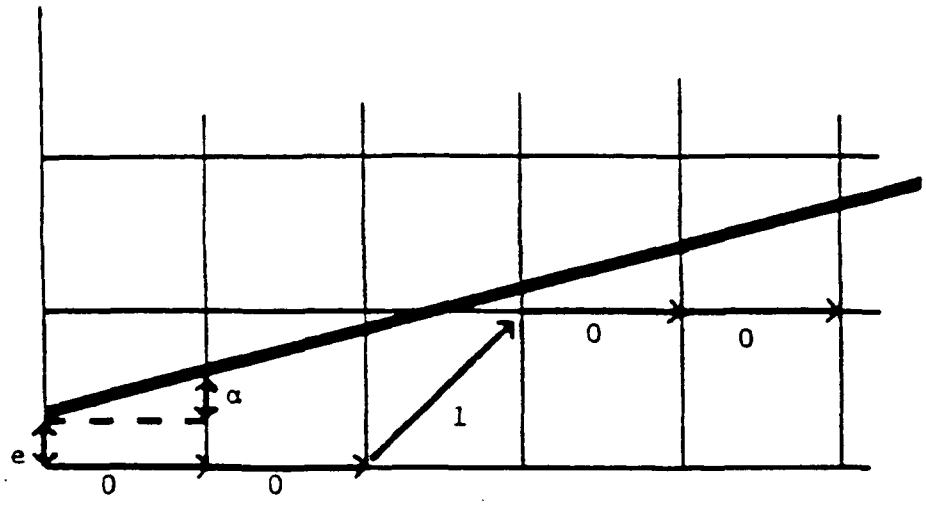


Figure 1 - Chain code of a digital line. The digitization of the dark diagonal line has pixels with lower lefthand vertices (0,0), (1,0), (2,0), (3,1), (4,1), (5,1). The resulting chain code indicated by the arrows is 00100.

Next, q is defined to be the smallest integer such that there exists an extension $c_{N+1}, c_{N+2},\ldots$, with $c_1,c_2,c_3,\ldots$ periodic with period q. Define p to be the number of ones in a period. The fourth parameter, s, provides a

normalization of the chain code for one period. Geometrically, s may be interpreted as follows. Any chain code corresponds to a line segment with rational slope. Along all such segments, select the slope p/q with $p \wedge q=1$ which has the minimum q. This q is the period. The standard chain code corresponding to the first period of ths chain code is the chain code of the digitization of the first q pixels of the line through the origin, $y=(p/q)x$. The ith element c_i, of this chain code is given by

$$c_i = \lfloor i(p/q) \rfloor - \lfloor (i-1)(p/q) \rfloor, \quad i=1,2,\ldots,N$$

The parameter s, of a code string of length N, is defined by the condition that the standard code string of p/q started at the (s+1)th element of the original chain code. Given the paramenters N,q,p,s of a code string, the ith element of the original code string can be obtained by

$$c_i = \lfloor (i-s)(p/q) \rfloor - \lfloor (i-s)(p/q) \rfloor, \quad i=1,2,\ldots,N$$

The parameters satisfy the constraints $0 \leq p \leq q \leq N$ and $0 \leq s \leq q-1$. A point which will be particularly important for the registration problem is that there are contraints on the parameters other than the above inequalities. These additional contraints are described in [Be et.al.]. Our interest in these matters stems from the need to enumerate the digital lines satisfying various conditions. If it were not for these messy constraints, the enumeration problems would often be straightforward. Without these additional constraints for fixed N, we would obtain all digital line segments of length N by independently varying s,p,q subject to the constraints $0 \leq p \leq q \leq N$ and $0 \leq s \leq q-1$.

We now give an example of the computation of the parameters for a chain code.

EXAMPLE: Chain Code 10010100
N = 8: there are 8 digits in the code
q = 5: the above code is part of the infinite code
. . . 100101001010010 . . .
p = 2: the number of 1's in the period 10010 is 2
s = 1: The standard codestring of 2/5 is 00101. The standard codestring starts at the 2nd elements of the chain code. Hence s = 1.

Since the smallest period plays an important role, let us point out two ways of computing it. The first one might be easier to use for long strings with the help of the FFT, the second one is very convenient for direct computation in short strings.

For the first algorithm extend the chain code to the right, with period, N i.e., $c_{i+N} = c_i$. Then

(1) $q = \inf \{j: 1 \le j \le N \text{ such that } 1/N \sum_{i=1}^{N} (-1)^{c_i + c_{i+j}} = 1\}$.

Notice that the maximum value of the average in the definition of q is precisely 1. In the second algorithm, we extend the code chain in both directions by zeroes and consider

$$q = \inf \{j: 1 \le j \le N \text{ such that } 1/N \sum_{i=1}^{N} (-1)^{c_i + c_{i+j}} = 1\}.$$

with the understanding that if the set of j's is empty we take q=N. What this really means is that we slide successively to the right of the chain code and compare the tail end of the original chain code with the first portion of the shifted chain code, the value q corresponds to the first perfect match, if there are no matches then q=N.

The primary result of [Do-Sm] is a description of the set of all lines whose digitization over the interval [0,N] is a set of pixels specified by a chain code. This result is of great importance for our registration accuracy results since it provides a hold on the errors which may arise by aproximating the true edge by a feasible edge. The set of lines is described by a quadrilateral in the (e,α)-plane where e is the y-intercept of a line and α is

the slope. We will call this plane the dual plane. The proof of the following formulas can be found in [Be et. al.].

Define functions F and L by:

(2) $F(s) = s - \lfloor s/q \rfloor q$

and

(3) $L(s) = s + \lfloor (N-s)/q \rfloor q$

Let ℓ be defined by the equation:

(4) $1 + \lfloor \ell(p/q) \rfloor - \ell(p/q) = 1/q$ and $0 \leq \ell \leq q$,

or what is the same, by the fact that $\ell p \equiv -1 \pmod{q}$. The set of feasible lines is a convex quadrilateral in (e,α)-space with vertices A, B, C, D given by

(5) $A = (\lfloor F(s)p/q \rfloor - F(s)p^+/q^+, p^+/q^+)$

(6) $B = (\lceil F(s)p/q \rceil - F(s)p/q, p/q)$

(7) $C = (1 + \lfloor F(s+\ell)p/q \rfloor - F(s+\ell)p/q, p/q)$

(8) $D = (1 + \lfloor F(s+\ell)p/q \rfloor - F(s+\ell)p^-/q^-, p^-/q^-)$

where

(9) $q^+ = L(s+\ell) - F(s)$, $p^+ = (pq^+ + 1)/q$

(10) $q^- = L(s) - F(s+\ell)$, $p^- = (pq^- - 1)/q$

The above expressions for the vertices of the feasible quadrilateral will be discussed in greater detail in later sections. We note here that none of the vertices A, C, D nor the points in the two sides of the quadrilateral determined by them correspond to lines that have the chain code (N,q,p,s) after digitization. It is also very important to note that (since we are working with lines of non-negative slope < 1 and non-negative ordinate to the origin < 1) the quantities p^+, q^+, q^- are strictly positive, while $p^- \geq 0$ (in fact, from (10) it follows that $p^- = 0$ only if $p = q^- = 1$). This remark, which is omitted in [Do-Sm], is crucial to provide a correct count of all distinct digitial lines of length N.

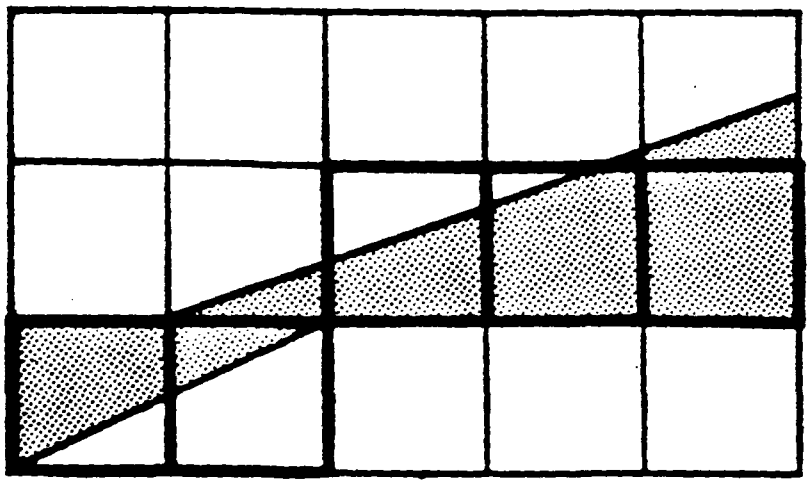


Figure 2 Feasible region for a digital line. The digital line consisting of those pixels with darkened boundaries has the shaded area as its feasible region.

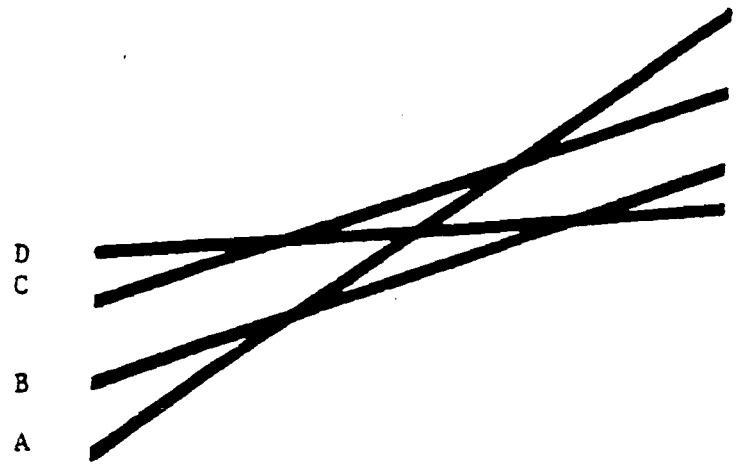


Figure 3 Intersections for the feasible region. The four boundary lines A, B, C, and D of a feasible region are shown. The intersection of A and D always lies between the parallel lines B and C. These lines in the x,y space correspond to the vertices A,B,C,D of the feasible quadrilateral in the (e,θ) parameter space.

Section 3.

PROBABILISTIC EDGE ANALYSIS

This section presents analytical results on the error analysis of the digital line approach to edge detection. Previous results on error bounds for offset estimation accuracy are reviewed. An asymptotic formula for the number of digital lines of a given length, which was conjectured previously, is proved and corresponding asymptotic error analysis is given.

A worst case bound on registration accuracy using digital edge was described in [Be et.al.]. More realistic error information can be obtained using probability. In this section we consider the question of obtaining probabilistic information on the registration error assuming the real world edge giving rise to the digital edge is generated by a natural distribution on edges. We have procedures for estimating these probabilities, but due to the considerable computational cost involved in evaluating these in special cases, we prefer to first seek analytical simplifications.

Many probabilitic questions pertinent to the geometric accuracy question can be formulated. In this section we consider the problem of determining the probability, that the actual registration error will not exceed a specified level. We wish to determine, for any acceptable error level in the estimated offset between sensed and reference image, what is the probability that a random edge will result in a digitization which permits estimation to better than that error level. Though a simple formula for these probabilities as a function of digital line length is not available, a procedure for calculating these probabilities for any given line length, N, is described and results for the case N=10 are presented. In addition we present an asymptotic expression for the error.

The basic approach to computing the error probabilities is quite simple. A probability density function is given on the set, A, of all lines with slope between 0 and 1, going through the pixel with lower left vertex (0,0). Since a line has only one chain code, the sets of lines with different chain codes gives a partition of the set A. Hence the density on lines induces a density on chain codes. For a chain code with period q, the maximum error is 1/2q as was shown in [Be et. al.]. Thus for any specified error h, we must calculate the probability of the following set, B, of line chain codes.

(1) $B = \{(N,q,p,s): \ 1/2q < h\}$

The set of all linear chain codes of length N can be enumerated. For each chain code in B, the corresponding feasible quadrilateral can be calculated as in Section 2. The density function on lines can then be integrated over the quadrilateral and the sum of these integrals over all members in B computed. This sum yields the desired probability.

The problem of enumerating linear chain codes of lines through the origin was discussed in [R-W] where also an algorithm for generating the set of linear chain codes was presented. We have not found any estimates in the literature of the number of chain codes of a given length. The problem is that the shortest period of the digital line of length N corresponding to a line

(2) $y = (p/q)x+m/q$

might be strictly smaller than q. Since such lines generate all the possible digital lines and we can associate to each a code (N,q,p,s), the problem reduces to characterize those values of s for which this code does not coincide with $(N,\bar{q},\bar{p},\bar{s})$ with $\bar{q} < q$. The answer lies in the following.

Proposition 1: Given a code (N,q,p,s), the necessary and sufficient condition that it does not coincide with a code of strictly smaller period is that $q^+>0$ and $q^-<0$, where q^+,q^- are defined by (2.9) and (2.10).

Proposition 1 and its proof give us a way to compute the number $L(n,q)$ of digitial lines of length N and smallest period q. In fact $L(N,1)=1$ so we can consider $q>1$, then the situation $N-s<q$ can only arise if $N\leq q+s-1\leq 2q-2$, that is, $(N+2)/2\leq q$. Hence, if $q<(N+2)/2$, s can take arbitrary values and it follows that

(3) $L(N,q)=q\phi(q)$ for $2\leq q<(N+2)/2$

where $\phi(q)$ is the Euler function that counts the number of values p, $1\leq p\leq q$, $(p,q)=1$. This formula is clearly valid for $q=1$ since $\phi(1)=1$. In the remaining range of q we can use the fact that when p runs over all the values considered in $\phi(q)$, so does ℓ, where we remind the reader ℓ is defined by (2.4). We fix ℓ and divide the range of s into two classes

(4) $0\leq s\leq N-q, N-q+1\leq s\leq q-1$

The second class is not empty since we are assuming $N+2\leq 2q$. In the first class every line has smallest period q, this accounts for $N-q+1$ lines. In the second class we have two subclasses, $s+\ell<q$ and $q\leq s+\ell$. The first one cannot introduce any lines of period q due to the condition $q^-<0$. In the second one we have to consider whether

(5) $N-(s+\ell-q)\geq q$

or not. Only if this inequality is true we get new lines (due to the condition $q^+>0$). Hence we must have

(6) $\max\{q-\ell,\ N-q+1\}\leq s\leq \min\{q-1, N-\ell\}$

which gives us $1+\min\{\ell-1,\ N-q,\ 2q-N-2,\ q-\ell-1\}$ lines (notice that this minimum

is non-negative). Therefore, in this range of values of q we have

(7) $L(N,q) = (N-q+2)\phi(q) + \sum_{\ell} \min\{2q-N-2,\ q-\ell-1,\ \ell-1,\ N-q\}$

where the sum takes place over all values ℓ, $1 \leq \ell \leq q-1$, $\ell \wedge q = 1$.

Proposition 2: Let L(N) be the number of digital lines of length N with both slope and y-intercept between 0 and 1. Then

$$(8) \quad L(N) = \sum_{q=1}^{\lceil \frac{N}{2} \rceil} q\phi(q) + \sum_{q=\lceil \frac{N}{2} \rceil + 1}^{N} (N-q+2)\phi(q)$$

$$+ \sum_{q=\lceil \frac{N}{2} \rceil + 1}^{N} \sum_{\substack{\ell=1 \\ (\ell,q)=1}}^{q-1} \min\{2q-N-2,\ q-\ell-1,\ \ell-1,\ N-q\}$$

Since this expression is a little bit hard to work with, we can use upper and lower estimates, $L^{**}(N,q) = q\ (q)$, $L_{*}(N,q) = (N-q+2)\ (q)$ for q in this range. Finally, setting L(N) = total number of digital lines of length N, we get the estimates

$$(9) \quad L_{*}(N) = \sum_{q=1}^{\lceil N/2 \rceil} q\phi(q) + \sum_{(N/2)+1=q}^{N} (N-q+2)\phi(q)$$

$$<= L(N) <= L^{**}(N) = \sum_{q=1}^{N} q\phi(q)$$

Using the above formulas we can produce the following table for N=10.

q	$\phi(q)$	$L_*(N,q)$	$L(N,q)$	$L^*(N,q)$
1	1	1	1	1
2	1	2	2	2
3	2	6	6	6
4	2	8	8	8
5	4	20	20	20
6	2	12	12	12
7	6	30	36	42
8	4	16	20	32
9	6	18	22	54
10	4	8	8	40
TOTAL:		21	135	217

We notice that L(N) is fairly close to $L_*(N)$ and very different from $L^{**}(N)$. $L^{**}(N)$ would have been the count if no digital lines drop their period when considered to have finite length. A better upper bound function $L^*(N,q)$ can be defined as follows:

$$
\begin{aligned}
L^*(N,q) &= L(N,q) \quad 1 <= q <= \lceil (N/2) \rceil \\
L^*(N,q) &= L_*(N,q) + (2q-N-2)(\varphi(q)-2), \quad (N/2)+1 <= q \\
&\quad <= (2/3)N + 2/3 \\
L^*(N,q) &= L_*(N,q) + (N-q)(\varphi(q)-2), \quad (2/3)N+2/3 < q \\
&\quad <= N
\end{aligned}
\tag{10}
$$

The choice is motivated by choosing the smallest of the two terms independent of ℓ in the minimum that appears in (7). Since the values $\ell=1$, $\ell=q-1$ make this minimum zero we only have $(\phi(q)-2)$ terms in the sum. We also note that $L^*(N,q)=L_*(N,q)$ for $q=(N/2)+1$ (if this value is an integer) and for $q=N$. For

N=10, we have only three values to compute

(11) $L^*(N,7) = 38,\ L^*(N,8) = 20,\ L^*(N,9) = 22$

which gives $L^*(10) = 137$ in this case, a very good approximation (We have used $L^*(N) = \sum_{q=1}^{N} L^*(N,q)$.

Proposition 3:

The following asymptotic development for L(N) holds:

(12) $L(N) = \frac{N^3}{\pi^2} + O(N^2 \log N)$.

We can compare this proposition with the asymptotic behavior of the asymptotic behavior of the upper and lower bounds that were proven in [Be et. al.]:

(13) $L(N) = (3/4\pi^2)N + O(N^2 \log N)$ $0.076N^3$

(14) $L(N) = (10/9\pi^2)N + O(N^2 \log N)$ $0.112N^3$

We have computed L(N) and L'(N) (the leading term of the asymptotic formula 12) for N=100 and found the following values

(15) $L(N) = 104,359$

(16) $L'(N) = 104,949$

The relative error between these two values is only 0.5%. In order to prove Proposition 3, we need to introduce some preliminary lemmas.

Lemma 1

$$\sum_{d|n} \frac{|\mu(d)|}{d} = O(\log\log n)$$

$$\sum_{d|n} \frac{1}{d} = O(\log\log n)$$

Proof: The first sum is over the divisors that are square-free; it coincides with $\prod_{\substack{p\mid n \\ p \text{ prime}}} \left(1 + \frac{1}{p}\right)$. This is clearly as large as

possible if n is itself the product of the first r primes, $n = p_1 \cdots p_r$. We now estimate r and p_r. We have

$$\log n = \sum \log p_i \geq C\, p_r ,$$

by [Ha-Wr, Theorem 414]

Also $p_r \approx r \log r$, by the Prime Number Theorem

$$\therefore \log n \geq C r \log r$$

$$\therefore r \leq C \frac{\log n}{\log\log n}$$

Now $$\log \prod_{p\mid n} \left(1 + \frac{1}{p}\right) \leq \sum_{p \leq p_r} \frac{1}{p} \cong \log\log p_r$$

Thus $$\prod_{p\mid n} \left(1 + \frac{1}{p}\right) = O(\log\log n)$$

To prove the second estimate, one needs to show that:

$$\sum_{d\mid n} \frac{1}{d} \Big/ \sum_{d\mid n} \frac{|\mu(d)|}{d} \leq C$$

We have $$\sum_{d\mid n} \frac{1}{d} \Big/ \sum_{d\mid n} \frac{|\mu(d)|}{d} \leq \prod_{p\mid n} \left(\frac{1 + \frac{1}{p} + \frac{1}{p^2} + \cdots}{1 + \frac{1}{p}} \right)$$

$$\leq \prod_{p} \left(\frac{1 + \frac{1}{p} + \frac{1}{p^2} + \cdots}{1 + \frac{1}{p}} \right)$$

$$= \prod_p \left(1 - \frac{1}{p}\right)^{-1} \left(1 + \frac{1}{p}\right)^{-1}$$

$$= \prod_p \left(1 - \frac{1}{p^2}\right)^{-1} = \sum_1^\infty \frac{1}{n^2} = \frac{\pi^2}{6} \leq 1.65$$

Let $F(x) = \sum_{\substack{(\ell,n)=1 \\ \ell \leq x}} 1 = x - \sum_{\substack{p|n \\ (p \leq x)}} \left[\frac{x}{p}\right] + \sum_{\substack{p|n \\ q|n \\ p \neq q \\ pq \leq x}} \left[\frac{x}{pq}\right] - \sum \left[\frac{x}{pqr}\right] + \cdots$

$$= x\left(1 - \sum_{p|n} \frac{1}{p} + \sum_{\substack{p|n \\ q|n}} \frac{1}{pq} \cdots\right) + \text{error}$$

$$\prod\left(1 - \frac{1}{p}\right) = \frac{\phi(n)}{n}$$

$$\text{error} \leq \sum_{\substack{p|n \\ p \leq x}} \frac{1}{p} + \sum_{\substack{p|n \\ q|n \\ p \neq q \\ pq \ x}} \frac{1}{pq} + \cdots \leq \sum_{\substack{d|n \\ d \leq x}} \frac{|\mu(d)|}{d} \leq \sum_{d|n} \frac{|\mu(d)|}{d} = O(\log\log n)$$

By Lemma 1

The distribution function F(x) of the number of ℓ's relatively prime to n, for $x \leq n$ is given by

$$F(x) = \frac{\phi(n)}{n} x + O(\log\log n)$$

We obtain the following corollary:

$$\sum_{\substack{\ell \leq n \\ a \leq \ell \leq b \leq n}} = F(b) - F(a) = (b-a)\frac{\phi(n)}{n} + O(\log\log n)$$

$$\sum_{\substack{\ell \mid n \\ a \le \ell \le b}} \ell = \int_a^b x\,dF = xF(x)\Big|_a^b - \int_a^b F(x)\,dx$$

$$= x^2\frac{\phi(n)}{n}\Big|_a^b + O(\log\log n)\,(b-a) - \frac{\phi(n)}{n}\int_a^b x\,dx + (b-a)\,O(\log\log n)$$

$$= \frac{1}{2}x^2\,\frac{\phi(n)}{n}\Big|_a^b + (b-a)\,O(\log\log n)$$

$$= \frac{b^2 - a^2}{2}\,\frac{\phi(n)}{n} + (b-a)\,O(\log\log n)$$

Proof of Proposition 3:

First we want to deal with the term

$$\sum_{\ell} \min\left\{2q-N-2,\ q-\ell-1,\ \ell-1,\ N-q\right\}$$

Where q has been fixed in the range $\left\lceil\frac{N}{2}\right\rceil + 1 \le q \le N$, and the summation takes place over ℓ, $1 \le \ell \le q$, $(\ell, q) = 1$. Clearly we can assume $1 < \ell < q$ otherwise the corresponding term is zero. First we divide the range of q according to whether $N-q < 2q-N-2$ or not.

In the first case $2N + 2 < 3q$ so $q > \left\lceil\frac{2N+2}{3}\right\rceil$.

If $\left\lceil\frac{N}{2}\right\rceil + 1 \le q \le \left\lceil\frac{2N+2}{3}\right\rceil$, and we graph the minimum as a function of ℓ we have a trapezoid. That is, for small ℓ the minimum is $\ell-1$, for ℓ near q the minimum is clearly $q - \ell - 1$ and in the middle range we have 2q-N-2, and we only have to find the cut off points:

$$\ell - 1 \le 2q-N-2 \Rightarrow 1 < \ell \le 2q-N-1$$

$$2q-N-2 \ge q-\ell-1 \Rightarrow q > \ell \ge N-q+1$$

The middle range is compatible because of the assumption on q that

$$2q-N-2 < N-q.$$

The sum

$$\sum_{2}^{2q-N-1} (\ell-1) + \sum_{2q-N}^{N-q} (2q-N-2) + \sum_{N-q+1}^{q-1} (q-\ell-1)$$

can be written using the corollary to Lemma 1

$$\frac{\phi(q)}{q} (2q-N)(N-q) + O(N \log\log N)$$

Over the remaining range of q we get exactly the same expression. Hence

$$L(N) = \sum_{q=1}^{\left\lceil \frac{N}{2} \right\rceil} q\phi(q) + \sum_{\left\lceil \frac{N}{2} \right\rceil + 1}^{N} (N-q+2)\phi(q) + \sum_{\left\lceil \frac{N}{2} \right\rceil + 1}^{N} \left(\frac{\phi(q)}{q} (N-q)(2q-N) \right) + O(N \log\log N)$$

Now recall that the distribution function Φ of (q) is known to have the asymptotic behavior [Ha-Wr, Theorem 330]

$$\Phi(x) = \sum_{q \leq x} \phi(q) = \frac{3x^2}{\pi^2} + O(x \log x)$$

and hence the asymptotic behavior of L(N) can be computer from the above expression using Stieltjes integrals and integration by parts as we did with the Corollary of Lemma 1

$$\int_{1}^{\frac{N}{2}} x \, d\Phi(x) = \frac{1}{\pi^2} \frac{N^3}{4} + O(N^2 \log N)$$

$$\int_{\frac{N}{2}}^{N} (N+2-x) \, d\Phi(x) = \frac{1}{2} \frac{1}{\pi^2} N^3 + O(N^2 \log N)$$

$$\int_{\frac{N}{2}}^{N} \frac{(N-x)(2x-N)}{x}\, d\Phi(x) = \frac{3}{\pi^2}\left[\frac{2}{3}\left(N^3 - \left(\frac{N}{2}\right)^3\right) - \frac{N^3}{2}\right] + O(N^2 \log N)$$

$$= \frac{N^3}{4\pi^2} + O(N^2 \log N)$$

Using these three integrals together and noting the discarded term is only $O(N^2 \log\log N)$, we have

$$L(N) = \frac{N^3}{\pi^2} + O(N^2 \log N).$$

We are now ready to study the asymptotic behavior of the error in the offset estimate. Let us recall that for a given period q, the minimum width of the channel parallel to the line B (Section 2) is 1/q. We set

(17) $$S(N) = \sum_{q=1}^{N} (1/q)L(N,q)$$

Then the average offset error incurred by using the line parallel to B passing through the middle of the channel is given by

(16) $E(N) = ((1/2)S(N))/L(N)$

when we use the uniform distribution on digital lines.

Proposition 4: The asymptotic behavior of S(N) and E(N) is given by

(17) $S(N) = (6(1-\log 2)N^2)/\pi^2 + O(N \log N)$

(18) $E(N) = (3(1-\log 2))(1/N + O(\log N/N^2)$

Hence on the average we expect an offset error of approximately 0.92/N.

Proof

We have from (7)

$$S(N) = \sum_{q=1}^{N} \frac{1}{q} L(N,q)$$

$$= \sum_{q=1}^{\left[\frac{N}{2}\right]} \phi(q) + \sum_{\left[\frac{N}{2}\right]+1}^{N} \frac{N-q+2}{q} \phi(q) + \sum_{\left[\frac{N}{2}\right]+1}^{N} \frac{\phi(q)}{q^2} (N-q)(2q-N)$$

$$+ O(N \log N \log\log N)$$

Now we can show as before

$$\int_{\frac{N}{2}}^{N} \frac{N-x-2}{x} \, d\Phi(x) = \frac{3}{\pi^2} \frac{N^2}{4} + O(N \log N)$$

and

$$\int_{\frac{N}{2}}^{N} \frac{(N-x)(2x-N)}{x^2} \, d\Phi(x) = \frac{3}{\pi^2} N^2 \left(\frac{3}{2} - 2 \log 2\right) + O(N \log N)$$

Hence

$$S(N) = \frac{3}{\pi^2} N^2 \left[2 - 2 \log 2\right] + O(N \log N \log\log N)$$

and

$$E(N) = \frac{1}{2} \frac{S(N)}{L(N)} = \frac{3(1-\log 2)}{N} + O\left(\frac{\log N \log\log N}{N^2}\right)$$

$$\approx 0.92 \frac{1}{N}$$

Section 4.

EDGE DETECTION

One approach taken to edge detection in an earlier phase of our study on subpixel accuracy was to search for the edge using a hypothesize and test procedure. This procedure, which is described in [Be et.al.], proved accurate in experiments. Unfortunately, it is difficult to evaluate the process analytically, since the effect of noise on the search is difficult to quantify.

A simple approach to the subpixel detection of edges which is efficient and can be easily analyzed is described in this section. This technique is based on the idea of matching the moments of a digital image window with those of a continuous scene with an ideal edge in order to estimate the edge position. This approach of matching moments for edge detection first appeared in a paper of Tabatabai and Mitchell [1984]. Our assumptions that the edge location is approximately known and that the edge orientation is known provides simplifications which permit more complete analysis of the algorithm performance. In addition, the assumptions enable us to make additional adjustments for digitization error.

The basic approach to edge detection taken by Tabatabai and Mitchell is to set the first three observed moments of the image equal to the first three moments of a continuous image with an ideal step edge. In their case, the slope and y-intercept of the edge are unknown as are the grey levels on the two sides of the edge. They use a digital disk for a window and write the first three moments of the real edge in terms of three parameters: the grey levels, h1 and h2, on the two sides of the edge and the area, A, on one side of the edge. They then set these three moments to be equal to the first three moments of the observed image amd solve for h1, h2, and A.

One desirable feature of the Tabatabai-Mitchell approach is that it is unnecessary to know the average grey levels on the two sides of the edge before estimating the edge location to subpixel accuracy. For the purposes of the present study, we assume the edge position is known to within a pixel, so unless the areas on the two sides are small, this additional flexibility may not be very useful. On the other hand, if the regions abutting the edge have relatively few pixels, it may be desirable to use the mixed edge pixels in estimating the region grey levels for use in edge detection. In the procedure described in this section, we assume that the grey levels for the region are estimated without using the mixed pixels.

One problem with the Tabatabai-Mitchell approach is that it is based on the assumption that the digital moments and real moments are equal if no noise is present. While the first moments are the same, it can be easily seen that other moments do not agree. We have not yet been able to develop an exact formula to correct for this discrepancy, but we have been able to derive an empirical correction formula which works well.

We now describe a procedure for detecting straight edges to subpixel accuracy given that the orientation of the edge is known and given that the mean grey levels on the two sides of the edge have been estimated. This algorithm draws heavily on the work of Tabatabai and Mitchell [1984], adapting it to make more effective use of the assumptions on the current problem.

The Simple Moment Edge Detector (SMED) seeks to find a single edge in an nxn square window. A window width of ten was selected for the experimental study. The window has lower left hand coordinates of (0,0) and upper right hand coordinates (10,10). For simplicity of experimentation, we assumed that the edge is given by a line $y=mx+b$ where $m>0$, $0<=b<=10$ and $0<=10m+b<=10$. Let the grey level above the edge be h1 and the grey level below the edge be h2.

Let A1 denote the area above the edge and let A2=100-A1 be the area below the edge. The ith moment, mi, of the real edge is defined by

(1) $$mi=A1*h1^{i}+A2*h2^{i}.$$

The digital moments are computed in a similar fashion. Let xij denote the pixel whose lower left hand corner has coordinates (i,j). Then the kth moment, mek, is defined by

(2) $$mek=\Sigma(xij)^{k}.$$

The digital edge is formed by assigning, to each pixel not intersected by the line, the corresponding grey level from the continuous image and by, assigning to each pixel the line goes through, a weighted average of the grey levels h1 and h2. The weights are the fractions of the area of the pixel above and below the line.

The slope of the line, m, is assumed known and the grey levels h1 and h2 are assumed to be estimated from the data prior to the calling of the procedure SMED. For the present analysis, we assume the estimates of h1 and h2 are exact. Thus the only parameter to be estimated is the y-intercept of the line. In the noise free case, the y-intercept can be written as a function of m, h1, h2, and any one of the moments. Let yi denote the y-intercept. Then

(3) $$A2=h2*yi+n^{2}m.$$

The kth moment of the real image is given by

(4) $$m1=h1^{k}*A1+h2^{k}*A2.$$

Equating the real and observed kth moments we get

(5) $$h1^{k}A1+h2^{k}A2=\Sigma xij^{k}.$$

Since the sum of the areas is n^{2}, we have

(6) $$A1+A2=n^{2}.$$

Substituting (6) in (5) and simplifying, we get

$$(7) \qquad A2 = \frac{1}{h2^k - h1^k} \left(\sum xij^k - h1^k n^2\right)$$

From (3) and (7), we get

$$(8) \qquad y1 = \frac{\frac{1}{h2^k - h1^k}\left(\sum xij^k - h1^k n^2\right) - n^2 m}{h2^k - h1^k}$$

This provides us with an estimate of y in terms of the known parameters and the kth moment of the observed data.

An expression for the y intercept of an edge in terms of the first or second moment of the observed window has been developed. Unfortunately, as indicated above, the higher digital moments do not agree with the higher continuous moments. This discrepancy will result in errors in the estimation of a y-intercept, even in the absence of noise. This problem is explored in the next section and an analysis of the error in the y-intercept estimate is given in the following section.

A variation of the SMED will be described in Section 6. In this variation, only pixels near the edge will enter into the moment calculation. This has the advantage of using information only from those pixels we suspect of being mixed pixels containing the edge. This approach is more reasonable from a statistical point of view than using the entire square, but it is more difficult to compute.

Section 5.

DIGITAL MOMENTS

One problem in using the moment matching technique for subpixel accuracy in edge detection is the fact that even for noise-free images, the digital and continuous moments are not equal. Since the experimental investigation of many choices of moment exponent (including fractional exponents), line slopes, y-intercepts, and noise levels is costly, it is desirable to have a theoretical analysis of the effect of this error. It is also desirable to have some means for compensating for the discrepancy. We have not yet been able to develop a general theoretical analysis of the problem. In this section we introduce some empirical results and initial theoretical results.

The difference between digital and continuous moments can best be seen in the case of a single pixel. Let L be a line going through a pixel. Let the two regions into which the line divides the pixel have areas A1 and A2. Let the region with area A1 have constant grey level h1 and let the other region have grey level h2. We now assume the grey levels are fixed and first determine that value of A1 which results in the maximum discrepancy between real and computed moments and second determine that maximum resulting discrepancy. Note that there is no error if A1 or A2 is zero.

We define the computed second moment, c(A1), and the real second moment, r(A1) and define the error, e(A1) by

(1) $$e(A1)=r(A1)-c(A1).$$

The functions c and r and given by:

(2) $$c(A1)=(h1A1+h2A2)^2,$$

(3) $$r(A1)=h1^2A1+h2^2A2 .$$

Substituting $A2=n^2-A1$ into (2) and (3) and substituting these expressions into

(1), we get

(4) $$e(A1)=A_1(h1^2+h2^2-2h1h2) + A1^2(-h1^2-h2^2+2h1h2)$$

Differentiating e, setting the derivative to zero and solving, we see that the error e is maximized when A1=1/2. The resulting error is

(5) $$e(1/2)=1/4(h1^2+h2^2-2h1h2)$$

For n=10, h1=20, and h2=10, and a horizontal edge bisecting the window, the second digital moment is off by 12% from the second real moment. The resulting error in the y-intercept estimate can be much greater, depending on the parameters of the edge. Analytical calculations for higher order moments and for real exponents rather than integer exponents in the moment definition are harder to calculate, but we believe that the moment error effect grows with the size of the exponent.

Since the theoretical analysis of the digitization error in the second moment is difficult for lines with arbitrary orientation and y-intercept, we took an empirical approach to determining a correction term. Real edges with intercepts ranging from 0.1 to 0.9 in steps of 0.1 and slopes from 0.1 to 0.9 in steps of 0.1 were digitized. Grey levels of 10 and 20 were used on the two sides of the real edge. The window size was 10x10. For each slope the average error in the second moment was computed. This average was taken over all intercepts with that slope. The moment errors ranged from 167 at a slope of 0.1 to 220 at a slope of 0.9. The average error was monotonically increasing as a function of the slope. A linear function agreeing with the observed values at the extreme slopes was used to represent the moment error. The maximum difference between the value of the linear function evaluated at a given slope and the corresponding error in the second moment was about 5. Thus the linear approximation to the second moment provided a correction which yielded a second moment which was within approximately 3% of the correct

value. This contrasts with the uncorrected error of approximately 12% in the horizontal line case.

While the above empirical correction scheme was adequate for the experiments, a theoretical analysis of the error would be highly useful. Under our assumptions, the y-intercept can be estimated using the sum of the pixel values raised to any positive real power, not just the positive integers as one encounters in using the moments. Based on limited experimentation, it appears that the choice of exponent should vary with the slope of the line. To determine the optimal exponent, it would be useful to be able determine the exponent which results in the best y-intercept estimate for a given slope. This optimization, which will be discussed later, depends heavily on a knowledge of the dependence of the moment error as a function of the slope.

One aspect of the moment error analysis which may be approachable using the digital geometry techniques of the previous report [Be et. al.] is the estimation of the dependence of the moment error on the y-intercept for a fixed slope. Consider the behavior of the digitizations obtained by translating a line parallel to itself. The effect of this translation is to cause the chain code describing the digitization to change by rotating the portions of the chain code within each period. Thus the digitizations of the various lines tend to exhibit considerable similarity. Since the moment error is only dependent upon the relative areas above and below the edge in the mixed pixels, one might hope that translating the edge results in a set of pixels with approximately the same relative areas with the areas occuring in a possibly different order. The rotation of the chain codes suggests that it may be possible to derive bounds on the effect of line translation on the sequence of relative pixel areas. This topic appears a promising direction for future work. If the effect of translation on the moments can be bounded,

then the determination of the variation in moment error as a function of line slope and intercept may be simplified.

Section 6.

THEORETICAL INTERCEPT ESTIMATION ERRORS

We now compute the error distribution for the estimate of the y-intercept of an edge given its slope, the average grey levels above and below, and the observed grey levels. We consider a modification of the procedure outlined previously in which observations from the entire nxn window were used. The modified procedure is more realistic than the corresponding analysis based on the full window.

The new intercept estimation procedure uses a parallelogram instead of a rectangle where two sides of the parallelogram are parallel to the edge and the others are vertical. Assuming a prior registration which is accurate to within a pixel is available, many such parallelograms containing the edge can be constructed. The following analysis contains parameters which are a function of whichever parallelogram is selected.

The geometry of our parallelogram window is shown in Figure 1. The area below the edge L and above the bottom of the parallelogram is A1. The area A, of the parallelogram is equal to A1 + A2. The bottom of the parallelogram has height h and L has y-intercept y1. The area A can be easily computed:

(1) $A1 = n(y1-h)$

which can be rewritten as

(2) $y1 = A1/n + h$

We would like to set the real first moment of pixels in the parallelogram equal to the digital first moment. If h1 and h2 are the mean grey levels corresponding to A1 and A2, then the moment is h1A1 + h2A2. For pixels entirely inside the parallelogram, the contribution of these pixels to the first digital moment is just the sum of their grey levels. For pixels which

lie part inside and part outside the parallelogram, we take the contribution to be the area of the part of the pixel inside the parallelogram times the pixel grey level. Assume we have r pixels at least partially contained in the parallelogram and let w(i), i=1,...,r denote the area of the part of the ith pixel which lies inside the parallelogram. Then the first digital moment is defined to be $\Sigma\, w(i)xi$ where xi is the observed value of pixel i. (All summations in this section are for i=1,...,r.) Setting the real and computed first moments equal,

we get

$$(3) \qquad h1A1 + h2A2 = \Sigma\, w(i)xi.$$

Solving for A1, we get

$$(4) \qquad A1 = \frac{\Sigma\, w(i)xi - Ah2}{h1-h2}$$

Substituting into (2), we get

$$(5) \qquad y1 = \frac{\Sigma\, w(i)xi - Ah2}{n(h1 - h2)} + h$$

We now consider the modelling of noise in the above formulation. We assume the observed value of each pixel can be written as

$$(6) \qquad xi = yi + zi$$

where yi represents the noise free value and the $\{zi\}$ are identically distributed independent normal random variables with mean zero and variance σ^2.

The estimated value $\hat{y1}$ of y1, can be written, using (5) and (6) as

$$(7) \qquad \hat{y1} = \frac{\Sigma w(i)(yi+zi) - Ah1}{n(h1 - h2)} + h$$

It is easily seen from (7) that $\hat{y1}$ is an unbiased estimator of y1. The expression for y1 can be rewritten as

$$(8) \qquad \hat{y1} = y1 + \frac{\Sigma\, w(i)zi}{n(h1 - h2)}$$

The second term in (8) is a weighted sum of normally distributed random variables and is normal. Thus to completely characterize the error

distribution of $\hat{y1}$, we need only compute the variance of the second term. This variance can be easily computed yielding

$$(9) \qquad \hat{y1} \sim N\left(y1, \frac{(w(1)^2 + \ldots + w(r)^2)\sigma^2}{n^2(h1 - h2)^2}\right)$$

Defining the signal-to-noise ration S by

$$(10) \qquad S_N = 10\log\frac{(h1-h2)^2}{\sigma^2},$$

we see that a constant signal-to-noise ratio implies

$$(11) \qquad \sigma^2 = c(h1 - h2)^2,$$

where c is a constant.

For a constant signal-to-noise ratio, we have

$$(12) \qquad \hat{y1} \sim N\left(y1, \frac{(w(1)^2 + \ldots + w(r)^2)c}{n^2}\right)$$

Note that if one edge is horizontal, the parallelogram is a rectangle. If we fatten the rectangle to be the nxn square, then $r=n^2$ and $w(i)=1$ for all i. In this case

$$(13) \qquad \hat{y1} \sim N(y1, c)$$

For fixed S_N, the variance of $\hat{y1}$ is minimized by minimizing $w(1)^2+\ldots+w(r)^2$. For a fixed edge location, reducing the width of the parallelogram reduces the variance. Intuitively this merely says we should take the narrowest parallelogram that we are certain contains the edge.

The variation in the variance of $\hat{y1}$, for fixed parallelogram width and changing edge slope is difficult to determine analytically. If we assigned the area within the parallelogram a grey level of one, and the area outside a grey level of zero, then the second moment of the corresponding digitized image is $w(1)^2+\ldots+w(r)^2$. The second moment of the continuous image is trivial to calculate. Thus the determination of the variation of $w(1)^2+\ldots+w(r)^2$ with slope is equivalent to the determination of the digital moment discrepancy in the previous section and remarks there about methods attacking the problem apply.

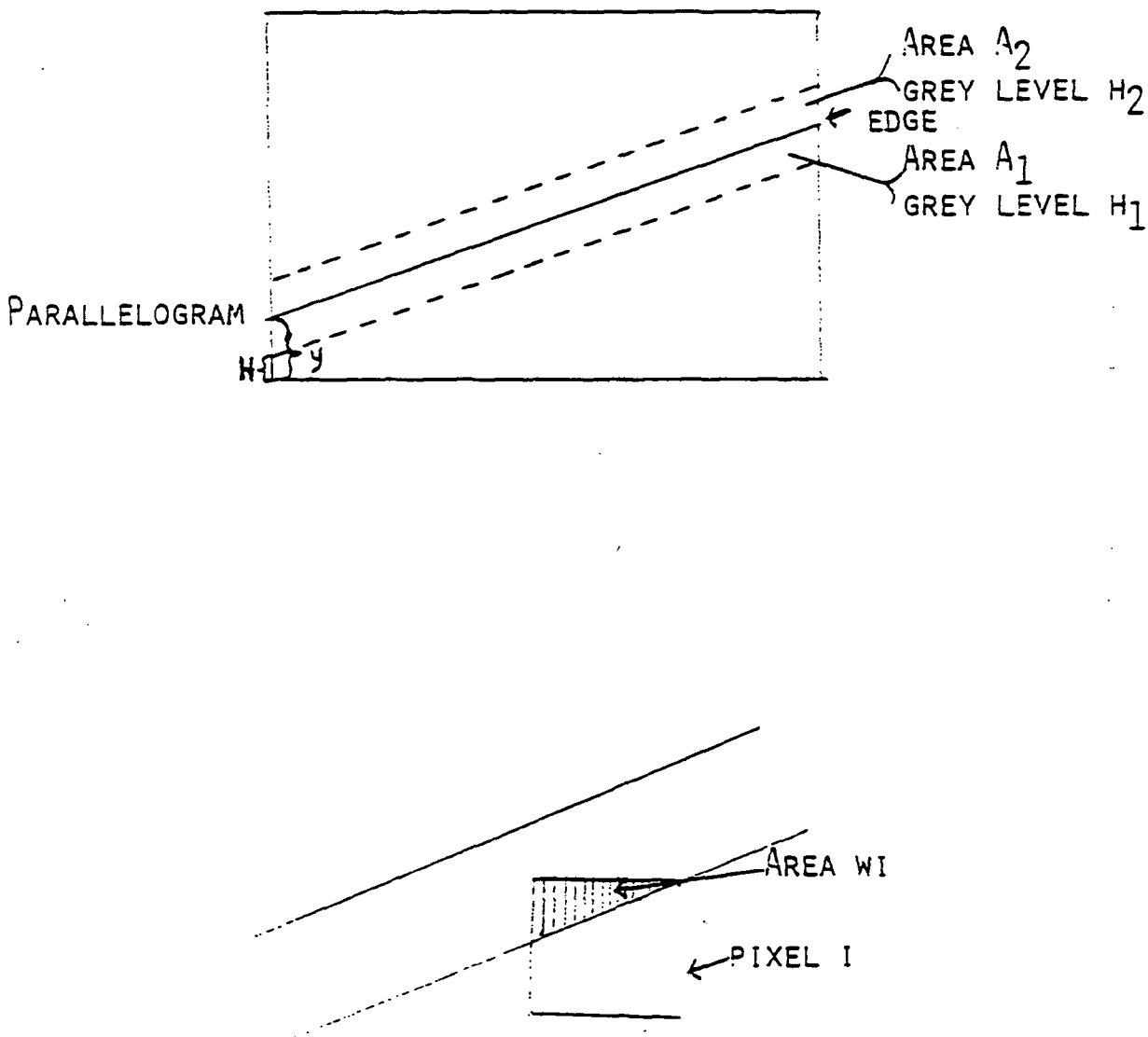


Figure 1. Parallelogram Mask

Section 7.

EXPERIMENTATION USING THE MOMENTS

Two sets of experiments were performed. One set used a square window of size 10X10. The second set used a parallelogram window of width 10 as described in Section 6. Three moments were considered: the first moment, second moment and a "square root" moment. The square root moment is calculated as follows:

$$M^{\frac{1}{2}} = A1*h1^{\frac{1}{2}} + A2*h2^{\frac{1}{2}}.$$

The first step of each set of experiments was to determine the dependence of the error in the estimated y-intercept on the true y-intercept and the slope. In both sets of experiments there was no appreciable dependence on the y-intercept. There was no appreciable dependence on slope in the square window set of experiments. There was a slight dependence on the slope in the parallelogram window set of experiments (see Figure 1) for the second and square root moments. Although the dependence is slight, it can lead to much larger errors in the estimate of the y-intercept which is also shown in Figure 1. Since the dependence appeared to be linear, a linear least squares fit was made to get a correction term for the digital moments. The second and square root moments were then corrected as follows:

$$M2 \leftarrow M2 + 328*slope + 958$$

$$M^{1/2} \leftarrow M^{1/2} - 7.27*slope - 12.4.$$

The linear corrections reduced the error between the real and digital moments to less than one percent of the real moment.

Experiments were then run to determine the error in the y-intercept estimate for the three types of moments using both types of windows. The slope was varied from 0 to 1 and three signal-to-noise ratios were used (6, 9 and 13). Forty iterations for each slope were performed and the average and

standard deviation for the error in the y-intercept estimation were found. Figures 2 through 8 present the results of the experiments.

The results can be summarized as follows. In both cases (square vs parallelogram windows), the first moment gave the least error in the y-intercept and the least standard deviation of the error. For a signal-to-noise ratio of 13, the average error in the y-intercept was about 1/10 of a pixel. As shown in Figure 3, the two types of windows had approximately the same average error, but the parallelogram window does lead to a significant reduction in the standard deviation. Figures 4 and 5 show the improvement that results through the use of the correction term for the digital calculation for the second and square root moments. Greater signal-to-noise ratios affect both the average error and the standard deviation of the error in the y-intercept estimate, as shown in Figures 6 and 7.

The results of these experiments are positive and indicate that further experimentation and analysis could lead to a fruitful, but simple procedure that would be useful for Landsat registration.

Figure 1.

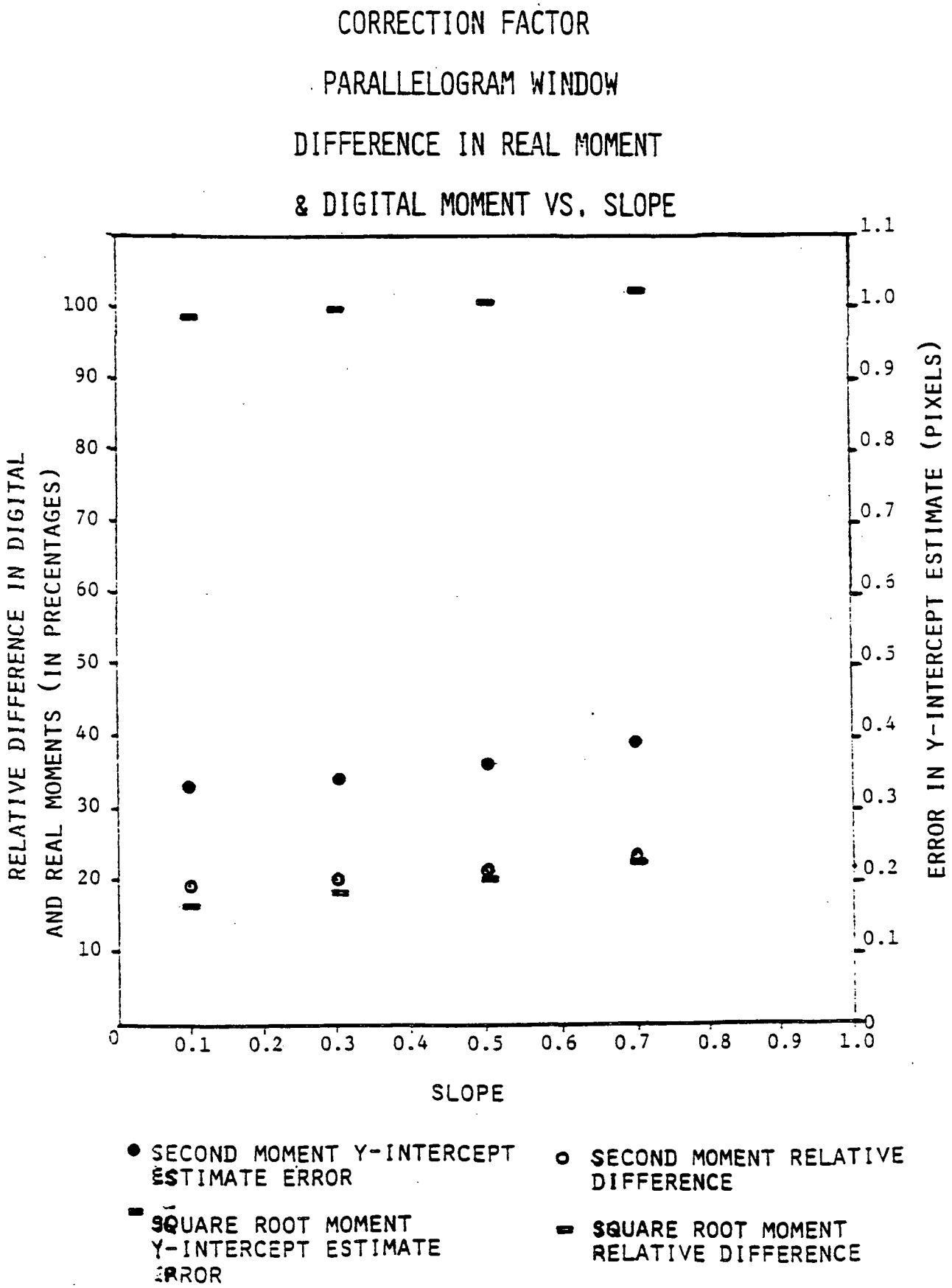

CORRECTION FACTOR
PARALLELOGRAM WINDOW
DIFFERENCE IN REAL MOMENT
& DIGITAL MOMENT VS. SLOPE
RELATIVE DIFFERENCE IN DIGITAL AND REAL MOMENTS (IN PRECENTAGES)
ERROR IN Y-INTERCEPT ESTIMATE (PIXELS)
SLOPE
0
10
20
30
40
50
60
70
80
90
100
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1.0
1.1
SECOND MOMENT Y-INTERCEPT ESTIMATE ERROR
SQUARE ROOT MOMENT Y-INTERCEPT ESTIMATE ERROR
SECOND MOMENT RELATIVE DIFFERENCE
SQUARE ROOT MOMENT RELATIVE DIFFERENCE

Figure 2.

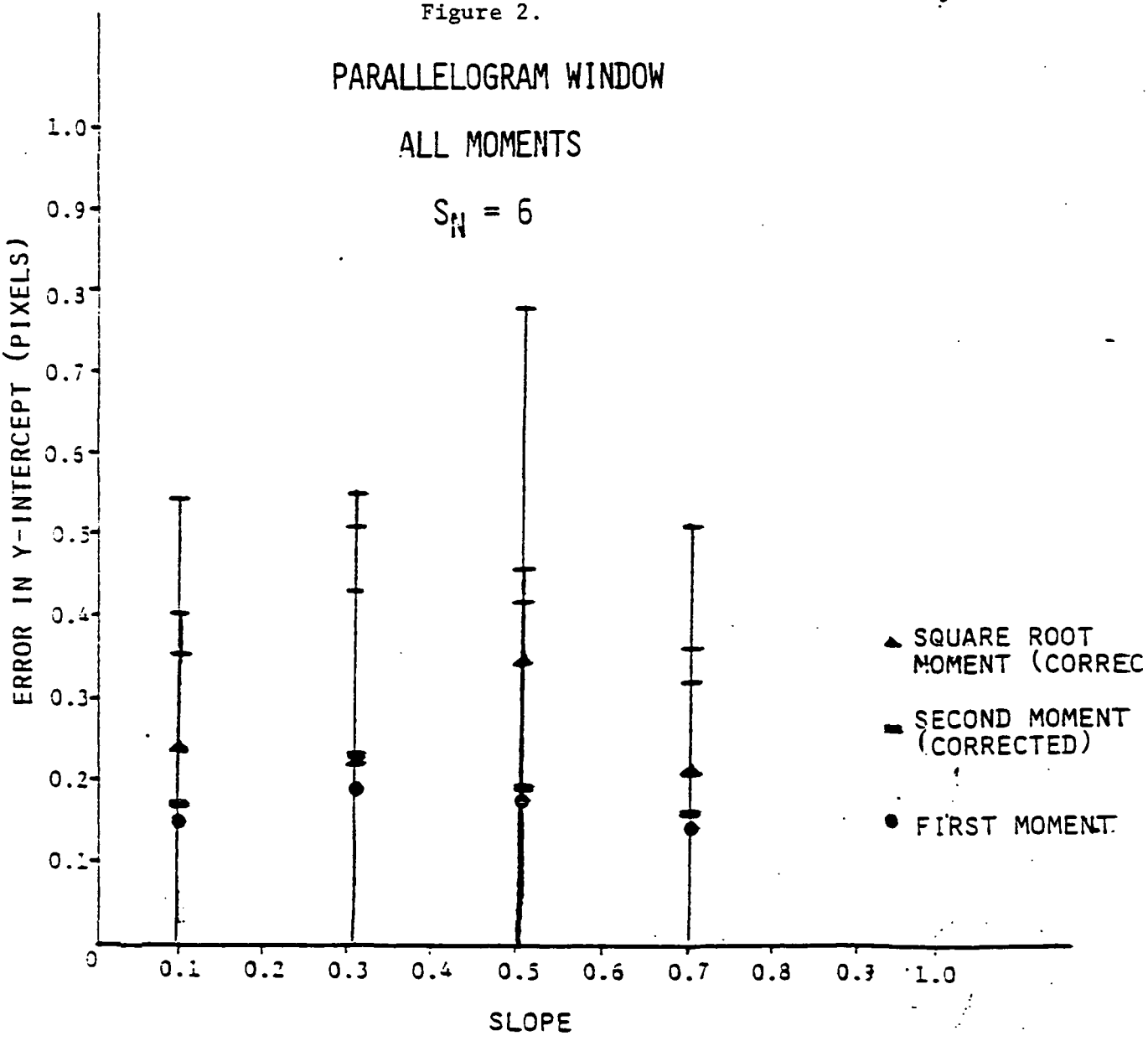

PARALLELOGRAM WINDOW
ALL MOMENTS
$S_N = 6$
ERROR IN Y-INTERCEPT (PIXELS)
1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1.0
SLOPE
SQUARE ROOT MOMENT (CORREC
SECOND MOMENT (CORRECTED)
FIRST MOMENT

Figure 3.

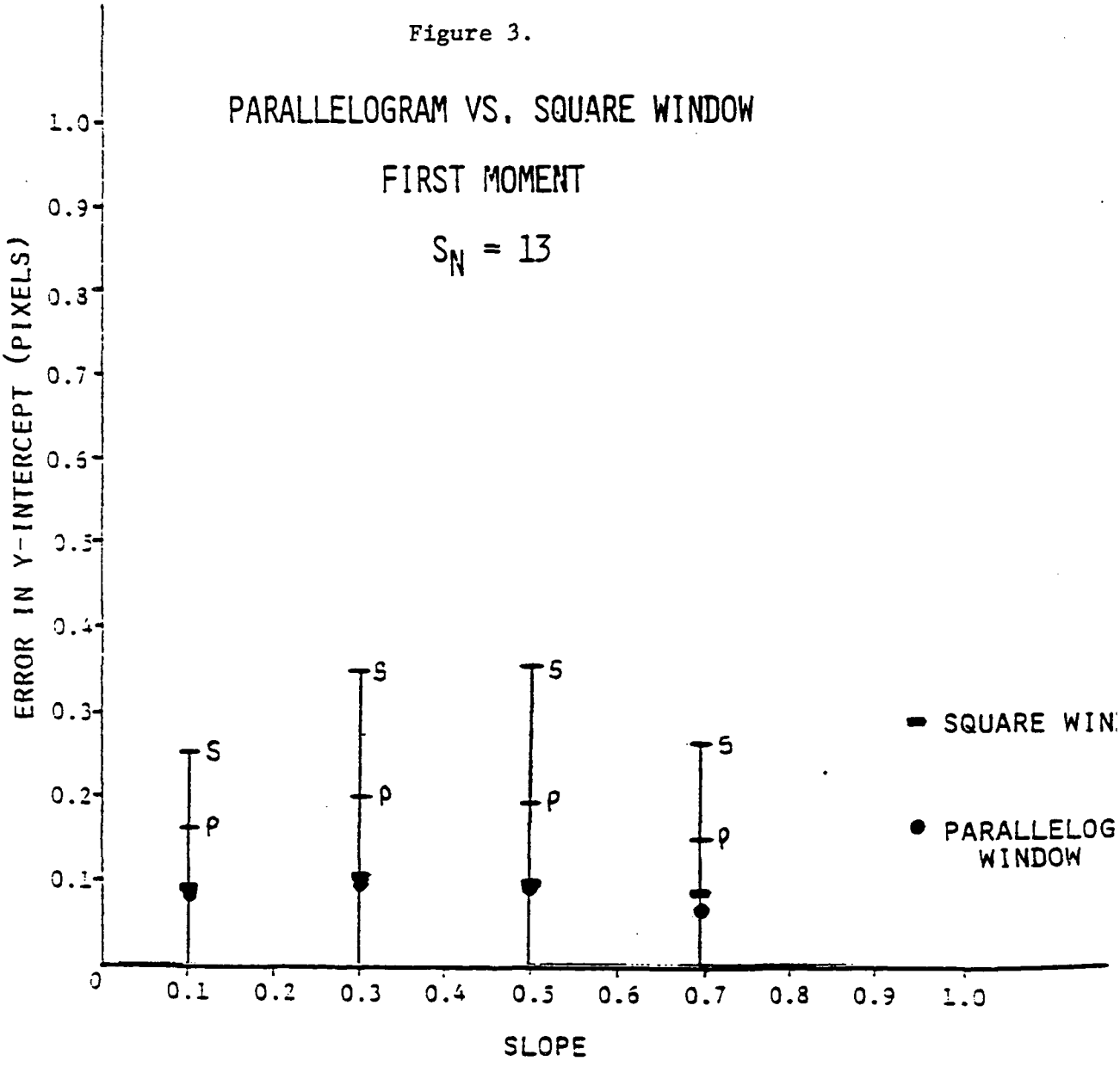

PARALLELOGRAM VS. SQUARE WINDOW
FIRST MOMENT
S_N = 13
ERROR IN Y-INTERCEPT (PIXELS)
1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
S
P
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1.0
SLOPE
SQUARE WIN
PARALLELOG
WINDOW

Figure 4.

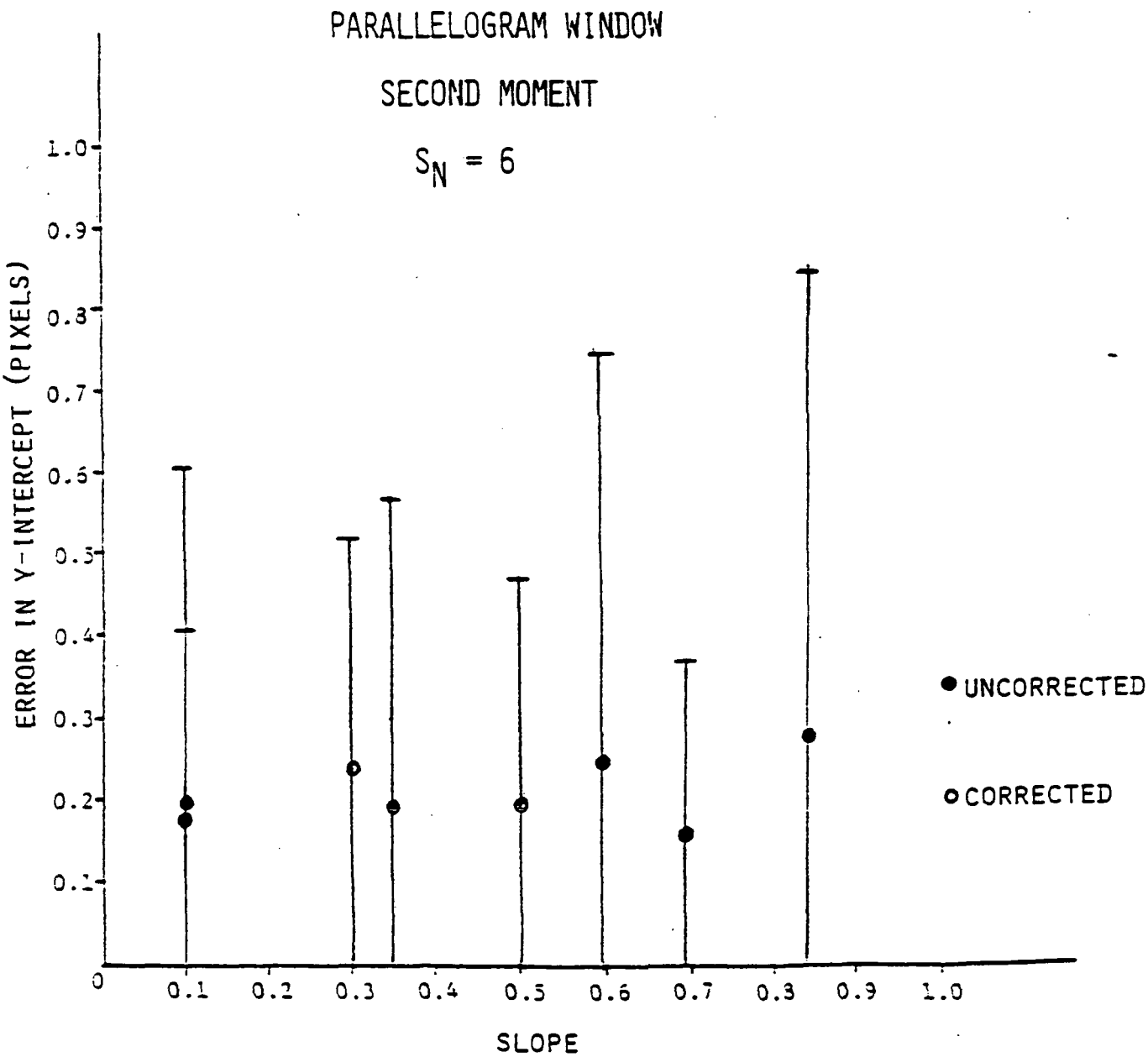

PARALLELOGRAM WINDOW
SECOND MOMENT
$S_N = 6$
ERROR IN Y-INTERCEPT (PIXELS)
1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1.0
SLOPE
UNCORRECTED
CORRECTED

Figure 5.

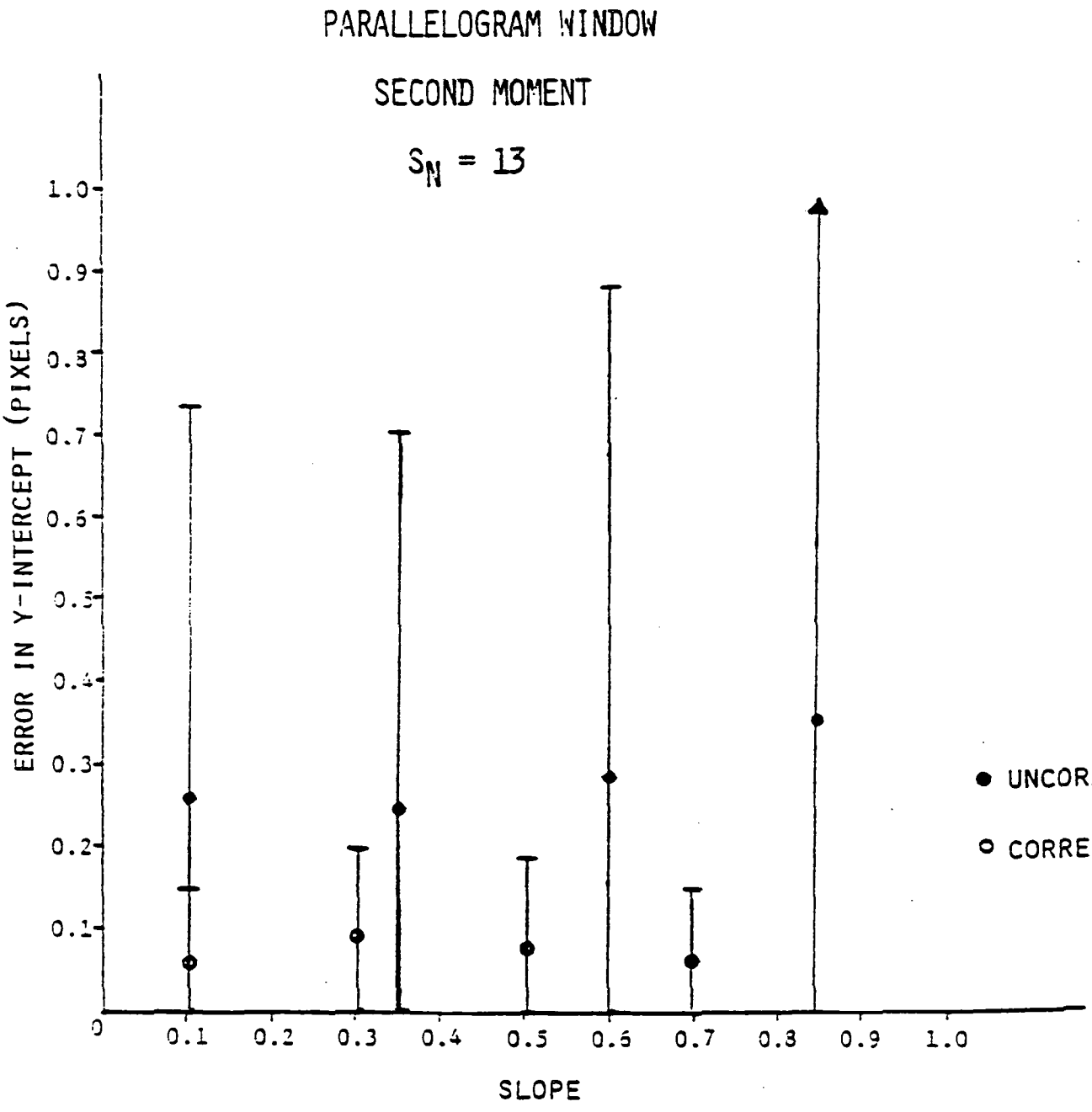

PARALLELOGRAM WINDOW
SECOND MOMENT
S_N = 13
ERROR IN Y-INTERCEPT (PIXELS)
1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1.0
SLOPE
UNCOR
CORRE

Figure 6.

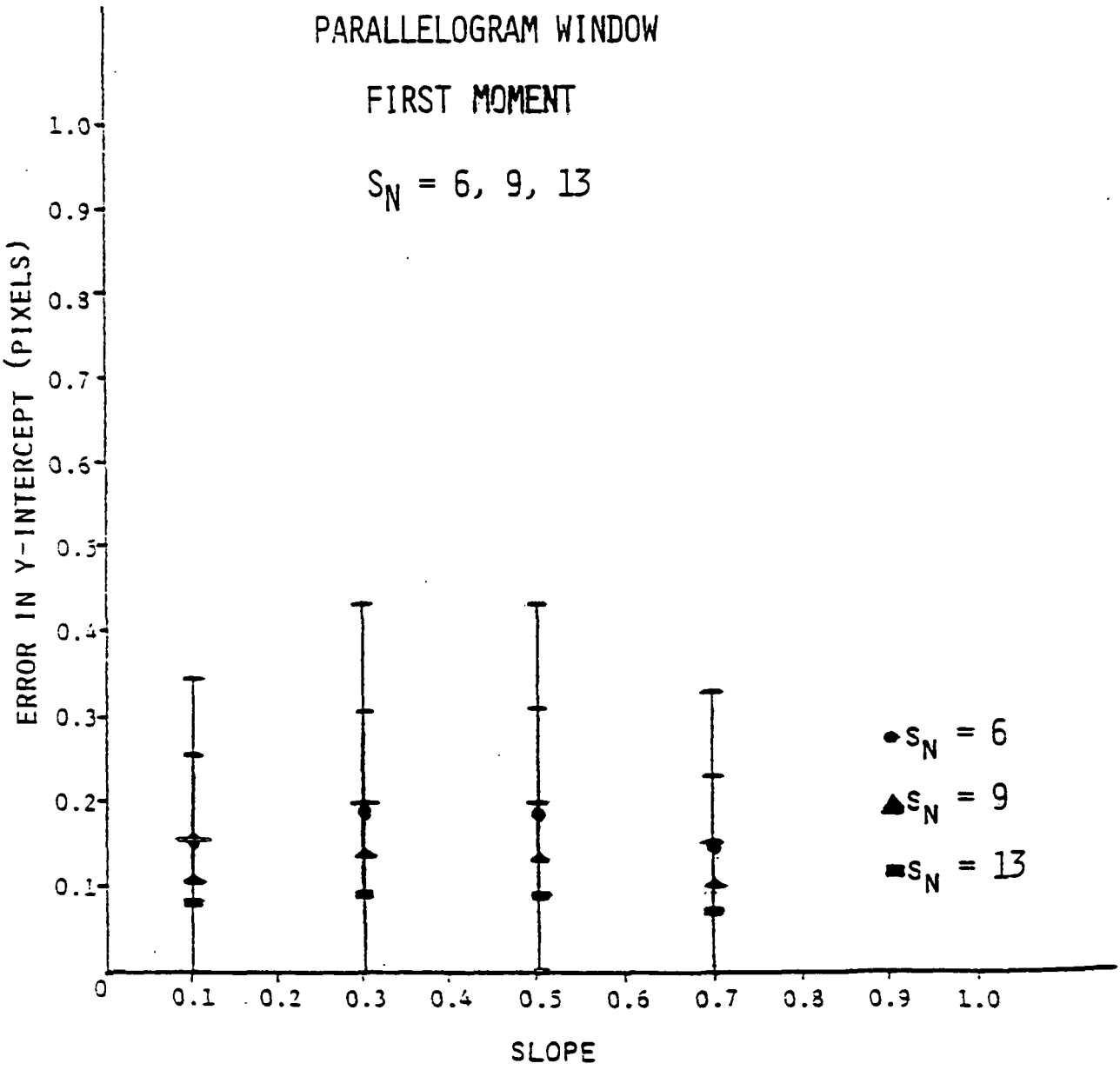

PARALLELOGRAM WINDOW
FIRST MOMENT
S_N = 6, 9, 13
ERROR IN Y-INTERCEPT (PIXELS)
1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1.0
SLOPE
S_N = 6
S_N = 9
S_N = 13

Figure 7.

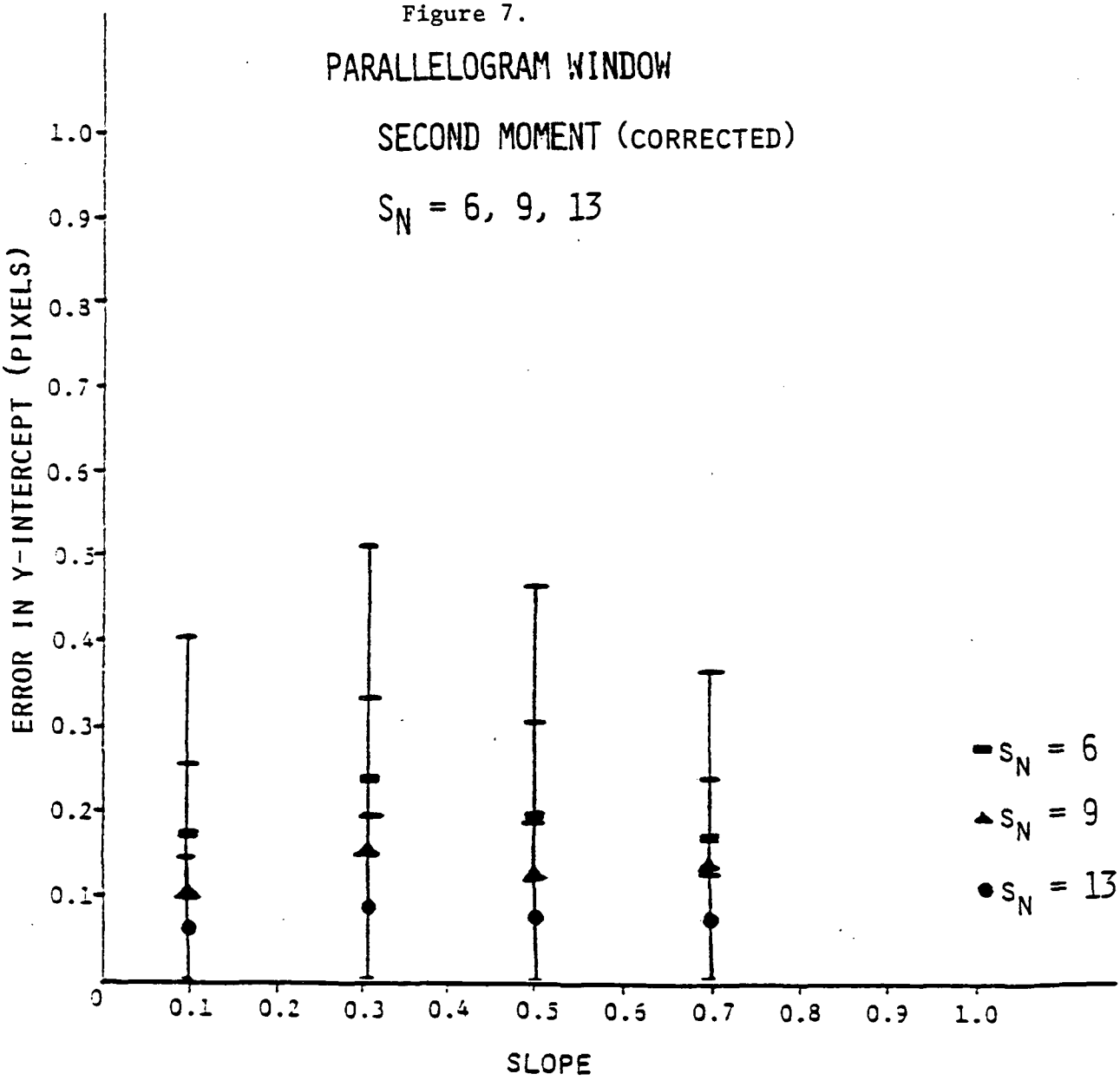
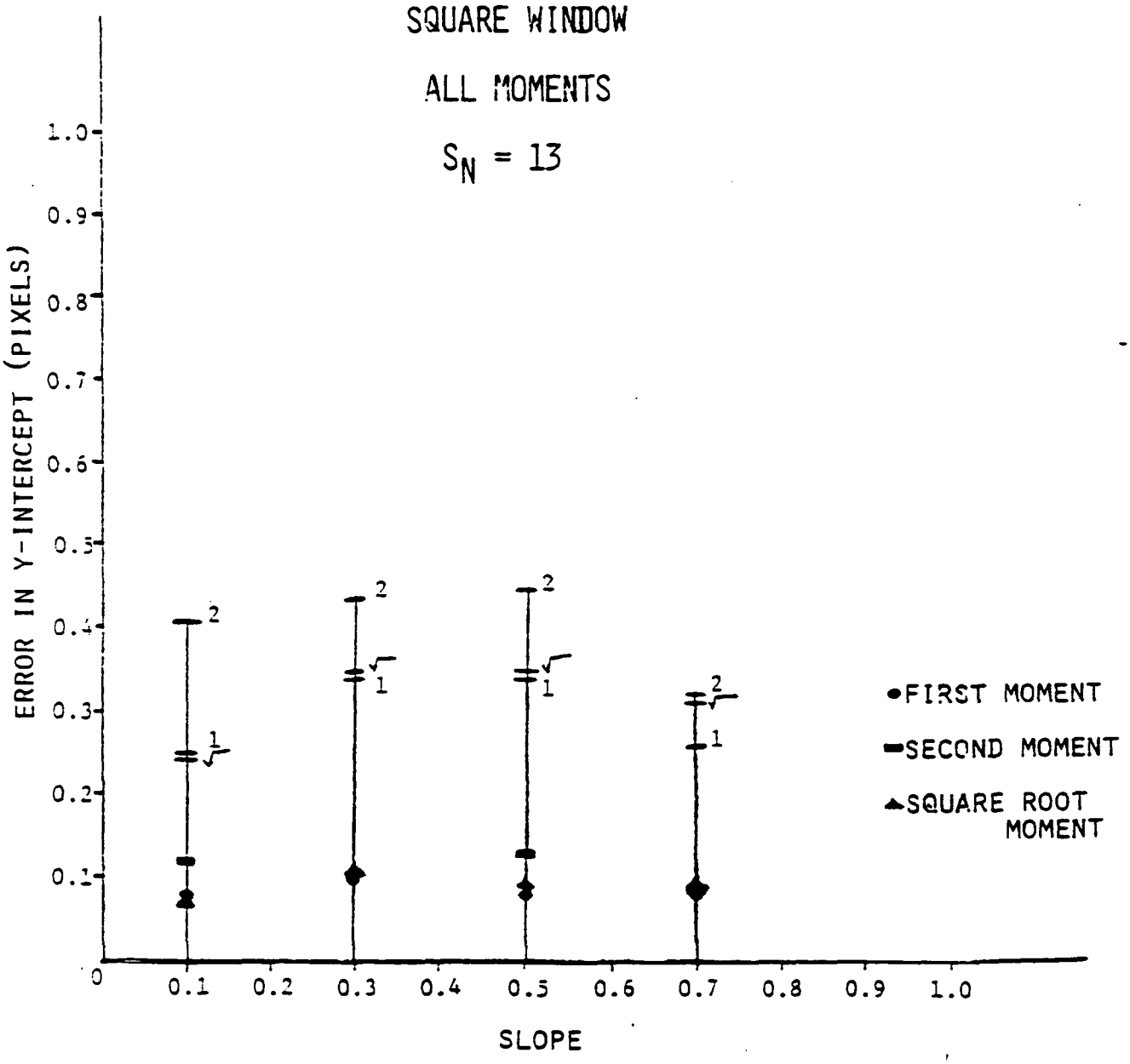

PARALLELOGRAM WINDOW
SECOND MOMENT (CORRECTED)
S_N = 6, 9, 13
ERROR IN Y-INTERCEPT (PIXELS)
1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1.0
SLOPE
S_N = 6
S_N = 9
S_N = 13

Figure 8.


SQUARE WINDOW
ALL MOMENTS
S_N = 13
ERROR IN Y-INTERCEPT (PIXELS)
1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1.0
SLOPE
FIRST MOMENT
SECOND MOMENT
SQUARE ROOT MOMENT

Section 8.

SUMMARY AND CONCLUSIONS

This section summarizes the work done in the three year study of subpixel accuracy. We note which parts of this work were done in the third year of the project. The fundamental question addressed in this work was that of understanding the problem of achieving subpixel accuracy in image registration. At the time we began our study, several algorithms for achieving subpixel accuracy had been implemented and tested for use with Landsat imagery. Ground truth of sufficient accuracy to test the claims made for the algorithms was often not available. Our study was motivated by the lack of theoretical tools for approaching the analysis of subpixel accuracy.

Two main classes of approaches were pursued in our study, edge-based techniques and correlation-based techniques. The primary focus in the edge-based techniques was on achieving subpixel accuracy in edge detection. A match between edges in a sensed image and a high resolution control chip representing the scene could then be used to estimate a registration transformation.

Several classes of subpixel edge detection procedures were explored. The first problem studied was the estimation of the position of an edge from a set of pixels forming a digital line. Many edge detection procedures are only concerning with extracting the set of pixels which constitute an edge, and not with the problem of determining a subpixel edge. Since registration algorithms capable of registering a Landsat image to within approximately one pixel were considered reliable and since rotational uncertainty was a minor problem, we assumed that the subpixel edge detector would know the position of the edge to within a pixel and that the slope of the edge was known.

Given the digitization of an edge, we looked for a real edge position

which was, in some sense, most central among the real edges which could give rise to that digitization. Using work of [Do-Sm] which characterized the set of all lines having a given digitization, we were able to derive an upper bound on the positional error estimate of the edge as a function of the parameters of the digital line. By using the unique translation and rotation invariant probability measure on the set of real lines, we were able to determine upper and lower bounds on the expected worst error in edge location estimation. The worst error refers to the upper bound for the location estimate given a single digital line. The expected value is then over the set of all digital lines.

The tightness of the bounds on the expected worst case error were difficult to estimate. For any particular edge length, the full probability distribution of the worst case error could be computed. In [Be et. al.], this computation was done for an edge length of ten. In that computation it was shown that the probability that the maximum error exceeded 0.25 pixels was only 0.0147.

An asymptotic error formula for the expected worst case error was conjectured in the second year of the project. The primary difficulty faced in proving the conjecture was the lack of an asymptotic formula for the number of digital lines of specified length. An exact formula for the number of digital lines of specified length was developed during the second year, but the formula was unwieldy. In the third year of the project, an asymptotic formula for the digital line count was developed. This result was then used to prove our conjecture on the asymptotic worst case expected error. The asymptotic expected worst case error was shown to be $0.92/N + O(\log N/N^2)$.

In the second year of the study, we explored means of using grey levels to gain a better edge position estimate than might be feasible with strictly

geometric information. In particular, we were interested in exploring the effect of noise in the estimation problem. A search procedure was developed to estimate the y-intercept of an edge. The search procedure employeed hill-climbing to evaluate the quality of an edge location estimate. An estimated edge position was used to generate a digital image which was then compared with the observed image. While this approach achieved a high level of estimation accuracy, it was time consuming and we were unable to develop any theoretical understanding of its performance.

A paper [Ta-Mi] appeared soon after the completion of the second year work, which developed a new approach to the extraction of edges to subpixel accuracy. This approach compares the observed digital moments of a circular window in an image with the corresponding moments in a continuous image containing an ideal edge with constant grey levels on the two sides of the edge. This is used to estimate the relative areas on the two sides of the edge in the observed image and ultimately to estimate the edge position. The above approach to edge detection did not make use of the power of the particular assumptions we have made in the present study. In particular, we assume that the edge orientation is known and the edge position is known to within a pixel. These assumptions led us to develop an algorithm in which we assume the grey levels on the two sides of the edge have been estimated prior to the subpixel edge detection process. This enabled us to estimate the areas below and above the edge from a single moment.

The effect on edge location accuracy of using different moments was studied. The first moment produced better results than either the 2nd moment or the 1/2 moment. It can easily be seen that the digital and the real moments are usually different except for the first moment. Empirical correction terms for this discrepancy were computed and resulted in a dramatic

increase in the accuracy of the y-intercept estimate, though the first moment performed best.

Two types of y-intercept estimation procedures were studied. In one, all pixels in an nxn window were used in estimating the edge location. While this approach uses pixels which are known not to be relevant to the problem, it is computationally simpler than the other approach studied. The second approach used a parallelogram with two sides parallel to the edge of interest. This approach necessitates computing the pixels which are intersected by the edges of the parallelogram and finding the areas on the sides of this intersection. By making the parallelogram narrow, it is possible to avoid using noisy grey levels from pixels which are not relevant to the edge location estimation problem. The two approaches produced similar mean estimation errors but the parallelogram approach resulted in a significantly smaller variance.

Since the 1st moment approach yields the exact y-intercept in the absence of noise, it was clear that some level of subpixel accuracy would be attainable even in the presence of noise. With a signal-to-noise ratio of six, the average error in the y-intercept estimate was less than 0.2 pixels.

The parallelogram and square approaches were analyzed used a Gaussian noise model. The y-intercept estimator was shown to be unbiased and normally distributed. The variance was computed in terms of the window width, the signal-to-noise ratio, and the areas of intersection between the pixels and the parallelogram.

A correlation approach to subpixel accuracy was analyzed in the study. An estimate for determining the error in using the peak of the cross-correlation between sensed and reference images as an estimate of the offset was developed. Simulations were used to determine the reliability of the error estimate and to determine the errors resulting from interpolation of

the correlation function to locate a subpixel peak. The level of subpixel accuracy as a function of the signal noise was analyzed using simulations.

Several approaches to the analysis of subpixel accuracy in registration were studied in this project. Theoretical predictions of subpixel accuracy using various models and simulation results were obtained. New results in probabilistic and enumerative problems in digital geometry were obtained in the process of developing error estimates.

REFERENCES

[Be et. al] Berenstein, C.A., Kanal, L.N., Lavine., Olson, E.C., Slud, E., Analysis of Subpixel Registration, in Proceedings of the Second Annual NASA Symposium on Mathematical Pattern Recognition and Image Analysis, Houston, TX, 1984, 489-593.

[Do-Sm] Dorst, L. and Smeulders, A.W.M., The estimation of parameters of digital straight line segments, Proceedings, 6th International Conference on Pattern Recognition (Munich 1982) 601-603.

[Ha-Wr] Hardy, G.H. and Wright, E.M., An Introduction to the Theory of Numbers, (Oxford at the Clarendon Press, 1971).

[Hy-Da] Hyde, P.D. and Davis, L.S., Subpixel edge estimation, Pattern Recognition, Vol. 16, No. 4 (1983) 413-420.

[La] Lavine, D., Kanal, L.N., Berenstein, C.A., Slud, E., Herman, C., Analysis of subpixel registration accuracy, in Proceedings of the NASA Sumposium on Mathematical Pattern Recognition and Analysis, Houston, Texas, June 1983, 327-412.

[Ro-We] Rothstein, J. and Weiman, C. Parallel and sequential specification of a context sensitive language for straight lines on grids, Computer Graphics and Image Processing 5 (1976) 106-124.

[S] Santalo, L.A. Integral Geometry and Geometric Probability,

Addison-Wesley, 1976.

[Ta-Mi] Tabatabai, A.J., Mitchell, O.R., Edge Location to Subpixel Values in Digital Imagery, IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI-6, No. 3, 1984, 188-201.

[W-R] Weiman, C. and Rothstein, J., Pattern recognition by retina-like devices, Technical Report No. OSU-CISRC-TR-72-8, Department of Computer and Information Science, Ohio State University, 1972.

Subpixel Translation-Registration of Random Fields

by Eric V. Slud
University of Maryland and L.N.K. Corp.

INTRODUCTION

Consider the problem of registering (i.e., finding an appropriate overlay by relative translation of) a sensed planar image with respect to a larger reference image supposed to contain it. In typical remote-sensing applications, both the sensed and reference images will be given, at the same resolution, as arrays of gray-level values, one value for each pixel. Both images will typically be noisy, due to minor changes in weather or ground features; to sensor characteristics; to preprocessing and detrending; and possibly also to nonlinear filtering of gray-level images, for example by edge-enhancers and thresholding.

The primary model assumptions for our discussion of this problem are:

(a) there exists underlying continuous sensed and reference images $Z_S(x)$ and $Z_R(x)$ *before* discretization into pixels, where $x=(x_1,x_2)$ are planar coordinates, such that $Z_R(.)$ and $Z_S(.)$ are jointly strictly stationary random fields (i.e., have translation-invariant statistics) with rapidly decaying dependence between the fields $(Z_R(x+y),\ Z_S(x+y))$ and $(Z_R(y),\ Z_S(y))$ as a function of $||x|| = (x_1^2+x_2^2)^{1/2}$ (see [2] for precise conditions and definitions: Z_R and Z_S must be ϕ-mixing with $\sum_{r=1}^{\infty} r\,\phi^{1/2}(r) < \infty$);

(b) there exists an unknown translation-parameter $\theta=(\theta_1,\theta_2)$, a known pixel width h, and a known kernel-function $K(.,.)$ such that the observed sensed and

reference gray-level arrays are

$$X_S(j,k) = h^{-2}\int_0^h\int_0^h K(s,t)Z_S(jh+\theta_1+s,\ kh+\theta_2+t)\,ds\,dt$$

$$X_R(j,k) = h^{-2}\int_0^h\int_0^h K(s,t)Z_R(jh+s,\ kh+t)\,ds\,dt$$

The interpretation of assumption (a) is as follows: we think of $Z_N(.)=Z_S(.)-Z_R(.)$ as the random noise-field superposed additively on the reference image to give the sensed image; to begin with, we assume that Z_R and Z_S (or equivalently, Z_R and Z_N) have jointly translation-invariant statistics, but we will find below that this requirement can be relaxed considerably as long as Z_N has translation-invariant statistics; in addition, it is important that dependence in Z_N dies off quickly as points become widely separated. We interpret (b) as describing the mechanism by which our analog sensed images Z_R, Z_S are discretized into pixels. In particular, since the coordinate-offset θ is the same throughout the reference and sensed images, there is considerable redundancy in the observable discretized images X_R, X_S for estimation of θ. There is therefore some hope of estimating θ from large images X_R and X_S to an accuracy better than 1 pixel. One of the main objects of this paper is to address this possibility quantitatively.

The fields Z_R and Z_S are of course assumed to be highly correlated images representing the same ground truth, and for identifiability of location it is quite important that the correlation between $Z_R(x)$ and $Z_S(x+y)$ be small except for y close to 0. The parameter θ is then identifiable in principle from large images

$(Z_R(x))_{|x_1|,|x_2|\leq Mh}$ and $(Z_S(y+\theta))_{|y_1|,|y_2|\leq Lh}$. To see whether and to what extent θ remains identifiable from pixel data $\{X_R(j,k)\colon |j|,|k| \leq M\}$ and $\{X_S(j,k)\colon |j|,|k| \leq L\}$ is precisely our problem. Note that the kernel function K models the *linear* transformation of a pixel image to a gray level. For simplicity (although all our results can be extended to general known K), and in apparent agreement with previous researchers, we assume in what follows that $K(s,t)=1$.

Our model assumptions are in some respects similar to, but substantially generalize, those of Mostafavi and Smith [5] (who were, however, interested also in the effects of affine distortion). In addition to (a), [5] assumed that $Z_R(.)$ and $Z_S(. + \theta)$ are directly observable and jointly Gaussian. This restrictive assumption is not necessary for an understanding of the asymptotic distribution theory, for large sensed images, of the maximum-correlation estimator θ^* for θ (see below). Moreover, Mostafavi and Smith do not take into account the transformation of Z_R, Z_S which renders only X_R, X_S directly observable. Thus their analysis, which we extend and improve in Section 1 of this report, only partially establishes the consistent maximum-correlation estimation of θ. By contrast, we derive bounds for each τ on the probability of mis-estimating θ (by the maximum-correlation method) by as much as τ pixels. We thereby justify what we call "neighborhood consistency" of registration for large sensed and reference images. In Section 2 we test the validity and stringency of our theoretical bounds via simulations of noise fields superimposed on real and on artificial reference images. Finally, we summarize and interpret our results in Section 3.

This research has been supported over several years by L.N.K. Corporation under NASA Contract and is an outgrowth of reports ([6], [7]) submitted to NASA. The author is grateful for many suggestions and comments to David Lavine and Dr. Laveen Kanal of L.N.K. Corporation.

1. Neighborhood-Consistency of Maximum-Correlation Estimation

The reason that we do not need to assume Gaussian distributions for gray-levels is simply that the fixed-offset "correlation" statistic for $Z_R(.)$, $Z_S(. + \theta)$ given by

$$(*) \qquad C(t) = (2T)^{-2} \int_{-T}^{T}\int_{-T}^{T} Z_R(x + t)Z_S(x + \theta)dx, \qquad T = Lh,$$

is asymptotically weakly convergent as a random process in $t = (s, t)$ as $L \to \infty$ to a Gaussian random field, under the precise condition of [De] on decay of dependence mentioned in (a). If $Z_R(.)$ and $Z_S(. + \theta)$ are directly observable, then a natural statistic to estimate θ is

$$\theta^* = \textit{maximizer of } C(.) \textit{ on } [-T, T]^2$$

The most easily interpreted figures of merit for this (and any other) estimator are of the form

$$Q(\tau) = P\{|\theta^* - \theta| \leq \tau\}$$

or

$$Q_{T_0}(\tau) = P[\sup\{C(x): ||x - \theta|| \leq \tau\} = \sup\{C(x): ||x|| \leq T_0\}]$$

where $||x||_\infty = \max(|x_1|, |x_2|)$ and T_0 is a fixed size of window inside which we may assume θ lies. We note that since Mostafavi and Smith [5] did not

treat $C(.)$ as a random field, they did not propose to evaluate quantities $Q_{T_0}(\tau)$ but rather to compare the asymptotically (in T) normal single-offset correlations $C(t)$ with either specified or "sidelobe" thresholds. That is, their probabilistic consideration of estimation-error depended solely on the (marginal) distributions of $C(t)$ values. On the other hand, evaluation of $Q_{T_0}(\tau)$ is clearly a problem about random processes - not simply finite-dimensional distributions - for which we now formulate an asymptotic solution, assuming (a).

Let $D(t)$ denote the expectation $EC(t)$. Joint stationarity of $Z_R(.)$ and $Z_S(. + \theta)$ implies

$$D(t) = E\{C(t)\} = E\{Z_R(t)Z_S(\theta)\}$$

which would be consistently estimated when T is large by the expression $C(t)$ in (*). (In other words, [De]'s conditions imply a law of large numbers for $C(t)$ for each t). The stationary covariance function

$$V(x - y) = Cov(C(x),C(y)) \sim T^{-2}\sigma(x - y) \quad as \quad T \to \infty$$

(which defines the asymptotic covariance $\sigma(.)$) can likewise be consistently estimated by a fourfold integral expression (cf. [5], where some simplifications occur if Z_R and Z_S are jointly Gaussian). The following result, the proof of which is sketched in the Appendix, bounds $1-Q_{T_0}(\tau)$ theoretically in terms of quantities derived from the joint distributions of Z_R and Z_S which we can hope to estimate consistently from data when T is large and (a) holds approximately.

Bound on Probability of Registration Error. Assume (a), (*) and fix $\tau>0$. For

simplicity, fix the units of measurement so that the pixel width h is 1. Assume T_0 and T are integers, and let

$$(1.1) \qquad H_\tau = \inf\{D(\theta) - D(t): \ ||t-\theta|| \geq \tau, \ ||t||_\infty \leq T_0\}.$$

Let $\Gamma > 0$, and let $\Psi(.)$ be a positive function such that $\int_1^\infty \Psi(e^{-u^2})du < \infty$ and $\Psi^2(u)\log(1/u)$ decreases as $u \downarrow 0$, and assume

$$(1.2) \qquad |C(t)-C(\theta)-D(t)+D(\theta)|/\Gamma \quad \text{and} \quad |C(s)-C(t)-D(s)+D(t)|/\Psi(||t-s||)$$

each have distribution functions $\leq (2/\pi)^{1/2}\int_0^x e^{-u^2}du$ for $||s||_\infty, ||t||_\infty \leq T_0$

Then

$$(B) \qquad 1-Q_{T_0}(\tau) \leq (\frac{2}{\pi})^{1/2}\{(8T_0+1)^2 + 379.2T_0^2\}\int_x^\infty e^{-u^2/2}du$$

whenever

$$x = H_\tau/[\frac{2}{\sqrt{2}-1}\int_1^\infty \Psi(2^{-u^2})du + \Gamma] \ \ is \ \geq 2.36$$

In this result, (1.2) holds automatically if $C(.)$ is Gaussian and

$$(1.3) \qquad \begin{cases} \Gamma^2 = \sup\{Var(C(t) - C(\theta)) : ||t|| \leq T_0, \ ||t-\theta|| \geq \tau\} \\ \Psi^2(u) = \sup\{Var(C(t) - C(s)) : ||s-t|| \leq u, ||s-\theta|| \geq \tau, \ ||t-\theta|| \geq ta||\} \end{cases}$$

The approximate joint Gaussian distributions of $C(.)$ for large T followed from the ϕ-mixing Central Limit Theorem of [2], and some variants of that Theorem do not require strict stationarity (of Z_R, Z_S) but only rapidly decaying dependence with marginal distributions (of $Z_R(t), Z_S(t)$) not varying too rapidly with t. Therefore we can expect, for moderately large T and realistic reference images

Z_R with only approximately translation-invariant statistics, and for $Z_N(.)$ stationary and approximately independent of $Z_R(.)$, that the foregoing bounds on error-probabilities should remain approximately valid. It will be the task of our next Section to test the stringency and validity of (B) for realistic and artificial examples by Monte Carlo simulation.

2. Simulation Study of Registration Error-Probabilities

In this Section, we describe the purpose, design, and numerical results of a Monte Carlo simulation study of maximum-correlation translation-registration of some realistic and some artificial random fields sensed with a fixed offset and independent stationary noise. The general objectives of the study were

(a) to compile empirical distributions for Euclidean distances and $||\hat{\theta}-\theta||$ and $||\theta^{LS}-\theta||$ under various conditions, where $\hat{\theta}$ denotes the pixel-vertex where $C(.)$ is largest, and where θ^{LS} denotes the location t of the maximum for the least-squares quadric surface approximating $C(x)$ at the nine points (j,k) with

$$j \text{ and } k = -1,0,1;$$

(b) to compare the performance of $\hat{\theta}$ and θ^{LS} with a view to examining the feasibility of subpixel registration;

(c) to gain information on how large the standard deviation of additive noise must be compared to gray-level standard deviation in various reference images before pixel-level and subpixel registration (estimation of θ) is seriously degraded;

(d) to check the validity and usefulness of the theoretical results of Section 1 for 35 by 35 reference images, window size $T=L=10$, and $T_0=5$, where the pixel

size h is 1.

Design of the Study

We specify now exactly what was computed in our study. To begin, we fixed six reference images Z_R, each on the 35 by 35 grid of pixel vertices $\{(j,k) : \max(|j|,|k|) \leq 17\}$. The first three were artificially constructed:

for *image* 1, $Z_R(j,k) = 55.0 - 1.5 * (|j| + |k|)$, $j,k = -17,-16,\ldots,+17$;

for *image* 2, $Z_R(j,k) = \begin{cases} 0 & \text{if } \max(|j|,|k|) \geq 3 \\ 40. & \text{if } \max(|j|,|k|) \leq 2 \end{cases}$;

for *image* 3, $Z_R(j,k) = \begin{cases} 20. & \text{if } \max(j,k) \leq 0 \\ 10. & \text{if } \max(j,k) > 0 \end{cases}$

The remaining three (numbered 4, 5, and 6) were real 35 by 35 gray-level arrays chosen more or less arbitrarily from an 80 by 125 LANDSAT image of a rural (United States) scene including cultivated fields, some wooded areas, and some roads. Before further processing, each of the six reference arrays was centered and scaled to have average value 0 and $\sum_{j=-17}^{17} \sum_{k=-17}^{17} Z_R^2(j,k) = 1$.

Some further assumption was of course required to define the continuous variation of Z_R (and similarly, of Z_S or $Z_N = Z_S - Z_R$) within pixels. For a point $t = (t_1, t_2)$ in the plane, we define $[t] = ([t_1],[t_2])$ and $\{t\} = t - [t]$ where $[x]$ is the greatest-integer function of x. Also let $e_1 = (1,0)$, $e_2 = (0,1)$, and $e = (1,1)$. Consider the following two model-assumptions for a random field Z: for $t = (t_1, t_2) \in R^2$,

$$(M1) \quad Z(t) = (1-\{t_1\})(1-\{t_2\})Z([t]) + (1-\{t_1\})\{t_2\}Z([t]+e_2) +$$

$$\{t_1\}(1-\{t_2\})Z([t]+e_1)+\{t_1\}\{t_2\}Z([t]+e)$$

or

$$(M2) \qquad Z(t) = Z([t]).$$

Assumption $(M1)$ would mean that Z at a point t interior to a given pixel J takes a value which is a weighted average of the values at the corners of J with weights proportional to the area of overlap of a unit square with lower-left corner t with squares whose lower-left corners are the four corners of J. Assumption $(M2)$ would mean that the field Z is homogeneous within each pixel $[j,j+1)\times[k,k+1)$. For the purpose of our study, we took $Z_N = Z_S - Z_R$ always to satisfy $(M1)$, with Z_R satisfying $(M1)$ in Study 1 described below and satisfying $(M2)$ in Study 2.

It remains to tell how the offset θ and the noise-process Z_N at lattice points were generated. On each iteration of each simulation, $Z_N(t)$ was defined for lattice-points t with $||t||_\infty \leq 17$ by

$$(2.1) \qquad Z_N(t) = \sum_{j=-1}^{1} \sum_{k=-1}^{1} \xi_{t_1+j,t_2+k}\, W(j,k)$$

where $\{\xi_{l,m}\}$ was a simulated array of independent identically normally distributed random deviates with mean 0 and variance σ^2 (another design-parameter in the study), and the $W(j,k)$ were fixed weights which took the form

$$W_1 = \begin{pmatrix} 1/36 & 1/9 & 1/36 \\ 1/9 & 1/4 & 1/9 \\ 1/36 & 1/9 & 1/36 \end{pmatrix} \quad \textit{in Study 1 where } (M.1) \textit{ was assumed for } Z_R$$

$$W_2 = \begin{pmatrix} 0 & 1/4 & 1/4 \\ 0 & 1/4 & 1/4 \\ 0 & 0 & 0 \end{pmatrix} \quad \textit{in Study 2 where } Z_R \textit{ satisfied } (M.2)$$

The offset-vector θ for each simulation-iteration was generated uniformly in $[0,1]^2$.

The correlation-statistic $C(.)$ was computed, for each lattice-point in the square $[-5,5]^2$, as follows. First, the expectation-term $D(t)$ was calculated as a sum rather than the integral in its definition from Section 1:

$$(2.2) \qquad D(t) = \frac{1}{(21)^2} \sum_{j=-10}^{10} \sum_{k=-10}^{10} Z_R(j,k) Z_R((j,k)+\theta-t)$$

This modification was made for two reasons: (1) although the integral could, under either assumption $(M1)$ or $(M2)$, be expressed as a weighted sum of terms $Z_R(x), Z_R(y)$, the weights would depend on θ, and it was computationally much easier to make use of the equally plausible definition (2.2); (2) in actual practice, in the absence of a validated model assumption like $(M1)$ or $(M2)$, (2.2) is the definition one would use, with sums similarly replacing integrals in the definition of $C(.)$. Then $C(t) - D(t)$ was calculated as

$$(2.3) \qquad C(t) - D(t) = \frac{1}{(21)^2} \sum_{j=-10}^{10} \sum_{k=-10}^{10} Z_R(j,k) Z_N((j,k) - t).$$

In this definition we have replaced $(4T^2)^{-1}$ for $T=10$ by $(21)^{-2}$ and modified some boundary terms, but (2.3) is otherwise the same as in its double-integral definition if $Z_N(.)$ had been made up of independent $N(0,\sigma^2)$ variables at lattice points and had been interpolated according to $(M.1)$ while Z_R was interpolated according to $(M1)$ or $(M2)$. [For example, under $(M.1)$ for both Z_R and Z_N,

$$\frac{1}{4T^2}\int\int Z_R(x) Z_N(x-t)\,dx \approx \frac{1}{(2\lfloor T/h \rfloor+1)^2} \sum_i Z_r(i)\{\frac{4}{9} Z_N(i-t)+$$

$$\frac{1}{9}(Z_N(i-t+e_1)+Z_N(i-t-e_1)+Z_N(i-t+e_2)+Z_N(i-t-e_2))+$$

$$\frac{1}{36}(Z_N(i-t+e)+Z_N(i-t-e)+Z_N(i-t+e_1-e_2)+Z_N(i-t+e_2-e_1))\}].$$

Results of the Study

Two simulation experiments were performed on the DEC 2060 at Cornell University, Study 1 with 450 iterations using weight-matrix W_1 and Study 2 with 250 iterations using weights W_2. For each iteration, one offset θ and one array $\{\xi_{jk}\}$ was generated for each of six reference images, and $D(t)$ and $C(t)-D(t)$ were calculated according to (2.2) and (2.3) with $\sigma=1$. Then for each of a number of different values of σ, the arrays $\{D(t)+\sigma\,(C(t)-D(t))\}_{t\,:\,t_1,t_2=-5,\ldots,+5}$ (correlation-statistic arrays corresponding to the noise-fields $\sigma\,Z_N(.)$ generated from the same random numbers) were used to calculate estimators $\hat{\theta}$ (the lattice-point t corresponding to the largest array element) and θ^{LS} (the maximum-point (x,y) for the least-squares quadric surface for the nine correlation-array values at $\hat{\theta}+(j,k)$, $j,k=-1,0,1$). In addition, a third estimator was defined as

$$\bar{\theta} = \hat{\theta} + .5*(sign\ (\theta_1^{LS}-\hat{\theta}_1),\ sign\ (\theta_2^{LS}-\hat{\theta}_2)),$$

in order to check whether any possible increase in accuracy of θ^{LS} over $\hat{\theta}$ might simply be ascribed to allowing θ^{LS} to take values in the interiors of pixels. For each reference image and each of seven values of σ, the empirical distribution functions $\hat{F}$ of $||\hat{\theta}-\theta||$, F^{LS} of $||\theta^{LS}-\theta||$, and $\bar{F}$ of $||\bar{\theta}-\theta||$ were tabulated, at intervals of 0.1 in Study 1 and of 0.125 in Study 2. (The empirical distribution function of a simulated quantity Q at the point x is simply the relative frequency with which the value Q is $\leq x$). For selected values of σ, the empirical distribution functions $\hat{F}$ and F^{LS} are displayed in Figure 1. In Table I

are exhibited, for selected σ and all six reference images in Study 1, the empirical upper-quartile values for the distances $||\hat{\theta} - \theta||$, $||\theta^{LS} - \theta||$, and $||\bar{\theta} - \theta||$ (that is, the smallest values x for which the respective empirical distribution function values exceeded 0.75), obtained by linearly interpolating the empirical distribution functions from Study 1. Further tabulation of the empirical distributions in Studies 1 and 2 is omitted because of the similarity of the results to Figure 1 and Table I.

3. Discussion and Interpretation of Results.

The results of our simulation experiments are summarized roughly in Table I, in which we remark:

(1) for all six reference images (but especially for the real images, numbers 4-6, and the smaller values of σ), the least-squares estimator θ^{LS} gives a noticeable improvement in accuracy over $\hat{\theta}$ in estimating θ; for all the images except number 1, the artificial estimator $\bar{\theta}$ (which is an attempt to bridge the gap between $\hat{\theta}$ and θ^{LS} by shifting $\hat{\theta}$ to the center nearest θ^{LS} of a pixel with vertex $\hat{\theta}$) is markedly worse than both $\hat{\theta}$ and θ^{LS}; thus, for the types of moving-average Gaussian noise fields studied, the subpixel improvement of $\hat{\theta}$ by θ^{LS} makes θ^{LS} the estimator of choice for θ (in the absence of more detailed geometric information about Z_R);

(2) Images 1 and 3 (both artificial, with strong geometric structure, and quite nonstationary) show very little advantage for θ^{LS} over $\hat{\theta}$, except for the smallest value of σ, and show very rapid loss of accuracy as σ increases (e.g., the upper-quartiles in Table I for $||\theta^{LS} - \theta||$ are larger for Images 1 and 3 than for the

other

images, with σ only half as large or less);

(3) the accuracy of $\hat{\theta}$ is relatively insensitive to the noise-level parameter σ for the real reference-images (4-6), and $||\hat{\theta}-\theta||_2$ is less than 0.5 pixel, for σ between 0.4 and 1.2, roughly 75% of the time; for these images, $||\theta^{LS}-\theta||_2$ has upper-quartile ranging from .2 to .5 pixels as θ ranges from .4 to 1.2, and the advantage of θ^{LS} over $\hat{\theta}$ deteriorates as σ gets longer than 1.0.

Indeed, Figure 1 and the tabulated empirical distribution functions in Studies 1 and 2 (not presented here) strongly support conclusions (1)-(3) as well as the following generalization: for images 2 and 4-6, when $||\hat{\theta}-\theta||$ is less than about 0.6 pixel, $||\theta^{LS}-\theta||$ is (stochastically) smaller than $||\hat{\theta}-\theta||$ by 0.1 pixel or more for small σ (but this advantage is diluted by larger σ). Quite generally, for all six images, there seems to be no advantage of θ^{LS} over $\hat{\theta}$ when $||\hat{\theta}-\theta||$ is 0.9 pixel or more.

We next discuss the accuracy of the empirically estimated numbers in Figure 1 and Table I. All the distribution function values p are with approximate probability $1-\alpha$ contained in the symmetric interval of length $p(1-p)\Phi^{-1}(1-\frac{\alpha}{2})/\sqrt{n}$ around the empirically estimated values, where Φ is the standard normal distribution function and n is the number of iterations in the simulation. With $n=450$, substituting $1/2$ for p, we find the conservative $(1-\alpha)$ quantiles for each t:

$$\textit{percentage points for } |F_{est}(t)-F(t)| = \begin{cases} .019 & \text{if } \alpha=.10 \\ .023 & \text{if } \alpha=.05 \\ .026 & \text{if } \alpha=.02 \end{cases}$$

In order to take account of our having estimated values $F(t)$ by empirical estimates $F_{est}(t)$ for many t simultaneously, the Kolmogoroff- Smirnoff approximate percentage points for $n=450$ are relevant:

$$\textit{percentage points for } \sup\{F_{est}(t)-F(t):\ 0<t<\infty\} = \begin{cases} .058 & \alpha=.10 \\ .064 & \alpha=.05. \\ .077 & \alpha=.01 \end{cases}$$

Finally, in Table I we have empirically estimated upper quartiles for random variables like $||\hat{\theta}-\theta||$. Although it is hard to assess the accuracy of the linear interpolation we have used, the ordinary binomial-normal confidence interval (with $n=450$) for any t near the upper quartile of $F(.)$ (with $F(t)$ near 3/4) yields $F(t)$ with 98% probability in the range $F_{est}(t) \pm .02$. Therefore, we can ascribe extremely high confidence to the first decimal place of the upper-quartile estimates, and if $F(.)$ (e.g. the df. of $||\hat{\theta}-\theta||_2$) were approximately linear within increments of .1 for x between 0 and 1.7, we could have approximately 98% confidence that the error in upper-quartile estimates would be at most .02.

It is striking that, when the standard deviation of superposed Gaussian noise is a fixed proportion of the "sample standard deviation" $(2T+1)^{-1}(\sum_{j,k} Z_R^2(j,k))^{1/2}$, the estimation of θ is actually more accurate for the real reference images (4-6) than for the highly structured artificial images (1-3). Clearly the variability within the reference image and the sharpness of the peak in $D(t)$ at θ interact in a nontrivial way in determining the feasible subpixel accuracy of estimation of the offset θ. We can productively unify the theoretical results of Section 1 with the simulation results of Section 2 by describing the features of the reference

image which seem to govern subpixel registration accuracy. An important aspect of this unification is the comparison of the theoretical bounds (B) of Section 1 with the simulated empirical distribution function for $| |\hat{\theta} - \theta| |$.

Inequality (B) of Section 1 says that the (upper bound for the) probability that $| |\hat{\theta} - \theta| | > \tau$ depends on the statistics of the reference image only through

$$x_* = x_*(\tau) = H_\tau/(\Gamma + \frac{2}{\sqrt{2}-1}\int_1^\infty \Psi(2^{-u^2})du)$$

where H_τ, Γ, and Ψ are given by (1.1) and (1.2). In our simulation studies, where $T=10$ and $T_0=5$, for each of six reference images the quantities H_τ, Γ, Ψ, and $\Psi(1.414)$ are given in Tables II and III. Only the values of $\Psi(u)$ for $0 \leq u \leq 1/2$ are relevant in calculating $x_*(\tau)$, and for purposes of approximate calculation we treat $\Psi(.)$ as being linear on [0,1], in which case $2(\sqrt{2}-1)^{-1}\int_1^\infty \Psi(2^{-u^2})du = 1.22\ \Psi(1)$. In further calculations, we therefore estimate $x_*(\tau)$ by $H_\tau/(\Gamma+1.22\ \Psi(1))$. Now according to (B), with $T_0=5$ and $T=10$, and the inequality $\int_x^\infty e^{-t^2/2}dt \leq e^{-x^2/2}/x$,

$$(3.1) \qquad P\{| |\hat{\theta} - \theta| | > \tau\} \leq 8900\ (e^{-(x_*(\tau))^2/2}/x_*(\tau)).$$

The right-hand side of (3.1) is approximately .75 for $x_*=4$ and .01 for $x_*=4.5$. We show in Table IV, for all six reference images, the smallest τ (interpolated between multiples of .7 pixels) for which $x_*(\tau) \geq 4$ when $\sigma=1$. [Note that reducing σ by the factor 1/2 does not change H_τ but multiplies both Γ and Ψ by 1/2, so that x_* is inversely proportional to σ]. Table IV already indicates why θ is harder to estimate for reference images 1 and 3 than for the others. A com-

parison between Tables I and IV indicates that while upper quartiles for $||\hat{\theta}-\theta||$ can of course not be reasonably predicted via the bound (B), nevertheless there is some value in the figure-of-merit $x_*(\tau)$ (estimated by $H_\tau/[\Gamma+1.22\ \Psi(1)]$) for discriminating those reference images for which θ is easier to estimate (2 and 4-6 in our cases).

Summary:

According both to theoretical inequalities and the simulation study reported here, automatic subpixel registration with respect to real gray-level reference images (assumed to be observed translated, with a stationary noise field added to the pixel gray-levels) seems quite feasible. The present simulation study —— one of the first systematic performance evaluations of the maximum-correlation method of image-registration and of a known effective variant based on maximizing a least-squares quadric surface locally approximating the (discrete) correlation-statistic near its (discrete) maximum —— shows that even if the additive noise has standard deviation as large as that of the 35 by 35 reference image, the upper quartile of the error in registration need be no more (and may be much less) than .25 to .5 pixel.

REFERENCES

[1] Adler, R., The Geometry of Random Fields, (Wiley, New York, 1981).

[2] Deo, C., A Functional central limit theorem for stationary random fields, Ann. Prob. 3 (1975) 708-715.

[3] Karlin, S. and Taylor, H.M., A first course in stochastic processes, (Academic Press, New York, 1975).

[4] Marcus, M., A bound for the distribution of the maximum of continuous Gaussian processes, Ann. Math. Statist. 41 305-309.

[5] Mostafavi, H. and Smith, F., Image correlation with geometric distortion. Part I: Acquisition performance. Part II: Effect on local accuracy, IEEE Trans. Aerospace Elec. Sys. AES-14, 487-493; 494-500.

Appendix. Proof of upper bound on misregistration probability

Let $Y(t)$ be a real-valued separable random field on $[0,2T_0]^d$ where $d \geq 2$ and $2T_0$ are integers, and let S be the complement in $[0,2T_0]^d$ of a convex set. Assume also that for $s, t \in S$, fixed Γ, and a non-decreasing continuous function Ψ satisfying $\int_1^\infty \Psi(e^{-x^2})dx < \infty$ and $\Psi^2(u)\log(1/u)$ decreasing in u, that

(A. 1) $|Y(t)|/\Gamma$ and $|Y(t)-Y(s)|/\Psi(||t-s||_2)$ *each have distribution functions (at* x*)* $\leq (2/\pi)^{1/2}\int_0^x e^{-u^2/2}du$.

Lemma A.1. Under the foregoing assumptions, whenever $x \geq (4d\log n)^{1/2}$, where $n \geq 2$ is a fixed integer,

$$P\{\sup\{|Y(t)| : t \in S\} \geq x(\Gamma + \frac{2}{\sqrt{2}-1}\int_1^\infty \Psi(n^{-u^2})du)\} \leq C(d,n)\int_x^\infty e^{-u^2/2}du$$

where

$$C(d,n) = (2/\pi)^{1/2}\{(2T_0n^2+1)^d + (2T_0n^2)^d \frac{d^{(3d-1)/2}}{3}(\frac{4d\log n}{4d\log n - 1})\sum_{p=1}^\infty 2^{-p/2}\frac{1}{[1-2^{-p-1}(\log d)/(\log n)]^{1/2}}\}$$

The proof, which we omit, is a direct imitation of the method of [4], using for each $t \in S$ a sequence $k(p)/c(p)$ of points in S such that $||c(p)t - k(p)||_\infty \leq 1$, where $k(p)$ has integer coordinates and $c(p) = 2T_0n^{2^p}$ for $p \geq 1$. We must remark that Marcus assumed his random process Gaussian although he used only the property (A.1) (in the one-dimensional case). Lemma A.1 is a simple generalization of the main results of [4] to the d-dimensional case.

Now specialize to the case $d=2$ and $n=2$, replace $[0,2T_0]^d$ by $[-T_0,T_0]^2$, and fix $\theta \in [-T_0,T_0]^2$ and $\tau>0$. Let $S = \{t \in [-T_0,T_0]^2 : ||t-\theta|| \geq \tau\}$, and put $Y(t) = C(t)-C(\theta)-D(t)+D(\theta)$, where C and D are as in Section 1. Then

$$P(\sup\{C(t) : ||t||_1 \leq T_0,\ ||t-\theta||_2 \geq \tau\} \geq C(\theta)) \leq$$

$$P(\sup\{Y(t) : ||t||_1 \leq T_0,\ ||t-\theta||_2 \geq \tau\} \geq$$

$$\inf\{D(\theta)-D(t) : ||t||_1 \leq T_0,\ ||t-\theta||_2 \geq \tau\})$$

and putting $H_\tau = \inf\{D(\theta)-D(t) : ||t||_1 \leq T_0,\ ||t-\theta||_2 \geq \tau\}$ and applying Lemma A.1 yields the bound (B) of Section 1.

Figure 1

These graphs display the simulated empirical distribution functions for $| |\hat{\theta} - \theta| |_2$ (lower curves) and $| |\theta^{LS} - \theta| |_2$ (upper curves) from Study 1 (n=450). For reference image 1, the graph corresponds to $\sigma = .2$; for image 2, to $\sigma = .4$; for image 4, to $\sigma = .4$; for image 5, to $\sigma = .8$; and for image 6, to $\sigma = 1.2$.

Table I

Triples of empirical 75th percentile values for $(\|\hat{\theta}-\theta\|_2, \|\theta^{LS}-\theta\|_2, \|\tilde{\theta}-\theta\|_2)$ from Study 1 (450 iterations), for each reference image and each of three values of σ.

σ	Image 1	σ	Image 2
.10	(.83, .64, .80)	.2	(.50, .24, .58)
.20	(1.45, 1.32, 1.31)	.4	(.58, .46, .73)
.30	(2.1, 1.86, 1.94)	.6	(.86, .8, 1.02)

σ	Image 3
.12	(.62, .55, .90)
.24	(.67, .64, .97)
.36	(.8, .82, 1.1)

σ	Image 4	Image 5	Image 6
.4	(.49, .26, .59)	(.48, .19, .55)	(.49, .19, .53)
.8	(.51, .36, .66)	(.50, .44, .64)	(.51, .28, .60)
1.2	(.57, .50, .75)	(.57, .52, .80)	(.55, .41, .68)

Table II τ vs. H_τ for six reference images

	Image 1	2	3	4	5	6
τ	H_τ					
.7	.014	.212	.034	.187	.244	.197
1.4	.027	.382	.069	.401	.434	.461
2.1	.062	.551	.103	.518	.530	.662
2.8	.098	.636	.103	.518	.608	.662
3.5	.150	.806	.137	.580	.643	.669
4.2	.202	.890	.172	.590	.678	.669
4.9	.235	.975	.172	.590	.678	.669
5.6	.329	1.0	.322	1.0	.734	.720
6.3	.399	1.0	.372	1.0	.770	.850
7.0	.469	1.0	.422	1.0	.866	.895

Table III
Γ, $\Psi(1)$, and $\Psi(\sqrt{2})$ values (for $\sigma = 1$)

	Γ	$\Psi(1)$	$\Psi(1.414)$
Image			
1	.0453	.0075	.0105
2	.0615	.0207	.0284
3	.0536	.016	.022
4	.0603	.030	.035
5	.0581	.025	.029
6	.0568	.028	.034

Table IV

Smallest τ (linearly interpolated from H_τ between multiples of .7 pixel) for which $x_*(\tau) \geq 4$, for six reference images and four values of σ.

Image	1	2	3	4	5	6
σ=						
1	7.0	1.3	5.5	1.4	1.1	1.1
.5	5.2	.6	3.7	.7	.5	.7
.25	3.4	.3	1.5	.35	.3	.3
.125	2.2	.1	.7	.2	.1	.2

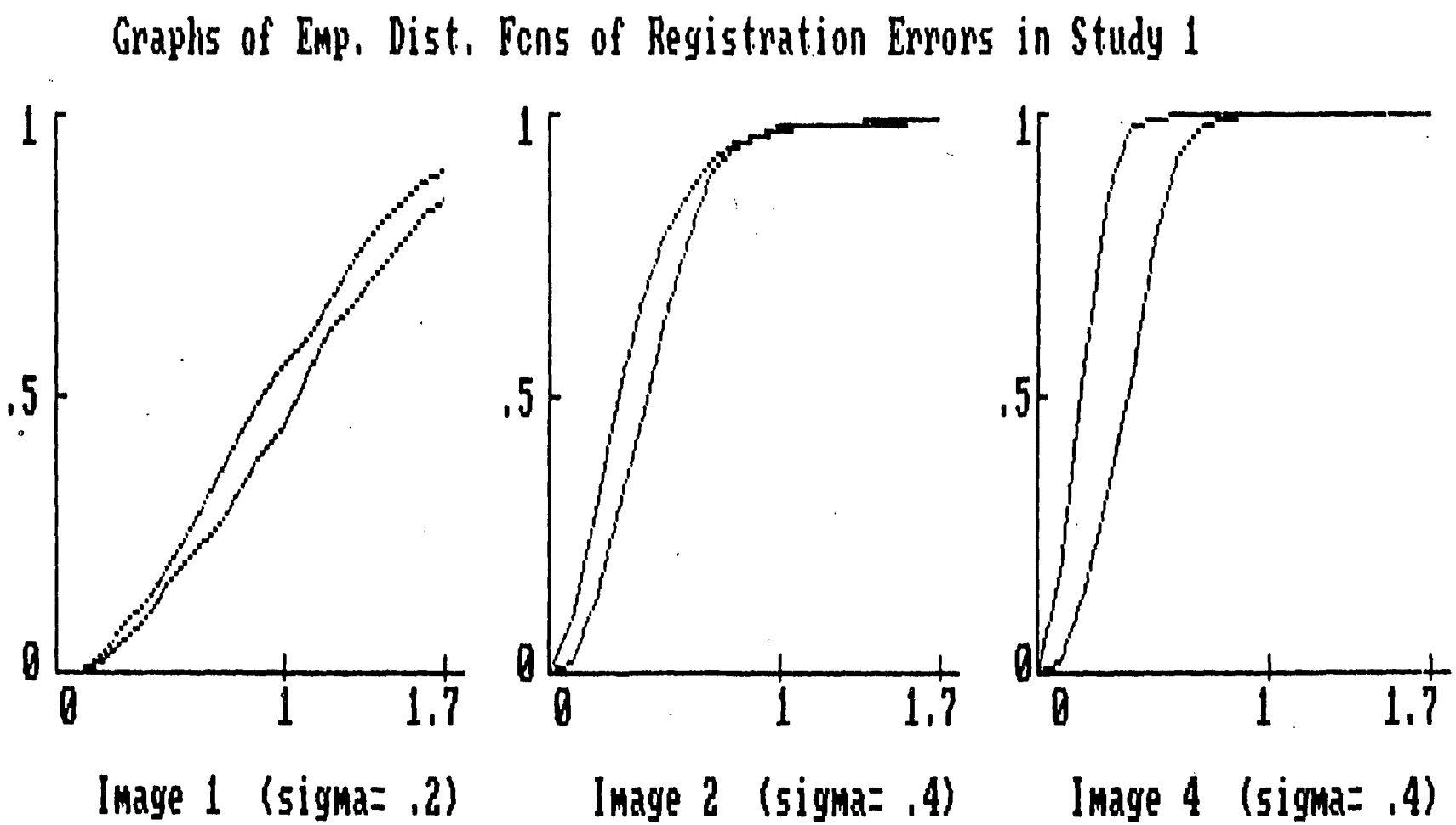


Fig. 1A. Upper curve is e.d.f. for least-sqaures method; lower is e.d.f. for max-correlation pixel vertex ; vertical scale is probability ; horizontal scale is distance in units of 1 pixel.

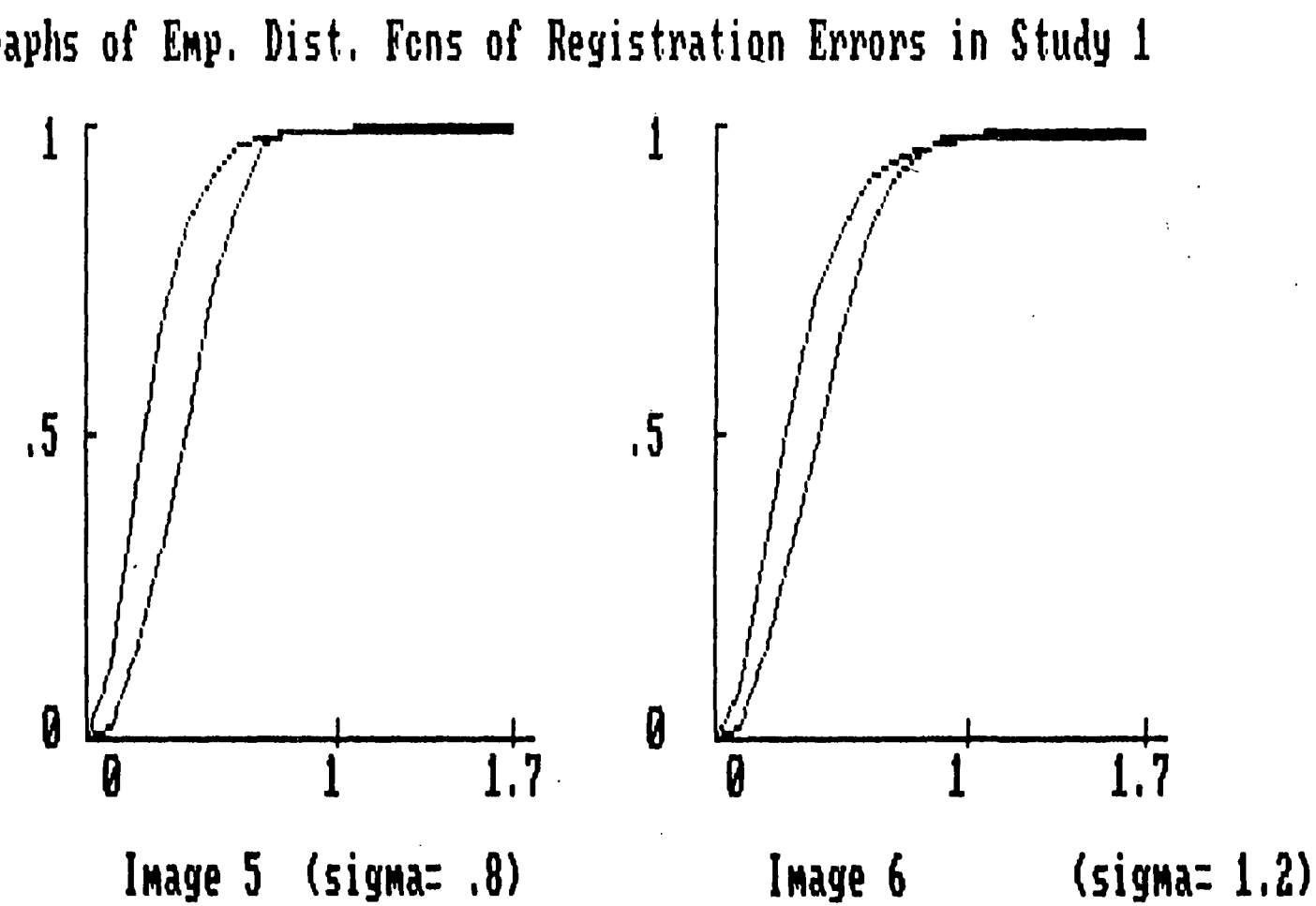


Fig. 1B. Upper curve is e.d.f. for least-sqaures method; lower is e.d.f. for max-correlation pixel vertex ; vertical scale is probability ; horizontal scale is distance in units of 1 pixel.

Recovery of Surface Shape from Multiple Images

Grahame B. Smith

Artificial Intelligence Center, SRI International
Menlo Park, California 94025

Abstract

The conventional approach to the recovery of scene topography from multiple images is based both on the identification of distinctive scene features and on the application of constraints imposed by the viewing geometry. We offer a new prescription for recovering a relative-depth map. We integrate image irradiance profiles to find dense relative-depth profiles. Our procedure neither matches image points (at least, not in the conventional sense) nor "fills in" data to obtain the dense depth map. Although there are outstanding problems associated with depth discontinuities and image noise, the technique is effective.

1. Introduction

The objective of classifying areas of the earth's surface according to attributes of that surface is central to the science of remote sensing. These attributes can be divided into two classes: those associated with the topological and geometrical nature of the surface, and those related to material composition, surface coverage and usage. A substantially different approach has been taken to ascertain the attributes of these two classes. While remotely sensed measurements must recover surface shape if they are to determine topological and geometrical properties of the surface, measurements designed to elicit data regarding material composition, surface coverage and usage have not usually sought to "understand" the shape of the surface. Such an understanding, however, may be vital for successful determination of those properties. We therefore address the problem of recovery of surface shape not only to establish the topological and geometric properties, but also to provide an underlying three-dimensional model to assist in recovering those other attributes of material composition, surface coverage and usage.

What information is needed to determine surface shape uniquely? Previously [1] we examined the shading information available in a single image. We concluded that there is not enough information in the shading to determine surface shape, although that information does constrain the possible shapes. Is there enough information in two or more images of the surface? Certainly the human visual system can fuse a stereo pair of images, but conventional approaches to stereo processing have not

The research reported herein was supported by the Defense Advanced Research Projects Agency under Contract MDA903-83-C-0027 and by the National Aeronautics and Space Administration under Contract NASA 9-16664. These contracts are monitored by the U.S. Army Engineer Topographic Laboratory and by the Texas A&M Research Foundation for the Lyndon B. Johnson Space Center.

provided a completely automatic procedure for doing so. Most conventional stereo processing systems require corrective human intervention when a deviant surface shape is produced. This paper takes an alternative approach to processing two or more images in an effort to understand the nature of multi-image interpretation.

First we shall examine conventional stereo methods to determine where different procedures might be warranted. Then we present an alternative for the more demanding aspects of the conventional approach. Finally we present the results we have obtained and discuss their implications.

2. Conventional Stereo Processing

The conventional approach to recovering scene topography from a stereo pair of images (or from a motion sequence) is based on the identification and matching of distinctive scene features and on the satisfaction of constraints imposed by the viewing geometry. Typically, three steps are required: determination of the relative orientation of the two images, computation of a sparse depth map, and derivation of a dense depth map for that scene.

In the first step, points corresponding to unmistakable scene features are identified in each of the images. The relative orientation of the two images is then calculated from these points. This is, in part, an unconstrained matching task. Corresponding image features must be found. Without a priori knowledge, such a matching procedure knows neither the approximate location (in the second image) of a feature found in the first image, nor the appearance of that feature. We may often assume that appearance will vary little between images and that they were

taken from similar positions relative to the scene, but this assumption is based on a priori knowledge of the acquisition process.

Recovery of the relative orientation of the images reduces the computation of a sparse depth map from unconstrained two-dimensional matching to constrained one-dimensional matching. The quest for a scene feature identified in the first image is reduced to a one-dimensional search along a line in the second image. Identification of this feature in the second image makes it possible to calculate the feature's disparity, and hence its relative scene depth.

Identification of corresponding points in the two images is based primarily on correlation techniques. Area-based correlation processes may be applied directly to the raw image irradiances or to images that have been preprocessed in some manner. For example, edges (identified by the zero crossings of the Laplacian of their image irradiances) have been used in obtaining correspondences.

The outcome of this second step is a sparse map of the scene's relative depth at those points that were identified in both images of the stereo pair.

A sparse depth map does not define the scene topography. The third and final step in recovering the topography of the scene is "filling in" this sparse map to obtain a dense depth map of the scene. Typically, a surface interpolation or approximation method is used as a means of calculating the dense depth map from its sparse counterpart. A surface approximation model may be formulated to provide desirable image properties (such as the lack of additional zero crossings – in the Laplacian of the image irradiances – that are artifacts of the surface approximation model), but often the surface model is based on a priori requirements for the fitted surface, such as smoothness.

The problems encountered in the first two steps – recovery of the relative orientation of the images and computation of the sparse depth map – are dominated by the problems of image matching. False matches that arise from repetitive scene structures, such as windows of a building, or from image features that are not distinctive (at least, on the basis of local evidence) occur more frequently in the unconstrained matching environment than in the constrained environment. Fortunately, in recovering the relative orientation of the images, we can use redundant information in an effort to reduce the influence of false matches. This is not the case when the sparse depth map is computed. While constrained matching is less susceptible to false matches than is unconstrained matching, there is no redundant information that can be used to identify problems. Furthermore, we have little choice as to which features we may use for sparse depth mapping; if we choose not to use a feature, we cannot recover the relative depth at that scene point.

The selection of suitable features for determining image correspondence is difficult in itself. Correlation techniques embed assumptions that are often violated by the best image features. Area-based correlation techniques usually reflect the premise that image patches are of a scene structure that is all at one distinct depth, whereas edges that arise at an object's boundaries are surrounded by surfaces at different scene depths. Edge-based techniques are based on the assumption that an edge found in one image is not "moved" by the change in viewing position of the second image, whereas zero crossings found at boundaries of objects whose gradients are tangential to the line of sight contradict this assumption. These would seem minor problems, were it not for the accuracy required of the matching process. Typically, the spatial resolution of disparity measurements must be an order of

magnitude better than the image's spatial resolution. Matching appears to require distinct features whose properties are incompatible with the assumptions needed to implement the matching process.

The third step, derivation of a dense depth map from a sparse one, is barely adequate. While stereo pairs of images are used to compute the sparse depth map, they have generally been ignored when the dense surface is being filled in. The dense depth map should, in principle, serve as a potential basis for reproducing the stereo pair of images. The computation of the dense depth map should make explicit use of the stereo irradiance data.

While the first step, recovery of the relative orientation of the images, is not an easy problem it does have the advantage of redundancy. We assume in this paper that the relative orientation of the images has been computed. The most demanding steps are the final two: computation of a sparse depth map, and derivation of its dense counterpart. We offer a new prescription for these steps by combining them to recover a dense relative-depth map of the scene directly from the image pair. We use image irradiance profiles as input to an integration routine that returns the corresponding dense relative-depth profile. Our procedure neither matches image points (at least, not in the conventional sense), nor does it "fill in" data to obtain the dense depth map.

First, we extract "corresponding" irradiance profiles from a stereo pair of images. This is the epipolar mapping that allows stereo reconstruction to be treated as a set of one-dimensional problems. Then we formulate the one-dimensional integration procedure that returns relative depth. This is the main result presented in this paper.

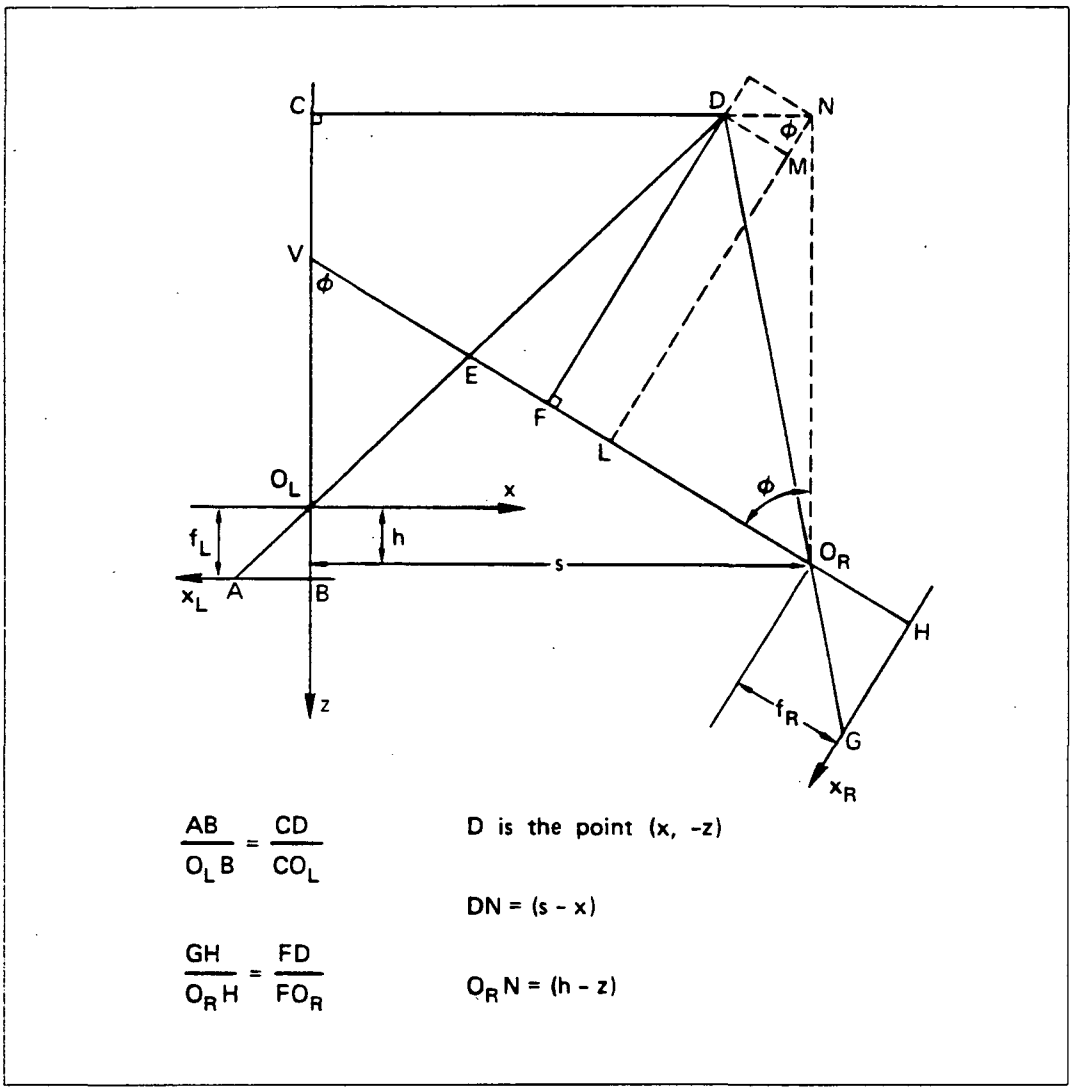


Figure 1 Geometrical Arrangement. The two-dimensional arrangement in the epipolar plane that contains the optical axes of the imaging systems.

While we phrase this presentation in terms of stereo reconstruction, it should be noted that there is no restriction on the positions from which acquisition of the two images occurs; they may equally well be frames from a motion sequence.

3. "Corresponding" Image Irradiance Profiles

The integration procedure takes two image irradiance profiles – one from the left image, one from the right – and computes the corresponding relative-depth profile of the scene. In this section we define "corresponding" irradiance profiles. These are basically the epipolar-mapping considerations, but they provide a means of introducing our notation and establishing the one-dimensional situation analyzed in the next section.

We could select any coordinate frame to describe scene depth, provided that we know the position and orientation of the optical systems relative to that frame. Without loss of generality, we shall select a particular frame based on the optical arrangement of the left imaging system. Scene depth recovered in this frame may be transformed into any desired frame of reference.

If two optical systems are pointed in arbitary directions this adds a level of complication that we wish to avoid in this presentation. We shall assume that the left and right optical systems are such that their optical axes intersect and that, consequently, these axes are coplanar. This restriction can be removed with minimal modification of the model presented [2]. However, clarity of explanation is gained by adding this restriction.

We consider a scene depth profile that is the intersection of an epipolar plane through the two optical centers and a point in the scene. Figure 1 illustrates the two-dimensional situation. The optical (lens) centers are points O_L and O_R. Two rays emanate from the scene point D and intersect the image planes of the left and right optical systems at points A and G respectively. The image plane coordinates are x_L and x_R. The world coordinate system we adopt is based on the optical

arrangement of the left imaging system. The optical axis of the left system defines the z axis. The positive z direction is from world to image, with the optical center of the left system, O_L, as the origin. The x coordinate axis lies in the plane and is parallel to the x_L axis.

The two irradiance profiles, one from the left and one from the right image, viewed as functions of the particular coordinates x_L and x_R, are our "corresponding" image irradiance profiles. We use these irradiance profiles to compute the scene depth profile associated with these irradiance profiles.

By rotating the epipolar plane about the axis through the two optical centers, we can build up the two-dimensional scene depth map by recovering the one-dimensional depth profiles.The circumstances depicted in Figure 1 are the same for any "corresponding" image irradiance profiles when these are described as functions of x_L and x_R. Consequently, the following analysis of the situation shown in Figure 1 is independent of the epipolar plane used. Once a depth profile of the scene has been recovered (by using the algorithm presented below), this profile can be related to others simply as a function of the angle between the epipolar plane and the optical axes of the imaging systems.

4. Recovery of Relative Depth

The geometrical arrangement presented in Figure 1 allows us to derive expressions relating the world coordinates of the scene to the image coordinates of its projection. The similar triangles ABO_L and CDO_L, along with those of GHO_R and FDO_R, allow us to write $\frac{AB}{O_LB} = \frac{CD}{CO_L}$, and hence

$$\frac{x_L}{f_L} = \frac{x}{-z} \quad . \tag{1}$$

Also $\frac{GH}{O_RH} = \frac{FD}{FO_R}$, but, $\frac{FD}{FO_R} = \frac{LN-MN}{LO_R+MD} = \frac{O_RN\sin\phi - DN\cos\phi}{O_RN\cos\phi + DN\sin\phi}$, $DN = (s-x)$, and $O_RN = (h-z)$, yielding

$$\frac{x_R}{f_R} = \frac{(h-z)\sin\phi - (s-x)\cos\phi}{(h-z)\cos\phi + (s-x)\sin\phi} \quad . \tag{2}$$

Solving Equations (1) and (2) for x and z, we obtain expressions for the world coordinates of a scene point in terms of image-measurable quantities and the imaging parameters that specify the relative orientation of the two images. The equations are the usual ones obtained from the stereo geometry:

$$x = x_L \frac{(x_R s - f_R h)\tan\phi + x_R h + s f_R}{(x_R x_L + f_R f_L)\tan\phi - x_R f_L + x_L f_R} \quad , \tag{3}$$

and

$$z = -f_L \frac{(x_R s - f_R h)\tan\phi + x_R h + s f_R}{(x_R x_L \cos\gamma + f_R f_L)\tan\phi - x_R f_L + x_L f_R} \quad . \tag{4}$$

Equations (3) and (4) form part of the algorithm we present. Equations (1) and (2) are used as part of our analysis of the image irradiance information available to us in the two images.

We now turn our attention to scene radiance. Rays of light emanate from a scene point and travel to their image projections. What is the relationship between the scene radiances of the rays that project into the left and right images respectively? Let us suppose that the angle between the two rays is small. The bidirectional reflectance function of the scene's surface will vary little, even when it is a complex function of the lighting and viewing geometry. Alternatively, let us suppose that the surface exhibits Lambertian reflectance. The scene radiance

is independent of the viewing angle; hence the two ray will have identical scene radiances, irrespective of the size of the angle between them. For the model presented here, we assume that the scene radiances of the two rays emanating from a single scene point are equal. This assumption is a reasonable one when the scene depth is large compared with the separation between the two optical systems, or when the surface exhibits approximate Lambertian reflectance. For temporally separated images this assumption is not valid. Such images will need to be recalibrated to remove the irradiance changes due to contrast and the like. For images in which the scene content has changed, such recalibration is not possible. We will consider recalibration further during the discussion. It should be noted that there are no assumptions about albedo (e.g., it is not assumed to be constant across the surface) and, in fact, it is not even necessary to know or calculate it. Since image irradiance is proportional to scene radiance, for corresponding image points we can write

$$I_L(x'_L) = I_R(x'_R) \quad .$$

I_L and I_R are the image irradiance measurements for the left and right images. It should be understood that these measurements at positions x'_L and x'_R are at image points that correspond to a single scene point.

Differentiating the above equation gives

$$\frac{dI_L}{dx}(x'_L) = \frac{dI_R}{dx}(x'_R) \quad ,$$

and hence

$$\frac{dI_L}{dx_L}(x'_L)\frac{dx_L}{dx} = \frac{dI_R}{dx_R}(x'_R)\frac{dx_R}{dx} \quad .$$

Expressions for $\frac{dx_L}{dx}$ and $\frac{dx_R}{dx}$ are obtained by differentiating Equations (1) and (2), as follows:

$$\frac{dx_L}{dx} = -\frac{f_L + x_L \frac{dz}{dx}}{z} \quad , \tag{5}$$

$$\frac{dx_R}{dx} = \frac{x_R \tan\phi + f_R + (x_R - f_R \tan\phi)\frac{dz}{dx}}{(h-z) + (s-x)\tan\phi} \quad . \tag{6}$$

Substituting these into the preceding equation and rearranging terms, we obtain an expression for $\frac{dz}{dx}$, namely,

$$\frac{dz}{dx} = -\frac{\left(\frac{dI_L}{dx_L} f_L (h - z + (s-x)\tan\phi) + \frac{dI_R}{dx_R} z(x_R \tan\phi + f_R)\right)}{\left(\frac{dI_L}{dx_L} x_L (h - z + (s-x)\tan\phi) + \frac{dI_R}{dx_R} z(x_R - f_R \tan\phi)\right)} \quad . \tag{7}$$

Note that, for clarity of expression, we have dropped the notation (x'_L) and (x'_R) that shows the value of the independent variable at which the image irradiance gradients are to be evaluated. All terms that involve the image irradiance are understood to be evaluated at corresponding image points.

We are now ready to outline an algorithm to recover scene depth:

1. Suppose we have a pair of corresponding image points, x_L and x_R.

   We use Equations (3) and (4) to calculate x and z for the scene point.

2. Equation (7) is used to calculate $\frac{dz}{dx}$ for this scene point.

3. Equations (5) and (6) are used to calculate a dx_R for a chosen dx_L.

4. The pair of points $x_L + dx_L$ and $x_R + dx_R$ are corresponding image points;

   Steps 1 to 3 may be repeated.

This, then, is an integration procedure that, given an initial pair of corresponding image points, proceeds along the two image irradiance profiles, maintaining the correspondence. As in other numerical integration procedures, we can adjust the

step size dx_L so that the scene's profile gradient, $\frac{dz}{dx}$, varies slowly between successive steps. In the following section we shall discuss the application of this algorithm to scene profiles that have discontinuities.

An obvious difficulty with the algorithm, as outlined, occurs when both $\frac{dI_L}{dx_L}$ and $\frac{dI_R}{dx_R}$ are zero; $\frac{dz}{dx}$ is indeterminate. A solution is still possible if the second derivatives of image irradiance are not zero as well. Differentiating $I_L = I_R$ twice gives us

$$\frac{d^2 I_L}{{dx_L}^2}\left(\frac{dx_L}{dx}\right)^2 + \frac{dI_L}{dx_L}\frac{d^2 x_L}{dx^2} = \frac{d^2 I_R}{{dx_R}^2}\left(\frac{dx_R}{dx}\right)^2 + \frac{dI_R}{dx_R}\frac{d^2 x_R}{dx^2} \quad ,$$

which reduces to

$$\sqrt{\frac{d^2 I_L}{{dx_L}^2}}\frac{dx_L}{dx} = \sqrt{\frac{d^2 I_R}{{dx_R}^2}}\frac{dx_R}{dx} \quad ,$$

when $\frac{dI_L}{dx_L}$ and $\frac{dI_R}{dx_R}$ are zero. Hence

$$\frac{dz}{dx} = -\frac{\left(\sqrt{\frac{d^2 I_L}{{dx_L}^2}} f_L(h - z + (s - x)\tan\phi) + \sqrt{\frac{d^2 I_R}{{dx_R}^2}} z(x_R \tan\phi + f_R)\right)}{\left(\sqrt{\frac{d^2 I_L}{{dx_L}^2}} x_L(h - z + (s - x)\tan\phi) + \sqrt{\frac{d^2 I_R}{{dx_R}^2}} z(x_R - f_R \tan\phi)\right)} \quad . \qquad (8)$$

When $\frac{dI_L}{dx_L}$ and $\frac{dI_R}{dx_R}$ are both zero, we adjust Step 2 of the algorithm to use Equation (8) rather than Equation (7). This allows integration through the peaks and troughs of image irradiance.

It should be noted that scene depth profiles of planar objects have zero image irradiance gradients and zero second derivatives. These situations must be detected and treated separately, for, except at the object's boundaries, there is no information available from which to assess orientation.

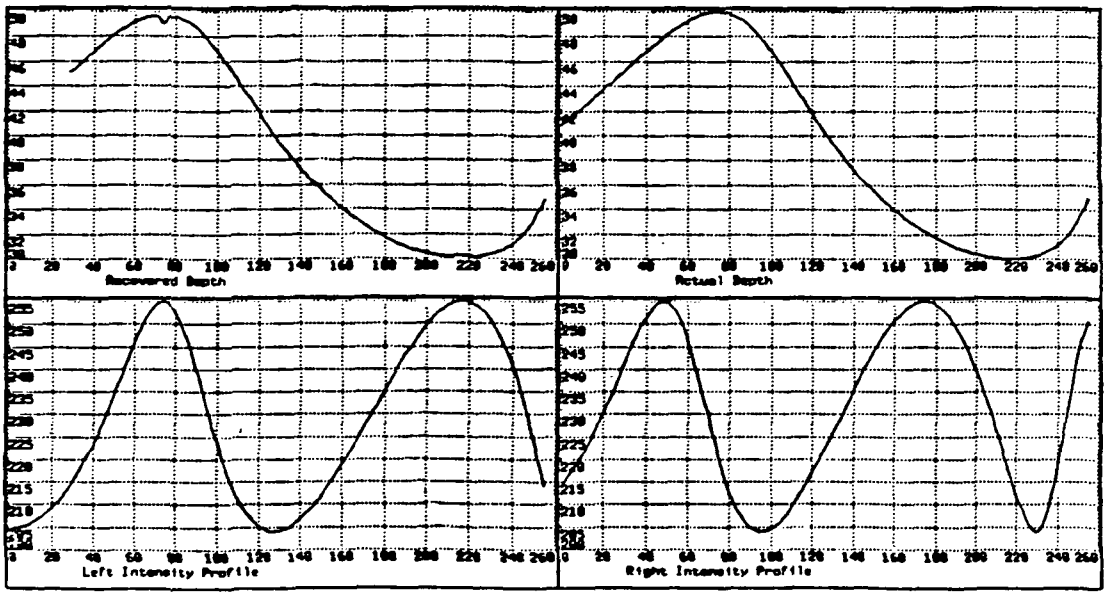


Figure 2 Depth Recovery: Ideal Case. At the upper left is shown the recovered depth from the two irradiance profiles depicted in the lower half of the figure. For comparison, the actual depth is shown at the upper right.

The integration routine uses the information available in the geometric distortion of perspective projection. It does not use the reflectance characteristics of the scene, nor does it need to know them. The method is based on the assumption that the scene radiances of two rays emanating from a single scene point (and entering the two optical systems) are identical. Spatial variations in albedo and lighting are inconsequential for this procedure.

5. Experimental Results and Discussion

The presented algorithm requires spatially continuous image irradiance profiles as input. To apply it to digital images, we must first construct spatially continuous profiles from their sampled counterparts. We employ simple modeling techniques, such as linear interpolation, for this purpose.

The result of applying the above algorithm to two synthetic, corresponding Lambertian image irradiance profiles is shown in Figure 2. The actual depth profile corresponding to the irradiance profiles is shown in the upper right portion of Figure 2. For this example, initial starting positions for the integration were selected near the center of each profile. These initial positions were corresponding points, with no error in the determination of their location. The integration process was applied in both directions from the initial point. The recovered depth is shown in the upper left corner of Figure 2.

A second example is shown in Figure 3. The image irradiance profiles were obtained by "painting" the previous surface with "pigment" of continuously varying albedo. In addition, three strips of different albedos were painted on the surface. The effect can be seen by examining the image irradiance profiles shown in the bottom half of Figure 3. The processes we applied to recover depth were twofold. First, we used a simple smoothing routine, based on moving average, to produce intermediate profiles. This rounded the step edges associated with the albedo strips. Next the integration procedure was performed. The result is shown in the upper left part of Figure 3.

You will notice small errors near the peaks and troughs of irradiance, where second-derivative information is being used. Furthermore, there are small errors associated with albedo edges. What is happening here is that the tracking mechanism that maintains point correspondence as it moves along the profiles is getting out of sync. The process is "self-correcting," however, a feature that we will exploit in the next example. Note that the continuously variable albedo change across the profiles has no influence on the resulting recovered depth.

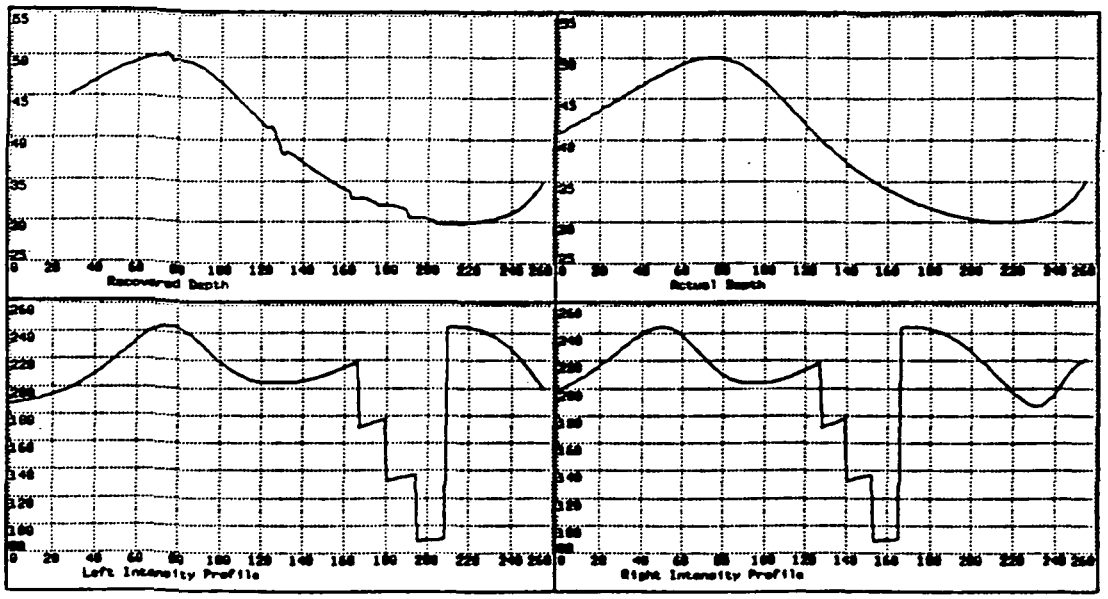


Figure 3 Depth Recovery: Painted Surface.

What would be the effect if the initial matched points were in error? We repeat the above procedure but select initial starting points that are mismatched by two pixels (the horizontal units in Figures 2, 3, 4 and 5). The left half of Figure 4 demonstrates the result achieved. The effect of the starting point error shows up as depth error at positions 120 to 130 on the horizontal axis. Note the swift correcting action, which suggests that the initial points are not critical for recovering depth. Clearly, this algorithm has a very special feature whose implication for stereo processing is far-reaching: approximate matches are all that is necessary for the recovery of scene depth.

The above examples have been based on synthetic images. We now turn our attention to real scenes that are full of discontinuities in the depth profile, as well as to real images that are not free of noise.

In the synthetic scene profile used in the preceding examples, we have used

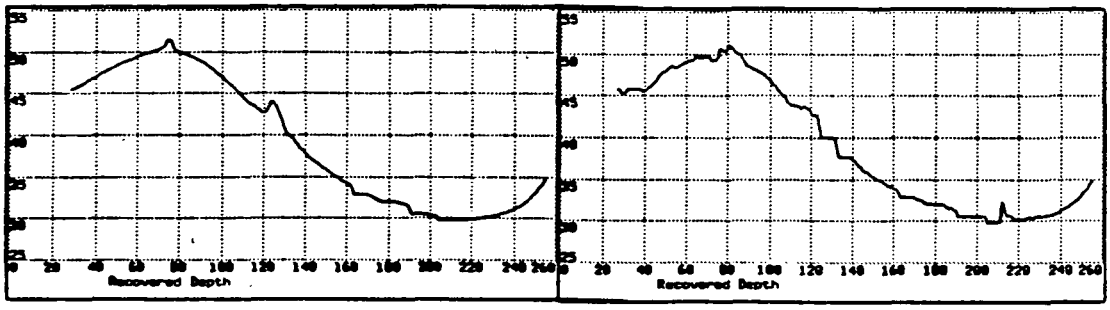


Figure 4 Depth Recovery: Mismatched Initial Points, and Noise Concerns.

continuous-depth profiles. For real scenes this is unrealistic. At an object's boundaries, discontinuities in depth are likely. Because the presented algorithm cannot integrate across these discontinuities, we need to be able to identify them. Let us suppose that we use zero crossings of the Laplacian of image irradiance as places at which depth discontinuities may occur. We shall apply our integration procedure, tracking along the image irradiance profiles until we come to a zero crossing in one of the image irradiance profiles.

If continuation implies that the scene depth gradient, $\frac{dz}{dx}$, varies slowly, we continue. A sudden change in gradient signals a depth discontinuity and the integration procedure is terminated. Note that the integration routine itself signals depth discontinuity if $\frac{dz}{dx}$ exhibits rapid change for arbitrarily small step sizes. This procedure also handles occlusion problems in which one view (hence its image irradiance profile) "sees" around an object that is occluded from the other view. Again we stop at the first zero crossing encountered in either of the image irradiance profiles, or when $\frac{dz}{dx}$ changes too rapidly. It should be noted that the above procedure does not require that the zero crossing from both image irradiance profiles be matched; rather, it simply requires their detection.

Of course, there is a price that must be paid: we now need to be able to

detect initial starting points for the integration procedure between adjacent zero crossings. The peaks and troughs of irradiance would seem appropriate, being invariant through most realistic image irradiance transformations that may occur during image acquisition. Furthermore, as these peaks and troughs of the two irradiance profiles match (considering that the value of irradiance should be identical at matched points), the opportunity exists for correcting the image irradiances for linear transformations in contrast. This allows for local contrast correction – an especially important recourse for image pairs that are temporally separated. A suggested procedure is to (1) detect the peaks and troughs in image irradiance, and also the zero crossings of the Laplacian of image irradiance; (2) match the peaks and troughs across the two images to provide initial points for integration;[1] (3) correct the image irradiance profiles for each profile section between peaks and troughs for a linear transformation in contrast; (4) then apply the integration procedure, terminating at rapid changes in $\frac{dz}{dx}$ or at zero crossings, if necessary. We are currently giving our attention to these matters.

A serious deficiency of the present algorithm is its sensitivity to noise – a disadvantage inherent in any procedure that makes use of image irradiance gradients. This sensitivity can be easily demonstrated with quantization noise alone. If the image irradiances shown in Figure 3 are quantized to 256 different levels, the results of applying the algorithm can be seen in the right half of Figure 4. This result should be compared with the one shown at the upper left of Figure 3. Noise is an undeniable problem. We have difficulty in recovering reliable depth estimates if

[1]We do not underestimate the difficulty of this step, but the basic assumptions implicit in correlation techniques are likely to be satisfied near peaks and troughs. Some mismatch error can be tolerated and as we can integrate through peaks and troughs of image irradiance, we have only to detect and match the "obvious" ones.

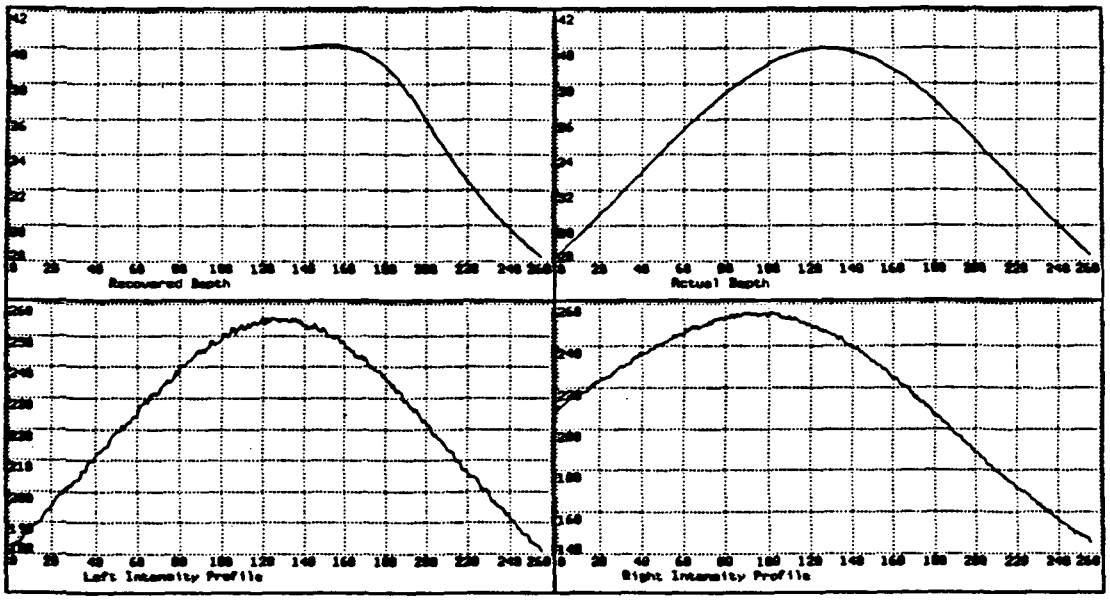


Figure 5 Depth Recovery: Alternative Expression for the Depth Profile Gradient.

the signal-to-noise ratio is less than a few hundred. This sensitivity is particularly apparent when the image irradiance gradient is small. Smoothing of the image irradiance profiles is at best inadequate.

An approach that is competent to deal with noise (although it has other deficiencies) is to replace Equation (7), which determines the depth profile gradient $\frac{dz}{dx}$ from image irradiance gradients, with an expression that involves irradiance integrals rather than derivatives. This expression is obtained by integrating the earlier expression

$$I_L(x'_L) = I_R(x'_R) \quad ,$$

with respect to the scene coordinate dx.

$$\int_a^{x'} I_L(x_L)dx = \int_a^{x'} I_R(x_R)dx \quad .$$

Changing the integration variable to image coordinates gives

$$\int_{a_L}^{x'_L} I_L(x_L)\frac{dx}{dx_L}dx_L = \int_{a_R}^{x'_R} I_R(x_R)\frac{dx}{dx_R}dx_R \quad ,$$

where a_L, a_R, and x'_L, x'_R are corresponding points in the left and right images. We replace $\frac{dx}{dx_L}$ and $\frac{dx}{dx_R}$ with Equations (5) and (6), then use this expression to compute $\frac{dz}{dx}$. For computation we replace the integral with finite sums. To calculate these finite sums we use an irregular grid that is positioned at the x'_L and x'_R points previously determined to be in "correspondence" as we integrated the profiles from the starting points, a_L and a_R.

Figure 5 shows the results obtained when we integrated from the center of the left irradiance profile (and from the corresponding point in the right image) to the right. In this example the surface reflectance is Lambertian and the albedo constant. Random noise has been added independently to each of the irradiance profiles. While the recovered depth profile in Figure 5 is reasonable, the integration procedure does not maintain good "correspondence" between its position in the left image and that in the right. Consequently, we cannot handle albedo boundaries with the competence of the previous technique. Some combination of these two approaches may have the desirable properties of maintaining good "correspondence" - thus, while insensitive to noise, be effective across albedo changes. We are actively exploring this problem in our current research. A solution is necessary if the presented algorithm is to become a viable technique for recovering scene depth from pairs of real images that cannot be preprocessed to remove noise.

6. Summary

We have presented a new approach to reconstruction of scene depth from a

pair of images. The technique does not depend upon matching of image features, at least not in the usual sense, nor does the necessary matching require great spatial accuracy. Furthermore, the features to be matched are more compatible than their traditional counterparts with the assumptions implicit in correlation techniques.

The results point to a technique that is capable of handling changes in both albedo and illumination. Furthermore, the technique directly yields a dense depth map of the scene.

We are exploring several related outstanding issues. Among these are the exploitation of depth discontinuities and the problem of reducing sensitivity to image noise.

Besides its direct use in remote-sensing applications, the recovery of scene geometry provides an underlying three dimensional model to assist in the reliable recovery of attributes of the earth's surface. Competent recovery of such surface attributes as material composition has not yet been achieved. Moreover, it is unlikely to be until the techniques we use are able to truly "understand" the shape of the earth's surface.

References

1. Smith, G.B., Shape from Shading: An Assessment, *Proceedings of NASA Symposium on Mathematical Pattern Recognition and Image Analysis*, Houston, Texas, 1983, pp 543-576.

2. Smith, G.B., Stereo Reconstruction of Scene Depth, *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, San Francisco, California, 1985, pp 271-276.

HYPOTHESIS INTEGRATION IN IMAGE UNDERSTANDING SYSTEMS

Vincent Shang-Shouq Hwang
Larry S. Davis
Takashi Matsuyama*

Center for Automation Research
University of Maryland
College Park, MD 20742

*Permanent address: Department of Electrical Engineering, Kyoto University Sakyo, Kyoto, Japan.

ABSTRACT

The goal of this research is to develop a robust control strategy for constructing image understanding systems (IUS). This paper proposes a general framework based on the integration of "related" hypotheses. Hypotheses are regarded as predictions of the occurrences of objects in the image. Related hypotheses are clustered together. A "composite hypothesis" is computed for each cluster. The goal of the IUS is to verify the hypotheses. We constructed an image understanding system, SIGMA, based on this framework and demonstrated its performance on an aerial image of a suburban housing development.

1. Introduction

A primary objective in computer vision research is to construct image understanding systems (IUS's) which can analyze images based on object models. Usually, an IUS analyzes images by constructing *interpretations* in terms of the object models given to the IUS. *Interpretation* refers to the mapping between objects (e.g., houses, roads) in the object model and image structures (e.g., regions, lines, points) in the image. During the analysis, an IUS needs to perform the following two types of tasks:

- segmentation : the task of grouping pixels together to construct image structures that can be associated with objects in the given model.

- interpretation : the task of constructing mappings between image structures and objects.

Segmentation is practical when sufficient knowledge is available about the image to be processed and the image structures to be computed. The base of knowledge increases as the interpretation process develops, leading to more constrained and therefore more reliable segmentation.

Many IUS's were constructed in the late 1970's ([Barr81], [Ball82], [Binf82] [Ball82].) Most systems integrate segmentation and interpretation using one of the following types of analysis.

1) Bottom-up analysis: the image structures are extracted from the image, and are interpreted as instances of the objects in the model. For example, when a large rectangular region is extracted, interpret it as a house.

2) Top-down analysis: the appearance of the object is first determined, and the associated image structures are extracted. For example, suppose an IUS wants to find a house; the IUS invokes the house model and establishes the descriptions of the specific image structures to be extracted from the image.

It is generally accepted that image understanding systems should incorporate both bottom-up and top-down analyses. Some systems use only one type of analysis. MSYS [Barr76] developed by Barrow and Tenenbaum used bottom-up analysis. Image structures are first segmented from the image. A set of initial labels are assigned to these image structures (based on height, homogeneity, etc.) Then, geometric constraints between labels are used to filter out inconsistent labelings. Bolles [Boll76], on the other hand, used top-down analysis. In his system , a goal is first constructed. The system then matches the goal, which is represented as a template, with the image. A similar approach is used in Garvey's [Garv76] system. Other systems (Hanson, Riseman [Hans78]; Matsuyama [Naga80]) incorporate both types of analysis but use ad hoc rules to determine which type of analysis is to be used at what stage during the analysis. Such systems often require a large set of domain dependent control knowledge to direct the analysis of the IUS.

It is the goal of this research to develop a robust control strategy for constructing image understanding systems, thus eliminating the need to use large amounts of domain specific control knowledge in specific applications. In this paper, we propose a general framework which enables IUS's to integrate both bottom-up and top-down analyses into a single flexible reasoning process. We construct an image understanding system, SIGMA, based on this framework and provide demonstrations of its performance on images of a suburban housing development.

1.1. Integration of hypotheses

Considering the following proposition:

> If a structure of type x is present in the scene having certain spatial properties, then there should exist a structure of type y having certain properties in the image.

It is often the case that what is known about x is not sufficient to completely characterize y (i.e., we might be able to predict its size and color, but perhaps not its orientation). In addition, there might be many x's, each predicting the occurrence of y, but each contributing different constraints on the properties of y.

For example, by locating a house in the image, one may predict the occurrences of other objects, e.g., neighboring houses. Furthermore, the discovery of a rectangular homogeneous region in the image may also generate

a prediction of a house. It is usually the case (depending on the object model) that each of these predictions provides some "cues" about the occurrence of a house and it is the integration of all these cues that may characterizes the occurrence of a house adequately enough to easily recognize it.

Let us call the predictions about the occurrences of objects in the image *hypotheses*. Suppose several hypotheses, which may be independently generated, are predictions about objects at the same location in the image. It is reasonable to assume that these hypotheses are predictions about the "same" object, although each may only constrain some subset of the properties of the object. By integrating these hypotheses, an IUS could construct a more complete description of the object and use it to direct a more effective and informed analysis.

1.2. An overview of the SIGMA image understanding system

Figure 1-2 shows the system architecture of the SIGMA image understanding system. The user provides object models to SIGMA, and the results of the analysis are available to the user through a query-answering module.

The image is first segmented by a general purpose low level vision system (LLVS). The segmentation results are recorded in the iconic/symbolic database. The high level vision system (HLVS) uses the object model either to interpret image structures already extracted or to direct the low level processes to search for image structures not yet discovered. During the

analysis, the HLVS incrementally constructs an interpretation network for the input image. A "goal" is given to the query-answering module (QAM). At the end of each analysis iteration, the QAM is activated and "matches" the current status of the analysis with the goal. This construction process continues until the "goal" is accomplished (i.e., a successful match between the current status of the analysis and the goal) or no more interpretations can be constructed. At this stage, the QAM provides the current status of the analysis. In the following subsections, we present each module of SIGMA in more detail.

1.2.1. The low level vision system

In SIGMA, the LLVS is formulated as a domain-independent goal-directed segmentation system. A goal, which is described by a list of constraints on the image structures to be computed, is given to the LLVS. The LLVS uses general segmentation techniques to extract such image structures. Other systems have been constructed to perform goal-directed segmentation - e.g., Selfridge [Self82] and Nazif & Levine [Nazi84].

Our approach differs from the approaches taken in these systems. We assume that many specialized methods are needed to extract image features from the image. An LLVS needs to select, from a pool, methods that best suit the task. Furthermore, new methods are frequently developed that can augment or replace the methods currently used by the LLVS. It is important to

design an LLVS so that adding methods to it is easy.

Our LLVS is based on a select-and-schedule strategy. When the LLVS is asked to verify some hypothesis, it first selects those methods which are applicable by matching the hypothesis against a decision table. Then, the LLVS schedules the selected methods according to their potential. If one method fails to verify the hypothesis, the next method will be tried until the hypothesis is verified or until all methods have been tried and have failed. This approach is similar to the "blackboard" method [Davi77] and the "contract net" idea [Smit78]; but the implementation here is simpler. For a detailed discussion of the LLVS, see [Hwan84].

1.2.2. The high level vision system

The high level vision system (HLVS) uses object models to interpret data recorded in the iconic/symbolic database and construct an interpretation network. The HLVS uses the integration of hypotheses principle to direct analysis. This is implemented by the following reasoning steps.

1) Hypothesis generation: the HLVS generates hypotheses about occurrences of objects in the image.

2) Hypothesis integration: the HLVS clusters "related" hypotheses together.

3) Hypothesis abstraction: the HLVS computes a "composite hypothesis" for each cluster.

4) Hypothesis verification: the HLVS selects hypotheses and verifies them by computing values for those attributes which are not completely

constrained.

The HLVS performs the reasoning iteratively. At the end of each iteration, the HLVS checks whether the "goal" is accomplished by activating the QAM. If the goal is accomplished or no more interpretations can be constructed, the construction process terminates and the status of the analysis is available through the QAM.

1.2.3. Query-answering module

Potentially, SIGMA constructs all possible interpretations for an image. However, SIGMA needs to select, among many interpretations, a good one as its conclusion. Instead of finding a "best interpretation", we model this selection process as a database query answering process. A program (QAM) was developed to answer simple queries about the interpretation network and to display the associated image structures.

The goal of the analysis is provided to the QAM as a query. Whenever the QAM is activated (by the HLVS), it matches the goal with the interpretations already constructed. If any interpretation that satisfies the goal is found, the QAM enters into an answer mode and provides a query-answering capability for selecting "good interpretations" and displaying the explanations for these interpretations.

1.3. Outline of the paper

We first present the knowledge representation paradigm used in SIGMA. In Section 3, we discuss a framework for performing hypothesis integration and abstraction. This is followed by a detailed description of the system constructed based on this framework. Conclusions are presented in the final section.

2. Representation of object models

2.1. What to represent?

The knowledge representation formalism determines a general framework for organizing the necessary knowledge into a knowledge base and supports a powerful inference mechanism for guiding the recognition of a specific scene. An appropriate knowledge representation tool can often simplify the task of transferring problem domain expert knowledge into knowledge bases in computer systems.

Consider the following house model:

> A house is "rectangular" or "L-shaped"; its area is larger than 1000 square feet but no larger than 2500 square feet. A house usually belongs to a group of houses which are on the same side of a road. Roads can be found near the house. Usually, the road is parallel or perpendicular to the house and a driveway connects the road to the house.

Based on how an IUS uses such a model to locate houses in a given image, one can categorize this scene knowledge into the following classes.

1) *What to look for.* This class of knowledge describes the appearances of objects (e.g., the type of image structures associated with objects.) In the house example, the appearance of the house is a homogeneous compact rectangular region. To locate houses, an IUS segments the input image and identifies as houses those regions which are rectangular and compact and whose sizes are between 1000 and 2500 square feet.

2) *Where to look.* This class of knowledge includes the geometric and topological relations between objects. The knowledge base might, for example, specify (based on connectivity, relative orientation, etc.) relations between

driveways, houses, and roads. An IUS might, if one of these objects is discovered (say a driveway), use this relation to initiate and constrain the search for other objects (e.g., a connected house and road) not yet discovered. An IUS might also use such relations to examine whether a house, a driveway, or a road already discovered satisfy the required relations.

3) *When to look.* This class of knowledge describes strategies regarding the application and confirmation of relations. One the one hand, we often want to postpone applying a specific piece of relational knowledge until sufficient information has been obtained to strongly suggest that the relation may be applicable. On the other hand, since the confirmation process often involves the searching of image structures associated with other objects, we might also want to postpone the confirmation of a specific relation until a sufficient description of the object to be searched is collected. For example, when the IUS generates a house hypothesis, instead of searching for an image structure associated with it immediately, the IUS might postpone the search until a sufficient description of the house (e.g., shape, intensity, etc.) is available.

A principal objective of this research is to develop a representation scheme which simplifies the task of capturing domain knowledge as a knowledge base for IUS's. This section presents the knowledge representation scheme used in the SIGMA system. Note that the scene model is used mainly by the HLVS (High Level Vision System) module in SIGMA.

2.2. Basic representation primitives

Our representation formalism is based on frame system theory [Mins75], semantic networks [Wino75] [Hend79], and an object oriented problem solving style [Stee79] [Wein80] [Gold83]. In SIGMA, object models are represented as a graph structure of nodes and arcs. Objects are described by "frames" (nodes in the graph structure) while relations between these objects are described by "rules" and "links" (arcs in the graph structure). In such a formalism, domain

knowledge is built around a set of objects and a set of operations that can be applied to them.

The basic entities of the representation are called *frames* and are used to model abstract objects in the problem domain such as "house" or "road". Each frame may have many associated descriptions that are defined by *slots*. Slots are similar to "property lists" in LISP. Each slot is a list which contains an indicator (i.e., name) and a value.

In addition to slots where values are recorded, we can also associate with frames all the knowledge which is used to compute values of slots. We represent this type of knowledge as *rules*.

Rules used in this context are procedural--i.e., the knowledge about how to compute values of slots is encoded in programs. As mentioned above, these "programs" are written using an object-oriented programming style.

Objects in the scene domain are often structured into hierarchies. It is often natural and convenient to preserve these hierarchies when we construct the scene model. *Links* are used to describe the hierarchical relations between objects.

One object hierarchy often used is the generalization/specialization hierarchy; CAN-BE and AKO links are employed to describe it. Link CAN-BE describes a frame and its specializations while link AKO describes a frame and its generalizations.

Properties are inherited through the AKO link. This usage is similar to the "property inheritance" in semantic networks ([Moor79], [Nils80].) All the knowledge recorded in frames that are linked to a father frame by the AKO link is inherited by that frame. For example, both the RECTANGULAR-HOUSE and the L-SHAPED-HOUSE have centroid, shape-description, front-of-house, and connecting-driveway slots. Also, both the RECTANGULAR-HOUSE and the L-SHAPED-HOUSE can use rule $F_{driveway}$ to compute the connecting driveway.

Often, the HLVS needs to reason across the CAN-BE link. For example, suppose the HLVS needs to compute the shape of a house. The HLVS is not able to do the computation since there is no such rule recorded in the HOUSE frame. Instead, the HLVS needs to reason about what specialization to choose, i.e., RECTANGULAR-HOUSE or L-SHAPED-HOUSE. The strategies for this type of reasoning are called *specialization strategies* and are encoded as rules and recorded in frames. Attaching such search strategies using CAN-BE links is similar to the process of "plan elaboration" in Garvey's system [Garv76]

As an example, suppose that there are two type of houses, rectangular and L-shaped, in community A. Every house has a driveway. However, each type of house has a different appearance. Suppose $F_{rectangle}$ is a rule which computes the shape description of a rectangular house, and $F_{driveway}$ is another rule which finds the driveway connecting to a rectangular house. Rule $F_{driveway}$ computes the driveway of a house. We can write the house model as

shown in Figure 2-1. In this model, the HOUSE frame is a generalization of the L-SHAPED-HOUSE frame and the RECTANGULAR-HOUSE frame while the L-SHAPED-HOUSE frame and RECTANGULAR-HOUSE frame are specializations of the HOUSE frame. Their hierarchical relations are shown in Figure 2-2.

2.3. Instantiation of a frame

Frames are the prototypes of objects. The SIGMA system uses frames as models to construct interpretations of the image by making instances of frames. An *instance* is a copy of a frame. The process of making instances is called *instantiation*. At instantiation, values can be assigned to slots. These values may be the "defaults" (specified in the frame definition) or may be computed using rules. Since all instances are recorded in the iconic/symbolic database in the HLVS as basic database entities, we use the term *Database Entities (DE's)* interchangeably with the term "instances" in the rest of the paper.

An important property of an object is its appearance. During the analysis, the HLVS needs to direct the LLVS (Low Level Vision System) to process the image and locate image structures which are associated with objects. Some objects' appearances are defined in terms of image structures that can be directly computed by the LLVS. Those frames which define such objects are called *primitive frames*. Frames which are not primitive are called

non-primitive frames.

Depending on what is known about the appearance of an instance, an instance can be in one of the following two states: *verified,* which indicates that the appearance of the instance is some already located image structure or is a function of the appearances of verified instances; and *hypothetical,* which indicates that the appearance of the instance has not been determined.

In addition to the appearances of objects, the HLVS also uses the iconic description of a frame during its reasoning. The iconic description specifies an area in the image and its definition is specified by a rule. During the hypotheses integration, the HLVS uses the iconic descriptions to reason whether two DE's are related (explained in Section 3). The use of iconic description in SIGMA is similar to the use of "functional areas" in Mckeown's SPAM aerial interpretation system [McKe84].

The values recorded in instances may be updated during the analysis. Every instance has a special numerical value which is called the *strength* of the instance. The method used to compute strength is described as a procedure, say $P_{strength}$, in the frame's definition. Upon instantiation, a strength is computed for each instance. Whenever the values recorded in an instance are updated, the strength of the instance is also recomputed by reevaluating $P_{strength}$. The HLVS uses such values to control its focus of attention mechanism.

Suppose one defines the appearance of a house (house frame) as a rectangular compact region and a row of houses (house-group frame) as the union of the appearances of all the houses in a house-group. Then the house frame is primitive while the house-group frame is non-primitive. In SIGMA, in order to locate a house-group, the HLVS first generates hypotheses about the location of member houses and then direct the LLVS to locate each house individually.

Now, suppose that the LLVS located a rectangular compact region, R_0. The HLVS will generate a house instance, H_1, whose appearance is R_0 and mark it as a verified instance. However, suppose the HLVS further generates neighboring house predictions for H_1, say H_2 and H_3. Both H_2 and H_3 are hypothetical instances since the appearances of these instances have not yet been determined from the image.

2.4. Representing relations between objects

A major portion of the scene domain knowledge involves relations between objects. However, these relations must be represented in forms that can be directly used by the HLVS. Our approach is influenced by production rules [Davi77] and the planning paradigm used in Garvey's vision system [Garv76].

Suppose we have the following house-road relation:

A road $road_0$ is *along* a house $house_0$ if the predicate along ($road_0$,$house_0$) is true.

There are at least two potential uses of this relation by the HLVS:

- HLVS uses the relation to check whether road $road_0$ is along house $house_0$.

- HLVS uses the relation to direct a search for a road along house $house_0$.

In order to support multiple uses of a relation by the HLVS, we use a test-hypothesize-and-act strategy to describe relations. A binary relation REL(O_1,O_2) between objects O_1 and O_2 is represented using two functional descriptions:

$$O_1 = F(O_2) \text{ and } O_2 = G(O_1).$$

Program F computes the object expected by object O_2 and is recorded in object frame O_2 as a rule. Program G computes the object expected by object O_1 and is recorded in object frame O_1 as a rule also.

As noted earlier, control knowledge for the use of relations and control knowledge for directing search are both required by the HLVS. We represent such knowledge as predicates associated with rules.

We present our rule representation scheme as follows:

A rule is composed of three parts:

<control-condition>
<hypothesis>
<action>.

<Control-condition> is a predicate. It indicates when a rule can potentially be applied. <Hypothesis> specifies the description of a desired object that is created when the <control-condition> evaluates to true. <Action> describes the code to be evaluated if <hypothesis> is verified. In general, <action> can add facts to or delete facts from the iconic/symbolic database of the HLVS.

The *house-road* relation can be written as a rule in the HOUSE frame as follows (Figure 2-3):

> To compute a road along house $house_0$, we always generate a hypothesis $road_x$ with the following slot values:
>
> road.orientation:
> greater than ($house_0$.front-of-house + 80 degrees) but less than ($house_0$.front-of-house 100 degrees).
> road.width:
> greater than ($house_0$.width * 0.3) but less than ($house_0$.width * 0.5).
> road.centroid:
> resides within REGION($house_0$.centroid + T($house_0$.front-of-house)).
>
> T(.) is a function. If the hypothesis $road_x$ is verified by some road $road_0$, then road $road_0$ is along house $house_0$.

Figure 2-4 shows a model for suburban housing developments. Objects are described by nodes (square) and relations are described by arcs. In this model, Rectangle and Picture-Boundary are the "primitive frames".

The HLVS makes use of the different parts of a rule to perform its reasoning. We discuss this in Section 4.

3. Integration of hypotheses

3.1. Introduction

Consider a binary relation REL(O_1,O_2) between two classes of objects, O_1 and O_2. This relation can be used as a constraint to recognize objects from these two classes by first extracting image structures which satisfy the specified appearances of O_1 and O_2, and then checking that the relation is satisfied by these candidate objects (Figure 3-1). In this *bottom-up* recognition scheme, analysis based on relations cannot be performed until image structures corresponding to objects are extracted.

In general, however, some of the correct image structures fail to be extracted by the initial image segmentation. So one must, additionally, incorporate *top-down control* to find image structures missed by the initial segmentation. Such top-down processes use relations to predict the locations of missing objects, as in the system described by (Garvey [Garv76], Selfridge [Self82])

As noted above, the use of relations is very different in the two analysis processes : consistency verification in bottom-up analysis and hypothesis generation in top-down analysis. An important characteristic of our hypothesis integration method is that it enables the system to integrate both bottom-up and top-down processes into a single flexible spatial reasoning process.

As will be described in Section 4, the HLVS first establishes local environments. Then, either bottom-up or top-down processes are activated depending on the nature of the local environment. The following sections describe the concepts and characteristics of this process.

3.2. The representation of database entities

All instances, hypothetical or verified, generated by the HLVS are recorded in a database. In the rest of this section, we use the term *database entity (DE)* to refer to instances recorded in the database. In addition, we use the term *hypothesis* to refer to instances in the hypothetical state.

The description of each DE consists of two parts. One part is the *iconic description*. This description is a region in the image which indicates where the DE may be located. It is generated by the rule which specifies the iconic description of the frame used to generate the DE.

The second part is the *symbolic description,* which includes the values filled into the slots of the DE, and the set of constraints imposed on these values. These constraints are represented by a set of linear inequalities in one variable (the slot name).

3.3. Consistency between a pair of DE's

"Related" DE's are integrated and analyzed together. In SIGMA, "relatedness" between DE's is defined in terms of "consistency" between pairs of DE's. A pair of DE's, DE_1 and DE_2, are said to be *consistent* if the following

conditions hold:

1) The iconic descriptions of the DE's must intersect. It is also possible to impose some requirements on the size and shape of the area of intersection.

2) The DE's are *compatible*. Let OP be the intersection arising from two DE's, and let F_1 and F_2 denote the frames from which DE_1 and DE_2 were copied. DE_1 and DE_2 are said to be *compatible* if F_1 and F_2 are linked by CAN-BE or AKO links. Otherwise, DE_1 and DE_2 are said to be *incompatible*. This will be explained in more detail in Section 3.5.

3) The constraints imposed on the attributes of the DE's must be *satisfiable*. Every DE has associated with it a set of linear inequalities in one variable that constrain the permissible values of the DE's attributes. A simple constraint manipulation system is used to check the consistency between the sets of inequalities by generating the solution space (also represented by inequalities) to the intersection of those sets. If this solution space is non-empty, then the constraints are *consistent*.

3.4. Formation of maximum consistent situations

Consistent DE's are combined into *situations*. These DE's are said to *participate* in the formation of a situation. The *P-set* of a situation is its set of participating DE's. Situation S_a is *less than* situation S_b if the P-set of S_a is a subset of the P-set of S_b. This ordering is used to structure all the situations into a *situation lattice*. Note that a single DE is also a situation. The rest of this section presents the algorithm used to form situations.

Two DE's are said to be *2-consistent* if they are consistent. In general, a set of DE's is said to be *n-consistent* if every possible subset of (n–1) of the DE's is (n–1)-consistent. Clearly, a set of DE's is n-consistent if and only if all possible pairs of DE's in the set are 2-consistent.

When a DE, say DE_{new}, is inserted into the iconic/symbolic database, the current situation-lattice is updated by first computing the set, U, that contains all DE's whose iconic descriptions intersect with the iconic description of DE_{new}. Then, we iteratively compute all lists of n-consistent DE's for those DE's in the set U. Each such list of n-consistent DE's forms the P-set of some situation. Algorithm 3-1 describes this process.

The *maximum consistent situations* are those situations which are the roots of the situation lattice.

Algorithm 3-1 : Updating the Situation Lattice

Step 1:	Suppose the newly inserted entity is DE_{new}. Compute the set U. N=2.
Step 2:	Compute the set, R, of all the N-consistent DE's for the DE's in U. Remove any which do not contain DE_{new}.
Step 3:	If R is empty, then exit. Otherwise, insert all the elements of R into the situation-lattice.
Step 4:	Increment N by 1. Construct all the pairs for elements in R. Represent each pair by the union of the members in each element. Remove any which is not N-consistent or does not contain DE_{new}. Set R to be the set of resulting N-consistent DE's.
Step 5:	Go to step 3.

Figure 3-2 shows an example of how the situation lattice is updated when a DE is inserted. Each DE is represented by a letter. A situation is represented by all the DE's in its P-set. Figure 3-2(a) shows the situation lattice before the insertion of DE_E and the iconic descriptions of the DE's. Suppose that the new DE, DE_E, is consistent with DE_A, DE_B and DE_D. The set U would then include

$$DE_A,\ DE_B,\ DE_C,\ DE_D,\ DE_E.$$

The first time that step 3 is evaluated, set R contains the following situations:

$$DE_{AE},\ DE_{BE},\ DE_{DE}.$$

The second time that step 3 is evaluated, set R contains the following situation:

$$DE_{ADE}$$

The updating stops at the third iteration. Figure 3-2(b) shows the situation lattice after the updating process.

When a DE, say DE_{remove}, is being removed from the iconic/symbolic database, the current situation lattice must also be updated. This can be done simply by removing all the situations in the situation lattice which are larger than DE_{remove}.

Suppose, for example, that DE_A is removed from the situation described in Figure 3-2(b). Figure 3-3 shows the resulting situation lattice.

It is possible that the number of situations in the situation lattice may grow exponentially. In practice, this does not happen since the number of participants in a situation is usually quite small, e.g., two or three.

3.5. Constructing the composite hypothesis

A situation is a collection of consistent DE's. The HLVS selects a situation and proposes a *composite hypothesis* which "summarizes" the constraints imposed on the attributes of all the participating DE's. The strategy for computing the composite hypothesis is specified by a procedure recorded in the frame's definition. (Note that two DE's are consistent only if they are instances of the same frame or instances of frames in the same generalization/specialization hierarchy. Therefore, all the participants in a situation must be instances of frames in the same generalization/specialization hierarchy. The procedure for computing the composite hypothesis is recorded in the most general frame.) This section presents some strategies for computing the composite hypothesis.

One simple strategy is to use the solution sets of all the constraints imposed on the attributes of all the participating DE's (explained in Section 3.4) as the constraint set of the composite hypothesis. The target object of the composite hypothesis is the most specialized object expected by all the

DE's.

Suppose that the constraint set of DE_1 is

target object = HOUSE,
house.centroid = (100,130),
230 < house.area < 300

while the constraint set of DE_2 is

target object = RECTANGULAR-HOUSE,
house.centroid = (100,130),
250 < house.area < 320,
house.region-contrast > 3.

Using this method, we generate the composite hypothesis for DE_1 and DE_2 as follows:

target object = RECTANGULAR-HOUSE,
house.centroid = (100,130),
250 < house.area < 300,
house.region-contrast > 3.

Another strategy is to take the union of all the solution sets of the constraints imposed on the attributes of all the participating DE's. Suppose, for example, that two hypotheses, DE_1 and DE_2, about a road have constraints on their starting and ending points as follows:

hypothesis DE_1,
target object = road,
road.end-points = $\{(100,100),(100,150)\}$.

hypothesis DE_2,
target object = road,
road.end-points = $\{(100,125),(100,180)\}$.

We may want to construct a road hypothesis whose constraint set is the union of these constraints on DE_1 and DE_2:

target object = road,
road.end-points = $\{(100,100),(100,180)\}$.

4. An implementation of SIGMA

4.1. Overview

The goal of SIGMA is to segment the image into image structures which correspond to the objects specified in the object model. Section 1.3 outlined the architecture of the SIGMA image understanding system. This section describes its implementation.

Figure 4-1 illustrates the different stages of the control of SIGMA. SIGMA first directs the LLVS to perform an initial segmentation of the image. A set of image structures are computed at this stage. At the second stage, the HLVS constructs partial interpretations based on the results of the initial segmentation. However, during the construction, the HLVS may direct the LLVS to compute more image structures. When all construction activities finish, SIGMA provides a query-answering module for selecting "good interpretations" and displaying the reasoning paths used to derive these interpretations. During the entire analysis, SIGMA maintains a database (the iconic/symbolic database) to record all the intermediate results generated at each stage.

The rest of this section discusses the implementation of SIGMA.

4.2. Description of goals

The Query-Answering Module (QAM) is activated by the HLVS at the end of each reasoning iteration. The goal of SIGMA is described as a query to

QAM. QAM matches the query with the interpretations already constructed. If any interpretation matches the goal, QAM enters into an answer mode and provides an interactive query-answering capability.

Suppose, for example, that the goal is to locate any road whose length is longer than 300 feet in the image and has at least two houses along it. This goal can be represented by the following query:

road(x) and (x.length > 300 feet) and (x.number-of-houses >2).

During the interpretation stage, whenever a road instance is constructed whose length is longer than 300 feet and has at least two houses along it (i.e., x is bound to some interpretation constructed by the HLVS), QAM will enter an answer mode and make the specific road instance that satisfies the goal available to an interactive program. One can use this program to traverse the *interpretation network* (the network which is constructed by the HLVS during the interpretation process), and display symbolic and iconic descriptions of the interpretations constructed.

4.3. The initial segmentation

SIGMA starts its processing by directing the LLVS to extract image structures. The schematic diagram of the initial segmentation process is shown in Figure 4-2. The set, I, which contains a list of hypotheses about primitive objects, is used to describe the goal of the initial segmentation pro-

cess.

The Initial Segmentation Controller (ISC) sequentially selects hypotheses from the set I and directs the LLVS to extract image primitives which satisfy these hypotheses. For each image primitive extracted, the ISC makes an instance of the frame of which the hypothesis is a copy, and then inserts the instance created into the iconic/symbolic database.

Suppose, for example, that we want to first extract all regions which might correspond to house groups and roads in the image. A set which contains the following hypotheses can be used as the set I:

hypothesis 1: /* extract compact and bright rectangles */
target object = rectangle,
in-window = whole image,
rectangle.elongatedness $\leq$ 10,
rectangle.compactness $<$ 18,
rectangle.region-contrast $>$ 3,
180 $<$ rectangle.area-of $<$ 400.

hypothesis 2: /* extract elongated rectangles */
target object = rectangle,
in-window = whole image,
7 $<$ rectangle.width $<$ 20,
rectangle.elongatedness $>$ 10,
rectangle.length $>$ 10,
rectangle.compactness $\geq$ 18,
rectangle.region-contrast $>$ 3.

The set I for the initial segmentation could, in principle, be computed from the scene model, since the appearances of objects are described in terms of the appearances of "primitive frames". The ISC could choose those primi-

tive frames whose appearances are salient (i.e., they can be located "easily" by the LLVS) as the I-set. However, this was not implemented in SIGMA; the I-set is simply given as part of the scene model.

4.4. Construction of partial interpretations

The schematic diagram of the processing involved in constructing partial interpretations is shown in Figure 4-3. The HLVS iterates the following steps in this stage:

(1) hypothesis generation,
(2) focus of attention,
(3) composite hypothesis construction,
(4) solution generation,
(5) action scheduling.

Detailed discussions of each step are presented in the following subsections.

4.4.1. Hypothesis generation

For each DE (hypothetical or verified) recorded in the iconic/symbolic database, the Iconic/Symbolic Database Manager (ISDM) evaluates all the rules that are "applicable".

Suppose I_0 is an instance of frame F. For each rule, say R_x, defined in frame F, the ISDM evaluates the <control-condition> part of rule R_x. If the evaluation result is true, the ISDM performs the following tasks:

(1) Compute the <hypothesis> part of rule R_x, and insert the computed hypothesis into the iconic/symbolic database.

(2) Insert the <action> part of rule R_x into the *Action List* which records all the actions waiting to be evaluated.

The actions in the action list are called *delayed actions*. For each delayed action, there is an associated hypothesis (computed at step 1) recorded in the iconic/symbolic database. Such a hypothesis is called the *cause of delay* of the action.

Note that for rules whose <hypothesis> part is nil, the <action> part is not put into the action list. Instead, the <action> is evaluated immediately. At the hypothesis generation stage, the ISDM evaluates, for each instance in the iconic/symbolic database, the <control condition> of every rule in the associated frame definition. (This strategy is not efficient. A more efficient strategy would evaluate only those <control condition> s whose values are affected by changes made to the attributes of the instance since the last time the <control condition>s were evaluated.)

The DE's in the iconic/symbolic database are combined into situations. All the situations are structured into the situation lattice. The Situation Lattice Database Manager (SLDM) updates the situation lattice whenever DE's are inserted into or removed from the iconic/symbolic database. The algorithm (3-1) for updating the situation lattice was presented in Section 3.4.

"Identical instances" may be created during the construction process of the HLVS. Two instances are *identical* if all the values filled in the slots of those instances are identical. It is necessary to detect identical instances and replace them by a single instance. This process is called *unification of instances,* and is performed during construction of composite hypotheses.

For example, a house group instance containing house instances H_0 and H_1 can be constructed from instance H_0 by first constructing a house group instance, say HG_0, which contains H_0 , and then expanding HG_0 to include house instance H_1 (see Figure 4-4(a)). An identical house group instance HG_1 can also be constructed from house instance H_1 (see Figure 4-4(b)).

One natural way to detect identical instances is to examine the P-set of a situation. For each situation selected by the focus of attention mechanism, the HLVS examines the instances in the P-set of the situation to find sets of identical instances.

The HLVS unifies identical instances as follows. All identical instances are first collected in a set, L. Then the HLVS selects one instance from the set L, say I_0. For each instance $I_x \in L$, the HLVS replaces every reference to I_x in the iconic/symbolic database by a reference to instance I_0.

Figure 4-5 illustrates the result of unifying HG_0 and HG_1 (assuming the HLVS chooses HG_0 as I_0).

4.4.2. Focus of attention

The focus of attention mechanism selects a situation with greatest strength from the situation lattice. If there are several situations with equal strength, the HLVS selects one arbitrarily.

For example, Figure 4-6 shows a situation lattice. There are two maximal consistent situations that can be selected (both situations have strength = 3). The HLVS can select either one (i.e., N_{10}, or N_{11}).

The situation selected by the focus of attention mechanism is given to the Composite Hypothesis Constructor to construct the composite hypothesis. The construction of composite hypotheses was discussed in Section 3.5.

4.4.3. Solution generation

The Solution Generator (SG) computes solutions for the composite hypothesis. The SG obtains/constructs instances to satisfy the composite hypothesis by one of the methods discussed in the following paragraphs.

First, the SG may discover an existing instance in the iconic/symbolic database that satisfies the composite hypothesis. In this case, the SG returns the instance found as the solution. In general, it may be necessary to search the iconic/symbolic database to find some instance which satisfies the composite hypothesis. However, since the composite hypothesis is constructed by taking the solution space of all the constraints imposed on the DE's participating in the situation (see Section 3.5), to find an existing instance which

satisfies the composite hypothesis, the SG needs only examine the P-set of the selected situation and use any instance in the P-set as the solution.

Suppose the SG cannot find any instance in the iconic/symbolic database that satisfies the composite hypothesis. There are two possibilities:

> (1) the target object of the composite hypothesis is a primitive object (such hypotheses are called *primitive hypotheses);*
>
> (2) the target object of the composite hypothesis is not a primitive object (such hypotheses are called *non-primitive hypotheses).*

In the first case, the SG first directs a top-down segmentation by providing to the LLVS the descriptions of the composite hypothesis. Then the SG creates instances based on the results of the LLVS. Finally, the instances created (if any) are returned as a solution.

In the second case, no top-down segmentation is performed. The SG simply returns the composite hypothesis as the solution.

4.4.4. Action scheduling

The Action Scheduler (AS) schedules the actions in the action list using the solution provided by the SG. Three possible types of solutions may be provided:

(1) nil,i.e., the hypothesis cannot be verified,
(2) an instance,
(3) a composite hypothesis.

In both the first and the second cases, the AS selects those <action>s in the action list whose "causes of delay" are in the P-set of the selected situation. Let the solution be I_0, the actions selected be $A_1, \ldots, A_n$, and their causes of delay be $H_1, \ldots, H_n$, respectively. The AS performs the selected actions sequentially:

(a) replace all the references to H_i in action A_i by I_0,

(b) evaluate A_i,

(c) remove H_i from the iconic/symbolic database, or update the attributes of H_i (we will discuss this in more detail in Section 4.5).

In the third case, the AS marks the composite hypothesis, say CH_0, as *partially processed* and inserts it into the iconic/symbolic database. The AS also marks the currently selected situation, say S_0, as *unconcluded.* The hypothesis CH_0 is said to be *derived from* the situation S_0. We will present a more detailed discussion of the effects of such processing in Section 4.4.4.1. Table 4-1 summarizes the terms defined in the previous paragraphs.

The removal of hypotheses from the iconic/symbolic database has the following side effects:

(1) If a hypothesis, say H_0, is removed from the database, then all the

Table 4-1. Glossary.

Primitive hypothesis:
A hypothesis whose target object is a primitive object.

Non-primitive hypothesis:
A hypothesis whose target object is a non-primitive object.

Unconcluded situation:
A situation which was selected by the focus of attention mechanism, but for which the Solution Generator cannot yet compute a solution.

Partially processed hypothesis:
A composite hypothesis, recorded in the iconic/symbolic database, which is computed for some unconcluded situation.

situations in the situation lattice whose P-sets contain H_0 are also removed from the situation lattice.

(2) If an unconcluded situation is removed from the situation lattice in (1), then the hypotheses which were derived from the situation are also removed from the iconic/symbolic database.

The updating of attributes of hypotheses is implemented by removing the original hypothesis and inserting a new hypothesis.

When all the actions selected are evaluated, the action scheduler terminates, and the next cycle of hypothesis construction begins.

4.4.4.1. Computing solutions for a non-primitive composite hypothesis

The SG does not directly propose solutions for a non-primitive composite hypothesis. Instead, a top-down parsing approach is used to compute the solution. Suppose the composite hypothesis constructed for a situation , say S_0, is CH_a. To compute the solution for CH_a, we first generate a set of hypotheses $H_i, 1 \leq i \leq n$ and compute the solution for each H_i. The solution for CH_a can be computed from the solutions for $H_i, 1 \leq i \leq n$.

To support such an approach, we associate with each non-primitive frame a *decomposition strategy* (represented as a rule) which describes how to generate a new set of hypotheses to be verified, and how to compute a solution for the non-primitive frame using the solutions for the generated hypotheses.

For example, the rule for the decomposition strategy of a RECTANGULAR-HOUSE frame is

Rule $R_{first\text{-}order\text{-}properties}$:
<control-condition> : true,
<hypothesis> :
  $H = F_0(RECTANGLE, self)$,
<action> :
  if H=nil then conclude(nil)
  else conclude(make-instance(RECTANGULAR-HOUSE,H)).

This rule indicates that a RECTANGULAR-HOUSE instance can be created

if a RECTANGLE instance which satisfies the attributes specified by F_0 is created.

As discussed in Section 4.4.4, the Action Scheduler (AS) marks the non-primitive composite hypothesis as partially processed and inserts it into the iconic/symbolic database. The AS also marks the situation selected as unconcluded. Partially processed hypotheses and unconcluded situations are processed by other modules of the HLVS in the following ways:

(1) If a situation, say S, is marked as "unconcluded", then all the situations in the situation lattice which are less than S are also marked as unconcluded. The focus of attention mechanism does not select any unconcluded situation. This strategy is based on the observation that if no conclusion can be drawn from the analysis of a situation, say S, then the analysis of all the situations which are "less than" S (i.e., composed of a subset of the instances of S) can be postponed.

For example, by marking situation N_{10} in Figure 4-6 as unconcluded, all the situations that are less than N_{10} are also marked as unconcluded (i.e., $N_i, H_i, 1 \leq i \leq 3$).

(2) The function "conclude" indicates that a solution, say I_{sol}, has been computed for an unconcluded situation, say S. Whenever this function is evaluated, the HLVS schedules S as the situation to be selected in the next iteration cycle and the solution proposed for the composite hypothesis of this situation is I_{sol}.

(3) Since a partially processed hypothesis, say H, is the composite hypothesis constructed for some unconcluded situation, S, H should not participate in the formation of new situations with any DE's in the P-set of S. HLVS uses the more efficient strategy of not allowing a partially processed hypotheses to participate in the formation of *any* situations.

(4) In the hypothesis generation process, only the rules which describe the decomposition strategy can be evaluated for partially processed hypotheses.

All the hypotheses generated are inserted into the iconic/symbolic database.

(5) The removal of a partially processed hypothesis from the iconic/symbolic database causes the removal of all the hypotheses in the database which are generated by the decomposition strategy.

Suppose, for example, that the situation N_{10} shown in Figure 4-6 is selected by the focus of attention mechanism and the composite hypothesis constructed, say CH_a, is:

target object : RECTANGULAR-HOUSE;
...

Since RECTANGULAR-HOUSE is not a primitive frame, the SG returns CH_a as the solution to the situation N_{10}. The AS marks N_{10} as unconcluded and inserts CH_a into the iconic/symbolic database.

At the subsequent hypothesis generation process, CH_a activates the rule $R_{first\text{-}order\text{-}properties}$ in the RECTANGULAR-HOUSE frame and creates hypothesis H_9:

target object : RECTANGLE;
...

Figure 4-7 shows the relation between CH_a and H_9 and the action which is delayed by H_9. The resulting situation lattice is shown in Figure 4-8.

Suppose a RECTANGLE instance, say I_R, is proposed to H_9 by the SG. The AS evaluates the action whose cause of delay is H_9 and:

(1) creates a RECTANGULAR-HOUSE instance, say I_{RH},

(2) evaluates the function "conclude".

The evaluation of the function "conclude" indicates to the HLVS that situation N_{10} is to be scheduled in the next iteration cycle and the solution proposed for CH_a is I_{RH}.

At the next iteration, the SG proposes I_{RH} to the hypotheses in the P-set of N_{10} (i.e., H_1, H_2, H_3). Those actions whose causes of delay are H_1, H_2, and H_3 are now evaluated by the Action Scheduler. Suppose H_1, H_2, and H_3 are removed after the evaluation of these actions. Figure 4-9 shows the resulting situation lattice. Note that this is usually the case when an appropriate solution is proposed to the hypotheses.

The processing of partially processed hypotheses and unconcluded situations are summarized in Table 4-2.

4.5. A taxonomy of actions

In this section, we discuss a taxonomy of the actions that are often used to specify the scene domain knowledge. The term action in this section refers to the activities described in the <action> part of a rule.

One type of action is the filling in of attributes of an instance. For example, a rule in the HOUSE-GROUP frame is:

Table 4-2. Summary.

Unconcluded situation:
- Will not be selected by the focus of attention mechanism.
- If a solution is proposed by the SG for some unconcluded situation, the HLVS schedules that situation in the next iteration cycle.

Partially processed hypothesis:
- A composite hypothesis for some unconcluded situation.
- Recorded in the iconic/symbolic database.
- Does not participate in the formation of any situations.
- Removal of a partially processed hypothesis, *H*, causes the removal of all the hypotheses generated by *H*.

<control-condition> : true
<hypothesis> : H = AR(self,ROAD),
<action> : self.along-road = H.

This rule specifies that if a ROAD instance which satisfies H is found, fill it in the slot "along-road" of the HOUSE-GROUP instance.

In addition to filling in attributes, actions often create new instances or unify several instances (as described in Section 4.4.1). Such actions are described by two functions:

- "make-instance" : create an instance and insert it into the iconic/symbolic database;

- "unify-instance" : unify a list of instances in the iconic/symbolic database into a single instance.

For example, a rule in the RECTANGLE frame is:

<control-condition> : IS-RECT-HOUSE(self)
<hypothesis> : nil,
<action> :
make-instance(RECTANGULAR-HOUSE,F(self)).

This rule describes the following piece of knowledge:

"If a RECTANGLE instance which satisfies the IS-RECT-HOUSE criteria is created, then create a RECTANGULAR-HOUSE instance using function F and insert it into the iconic/symbolic database."

Similarly, the following piece of knowledge:

"If more than one HOUSE-GROUP instance is filled in the "belongs-to" slot of a HOUSE instance, replace it by another HOUSE-GROUP instance which is created by the function COMBINE-H."

can be described by the following rule in the HOUSE frame:

<control-condition> :
if number-of-elements(self.belongs-to) > 1,
<hypothesis> : nil,
<action> :
unify-instance(self.belongs-to,COMBINE-H(self.belongs-to)).

Another class of actions deals with the removal of hypotheses and the updating of the attributes of hypotheses. Usually, hypotheses are removed by the Action Scheduler after the Solution Generator proposes solutions to them. However, instead of always removing hypotheses when no acceptable solution is found, we may want to update the attributes of the original hypotheses when more information is available. The function "update" is used to describe the updating of the attributes of a hypothesis.

For example, consider the following rule:

<control-condition> : ...
<hypothesis> : H = F(self)
<action> :
    if H = nil then update(H,CS_1)
    else ...

The action specifies that if the solution proposed for H is nil, then the AS replaces some attributes of hypothesis H by CS_1. However, H is not removed from the iconic/symbolic database. The <action> part is inserted again into the action list (its cause of delay is H.)

There is yet another category of actions which specifies the constraints on the evaluation of multiple rules. We describe this type by an example.

Any instance of a HOUSE-GROUP frame can be "along" at most one ROAD instance. Given a HOUSE-GROUP instance, say I_{HG}, we may not yet know the location of the road along I_{HG} ,i.e., at location F_l or at location F_r (see Figure 4-10). One strategy is to create hypotheses about a ROAD at

both locations. However, once one hypothesis is verified, the other hypothesis must be removed.

The above knowledge is represented as follows:

Rule R_1.
<control-condition> : true,
<hypothesis> : $H_1 = F_l$(self),
<action> : self.along-road = H_1,

Rule R_2.
<control-condition> : true
<hypothesis> : $H_2 = F_r$(self),
<action> : self.along-road = H_2.

In addition, the following rule for the HOUSE-GROUP frame constrains the simultaneous evaluation of R_1,R_2:

Rule $R_{control}$.
<control-condition> :
not-null(anyone(R_1,R_2)),
<hypothesis> : nil,
<action> :
remove-all(anyone(R_1,R_2)).

where anyone(R_1,R_2)=
if is-evaluated(R_1) then R_2
else if is-evaluated(R_2) then R_1
else nil

The above rule specifies that whenever one of the <action> parts of the rules R_1 or R_2 is evaluated, rule $R_{control}$ is evaluated which causes the removal of all the hypotheses that are created by the evaluation of R_1.<hypothesis>

or R_2.<hypothesis>.

Suppose a HOUSE-GROUP instance is created. The instance activates rules R_1 and R_2 and generates two hypotheses about the ROAD object. Whenever the SG proposes a ROAD instance to one of the hypotheses, the AS evaluates one of the delayed actions, and causes the removal of the other hypothesis.

We summarize the actions discussed in this section in Table 4-3.

4.6. Pursuing alternative hypotheses

It is possible that several hypotheses may be generated at the same time. This can be represented as the following rule:

Table 4-3. A taxonomy of actions

Action Type	Example
Attributes	Filling in of attributes in an instance.
Instances	Create instances. Unify instances.
Hypotheses	Remove hypotheses. Update hypotheses.
Rules	Constrain the simultaneous evaluation of several rules.

if <control-condition> then
    <hypothesis 1> <action 1>
or
    <hypothesis 2> <action 2>
or
    <hypothesis n> <action n>

Whenever <control-condition> evaluates to true, all of the <hypothesis>s can be generated. These hypotheses are called *alternative hypotheses* and we assume that at most one of the hypotheses is in fact true. However, it is difficult to decide which one should be pursued first, since a promising selection may turned out to be incorrect as new facts (generated by resegmentation) are obtained.

In SIGMA, all the alternative hypotheses are generated and participate in the hypothesis integration process. However, the associated actions of these alternative hypotheses are not evaluated (put in the delayed-action queue). When any one of the alternative hypotheses is verified, it is left to the associated action to decide whether other alternative hypotheses should be pruned. On the one hand, this strategy allows multiple alternative hypotheses to be pursued simultaneously. On the other hand, expert domain knowledge, which can be described in a rule, can be used to prune unpromising hypotheses when enough facts are known.

4.7. The selection of good interpretations

Potentially, SIGMA could construct all possible interpretations for the image. It is natural to require that no region be interpreted as two different objects in the scene model. However, in SIGMA, a region may be interpreted as several objects (e.g., an elongated region might be interpreted both as a road or a driveway). Intersecting image structures may be used to construct DE's whose iconic descriptions should never intersect. A pair of DE's whose iconic descriptions intersect while the scene model specifies otherwise are called *conflicting* DE's. The associated interpretations are called *alternative* interpretations.

For a set of conflicting DE's, we need to select a DE which "best" interprets the image. It is possible to design an algorithm to select such "best" interpretations. However, we did not investigate this issue in SIGMA. Instead, we model the final selection process as a database query answering process. A program (QAM) was developed to answer simple queries about DE's in the interpretation network and to display the iconic descriptions of the DE's selected.

5. Examples

5.1. Introduction

This section presents detailed examples of the application of SIGMA to the analysis of a high resolution aerial image to locate houses, roads, and driveways in a suburban scene.

We first present an example of the initial segmentation process. Then we discuss how the HLVS analyzes the image in several typical situations. Finally, we show the results of analysis by SIGMA on an aerial image.

5.2. Initial segmentation

The image used in the example is a 250 * 140 window of an aerial image (Figure 5-1) with intensities in the range of 0 to 63. The scene contains houses, roads, and driveways.

5.2.1. Initial segmentation goals

We want to locate houses and roads in the image. Since their appearances are either compact rectangles or elongated rectangles, and they are usually brighter than the background, the following hypotheses are used as the I-set of the initial segmentation process:

/* extract compact and bright rectangles */
hypothesis H_{blob}:
target object = rectangle,
in-window = whole image,
rectangle.elongatedness $\leq$ 10,
rectangle.compactness $<$ 18,
rectangle.region-contrast $>$ 3,
180 $<$ rectangle.area-of $<$ 360.

/* extract bright and elongated rectangles */
hypothesis H_{ribbon}:
target object = rectangle,
in-window = whole image,
8 $<$ rectangle.width $<$ 20
rectangle.elongatedness $>$ 10,
rectangle.length $>$ 10,
rectangle.compactness $\geq$ 18,
rectangle.region-contrast $>$ 3.

5.2.2. Verifying hypothesis H_{blob}

The Initial Segmentation Controller (ISC) first selects hypothesis H_{blob}. The ISC activates the LLVS to compute image primitives that satisfy hypothesis H_{blob}. The LLVS selects the following segmentation operators arranged in descending order of their priorities as follows:

Blob finder
Upper threshold method

The Ribbon finder and the Lower threshold method are not selected since their selection criteria evaluate to false.

The LLVS activates the Blob finder first. The Blob finder first convolves the original image with a Laplacian operator. Then it computes the positive connected regions in the convolved image (Figure 5-2). The regions computed by the Blob finder which satisfy the constraints of H_{blob} are shown in Figure 5-3.

Since the set of results computed by the Blob finder is not empty, the LLVS returns the computed regions to the HLVS. The Upper threshold method is not evaluated.

5.2.3. Verifying hypothesis H_{ribbon}

The ISC then selects hypothesis H_{ribbon}. The ISC activates the LLVS to compute regions which satisfy hypothesis H_{ribbon}. The segmentation operators selected by the LLVS for this task arranged in descending order of their priorities are as follows:

Ribbon finder
Upper threshold method

The Blob finder and the Lower threshold method are not selected since their selection criteria evaluate to false.

The LLVS activates the Ribbon finder first. The Ribbon finder first computes the skeletons of the positive regions shown in Figure 5-2. The resulting skeletons are shown in Figure 5-4.

Finally, the Ribbon finder decomposes these skeletons and computes the skeletons for the ribbons. Figure 5-5 shows the skeletons of the ribbons computed by the Ribbon finder which satisfy the constraints of hypothesis H_{ribbon}.

Since the set of results computed by the Ribbon finder is not empty, the LLVS returns the computed regions to the HLVS. The Upper threshold method is not evaluated.

5.2.4. Generating instances

The ISC collects the results computed by the LLVS, creates RECTANGLE instances, and inserts them into the iconic/symbolic database.

There are 26 RECTANGLE instances created at this stage. Figure 5-6 shows the iconic descriptions of these instances. Note that some of the instances intersect.

5.3. Constructing partial interpretations

A situation is classified into one of the following classes based on how the Solution Generator computes its proposed solution:

Case 1: The SG discovers an existing instance in the iconic/symbolic database which satisfies the given composite hypothesis.

Case 2: The SG cannot find any instance in the iconic/symbolic database which satisfies the given composite hypotheses. The composite hypothesis is non-primitive.

Case 3: The SG cannot find any instance in the iconic/symbolic database which satisfies the given composite hypothesis. The composite hypothesis is primitive.

Case 4: The SG obtains the solution from the previous iteration (i.e., the solution for an unconcluded situation is now computed.)

5.3.1. Case 1--Discovering an existing instance

Consider the situation shown in Figure 5-7. The relations between the DE's shown in this figure are described in Table 5-1.

Figure 5-8 shows the portion of the interpretation-network which is related to this situation.

Assume the focus of attention mechanism selects situation S_1 whose P-set is as follows:

$$\left\{DE_1, DE_3, DE_7\right\}.$$

Suppose the composite hypothesis, say CH_a, computed for S_1 is :

target object = ROAD,
...

Since the P-set of the situation S_1 contains an instance, DE_7, the SG proposes it as the solution to the composite hypothesis constructed for this situation.

The AS activates those actions whose causes of delay are DE_1 and DE_3 respectively. Figure 5-9 shows the resulting interpretation network. Note that hypotheses DE_2 and DE_4 are removed. This is caused by a control rule in the HOUSE-GROUP frame which specifies that each HOUSE-GROUP instance can be along at most one road.

5.3.2. Case 2--Decomposing a hypothesis

Consider the situation shown in Figure 5-10. The relations between the DE's shown in this figure are described in Table 5-2.

Figure 5-11 shows a portion of the interpretation network related to this situation.

Assume the focus of attention mechanism selects the situation S_1 whose P-set is

$$\{DE_1, DE_2\}.$$

Assume the composite hypothesis, say CH_a, computed for S_1 is

target object = DRIVEWAY,
...

The SG cannot find any existing instance that satisfies CH_a. Since CH_a is non-primitive, the AS marks it as partially processed and inserts it into the

iconic/symbolic database.

At the subsequent iterations, CH_a activates the rule $R_{first-order-properties}$ of frame DRIVEWAY to generate hypothesis DE_3:

databaseentity DE_3:
target object : RECTANGLE,
...
end–database–entity.

Suppose the action which is delayed by DE_3 is $A_{first-order-properties}$. We will revisit this example in Section 5.3.4. Note that DE_3 can participate in the formation of situations with other DE's in the iconic/symbolic database. Figure 5-12 shows the resulting interpretation network after DE_3 and CH_a are inserted into the iconic/symbolic database. Note that CH_a is marked as partially processed hypothesis. Table 5-3 summarizes the relations between the DE's, action $A_{first-order-properties}$, and S_1.

5.3.3. Case 3--Directing the segmentation

Suppose the composite hypothesis, say CH_a, given to the SG is primitive. The SG activates the LLVS to compute regions which satisfy the constraints provided by the SG. The regions computed by the LLVS are used by the SG to create RECTANGLE instances. The SG then proposes those created instances which satisfy the constraints of CH_a as solutions. If no instance is computed, the SG proposes nil as the solution. We illustrate the process used

by our system in the following two examples.

Suppose the composite hypothesis, say CH_a, given to the SG is:

target object = RECTANGLE,
in window : W_1,
rectangle.elongatedness $\leq$ 10,
rectangle.compactness $<$ 18,
275 $<$ rectangle.area-of $<$ 325.

The window W_1 is shown in Figure 5-13.

The LLVS first activates the Blob finder and fails to compute any region. Then the LLVS activates the Upper threshold method to compute regions. A region is successfully computed by setting the threshold value at 24. Figure 5-14 shows some of the intermediate results of the segmentation process. The measurements (the area and the compactness of a region) are shown for the largest region extracted at each specified threshold value.

The LLVS returns the computed region to the SG. The SG checks the features of the region and creates a RECTANGLE instance DE_{RECT} and propose it as the solution. Figure 5-15 shows the RECTANGLE instance created by the SG.

Suppose the composite hypothesis CH_a is again given to the SG. However, the window W_1 is as shown in Figure 5-16.

The LLVS activates the Blob finder, the Upper threshold method, and the Lower threshold method and cannot compute any region which satisfies

the given constraints. The LLVS returns "nil" to the SG. The SG then proposes nil as the solution.

5.3.4. Case 4--Analyzing an unconcluded situation

Consider the interpretation network shown in Figure 5-12. Suppose that at some other iteration the SG computes a solution, say I_0, for DE_3. Action $A_{first\text{-}order\text{-}properties}$ is now evaluated by the AS.

Two possible outcomes can be produced by the evaluation of $A_{first\text{-}order\text{-}properties}$. First, the evaluation of action $A_{first\text{-}order\text{-}properties}$ generates a solution, say I_1, for CH_a. This causes the HLVS to analyze the unconcluded situation S_1 in the next iteration. The SG will propose I_1 as the solution to CH_a, the composite hypothesis of S_1.

Figure 5-17 shows the resulting interpretation network in this case. The unconcluded situation S_1, the partially processed hypothesis CH_a, and the hypothesis DE_3 generated by the "decomposition method" are all removed.

Second, suppose no solution is generated by the evaluation of $A_{first\text{-}order\text{-}properties}$. Instead, the evaluation cause changes to be made to the attributes of DE_3. In this case, situation S_1 is removed from the situation lattice and new situations are constructed. Suppose DE_{3a} is the updated hypothesis. Figure 5-18 shows the resulting interpretation network in this case.

5.4. A complete example

In this section, we present the result of applying our image interpretation program to the image shown in Figure 5-1. No explicit goal is given to the system. The analysis terminates when all the hypotheses created are verified or refuted.

Figure 5-6 shows the RECTANGLE instances generated by the initial segmentation process. Figure 5-19 shows those RECTANGLE instances which are interpreted as RECTANGULAR-HOUSE instances (requiring that 200<rectangle.area-of<400) , and Figure 5-20 shows those RECTANGLE instance which are interpreted as VISIBLE-ROAD-PIECE instances (requiring that 6<rectangle.width<12). No RECTANGLE instances are interpreted as DRIVEWAY instances.

Instead of showing the processing of each situation by the program, we show only the processing of several interesting situations.

In the scene model, two HOUSE-GROUP instances are identical if they both share a common HOUSE instance and should be unified to a single instance. Figure 5-21(a) shows such an example. Let P_1 and P_2 denote two HOUSE instances, R_1 and R_2 two HOUSE-GROUP instances, and DE_i a HOUSE hypothesis.

Each HOUSE-GROUP instance creates hypotheses about more houses that belong to it. The process to unify the house groups is as follows:

(1) The situation whose P-set is

$$\{DE_1, P_2\}$$

is selected by the focus of attention mechanism.

(2) SG proposes HOUSE instance P_2 as the solution to the composite hypothesis of situation S_1. The evaluation of the action which is delayed by DE_1 fills P_2 in the "contains" slot of HOUSE-GROUP instance R_1.

(3) Since P_1 "belongs to" two HOUSE-GROUP instances at the subsequent iteration, the evaluation of a rule in HOUSE frame unifies R_1 and R_2.

Let us denote the resulting HOUSE-GROUP instance by R_1. Figure 5-22 shows the result of the analysis.

Figure 5-23 shows another example. Resegmentation of the image is required in this example. Let R_i denote a HOUSE-GROUP instance, P_i a HOUSE instance, DE_i a HOUSE hypothesis. Also let CH_i denote a partially processed hypothesis, and T_1 a RECTANGLE instance. These DE's are not shown in Figure 5-23. They are used later in this example.

The processes to activates the LLVS to process the image are as follows:

(1) Situation S_1 whose P-set is

$$\{DE_1, DE_3\}$$

is selected. Since the composite hypothesis (target object is HOUSE object) is non-primitive, a partially processed hypothesis, say CH_1, is generated.

(2) At the next iteration, the evaluation of the rule $R_{specialization-strategy}$ of the HOUSE frame generates a hypothesis DE_5 whose target object is RECTANGULAR-HOUSE (Figure 5-24(a)).

(3) Situation S_2 whose P-set contains DE_5 is selected. Again, a partially-processed hypothesis, say CH_2, about RECTANGULAR-HOUSE is generated.

(4) At the following iteration, the evaluation of the rule $R_{first-order-properties}$ of RECTANGULAR-HOUSE frame generates a hypothesis DE_6 whose target object is RECTANGLE (Figure 5-24(b)).

(5) The SG activates the LLVS to segment the image. A region is computed by the LLVS (see Figures 5-13, 14, 15). The SG creates a RECTANGLE instance T_1.

(6) The evaluation of the <action> of $R_{first-order-properties}$ creates a RECTANGULAR-HOUSE instance P_4. Since a solution is now ready for the unconcluded situation S_2, the HLVS schedules it to be processed next. Afterwards, since a solution is now ready for the unconcluded situation S_1, the HLVS schedules it to be processed next. Now, the actions delayed by DE_1 and DE_3 can be evaluated. The resulting interpretation network is shown in Figure 5-24(c).

(7) P_4 "belongs to" two HOUSE-GROUP instances. At the subsequent iteration, the evaluation of a rule in the HOUSE frame unifies R_1 and R_2.

Figure 5-25 shows the resulting HOUSE-GROUP instance.

In the scene model, every ROAD instance is smoothly extended from one ROAD-TERMINATOR instance to another ROAD-TERMINATOR instance. A ROAD-TERMINATOR is defined to be the boundary of the image. We present an example in the following paragraphs.

The extension of ROAD instances is similar to the merging of two HOUSE-GROUP instances discussed above. Figure 5-26 shows two ROAD instances R_1 and R_2. P_1 and P_2 are two ROAD-PIECE instances. DE_i denotes a ROAD-PIECE hypothesis. The extending of ROAD instance R_1 activates the merging of R_1 and R_2 into one ROAD instance (Figure 5-27).

Figure 5-28 shows another case. R_1 and R_2 are two ROAD instances. DE_1 is a ROAD-PIECE hypothesis generated by R_1. Since R_2 is not "connected" to R_1, hypothesis DE_1 is modified as shown in Figure 5-29.

Figure 5-30 shows yet another case. Road instance R_1 cannot be extended any longer. When this is detected, the original ROAD-PIECE hypothesis is removed and a ROAD-TERMINATOR hypothesis is generated.

Figure 5-31 shows another example. Let DE_r denote a ROAD instance, DE_h a HOUSE instance, DE_{re} a RECTANGLE instance, and DE_i a DRIVE-WAY hypothesis. House instance DE_h and ROAD instance DE_r create hypotheses DE_1 and DE_2 about the DRIVEWAY object respectively. There is no DRIVEWAY instance in the iconic/symbolic database which satisfies these hypotheses. However, there is a RECTANGLE instance, DE_{re}, which, if interpreted as a DRIVEWAY object, would satisfy these hypotheses. Note that DE_{re} is not interpreted as a DRIVEWAY object, a VISIBLE-ROAD-PIECE, or a RECTANGULAR-HOUSE since there are not enough distinguishing features of DE_{re} to make these interpretations.

The HLVS performs the analysis as follows:

(1) A composite hypothesis, CH_1, is first constructed for the situation whose P-set is

$$\left\{DE_1, DE_2\right\}.$$

(2) A hypothesis, DE_3, about the RECTANGLE object is created by the composite hypothesis CH_1.

(3) DE_{re} satisfies DE_3. A DRIVEWAY instance DE_{dr} is created by the <action> part of the rule $R_{first-order-properties}$ of the DRIVEWAY frame. The DRIVEWAY instance DE_{dr} satisfies both DE_1 and DE_2. Figure 5-32 shows the resulting interpretation network after DE_1 and DE_2 are removed.

The resulting interpretation network is shown in Figure 5-33. The iconic descriptions of the instances created during the analysis are shown in Figures 5-34 and 5-35.

Finally, we present two examples of the final selection stage of the program. Figure 5-36(a) shows a ROAD instance whose length is longer than 100. Instances of related objects are shown in Figure 5-36(b),(c), and(d).

Figure 5-37(a) shows a HOUSE-GROUP instance with more than four houses. Instances of related objects are shown in Figure 5-37(b) and (c).

6. Conclusions

This paper has described a model for the development of image understanding systems that involves representing scene domain knowledge using frames and controlling the actions of the system by hypothesis integration. Using such a framework, we developed a flexible image understanding system called SIGMA which performs both top-down(goal-oriented) image analysis and bottom-up construction of composite image structures, and demonstrated the system's performance on an aerial image of a suburban scene.

Developing computer systems for visual applications is one way to investigate how humans see, and also to make computers more useful. As pointed out by many researchers [Hall79], [Binf82], image analysis systems usually consist of several types of modules: low level vision modules(e.g., segmentation) and high level vision modules(e.g., matching, inference). This research leads to the conclusion that a powerful vision system should rely on a balance of performance between these two types of modules. The low level modules should provide descriptive information about the image to the high level modules and the high level modules should provide "hints" about image structures to the low level modules. This research is only a small step toward the construction of general vision systems.

REFERENCES

[Ball82]

D. H. Ballard and C. M. Brown, *Computer Vision,* Prentice-Hall, Inc., Englewood Cliffs, N.J., 1982.

[Barr76]

H. G. Barrow and J. M. Tenenbaum, *MSYS:A system for reasoning about scenes,* Tech. Note 121, SRI International, Menlo Park, CA, April 1976.

[Barr81]

H. G. Barrow and J. M. Tenenbaum, Computational vision, *Proc. of the IEEE* **69**, 5, pp. 572-595, May 1981.

[Binf82]

T. O. Binford, Survey of model-based image analysis systems, *The International Journal of Robotics Research* **1**, 1, pp. 18-64, Spring 1982.

[Boll76]

R. C. Bolles, *Verification vision within a programmable assembly system,* Memo AIM-295, Stanford Artificial Intelligence Laboratory, Dec. 1976.

[Davi77]

R. Davis and J. King, An overview of production systems, in *Machine Intelligence 8:Machine representations of knowledge,* ed. Elcock and Michie, Wiley, NY., 1977.

[Garv76]

T. D. Garvey, *Perceptual strategies for purposive vision,* Tech. Note 117, SRI International, Menlo Park, CA, Sept. 1976.

[Gold83]

A. Goldberg and D. Robson, *Smalltalk-80, the Language and its Implementation,* Addison Wesley, Reading, MA., 1983.

[Hall79]

E. L. Hall, *Computer Image Processing and Recognition,* Academic Press, New York, 1979.

[Hans78]

A. R. Hanson and E. M. Riseman, VISIONS: A computer system for interpreting scenes, pp. 303-333 in *Computer Vision Systems,* ed. A. Hanson and E. Riseman, Academic Press, New York, 1978.

[Hend79]

G. G. Hendrix, Encoding knowledge in partitioned networks, pp. 51-92 in *Associative Networks: Representation and Use of Knowledge by Computers,* ed. N. V. Findler, Academic Press, New York, 1979.

[Hwan84]

S. S. Vincent Hwang, Evidence accumulation for spatial reasoning in aerial image understanding, Ph. D. thesis, University of Maryland, College Park, Dec. 1984.

[McKe84]

D. M. McKeown, Jr., W. A. Harvey, and J. McDermott, *Rule based interpretation of aerial imagery,* Proc. of IEEE Workshop on Principles of Knowledge-Based Systems, Denver, Colorado, Dec. 1984.

[Mins75]

M. Minsky, A framework for representing knowledge, pp. 211-277 in *The Psychology of Computer Vision,* ed. P. H. Winston, McGraw Hill, New York, 1975.

[Moor79]

R. C. Moore, *Reasoning about knowledge and action,* Tech. Note 191, AI Center, SRI International, Menlo Park, CA, 1979.

[Naga80]

M. Nagao and T. Matsuyama, *A Structural Analysis of Complex Aerial Photographs,* Plenum Press, New York, 1980.

[Nazi84]

A. M. Nazif and M. D. Levine, Low level image segmentation:an expert system, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **6**, pp. 555-577, Sept. 1984.

[Nils80]

N. J. Nilsson, *Principles of Artificial Intelligence,* Tioga, Palo Alto, CA, 1980.

[Self82]

P. G. Selfridge, Reasoning about success and failure in aerial image understanding, Ph. D. thesis, University of Rochester, 1982.

[Smit78]

R. G. Smith and R. Davis, *Distributed problem solving: The contract net approach,* Report No. STAN-CS-78-667, Computer Science Department, Stanford University, Sept., 1978.

[Stee79]

L. Steels, Reasoning modeled as a society of communicating experts, Ph. D. thesis, TR-542, Artificial Intelligence Laboratory, MIT, June, 1979.

[Wein80]

D. Weinreb and D. Moon, *Flavors: Message passing in the LISP machine,* Memo. 602, MIT Artificial Intelligence Laboratory, MIT, Nov. 1980.

[Wino75]

T. Winograd, Frame representations and the declarative/procedural controversy, pp. 185-210 in *Representation and Understanding,* ed. D. G. Bobrow and A. M. Collins, Academic Press, New York, 1975.

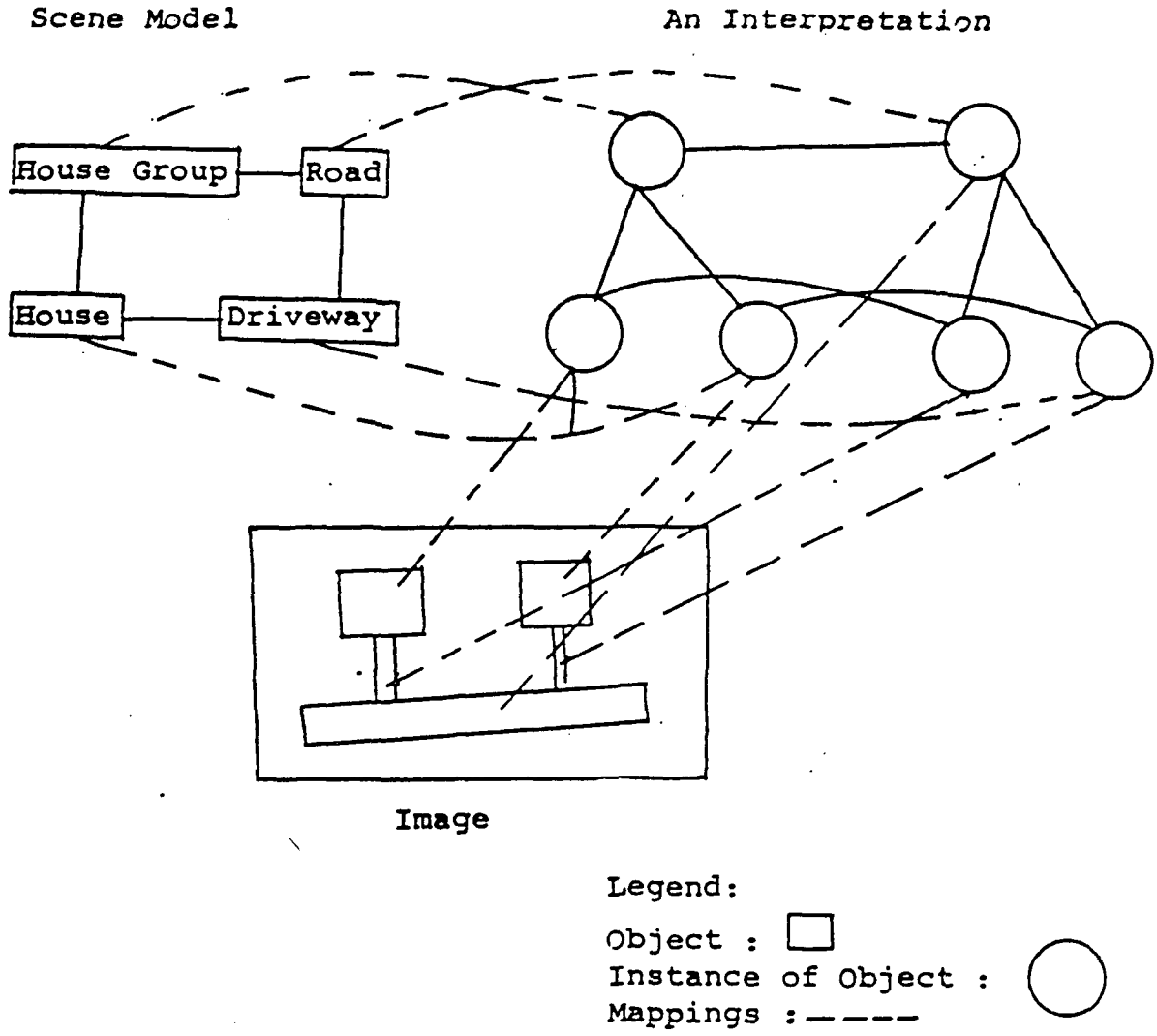


Figure 1-1. Mappings between the scene and the image.

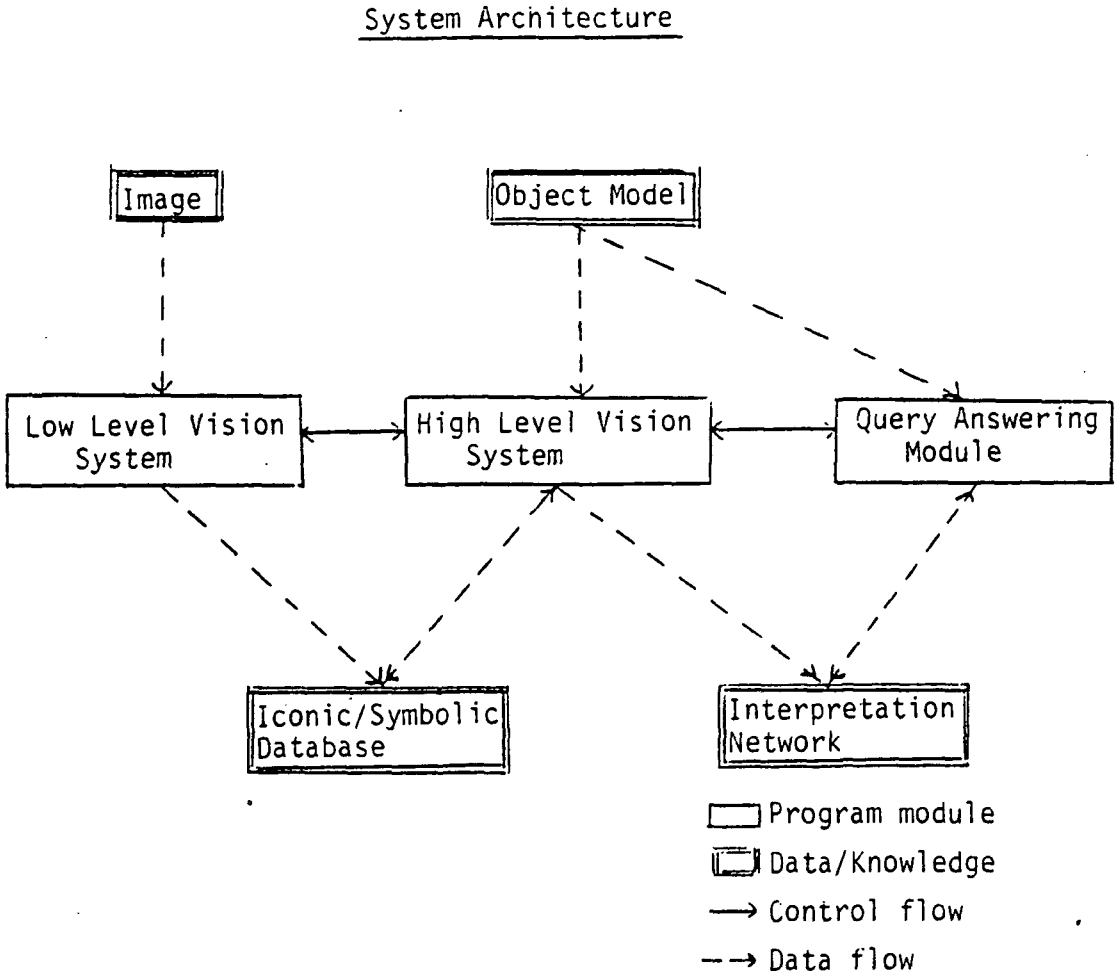


Figure 1-2. System architecture for the SIGMA image understanding system.

frame RECTANGULAR-HOUSE;
    rules :
        $F_{rectangle}$;
    links :
        AKO : HOUSE;
end-frame

frame L-SHAPED-HOUSE;
    rules :
        $F_{L-shape}$;
    links :
        AKO : HOUSE;
end-frame

frame HOUSE;
    slots :
        centroid;
        shape-description;
        front-of-house;
        connecting-driveway;
    rules :
        $F_{driveway}$;
    links :
        CAN-BE : RECTANGULAR-HOUSE, L-SHAPED-HOUSE;
end-frame

Figure 2-1 Frame definitions for HOUSE, RECTANGULAR-HOUSE, and L-SHAPED-HOUSE.

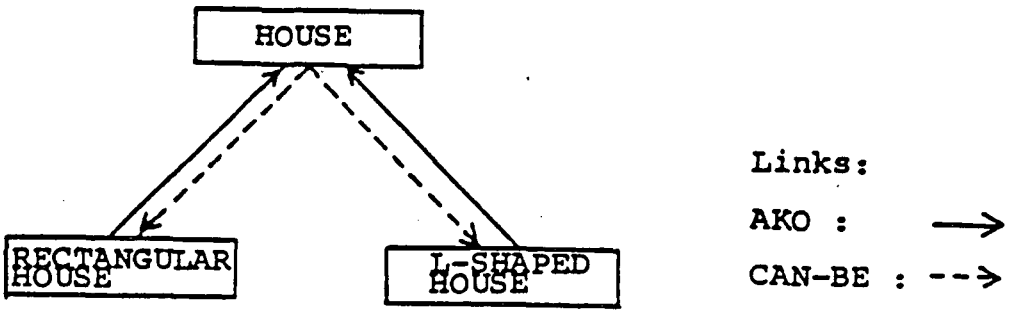


Figure 2-2 Links between HOUSE, RECTANGULAR-HOUSE and L-SHAPED-HOUSE frames.

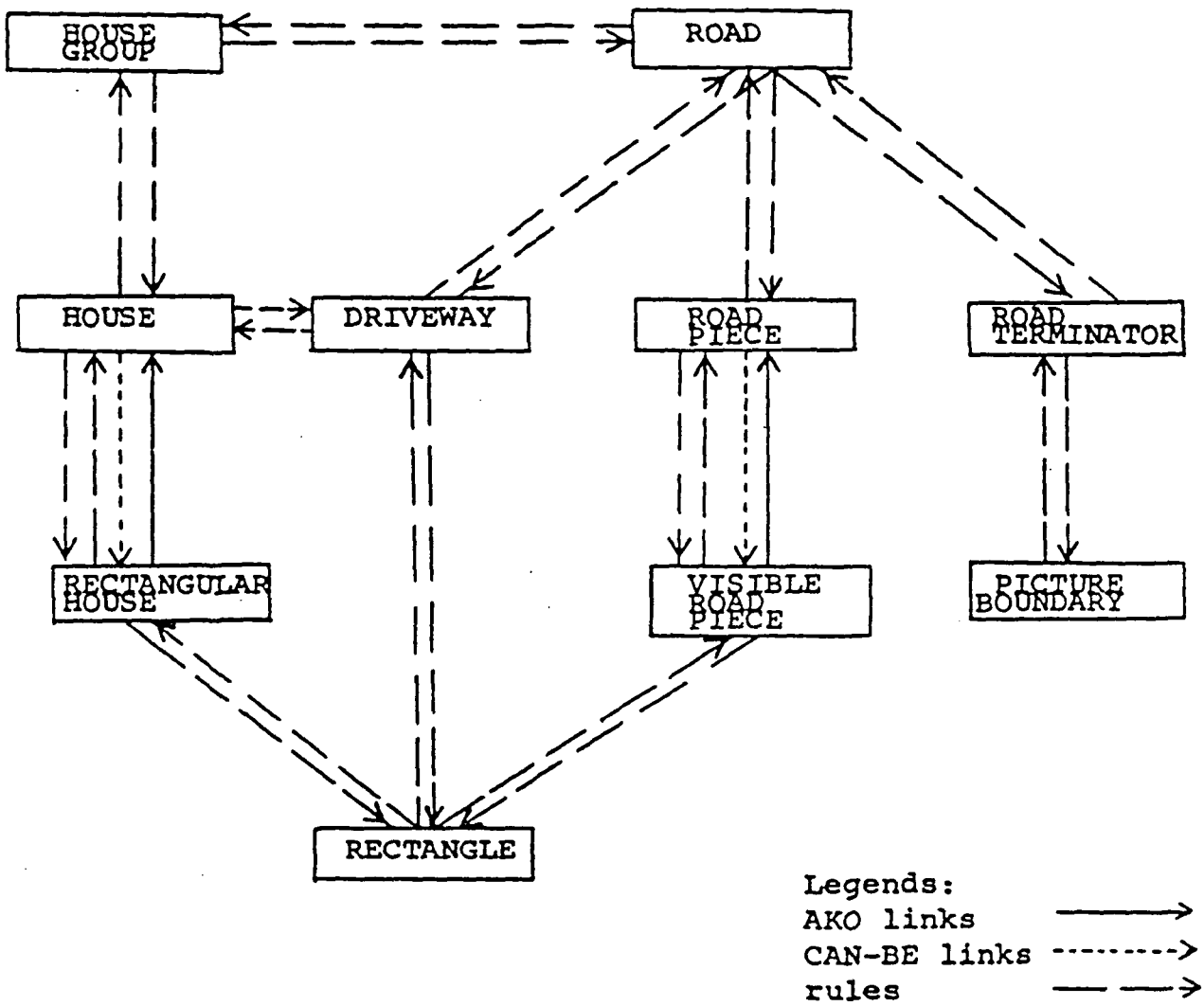


Figure 2-3 A model of a suburban housing development.

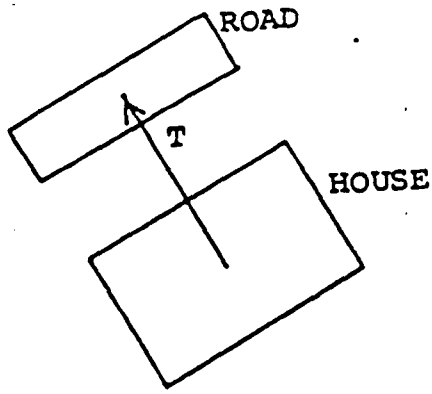


Figure 2-4. Pictorial description of house-road relation.

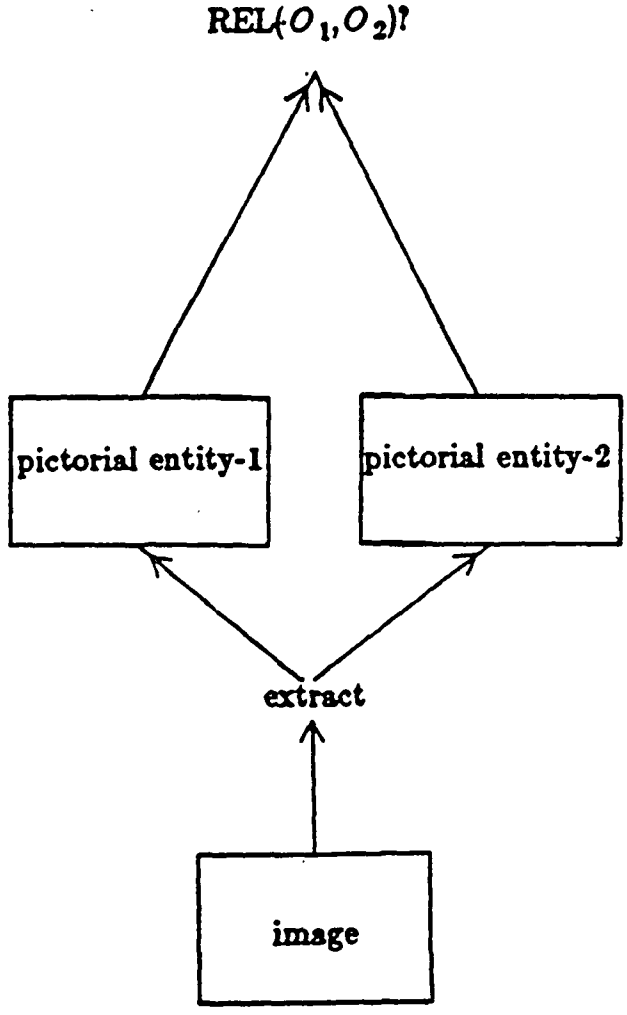
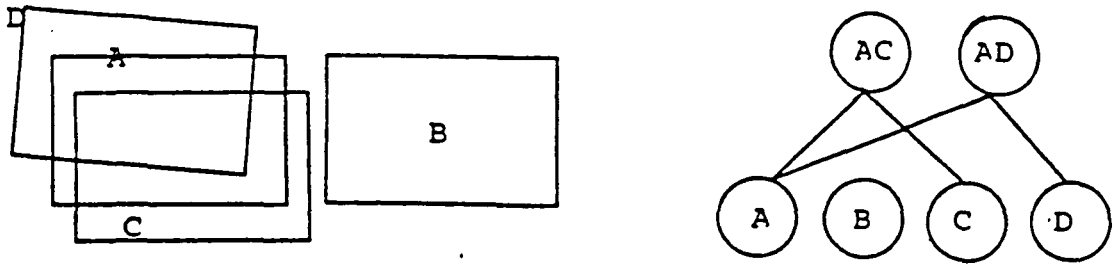
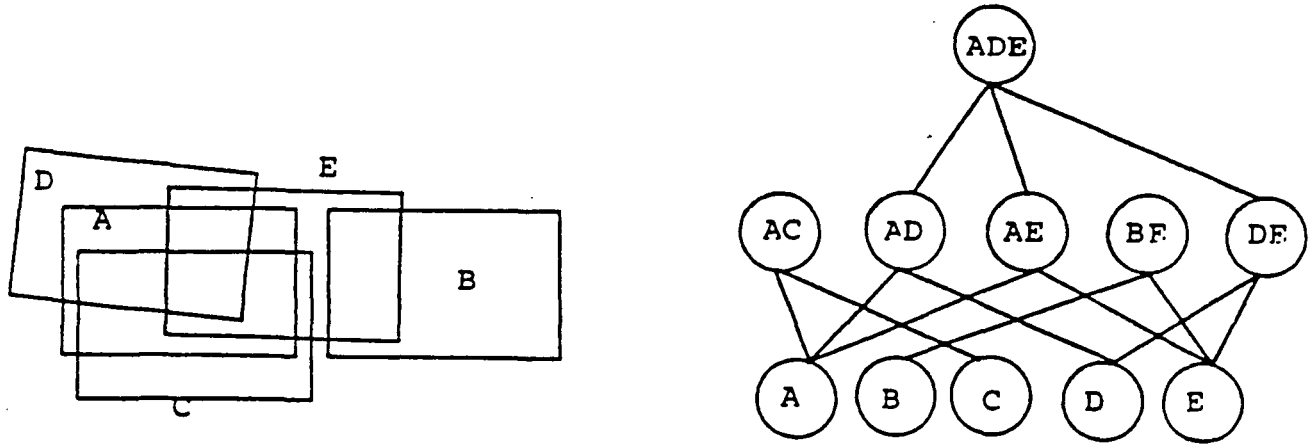


Figure 3-1. Using a relation as a constraint.



iconic descriptions **situation lattice**

Figure 3-2(a). The situation lattice before the insertion.



iconic descriptions **situation lattice**

Figure 3-2(b). The situation lattice after the insertion.

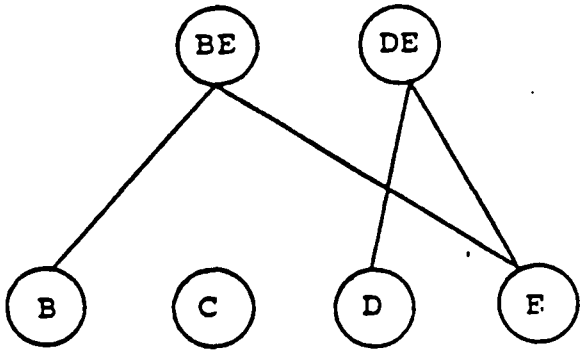


Figure 3-3. The situation lattice after the removal of A.

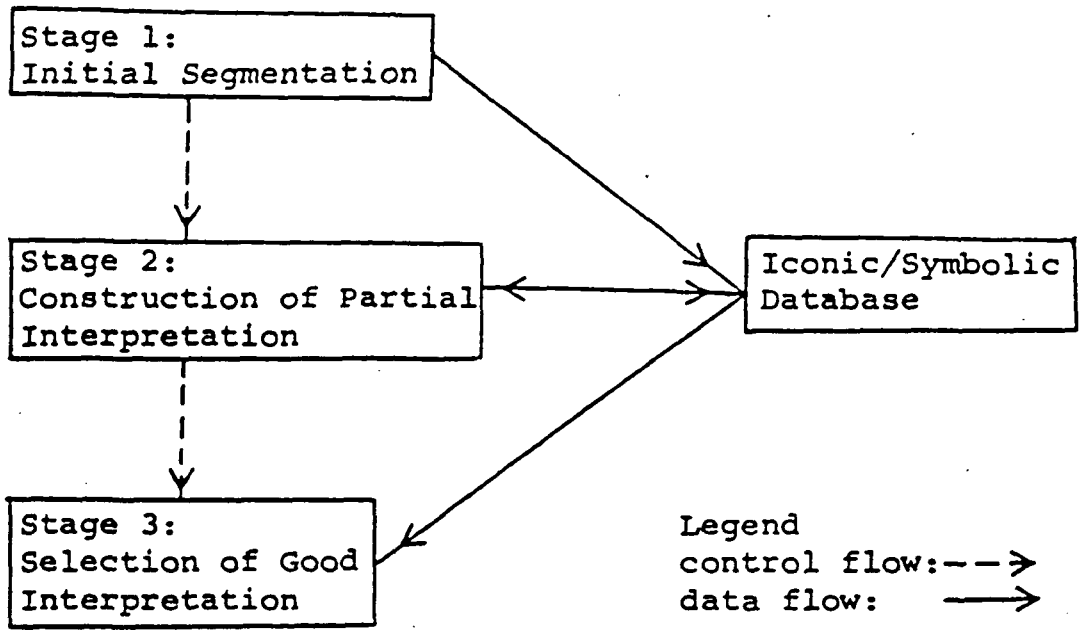


Figure 4-1. The stages of the control of SIGMA.

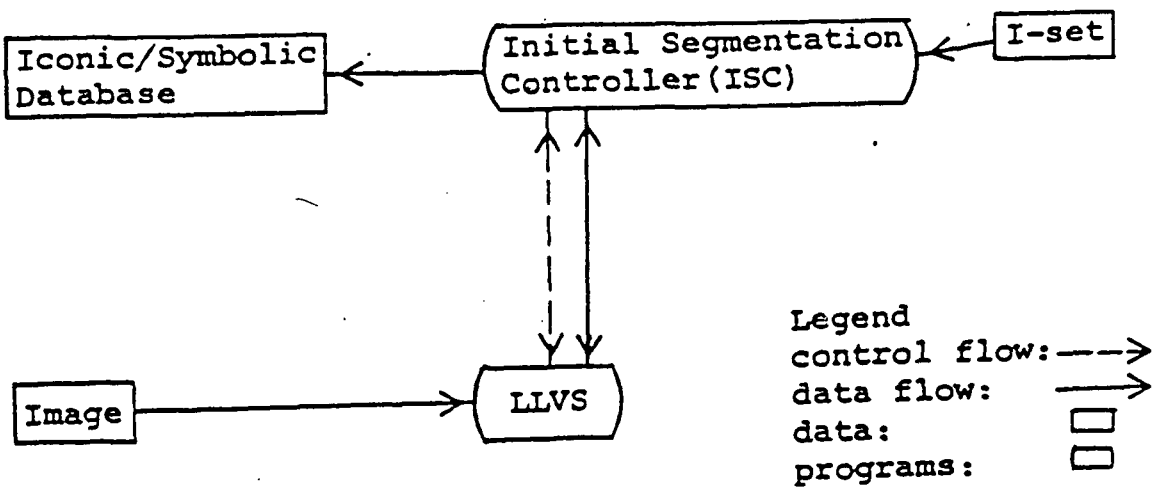


Figure 4-2. The schematic diagram of the initial segmentation process.

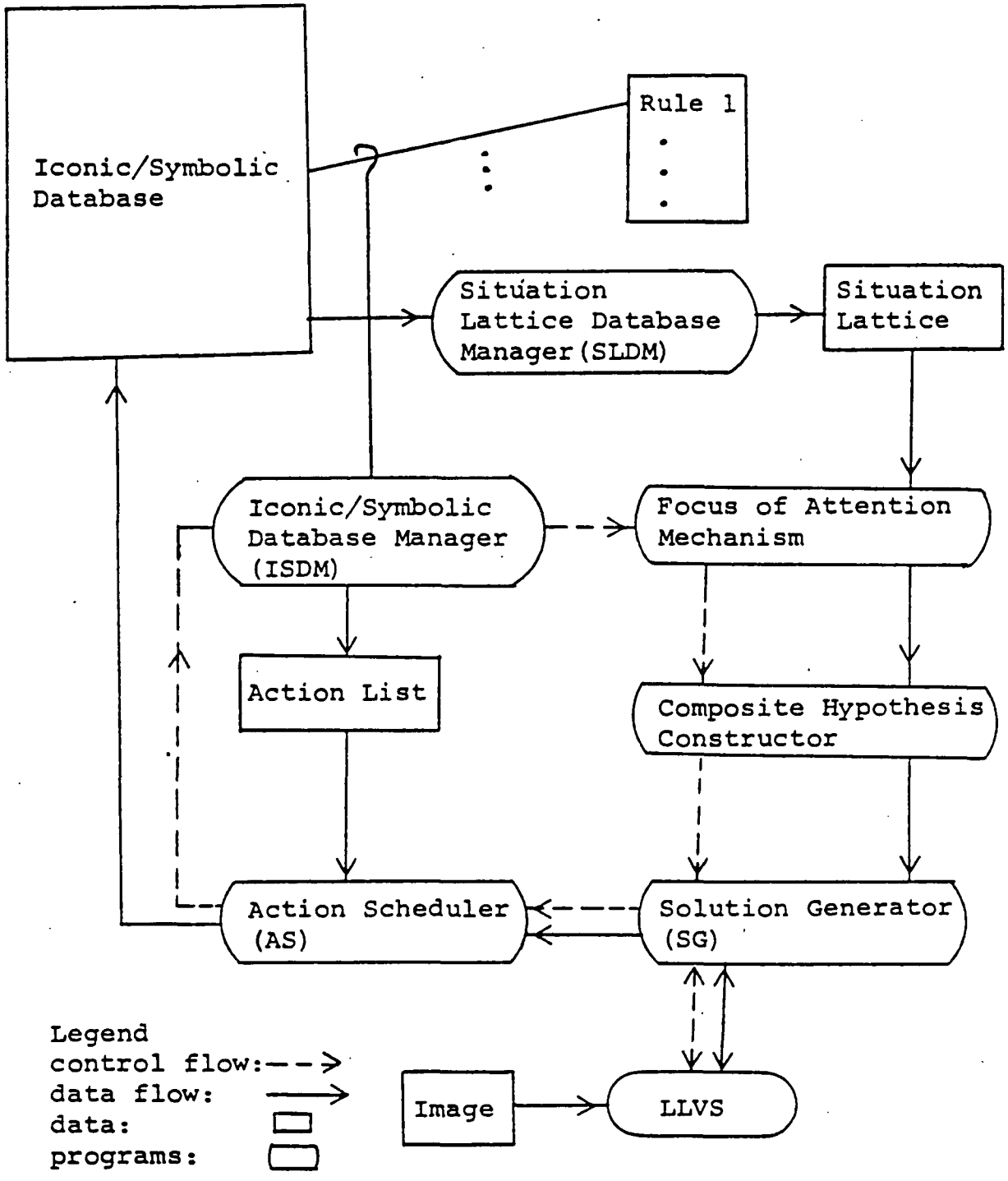
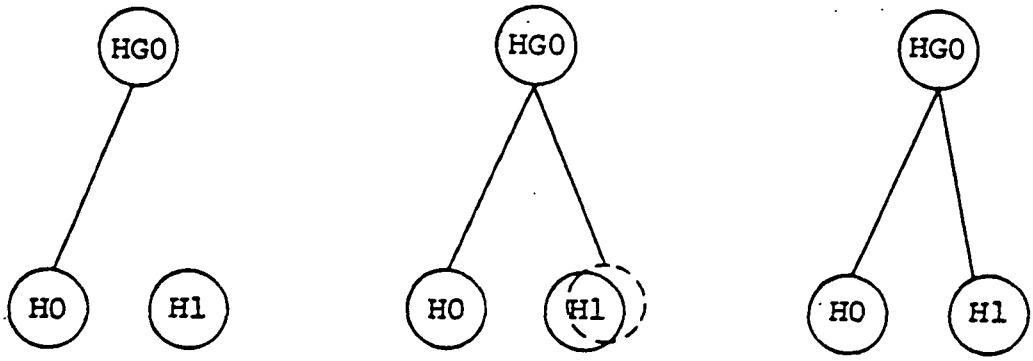


Figure 4-3. The schematic diagram of the interpretation stage.

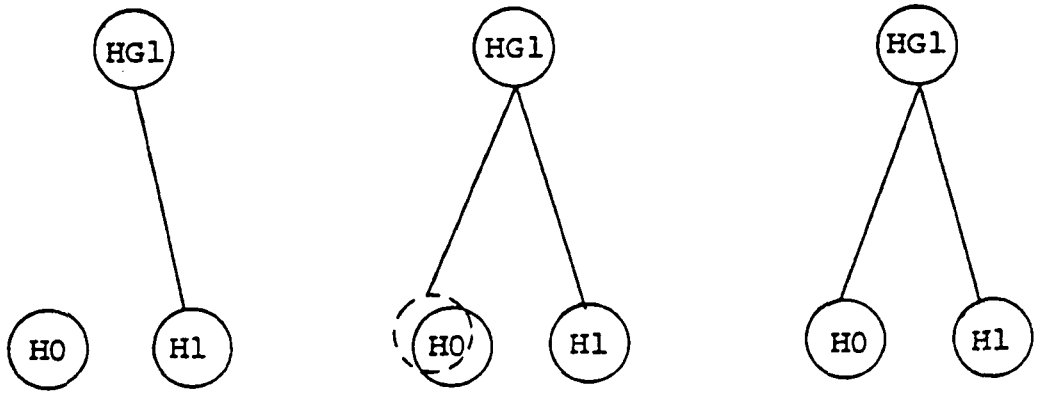


a house group instance containing H_0 is created

generate hypothesis about possible house in HG_0

fill H_1 in instance HG_0

Figure 4-4(a). Reasoning steps for constructing HG_0.



a house group instance containing H_1 is created

generate hypothesis about possible house in HG_1

fill H_0 in instance HG_1

Figure 4-4(b). Reasoning steps for constructing HG_1.

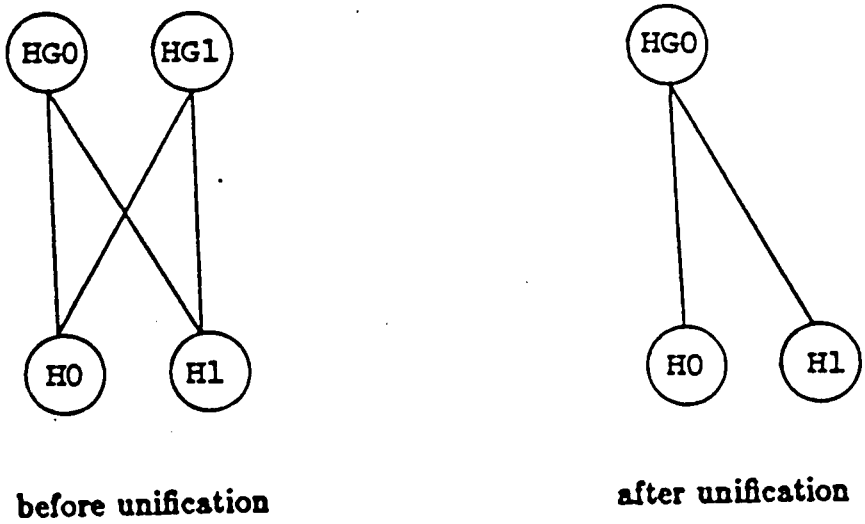


Figure 4-5. Unification of identical instances.

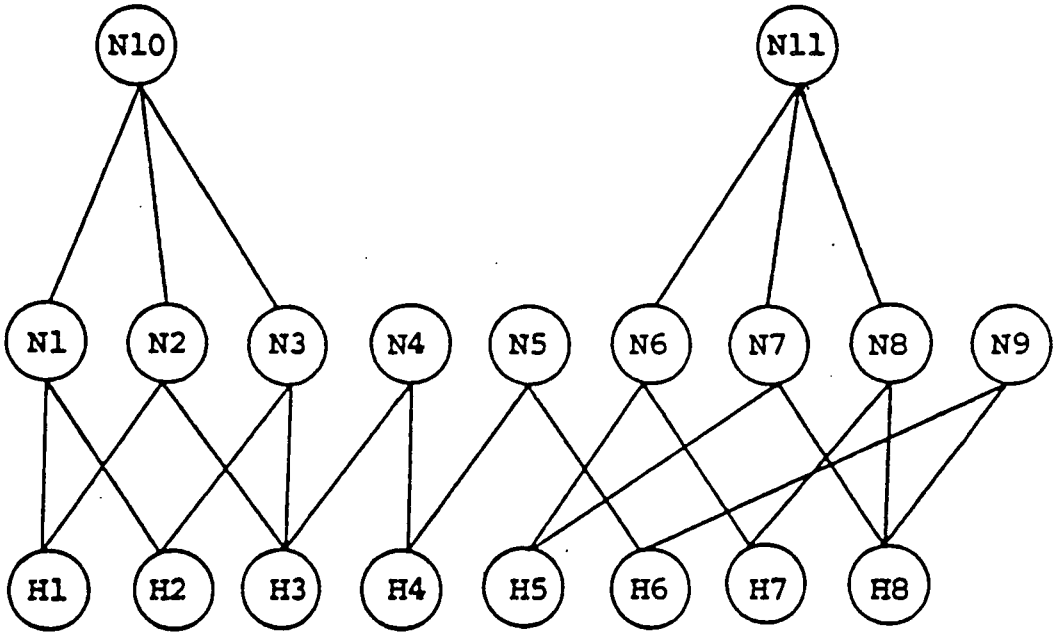
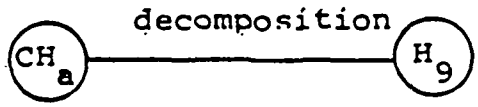


Figure 4-6. A situation lattice.



Target object of CH_a:
RECTANGULAR-HOUSE

Target object of H_9:
RECTANGLE

Delayed-action:
if H =nil then conclude(nil)
else conclude(make-instance(RECTANGLE-HOUSE,H)).

Figure 4-7. Decomposition of CH_a.

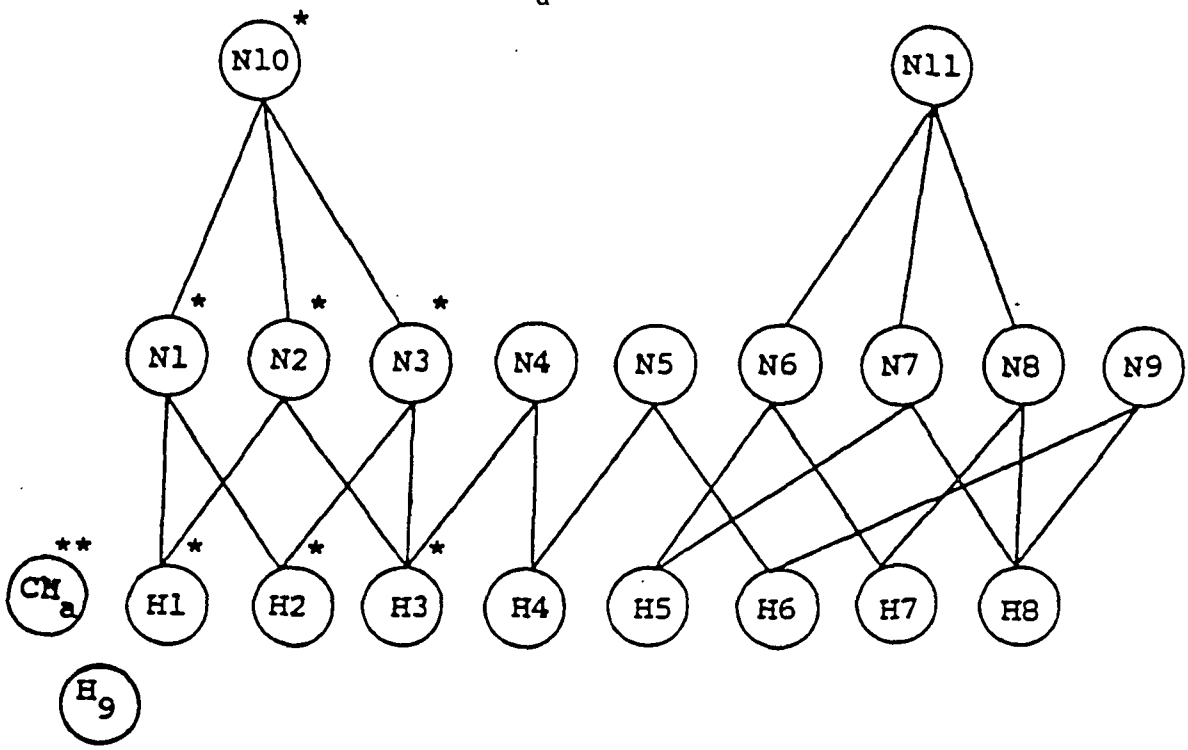


Figure 4-8. The resulting situation lattice.

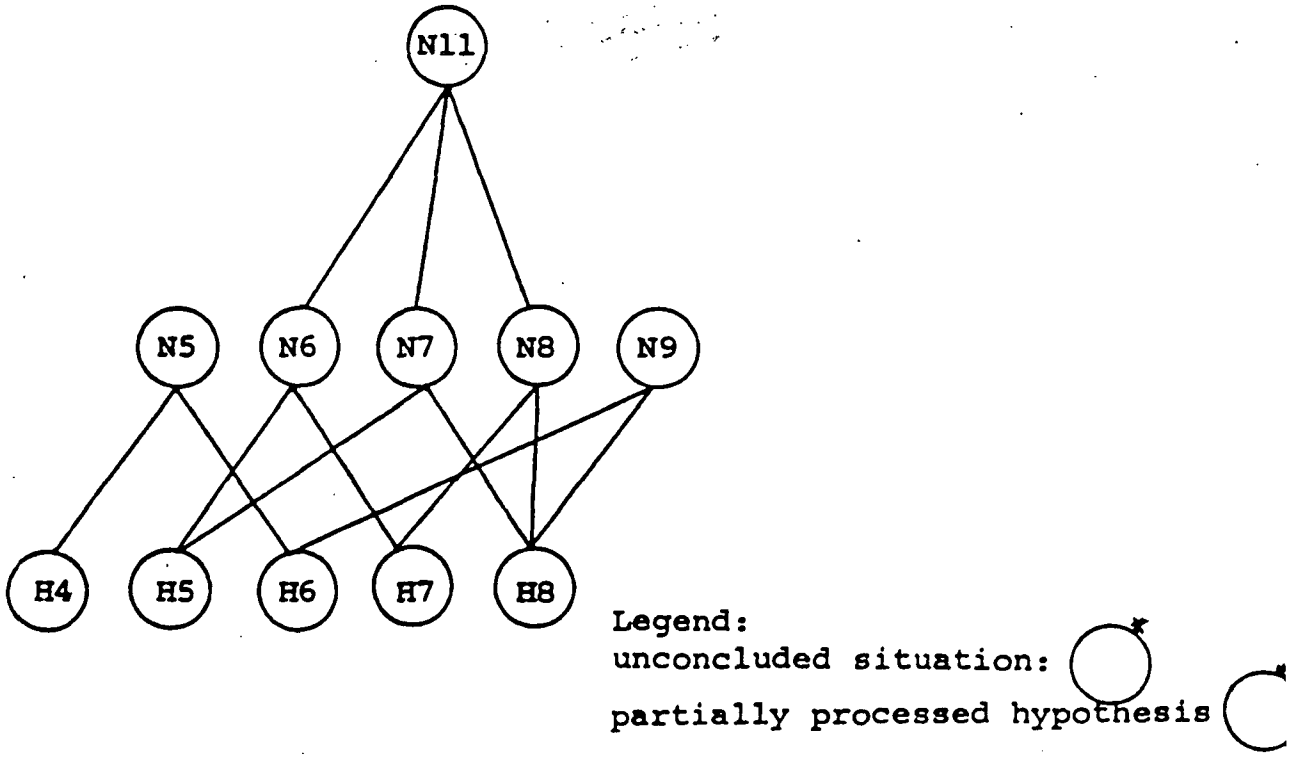


Figure 4-9. The situation lattice after actions are evaluated.

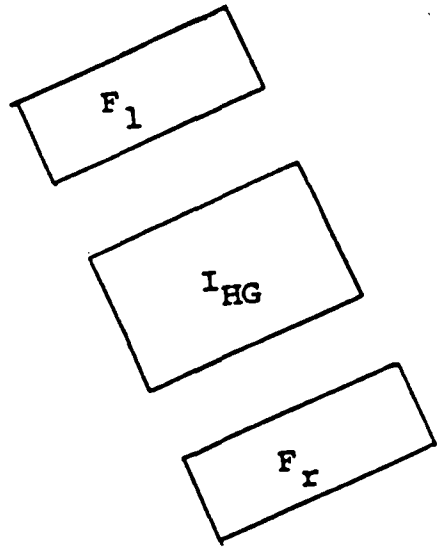


Figure 4-10. Possible road locations along I_{HG}.

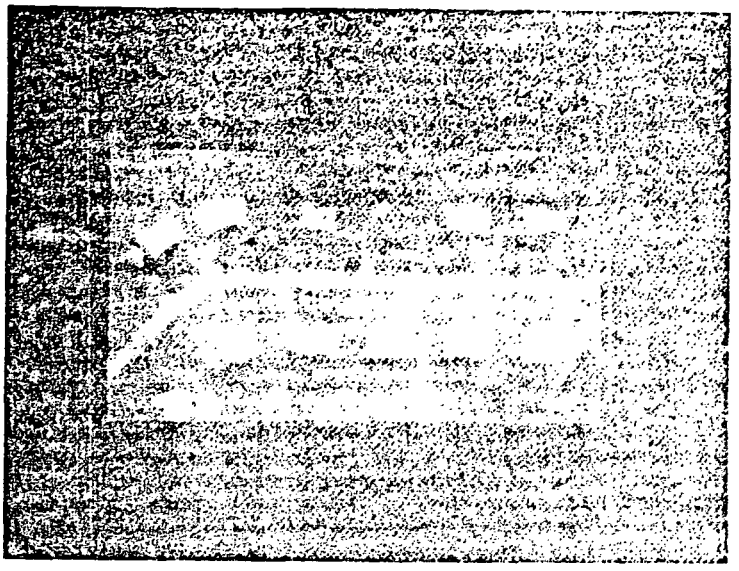


Figure 5-1. An aerial image.

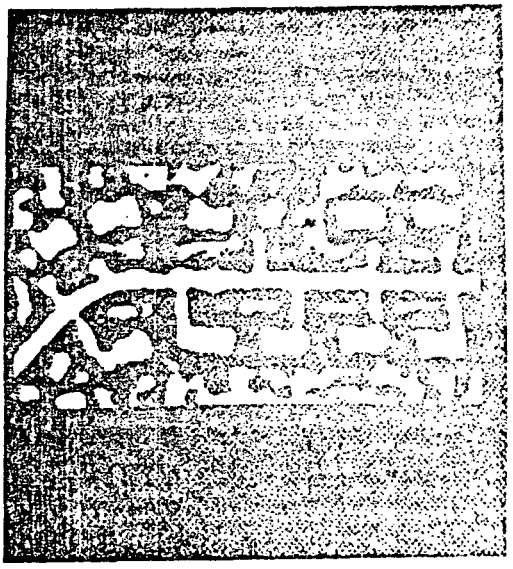


Figure 5-2. Positive connected regions.

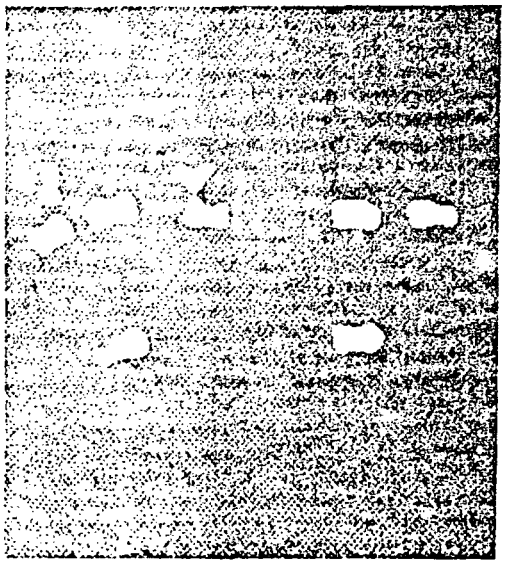


Figure 5-3. Blobs extracted by Blob-finder.

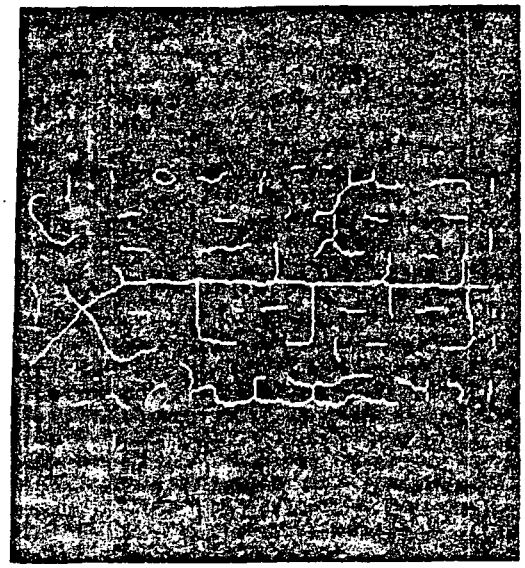


Figure 5-4. Skeletons of the connected components.

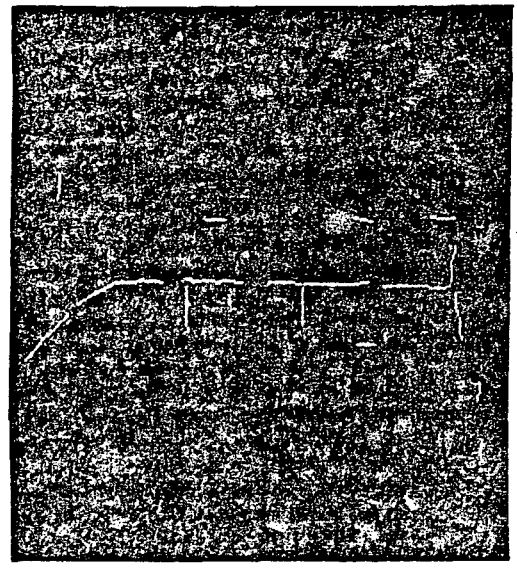


Figure 5-5. Skeletons of the ribbons extracted by Ribbon-finder.

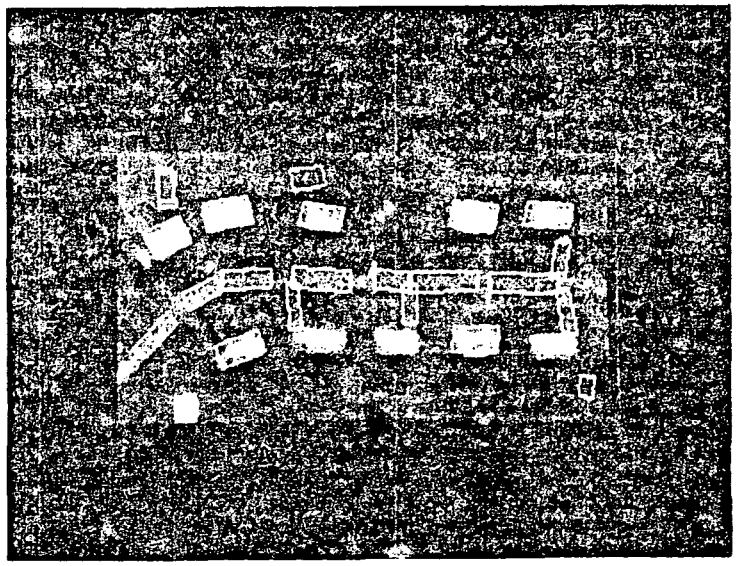


Figure 5-6. Iconic descriptions of the RECTANGLE instances generated based on the initial segmentation process.

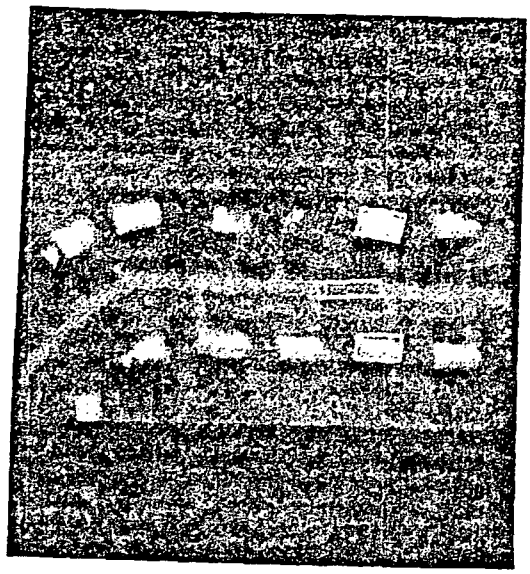


Figure 5-7(a). An example (see Section 5.3.1.)

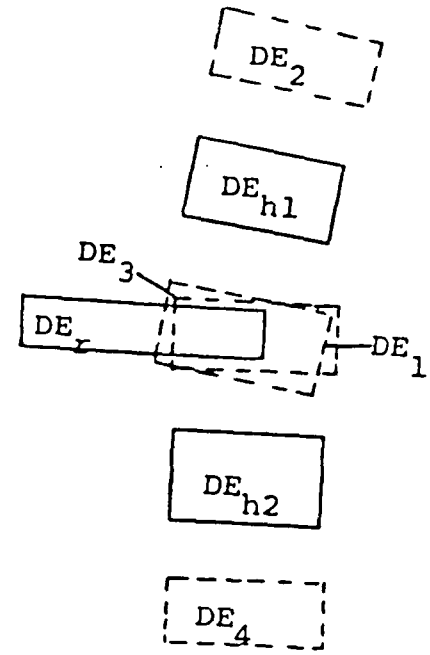


Figure 5-7(b). A depiction of the situation.

DE's	Type	Generated-by
DE_r	ROAD instance	
DE_{h1}	HOUSE-GROUP instance	
DE_{h2}	HOUSE-GROUP instance	
DE_1	ROAD hypothesis	DE_{h1}
DE_2	ROAD hypothesis	DE_{h1}
DE_3	ROAD hypothesis	DE_{h2}
DE_4	ROAD hypothesis	DE_{h2}

Table 5-1. The descriptions of the DE's.

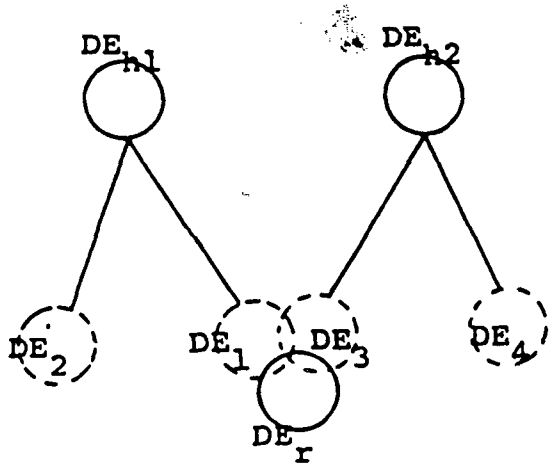


Figure 5-8. Portion of the interpretation network related to the situation.

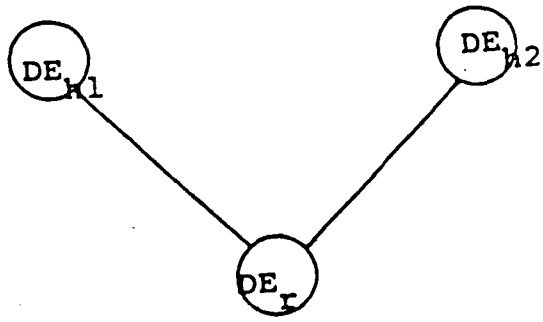


Figure 5-9. Resulting interpretation network.

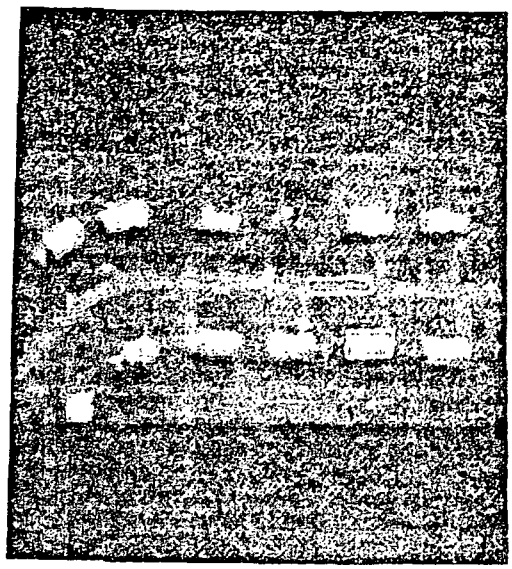


Figure 5-10(a). An example (see Section 5.3.2.)

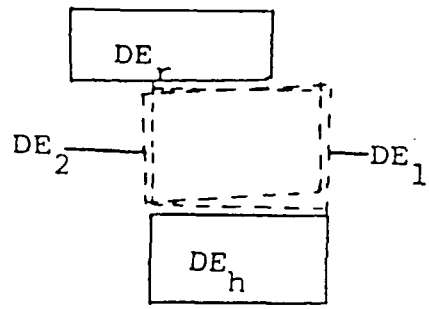


Figure 5-10(b). A depiction of the situation.

DE's	Type	Generated-by
DE_r	ROAD instance	
DE_h	HOUSE instance	
DE_1	DRIVEWAY hypothesis	DE_h
DE_2	DRIVEWAY hypothesis	DE_r

Table 5-2. The descriptions of the DE's.

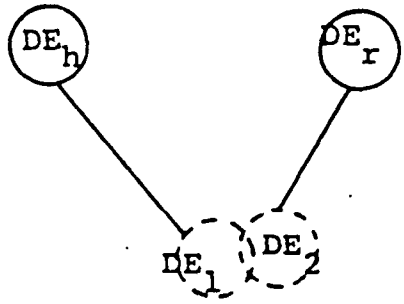


Figure 5-11. Portion of the interpretation related to the situation.

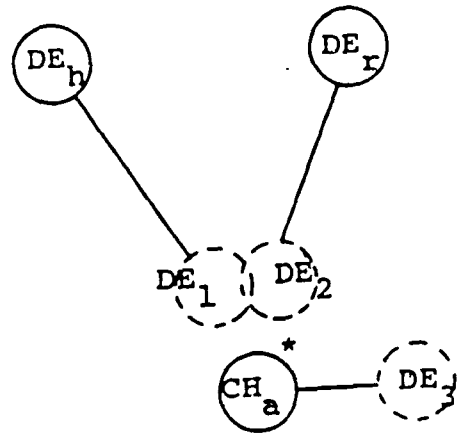


Figure 5-12. Resulting interpretation network.

Action	Cause-of-delay
$A_{first-order-properties}$	DE_3

Unconcluded-situation	Composite hypothesis
S_1	CH_a

Table 5-3. Relations between the DE's, action $A_{first-order-properties}$ and S_1.

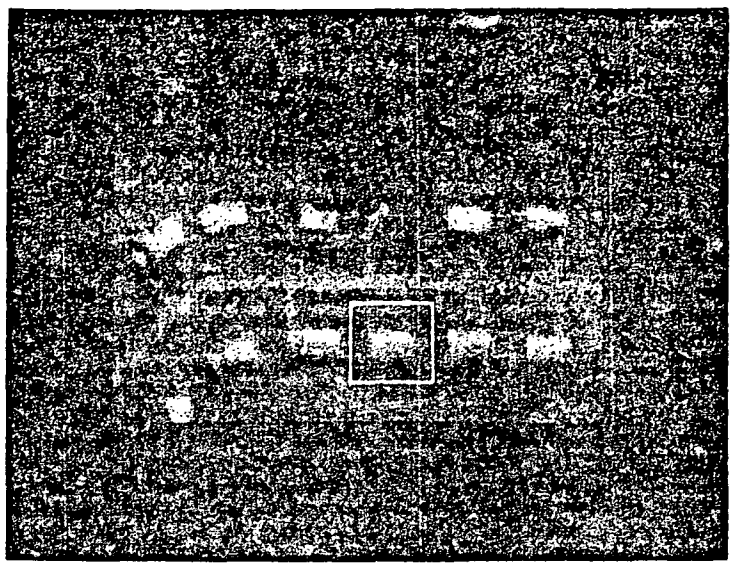


Figure 5-13. A window generated by the HLVS.

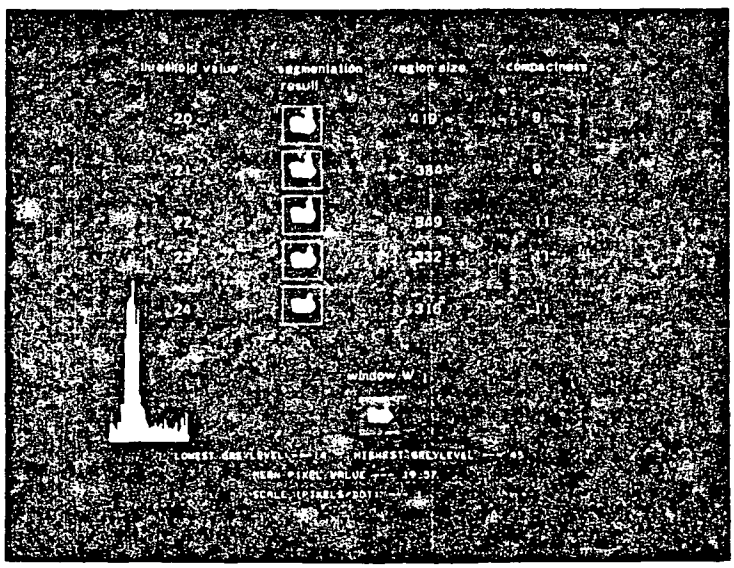


Figure 5-14. Intermediate results of the LLVS.

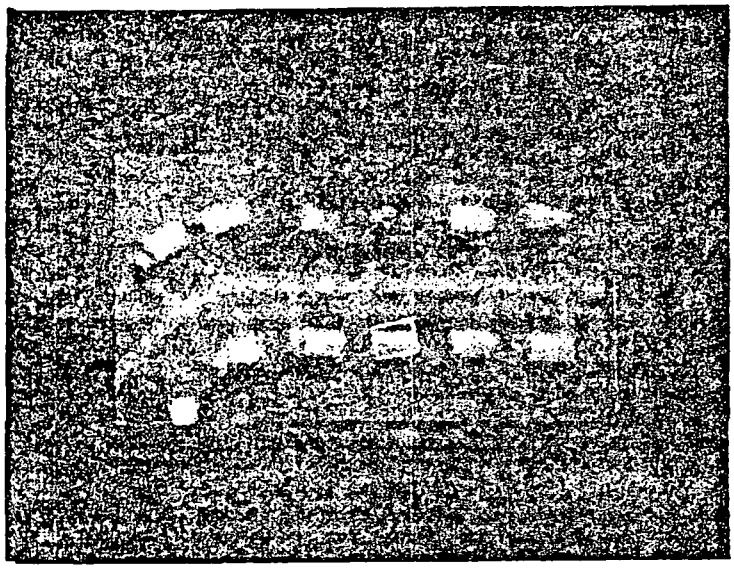


Figure 5-15. The RECTANGLE instance generated by the HLVS (based on the results computed by the LLVS).

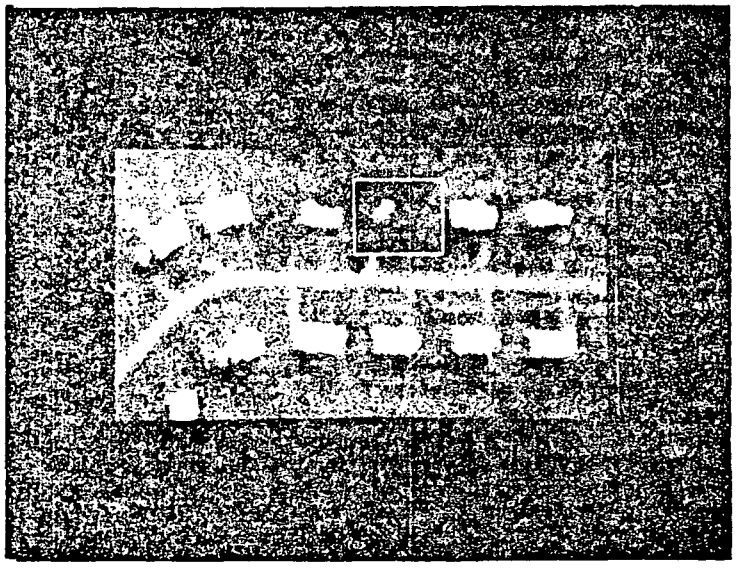


Figure 5-16. Another window generated by the HLVS.

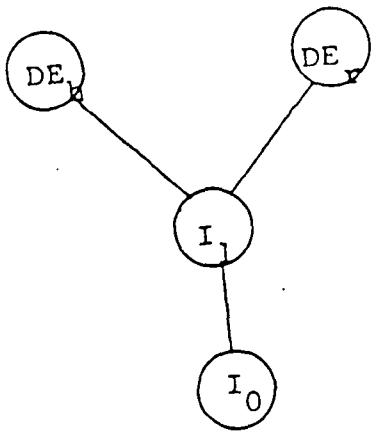


Figure 5-17. Resulting interpretation network (when a solution has been generated).

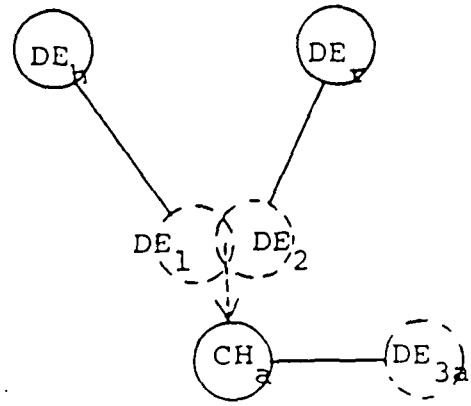


Figure 5-18. Resulting interpretation network (when no solution has been computed).

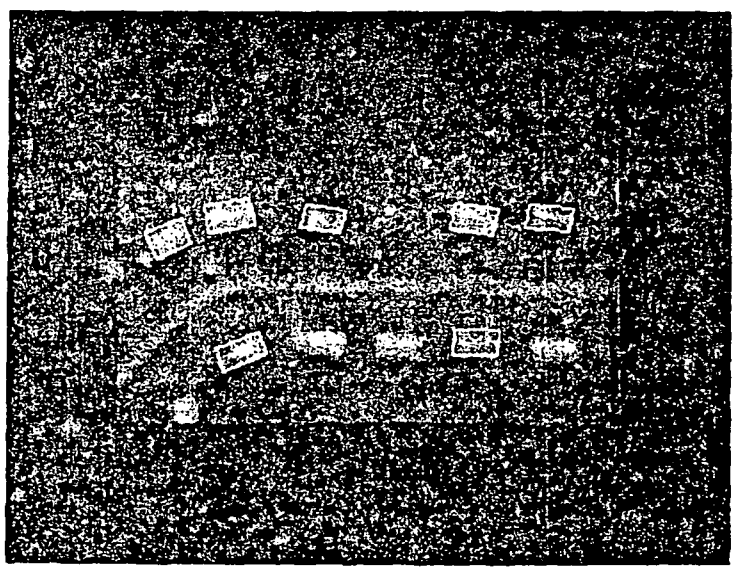


Figure 5-19. Initial set of RECTANGULAR-HOUSE instances.

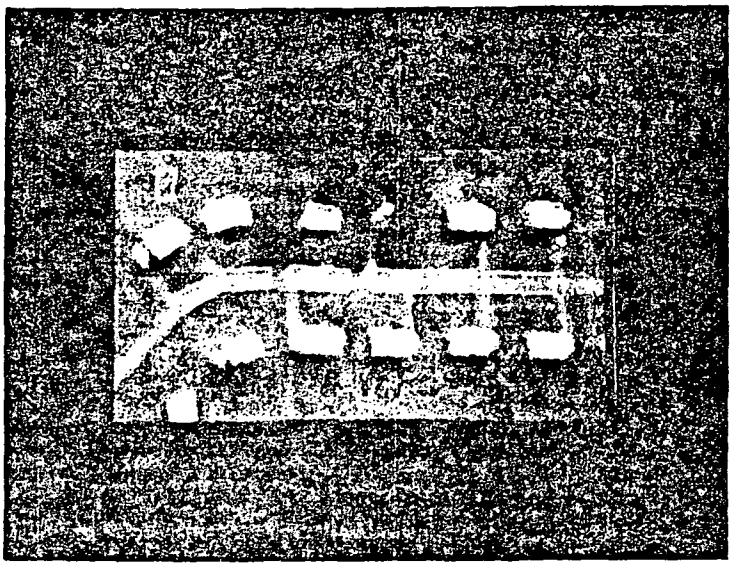


Figure 5-20. Initial set of VISIBLE-ROAD-PIECE instances.

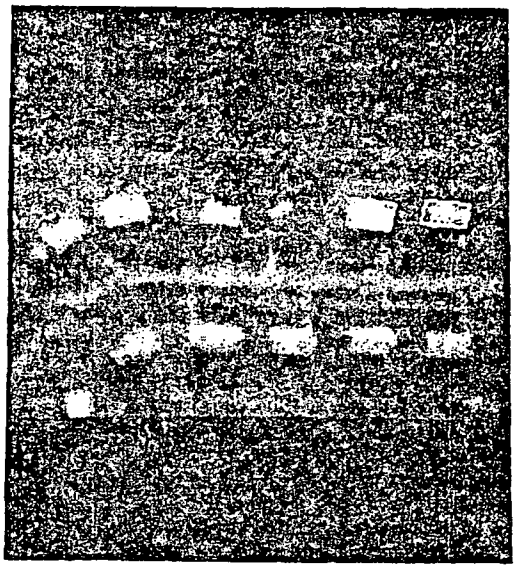


Figure 5-21(a). Two HOUSE-GROUP instances (see Section 5-4).

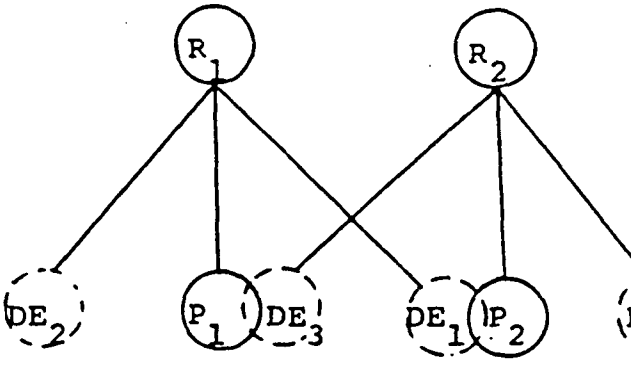


Figure 5-21(b). Portion of the interpret network related to the situation.

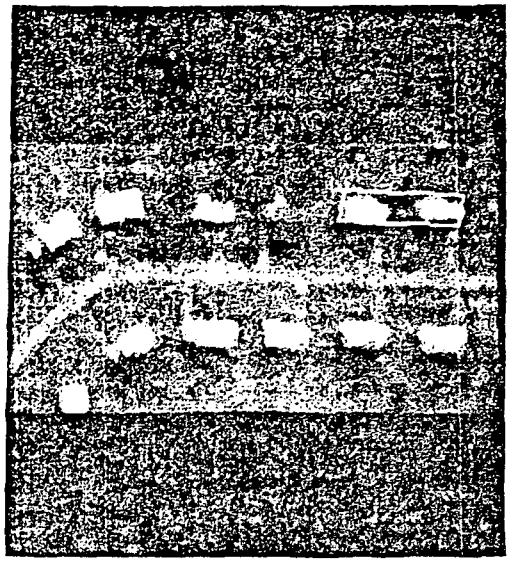


Figure 5-22(a) Resulting HOUSE-GROUP instance R_1.

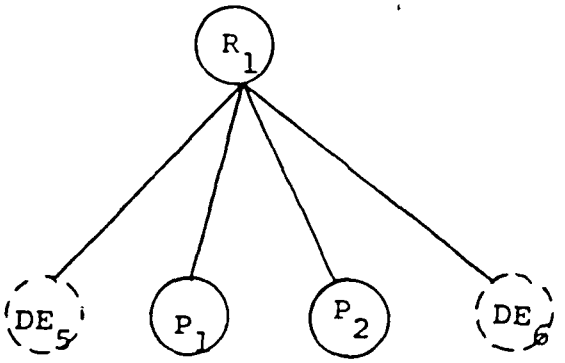


Figure 5-22(b). Hypotheses generated by R_1.

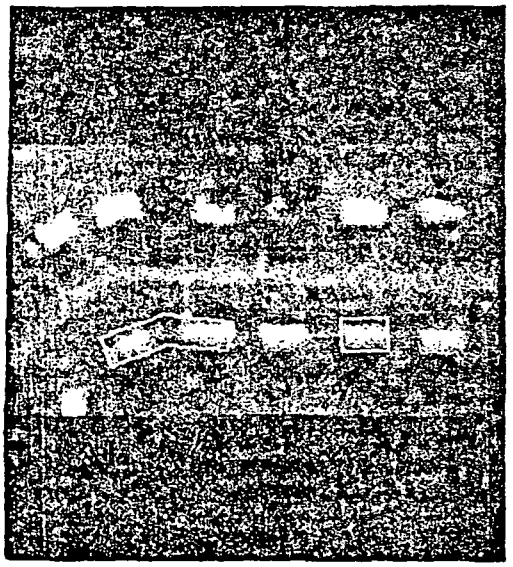


Figure 5-23(a). Two HOUSE-GROUP instances (see Section 5-4).

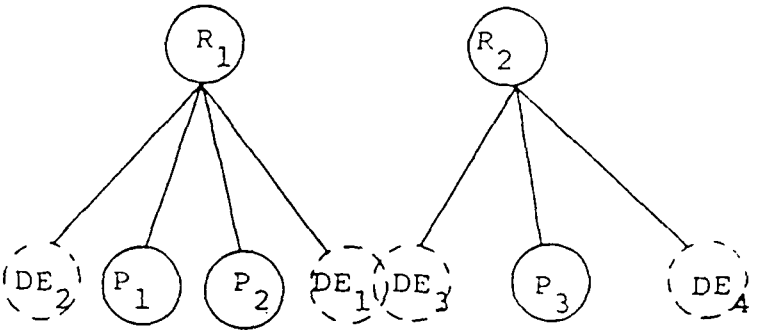
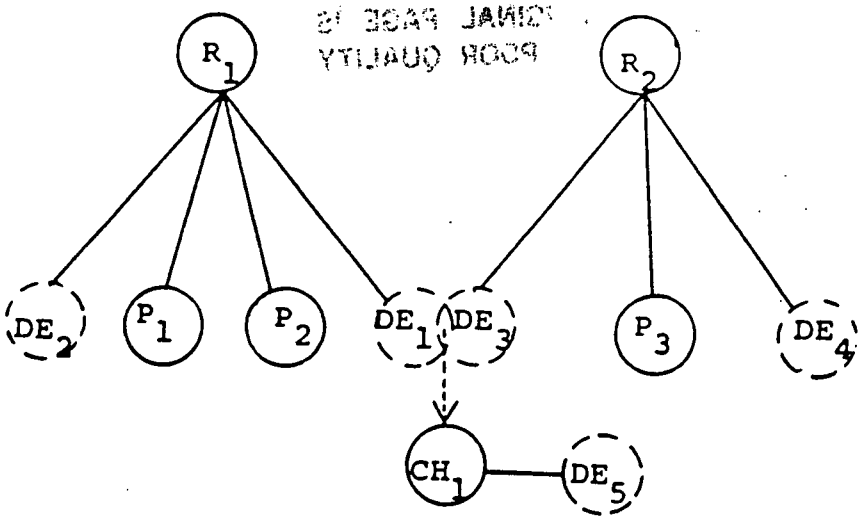
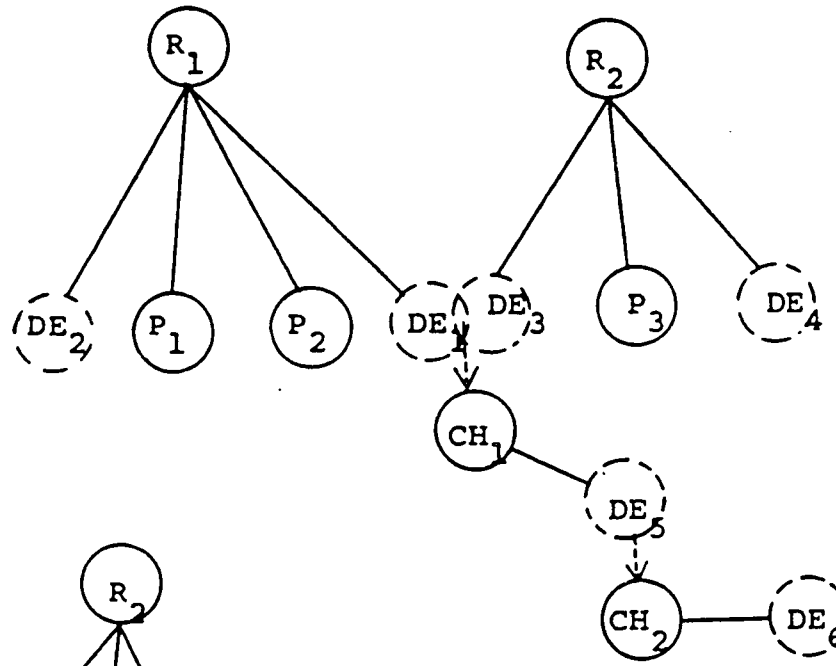


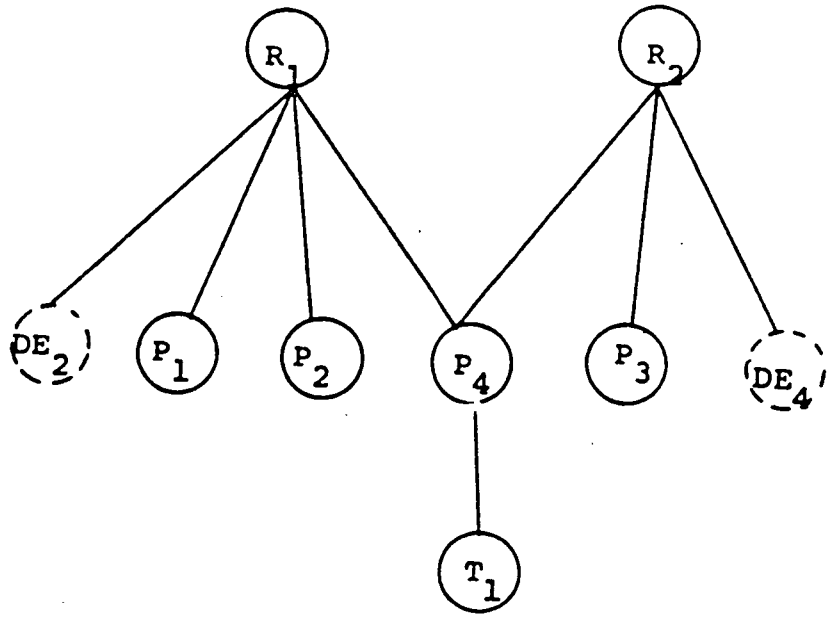
Figure 5-23(b). Portion of the interpretation network related to the situation.



5-24(a).

5-24(b) --





5-24(c).

Figure 5-24. Snapshots of the interpretation network related to R_1 and R_2 (see Figure 5-23) at various stages of the processing.

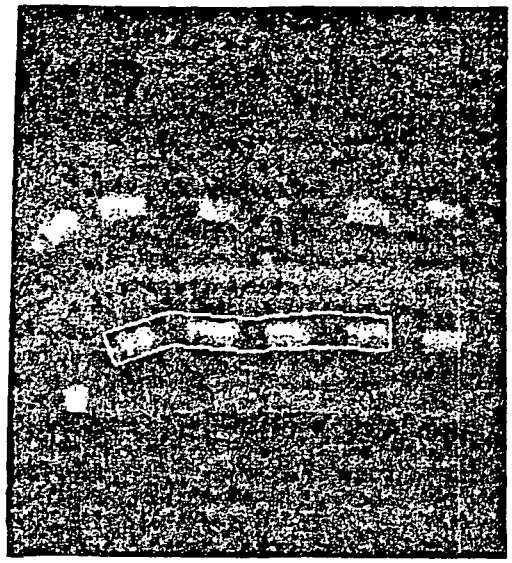


Figure 5-25(a). Resulting HOUSE-GROUP instance.

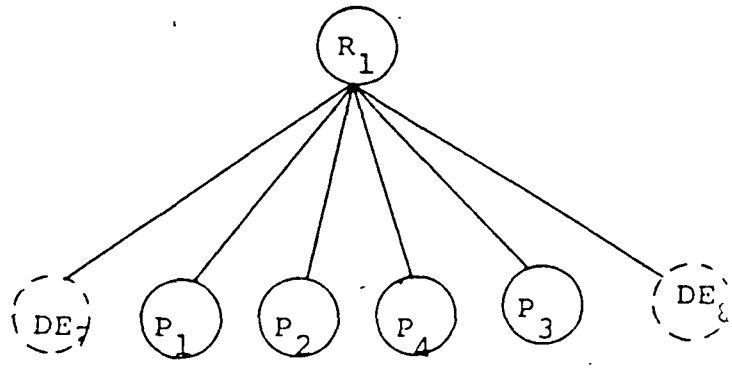


Figure 5-25(b) Resulting interpretation network.

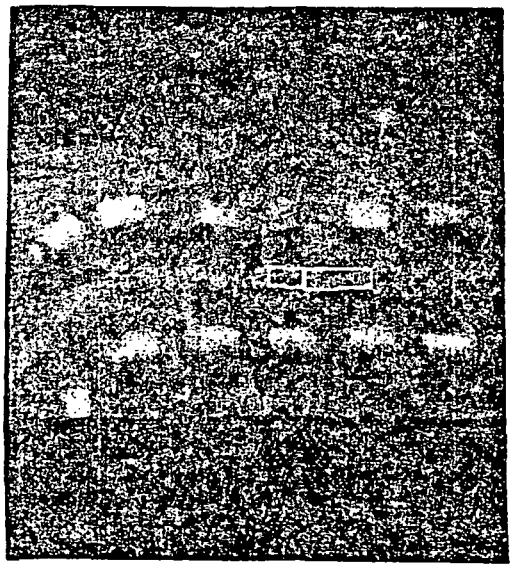


Figure 5-26(a). Two ROAD instances (see Section 5-4).

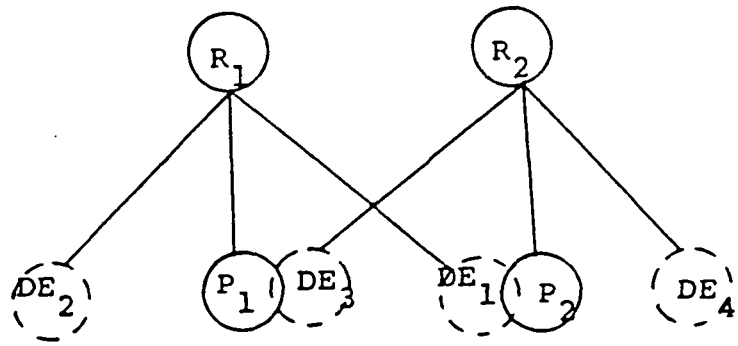


Figure 5-26(b). Portion of the interpretation network related to the situation

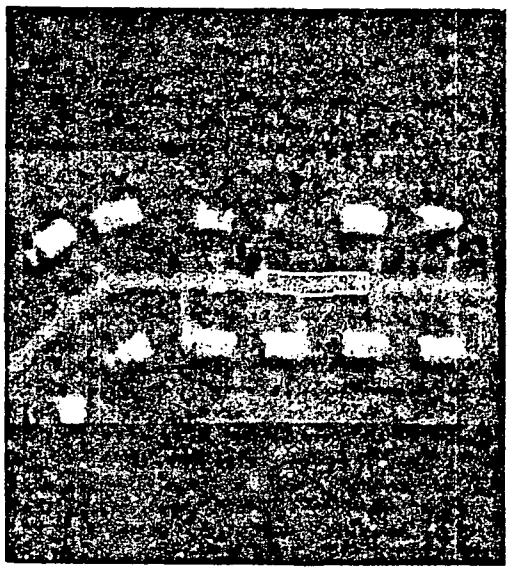


Figure 5-27(a). Resulting ROAD instance.

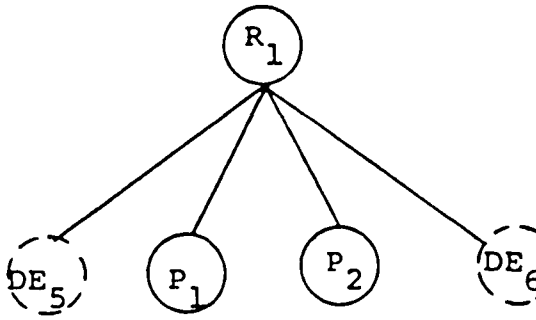


Figure 5-27(b). Resulting interpretat network.

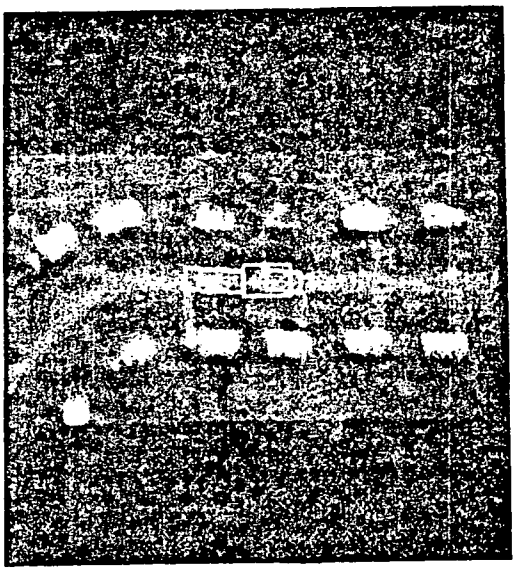


Figure 5-28(a). Two ROAD instances (see Section 5-4).

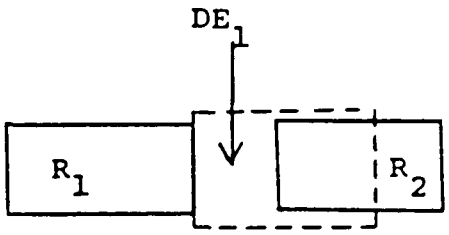
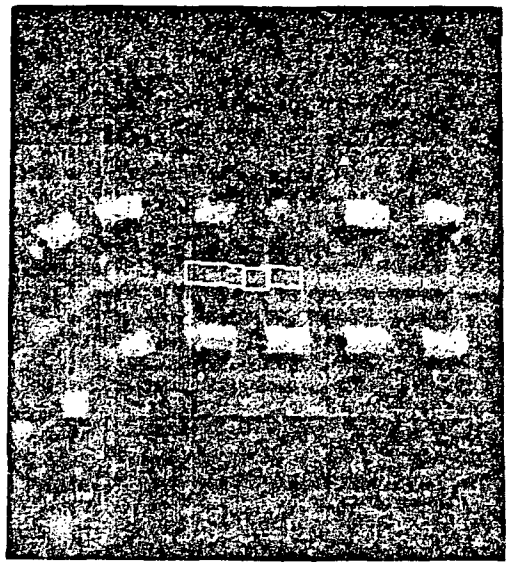


Figure 5-28(b). A depiction of the situation.



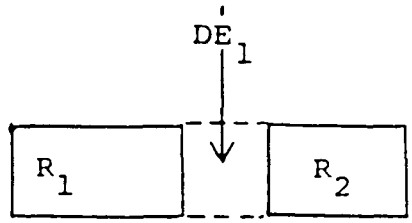


Figure 5-29. Hypothesis DE_1 has been modified.

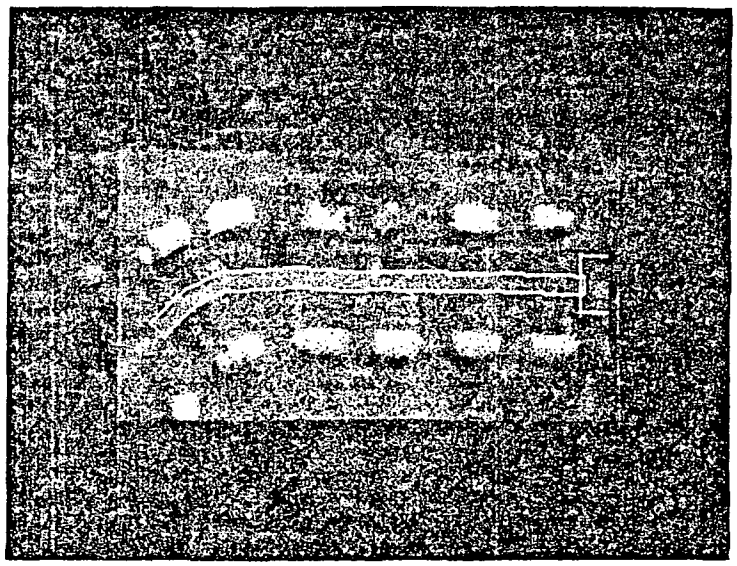
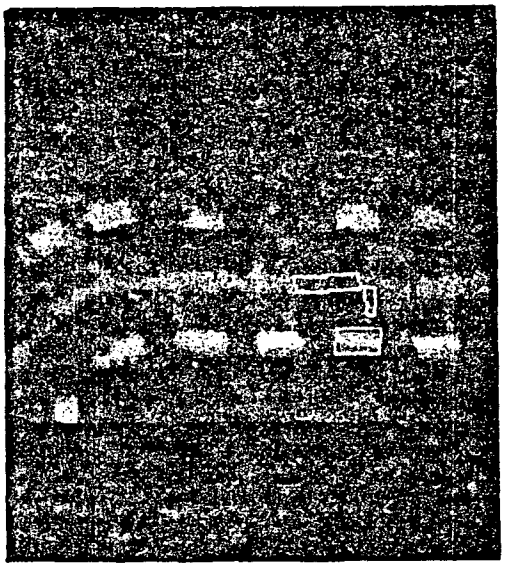


Figure 5-30. A ROAD-TERMINATOR hypothesis has been generated.



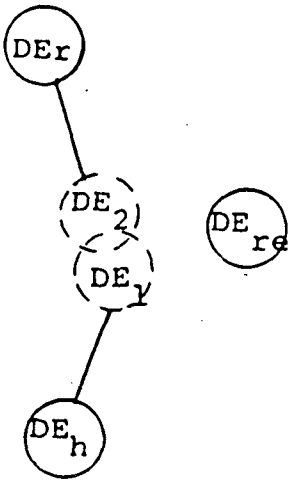


Figure 5-31. Iconic description of a situation and its interpretation network (see Section 5-4).

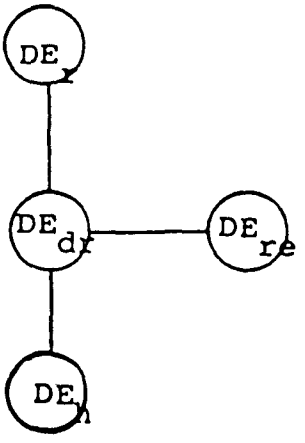


Figure 5-32. Resulting interpretation network.

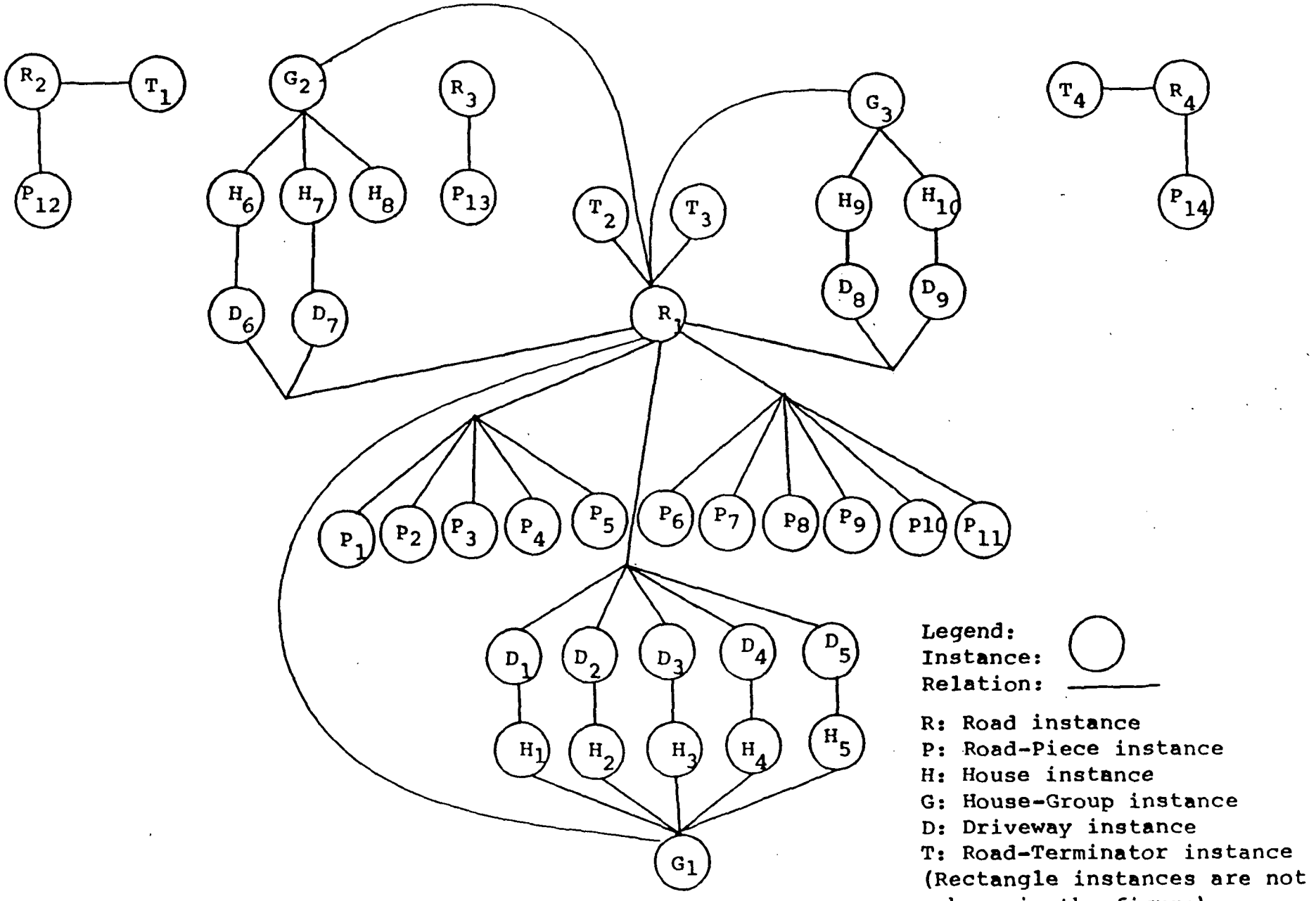


Figure 5-33. Final interpretation network.

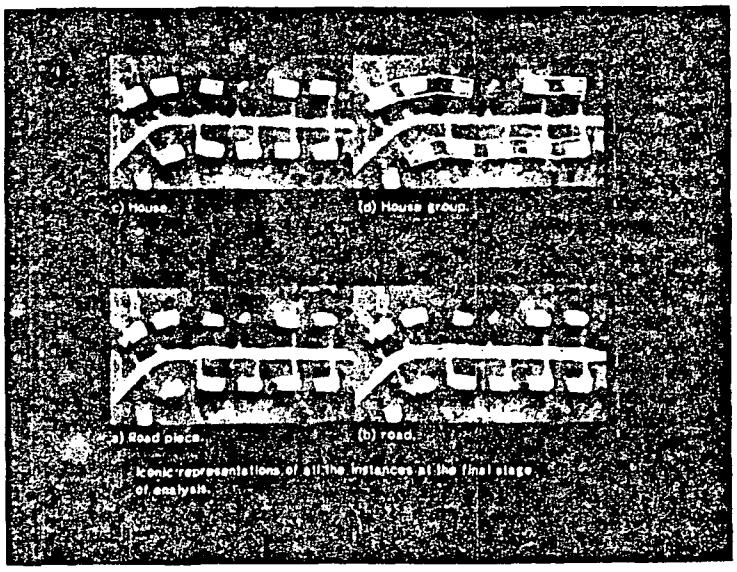


Figure 5-34. Final results.

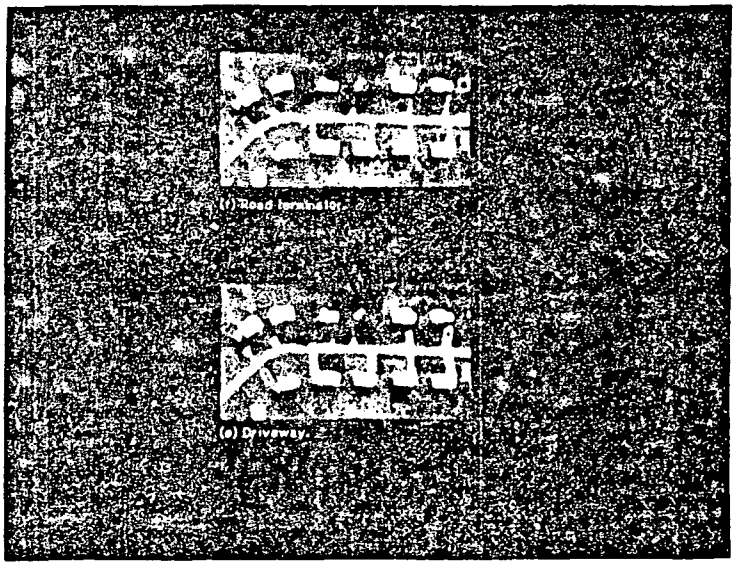


Figure 5-35. Final results (cont.).

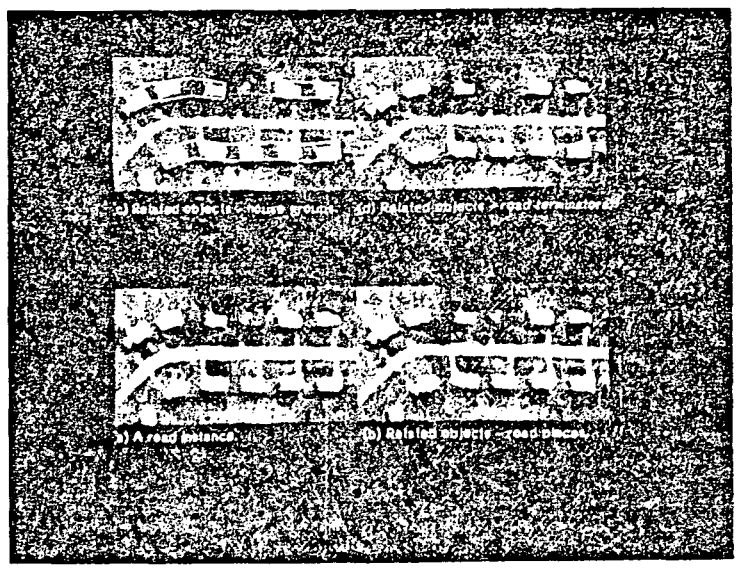


Figure 5-36. Explanation of a ROAD instance.

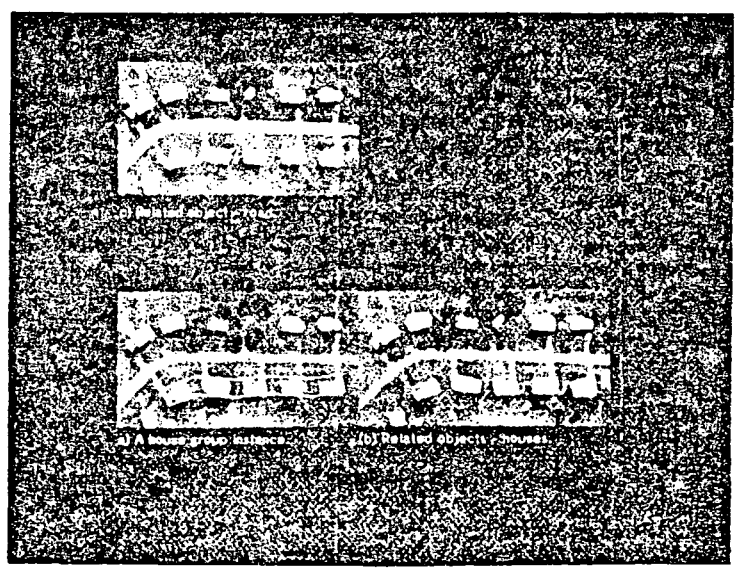


Figure 5-37. Explanation of a HOUSE GROUP instance.

INVESTIGATION OF CRITICAL ISSUES IN RECTIFICATION AND REGISTRATION OF SATELLITE SCANNER IMAGERY

by

Fidel C. Paderes, Jr.
Edward M. Mikhail

School of Civil Engineering
Purdue University
West Lafayette, Indiana 47907

NASA SYMPOSIUM ON
MATHEMATICAL PATTERN RECOGNITION AND IMAGE ANALYSIS
TEXAS A&M UNIVERSITY

June 10-11, 1985

ABSTRACT

An extensive mathematical model for rectification of satellite scanner data was developed. Using this model, factors affecting rectification accuracy were studied. Previous results included the effects of the following: (1) error in parameters singly and combined; (2) different mathematical models; (3) density of control points; (4) error in image coordinates; (5) error in ground coordinates of control points; (6) use of edge control in single image rectification; and (7) application of block adjustment. Current results include: (1) effect of errors in internal sensor geometry; (2) effect of error in weights of image and ground coordinates; (3) effect of different combinations of parameters defining the satellite position perturbations and the sensor orientation; (4) use of edge control in block adjustment; (5) study of rectification/registration sequence; (6) detection and identification of blunders; and (7) analysis of the potential for merging satellite scanner imagery and digital terrain model (DTM) data.

1. INTRODUCTION

The need for rapid and up to date acquisition of information pertaining to the earth and its atmosphere is increasing. One technology that shows promise in satisfying this need is the use of satellites to acquire remotely sensed data of the earth surface. Present day sensors on board satellites are capable of gathering enormous amounts of data in a timely fashion. Because of this, one pressing problem is the conversion of these data into useful information in an up-to-date and accurate manner.

Data from satellite sensors have found applications in many disciplines for identification, classification, and monitoring of earth features of interest. In all these applications, often there is need to integrate data from different sources including satellite data. This implies that all these data must be reduced into a common reference system which most often is earth based.

One type of sensor data which needs reduction to the earth surface in order to fully exploit its information content is the scanner type data. The process of defining the transformation required to relate scanner data arrays to the earth surface is called rectification. This process is an end in itself in the production and update of maps. In other applications, it is a necessary preprocessing step in order to obtain accurate results.

2. REVIEW OF LITERATURE

The earliest attempt at rectifying scanner data is through the use of polynomials to transform these data to the ground. With enough points of known image and ground positions called control points, this approach produces reasonable results with accuracy of up to a pixel in the image. Its main drawback is that positional accuracies are not uniform (Forrest, 1974; Trinder, 1976; Bahr, 1978; Dowman, 1981). An alternative approach called parametric, attempts to model the geometry of the scanning process itself. The simplest parametric model assumes that, within the image extent, the earth is flat and the satellite path is straight, which is often the case in conventional photogrammetric mapping (Kratky, 1972; Konecny, 1976; Dowman, 1981). The most comprehensive model considers the earth as an ellipsoid of revolution and the satellite path an ellipse (Mikhail, 1983; Paderes, 1983). In between, the earth can be assumed to be a sphere (Caron, 1975; Bahr, 1976; Sawada, 1981) and the satellite orbit a circle (Forrest, 1981; Levine, 1981; Synder, 1982). Deviation of the satellite position from the ideal can be assumed to be deterministic or random and modeled accordingly. The same is true for the attitude and azimuth of the sensor which ideally should be along the vertical and the flight path, respectively (Wiesel, 1984).

All the above-mentioned methods are solely based on ground control points. If the satellite position and sensor attitude

are known a-priori from other sources, i.e., satellite tracking data, then the transformation necessary for rectification can, in principle, be completely defined. Currently available tracking information cannot supply the required data with sufficient accuracy. Instead, these and other ancillary data are used in conjunction with control points to define the rectification process (Friedman, 1983).

3. MATHEMATICAL MODELING

We have developed a comprehensive parametric model which is based on the geometry of the scanner imaging process. This model assumes that the earth is an ellipsoid of revolution and that the path of the satellite is an ellipse. Deviation of the satellite position, sensor attitude, and sensor azimuth from the nominal are modeled as polynomial functions of time and any a-priori information regarding these deviations can be incorporated in this model. The model which is based on the premise that the ground point, the image point and the position of the satellite at the moment of sampling are collinear, is given by:

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = kM \begin{bmatrix} X - X_s \\ Y - Y_s \\ Z - Z_s \end{bmatrix}$$

where

$[x\ y\ z]^t$ is a vector defining the position of a point in the image coordinate system and is a function of the pixel row and column numbers, and the interior scanner geometry;

$[X\ Y\ Z]^t$ is the position of the corresponding ground point in a geocentric coordinate system;

$[X_s\ Y_s\ Z_s]^t$ is the satellite position, in a geocentric coordinate system at the moment of sampling and is a function of orbital parameters, deviation of the satellite position from the ideal and time;

M is a 3x3 matrix which brings the ground coordinate system parallel to the image coordinate system and is a function of

parameters defining the satellite orbit, the geometry of the earth, deviation of the satellite position from the ideal, the sensor attitude and azimuth and time;

k is a proportionality constant which varies from point to point.

This model can be used both for rectification and for creating simulated data which are very useful in the analysis of the rectification process.

4. MODEL VALIDATION

In order to study, analyze, and draw significant conclusions regarding the rectification process, an extensive set of image frames with suitable control data must be available. Furthermore, the control data must have known accuracy. To satisfy this requirement is time consuming and costly. An alternative which is both flexible and less expensive is to use simulated data. Assuming that the same model is used for both simulation and rectification, the main drawback of this approach is that rectification results are more accurate than they really are if the model used is not valid. This is because systematic errors introduced by the inadequacy of the model during simulation is canceled out in the rectification process. So, before using simulated data to study rectification, the relevant model must be validated.

Model validation requires at least a few real image frames with control data of known accuracy. These frames are rectified using only a part of the control data set. The remaining control can then be used as check points to independently verify the accuracy of rectification. The next step in validating a model is to produce simulated image frames similar in characteristics to the real ones. The given image coordinates from the real image frame, elevations of the corresponding object points, all exterior orientation parameters and constants of that frame are used in the model to calculate the horizontal coordinates of these object points. This consistent set of image and

corresponding ground coordinates are subsequently perturbed to realistically reflect the accuracy of the real control data set. The simulated frames are then rectified using the simulated control data set in exactly the same manner as the real frames. The last step in model validation is to compare the accuracy of rectification for the real image frames with the accuracy for the corresponding simulated frames. If there are no significant differences in accuracy between the real and its corresponding simulated frame, then the model is considered adequate.

We used two real image frames taken by LANDSAT 2 to validate our model. Precise estimates of the quality of the control data is not available but on the basis of the procedure used in obtaining the data, a reasonable estimate of the standard deviations of coordinates is as follows: 0.5 pixel in row, 0.5 pixel in column, 15 meters in Northing, 15 meters in Easting, and 15 meters in elevation. For manual methods of control point identification, which is the one we used, the best accuracy that can be expected is 1/3 pixel in row and 1/3 pixel in column (Bahr, 1976). For the first frame which covers Kansas, the standard deviations applied in simulation are 0.44 pixel in row, 0.40 pixel in column, and 15 meters for each ground position coordinates. The RMS planimetric error in rectification for the real frame is 64 meters and for the simulated frame is 62 meters using 81 control and 72 check points. The second frame covers Louisiana and the standard deviations applied in simulation are 0.40 pixel, 0.64 pixel, and 15 meters for row, column, and each ground

coordinate, respectively. Using 70 control and 122 check points, the RMS rectification errors are 68 and 61 meters for the real and simulated frames, respectively. A detailed discussion of the above experiment can be found in Paderes and Mikhail (1984). From these, it can be concluded that our model is adequate and that it may produce only a very small systematic error, if at all, when used for rectification.

5. EXPERIMENTAL RESULTS

We have carried out an exhaustive series of experiments using simulated data to understand and clarify problems regarding rectification. Simulation is a very powerful and flexible tool whenever it can be appropriately applied as is the case here. Previous results include the study of the effects of the following on rectification accuracy: (1) error in parameters singly and combined; (2) different mathematical models; (3) density of control points; (4) error in image coordinates; (5) error in ground coordinates of control points; (6) use of edge control in single image rectification; and (7) application of block adjustment. These results are discussed in detail in Mikhail and Paderes (1983), Paderes and Mikhail (1983), Paderes, Mikhail, and Forstner (1984), and Paderes and Mikhail (1984). New results which are reported in this paper include: (1) effect of errors in the internal sensor geometry; (2) effect of error in weights of image and ground coordinates; (3) effect of different combinations of parameters defining the satellite position perturbations and the sensor orientation; (4) use of edge control in block adjustment; (5) study of rectification-registration sequence; (6) detection and identification of blunders; and (7) analysis of the potential for merging satellite scanner imagery and digital terrain model (DTM) data.

5.1 Effect of Errors in Internal Sensor Geometry

Errors in sensor geometry are primarily due to variations in scanning speed which should be constant during pixel sampling. This error plus other sensor instabilities cause errors in pixel row and column numbers. Since one scan consists of very few rows relative to the number of columns, and the scanning action primarily affects the columns only, then row errors are very small compared to column errors. Assuming that errors in internal sensor geometry constitute the factor that limits observation accuracy, an experiment was designed to determine the rectification accuracies that can be expected. For this purpose, an image frame was simulated with 100 uniformly distributed control points and the same number of well-distributed check points. The ground positions of the control points were perturbed using the normal distribution with only a one meter standard deviation in each of the three coordinate directions. The pixel row numbers were perturbed using the normal distribution with standard deviation of 0.01 pixel. The pixel column numbers were also normally perturbed with a series of standard deviation as seen in Table 1. The image is then rectified and the accuracy computed using the check points. The experiment was repeated three times with a different "seed" for the random number generator used for deriving the errors applied to the observations. The average RMS planimetric error of rectification corresponding to the different image column standard deviations are shown in Table 1. It can be

seen that sub-pixel accuracy is possible only if the scanning speed can be controlled to a very high degree of accuracy.

5.2 Effect of Errors in Variances of Image and Ground Positions

Ideally, only the true variances of observations should be applied in an adjustment problem. In reality, difficulties in determining the true accuracy of observations prevent us from doing so, especially if more than one type of observation is involved. Rectification of scanner data is largely an adjustment problem, and at least two different types of observations are involved (i.e. image positions and ground coordinates). To study the effect of errors in variances, a nine frame block in three adjacent orbits with three frames per orbit was simulated. The center of the block is approximately at 60°N latitude resulting in about 60% sidelap between frames belonging to different orbits. Adjacent frames in a single orbit have an artificial 15% overlap. There are 506 control points uniformly distributed over the whole block. The ground positions were normally perturbed with a standard deviation of 15 meters in each of the three coordinate directions. The image positions were perturbed using a combination of uniform and normal distribution. The uniform distribution had a range of +0.5 to -0.5 pixel and the normal one had a standard deviation of 0.5 pixel. Each frame contains 100 check points. Different cases, where either the image or ground position variances but not both were multiplied by a different factor, were run. The RMS planimetric error for each frame was computed using the check points and averaged over all nine frames.

The results are shown in Table 2. The image position variances can be in error by a factor of 0.1 or greater while the ground position variances can be in error by a factor of 10 or smaller. This means that the image position can be assumed to be less accurate than they really are and the ground coordinates to be more accurate than they really are or even assumed fixed without affecting rectification accuracy.

5.3 Effect of Different Parameter Combinations

One major problem in the rectification of satellite scanner imagery using a model elaborate enough to adequately describe the scanning process is caused by the very weak geometry of the image. Because of this, the parameters in the model are correlated with each other. In practice, therefore, only a subset of the total parameter set can be recovered in the adjustment. The modeling approach we used, which alleviates this problem, is to divide the satellite position and sensor attitude and azimuth into two components, i.e., ideal and perturbed. The ideal satellite position can be derived from satellite tracking data, the ideal sensor attitude can be assumed to be always in the direction of the vertical and the sensor azimuth to be parallel to the orbital plane. If no tracking data are available, those ideal satellite position parameters which vary from orbit to orbit can be derived using the control data set itself, assuming no perturbations. Errors in ideal satellite position, sensor attitude, and sensor azimuth, from whatever source, can be compensated for by the perturbation parameters. Therefore, only those parameters

describing the perturbation components need be considered as unknowns in the rectification process. The deviation of the satellite position from the ideal has three components, the sensor attitude with respect to the vertical has two and the sensor azimuth has one. Of these six components, only four are independent because the sensor attitude is highly correlated with the along and across track components of the satellite position deviation. Nevertheless, for high accuracy applications, the parameters defining all six components should be recovered. This experiment is designed to determine when all six components can be recovered and compare the accuracy of this approach to the case when only four components are used in rectification.

A block similar to that used in section 5.2 was simulated. The initial approximations used in the adjustment for the parameters defining the six perturbation components are their true values plus a given error as shown in Table 3. Since a given component is modeled as a third degree polynomial function of time (having four parameters), the component standard deviation is actually twice the individual parameter standard deviation. For case 1, all the six components were exercised in the adjustment. For parameter weights smaller than their true weights times 125, the solution did not converge except when the error of the parameters are equivalent to 0.1 pixel or smaller. In the latter case, the solution did converge using the true parameter weights. For case 2, the along and across track satellite position deviation components were held fixed and the remaining four

components were given very small weights in the adjustment. The along and across track components were selected because among the six perturbation components, these two contribute the smallest error in rectification. All 506 points in the block were used as control for both cases. For computing the rectification accuracy, each frame had 100 check points. The RMS planimetric errors were averaged over all frames. The results are shown in Table 3. Case 1 is superior to case 2 when the error for each parameter is smaller than 0.1 pixel and the opposite is true when the error is greater than 1 pixel. This experiment shows that if the solution to the adjustment converges using the true parameter weights, exercising all the six components is superior to using only four.

5.4 Use of Edge Control In Block Adjustment

A block of nine frames in three adjacent orbits with three frames per orbit similar to that in Section 5.2 was simulated. There are 360 control and 140 tie points uniformly distributed over the whole block. The ground coordinates of the control points were perturbed by a 15 meter standard deviation in each of the three coordinate directions using the normal distribution. The tie points ground coordinates were similarly perturbed except that the horizontal positions have standard deviation of 1000 meters. The image position of both the control and tie points were perturbed using a combination of uniform and normal distribution. The uniform distribution, which takes care of truncation errors, had a range of +0.5 to -0.5 pixel. The normal

distribution had a standard deviation of 0.5 pixel both in row and column direction. A second set of control data representing control and tie edges were simulated in exactly the same manner as the control point set except that the image coordinates were perturbed in a different way. Instead of perturbing the image position along the row and column directions, they were perturbed along a randomly directed line. The standard deviation along the line was 10 pixels and that perpendicular to the line was 0.5 pixel. The perturbations along the row and column direction were then derived by rotation given the direction of the line. A third set of data consisting of check points were simulated. The image and ground coordinates of these points were self consistent and they were used for computing the accuracy of the rectification procedure. Each frame had a set of 100 check points which were independent of other frames.

By varying the number of control and tie points/edges a total of 12 cases of block adjustment were run; 7 using points and 5 using edges as control. The whole experiment was repeated three times with a different "seed" for the random number generator which computed the perturbations. For each case, by using the check points, the RMS planimetric error for each frame is computed. The RMS planimetric error is averaged over nine frames and three replications. The results are plotted in Figure 1. From the figure, it can be seen that approximately a pair of edge control is equivalent to a single point control. This is the theoretical limit because a point is the same as a pair of per-

pendicular edges. This conclusion seems to be in conflict with previous results for edges with random directions which indicated that about three pairs of edges is equivalent to a single control point (see Paderes, Mikhail, and Forstner 1984). The more accurate results of the present experiment can be explained by the fact that the error along the edges was only 10 pixels instead of being infinitely large as was previously assumed. A 10 pixel error in measuring edge position (taken as the mid-point of the edge of finite length), especially if it is short enough, is quite achievable. A practical means to locate the edge-points on the map is to first get their approximate locations in the image using a simple transformation with a few control points for the whole frame. After that, each edge-point is then manually shifted to lie on the edge.

5.5 Study of Rectification/Registration Sequence

Rectification has been defined as the transformation of the scanner images into the ground reference system or into a scaled representation of the terrain such as a map. Registration, on the other hand, is the transformation of one or more images into another image covering the same segment of the earth. The images to be registered can be taken by similar or entirely different sensors (multi-sensor) at approximately the same or vastly different times (multi-temporal). For proper registration, the relevant images should preferably be taken from approximately the same sensor locations although those that are not can in principle be taken care of if the terrain shape is known.

In theory, if images covering the same segment of the earth are rectified, they should then also be registered with respect to each other. Conversely, if these images are registered with respect to each other and if one of these is rectified, then the rest should also be rectified. At first glance, the process of registration is superfluous because rectification alone can produce both rectified and registered images. In practice, registration stands on its own since it is considered to be more accurate because it is easier to find common features between images, than between an image and the corresponding terrain segment or its representation. This is especially true if the images were taken by similar sensors under approximately the same conditions. Furthermore, if matching images is the sole object, registration is more efficient than rectification.

Like rectification, the first step in registration is the finding of common features between images to serve as control. Then the rest of the images are transformed into the arbitrarily selected reference image using a mathematical model. Ordinarily the resulting system of equations is over-determined, therefore a method of adjustment is necessary (e.g. least squares). Since registration involves a minimum of two images the resulting geometric model will be very complex. For cases similar to this, the usual approach is to use polynomials or other mathematical series. This approach to registration, which is the approach we used, is feasible because of the relative ease of finding common features between similar images.

We performed a series of five experiments to study the utility of rectification for registration purposes and the usefulness of registration for rectification. This study is feasible because of our extensive set of simulation programs.

Experiment I: Registration (Transformation) of Frame A to Frame B

Experiment I was designed to measure the accuracy of registering one image to another. The following are the steps in this experiment:

(1) Select a suitable set of nominal orbit, sensor and earth parameters and constants such that the resulting simulated images are located at approximately 60^{o} N latitude.

(2) Add a positive error equivalent to one pixel in the image to each of the nominal parameters and assign them to frame A.

(3) Select, on frame A, 16 control and 225 check points that are uniformly distributed throughout the whole frame.

(4) Compute in a forward simulation procedure the planimetric ground coordinates of the above image points, using the parameters of frame A, and assuming that their ground elevations are zero.

(5) Add a negative error equivalent to one pixel in the image to each of the nominal parameters and assign them to frame B. This step together with step (2) assures us that the two frames are overlapping each other, with only a few pixels difference.

(6) Using the ground coordinates from step (4) and the parameters of frame B, compute the image coordinates of the control and check points in frame B in a reverse simulation procedure. This step results in a consistent set of control and check points between frames A and B.

(7) Perturb the image coordinates of control points only in frame A selected in step (3) using a normal distribution with standard deviation of 0.1 pixel.

(8) Repeat step (7) for image coordinates of control points in frame B, which were computed in step (6).

(9) Compute the registration parameters needed to transform frame A to frame B using the simulated control points and a second degree polynomial model.

(10) Transform the image coordinates of the check points in frame A, computed in step (3), into frame B.

(11) Compute the rms of the position errors of check points in pixels from the differences between the computed check point position in step (10) and the ideal check point position from step (6).

(12) Repeat steps (1)-(11) five times with different perturbations applied to control data and compute the average rms error.

(13) Repeat steps (1)-(12) for 25, 49, 81 and 144 control points. A plot of the average rms check point position errors vs. the number of control points is shown as curve (1) in Figure 2.

Experiment II: Rectification of Frame A

Experiment II was done to determine the accuracy of single frame rectification. This experiment consists of the following steps:

(1) Repeat steps (1)-(4) in experiment I resulting in a consistent set of image and ground coordinates for 16 control points and 225 check points in frame A.

(2) Perturb the image coordinates of the control points only in frame A using a normal distribution with a standard deviation of 0.1 pixel in the row and column directions.

(3) Perturb the corresponding control point ground coordinates in each of the three coordinate directions using a normal distribution with 5m standard deviation.

(4) Compute the rectification parameters needed to transform frame A into the ground system using the perturbed control points and our rectification model via a least squares adjustment procedure.

(5) Transform the image coordinates of check points from step (1) into the ground in a forward simulation procedure using the computed rectification parameters in step (4).

(6) Compute the rms of the planimetric position errors in meters of check points from the differences between the computed check point position in step (5) and the ideal check point position resulting from step (1).

(7) Repeat steps (1)-(6) five times with different perturbations applied to control data and compute the average rms position error of check points.

(8) Repeat steps (1)-(7) for 25, 49, 81 and 144 control points. The results are plotted as curve (2a) in Figure 3.

(9) Repeat steps (1)-(8) using 0.5 pixel and 15m standard deviations in steps (2) and (3), respectively. Curve (2b) in Figure 3 is the plot of these results.

Experiment III: Independent Rectification of Frames A and B.

In experiment I frame A was registered to frame B. In experiment II frame A was rectified to the ground. If experiment II is repeated for frame B, then the two frames should be registered with respect to each other. These two rectifications, which result from experiment III, should be compared to the result from experiment I. The steps in this experiment III are as follows:

(1) Repeat steps (1)-(6) in experiment I which results in a consistent set of image and ground coordinates for 16 control and 225 check points in frames A and B.

(2) Repeat steps (2)-(4) in experiment II resulting in rectification parameters for frame A.

(3) Transform the image coordinates of check points in frame A into the ground using the computed rectification parameters in step (2) via a forward simulation procedure.

(4) Repeat step (2) for the rectification of frame B.

(5) Compute the image coordinates of check points in frame B using the computed ground coordinates in step (3) and the rectification parameters in step (4) via a reverse simulation procedure.

(6) Compute the rms of the check point image position errors in pixels from the differences between the image coordinates of check points computed in step (5) and the corresponding true positions in frame B computed in step (1).

(7) Repeat steps (1)-(6) five times with different perturbations applied to control data and compute the average rms position error of check points.

(8) Repeat steps (1)-(7) for 25, 49, 81 and 144 control points. The average rms errors vs. the number of control points are plotted as curve (3a) in Figure 2.

(9) Repeat steps (1)-(8), in this experiment, using 0.5 pixel and 15m standard deviations in step (2) for image and ground coordinate perturbations, respectively. The results are plotted as curve (3b) in Figure 2.

Experiment IV: Registration of Frame A to B, Followed by Rectification of B.

In experiment I frame A was registered and transformed to frame B. If experiment I is followed by rectification of frame B to the ground, then frame A is also rectified. The sequence of registration followed by rectification is then equivalent to a simple rectification of frame A. Experiment IV was performed to measure the accuracy of this sequence. The following steps were done in this experiment:

(1) Repeat steps (1)-(4) in experiment I resulting in a consistent set of image and ground coordinates for 16 control and 225 check points in frame A.

(2) Repeat steps (5)-(6) in experiment I which produces image coordinates in frame B for control and check points which are consistent with those in frame A.

(3) Repeat steps (7)-(10) in experiment I which results in transformed check point image coordinates from frame A to frame B.

(4) Repeat steps (2)-(4) in experiment II for frame B instead of frame A which results in rectification parameters for frame B.

(5) Transform the image coordinates of check points produced by step (3) into the ground using the rectification parameters computed in step (4) in a forward simulation procedure.

(6) Compute the rms of the planimetric position errors in meters of check points from the differences between the computed check point position in step (5) and the true check point planimetric ground position in step (1).

(7) Repeat steps (1)-(6) five times with different perturbations applied to control data and compute the average rms check point error.

(8) Repeat steps (1)-(7) for 25, 49, 81, and 144 points. The average rms check point position errors vs. the number of control points are plotted in Figures 3 and 4 as curve (4a).

(9) Repeat steps (1)-(8) using 0.5 pixel and 15m standard deviations in step (4) for image and ground coordinate perturbations, respectively. A plot of the results similar to those in step (8) is shown in Figures 3 and 4 as curve (4b).

Experiment V.

Experiment V is essentially experiment III except that the registration errors between frames are computed on the ground instead of in the plane of frame B. This was done to facilitate comparison between the results of this experiment and experiment IV.

This comparison is interesting because both experiments deal with a sequence of two processes. For completeness the steps in experiment V are as follows:

(1) Repeat step (1) in experiment III.

(2) Repeat steps (2)-(3) in experiment III for frame A resulting in transformed ground coordinates of check points.

(3) Repeat step (2) for frame B.

(4) Compute the rms of the differences between the computed check point planimetric ground position in step (2) and in step (3).

(5) Repeat steps (1)-(4) five times with different perturbations applied to control data and compute the average rms position difference.

(6) Repeat steps (1)-(5) for 25, 149, 81 and 144 control points. The results of this experiment are shown as curve (5a) in Figure 4.

(7) Repeat steps (1)-(6) using 0.5 pixel and 15m standard deviations in step (2) for image and ground coordinate perturbations, respectively. The results are also shown in Figure 4, as curve (5b).

The above series of five experiments essentially covered two major cases. Case (a) assumed that the image coordinates of common points for both rectification and registration have the same standard deviation of 0.1 pixel. This implies that

correspondence for both registration and rectification can be accomplished at the same level of accuracy. Case (b), on the other hand, assumed that the image coordinates of common points for registration have 0.1 pixel standard deviation, while those for rectification have 0.5 pixel standard deviation. This case stems from current practical considerations where correspondence between like images (thus registration) is determined to a higher degree of accuracy than for rectification.

From the results of these experiments, if the common points for rectification have the same accuracy and number as those for registration (case (a)), it can be concluded that rectification is superior to registration. Under the more realistic assumptions in case (b), it can be concluded that if the sole purpose is to register two similar images taken from nearly the same sensor location, then direct registration is better than the indirect approach of rectifying both images. On the other hand, if rectified images are the desired results, then rectification should be the only procedure used. If both rectified and registered images are desired, the pure rectification approach is as accurate as the combined registration-rectification approach, still under the assumptions of case (b).

5.6 Blunder Detection and Identification

In any system involving observed data like rectification and registration of satellite scanner imageries, the elimination of blunders in the observations is of utmost importance for reliable

and accurate results. Ideally, if the true errors of observation are known or can be computed, they can readily be tested for blunders. Unfortunately, this is not possible so, a traditional approach has been to attempt minimizing blunders before the adjustment and reduction of data. This usually involves a carefully designed observation scheme with repetitive measurements to assure that as few blunders as possible are left undetected. A limited version of this approach should always be applied but full implementation is seldom done because of cost considerations. Hence, blunders remain in many instances which can considerably degrade the quality of the resulting products.

An alternative approach is to do statistical testing on functions of true errors after data adjustment and reduction. It had been shown (see Mikhail, 1979), that the residuals, v, resulting from a least squares adjustment are related to the true errors, e, hence to blunders, by the following equation:

$$v = - Q_{vv} W e$$

where Q_{vv} is the cofactor matrix of residuals, v, and W is the inverse of the cofactor matrix of observations, Q. The cofactor matrix, Q, is related to the covariance matrix of observations, E, by the following relation:

$$E = \sigma_0^2 Q$$

where σ_0^2 is the a-priori reference variance. This equation cannot be solved for e, although all the other quantities are known after a least squares adjustment, because Q_{vv} is singular. Since

v is a linear function of e, then v or functions of v can be tested for blunders. This post adjustment approach is the technique which we applied for blunder detection and identification.

Blunder detection only requires that we determine whether the vector of observations has blunder(s) in it. In order to eliminate blunder(s) we have to go one step further and identify the specific elements of the vector of observations which have blunders. In this context, multivariate statistics which are functions of v are only useful for blunder detection but not for identification. Paradoxically, univariate statistics which are functions of individual elements of v has the best chance of identifying individual blunders.

One commonly used statistic for blunder identification is the normalized residual

$$v_i' = v_i / \sqrt{\sigma_0^2 \, q_{vivi}}$$

where v_i is a specific element of the vector of residuals, v, and q_{vivi} is the i^{th} diagonal element of the cofactor matrix of residuals, Q_{vv}. If the original vector of observations, l, is normally distributed, v_i' is also distributed normally with zero mean and unit variance. The method based on this statistic is known as data snooping (Baarda, 1968; Mikhail, 1979).

If σ_0^2 is not known, we can use the a-posteriori estimate of the reference variance, $\hat{\sigma}_0^2$, in its place resulting in

$$v_i'' = v_i / \sqrt{\hat{\sigma}_0^2 \, q_{vivi}}$$

the a-posteriori reference variance, $\hat{\sigma}_0^2$, can be computed using the equation

$$\hat{\sigma}_0^2 = v^t W v/r$$

where r is the redundancy of the adjustment. This statistic v_i'', assuming normally distributed observations, has a tau distribution. The tau distribution can be derived from the Student t distribution using the following relationship:

$$T = \sqrt{r}\, t(r-1)/\sqrt{r-1 + t(r-1)^2}$$

where T is the tau distribution, t is the Student t distribution, and r designates the degree of freedom or redundancy of the adjustment (Pope, 1975).

Another useful statistic is the partial quadratic

$$q = v_2^t Q_{v2v2}^{-1} v_2/\sigma_0^2$$

where v_2 is a sub-vector of residual vector, v, and Q_{v2v2} is the corresponding submatrix of cofactor, Q_{vv}. The statistic q is distributed as a $\chi^2(p)$, where p is the number of elements in v_2 which can never exceed r, the redundancy of the adjustment. If p = 1, the partial quadratic approach is equivalent to the data snooping approach (Stefanovich, 1978).

The v_i' and v_i'' statistics are ideally suited for blunder identification when the vector of observations, l, has only one blunder, because they essentially are univariate statistics. For more than one blunder in the observations, these statistics do not perform as well. To alleviate this problem, we developed a

sequential blunder identification strategy based on these two statistics where the blunders are identified and eliminated one at a time. A shortcoming of the present implementation of the strategy is that once an observation is eliminated, even if it has no blunder, it can not be returned into the adjustment. Since the statistic is most sensitive when there is only one blunder in the set of observations, the eliminated observations should be returned one by one into the adjustment and retested.

The statistic q is multivariate, hence it can usually detect but can not identify blunders in a subset of observations. Stefanovich (1978) developed a search strategy using the q statistic which can identify the subset of observations that contains only blunders. The strategy is based on the fact that the statistic q for the subset of observations with no blunders will pass the chi-squared test while the subset containing only blunders will fail the test. Any other subdivision of the set of observations will fail to satisfy the above conditions. A major drawback of this strategy is that the chi-squared test becomes insensitive when the redundancy is large.

Stefanovich's strategy and the sequential strategy we developed to identify blunders require the repetitive elimination of one or more observations from the adjustment. At first glance, this would require repetitive readjustments which would be costly. A closer look will show that the only quantities we use in the tests which varies with the number of observations in the adjustment are the residuals, v, and the cofactor of

residuals, Q_{vv}. The estimated reference variance, $\hat{\sigma}_0^2$, also varies, but this is essentially a function of v. The cofactor of observations, Q, which is used in computing for the estimated reference variance, $\hat{\sigma}_0^2$, and the a-priori reference variance, σ_0^2, are constants. It turns out that the quantities v, Q_{vv} and $\hat{\sigma}_0^2$ can easily be updated after eliminating some observations without readjustment (Stefanovich, 1978). Also, only a subset of v which contains the observations that are most likely to have blunders and the corresponding subset of Q_{vv} need be stored and updated.

To test the effectiveness of the strategies outlined above for identifying blunders in control points used in rectification of satellite scanner imageries, two simulated MSS image frames were created. Frame A has 25 control points frame B has 49, both control point sets being uniformly distributed. The coordinates of image points for both frames were perturbed using a normal distribution with standard deviation of 0.5 pixel and a uniform distribution with a range of +0.5 to -0.5 pixel to take care of truncation errors. The corresponding ground coordinates were assumed fixed without loss of generality because whatever errors the ground coordinates have, these can be compensated for at the image positions. The two frames are at approximately 60° N latitude.

The level of significance for the tests were: 0.0005, 0.0005, and 0.005 for data snooping, tau test, and chi-squared test, respectively. The first two are two-sided tests while the

last is one-sided. These values were selected such that there is no misidentification when the control data sets have no blunders. Three different numbers of blunders (1, 2 and 4) were tested for both frames. The blunders were introduced on the row coordinates of image points only. If a coordinate is identified as having a blunder, the whole point is eliminated. The single blunder was introduced near the middle. The two blunders were introduced along a diagonal and one quarter of the diagonal length from the corners. The four blunders were introduced along both diagonals in a manner similar to the two blunders.

Results of the experiment are shown in Table 4. Methods 1, 2 and 3 correspond to data snooping, tau test and chi-squared (or partial quadratic) test, respectively. Entries in the table are the smallest blunder for which a given strategy identified all blunders correctly without misidentification at the selected levels of significance. This implies that whenever blunders are larger than those shown, they are always detected. If smaller, they may or may not be detected. The upper entry corresponds to 25 control points while the lower entry corresponds to 49. The row entries vary with the number of blunders and the column entries vary with the methods.

The results show that post adjustment blunder identification is feasible especially for large blunders with magnitudes of 10 or more. The procedure is expected to work quite well if the number of control points, hence the redundancy, is high and vice versa. It worked quite well for 25 control points, where the

redundancy is 34 even when there are four blunders. Further tests are necessary to determine the lower limit in the number of control points for the procedure to work well. As expected the data snooping and the chi-squared test performed a little better than the tau test because the σ_0^2 is perfectly known. In this context, the performance of the tau which uses $\hat{\sigma}_0^2$ instead of σ_0^2 is very good.

5.7 Analysis of the Potential for Merging Satellite Scanner Imagery and DTM Data

Digital terrain models or DTMs are becoming more and more common. A DTM is a digital representation of the topography or shape of the terrain as opposed to the conventional graphical representation in terms of lines of equal elevation called contour lines. DTMs essentially consist of a collection of three-dimensional vectors representing the horizontal position and elevation of points. These points might be arranged in a regular grid which is more common, or they might be arranged in an arbitrary manner. The density of these points depends on the character of the terrain and the ultimate application of the resulting data. Sometimes, other planimetric features such as roads may also be incorporated into the DTMs.

Since the terrain is continuous, representing it as a collection of discrete points may not be sufficient to completely describe the terrain. This inadequacy becomes very apparent when terrain points other than those available in a DTM are required.

Because of this, the definition of DTMs is sometimes extended to include the procedure used for interpolating the elevations of these other points. As a further consequence of this discreteness, DTMs can be stored in a more compressed form using suitable interpolation models.

Relatively speaking, the shape of the terrain does not change compared to the planimetric features on its surface. Once collected, the elevation component of DTMs need not be updated for a relatively long period of time except in cases where more accurate terrain models are required. This relative stability of DTMs is a blessing because the shape of the terrain is often more difficult and time consuming to observe and measure. If the terrain shape were to change as much as its planimetric features, the resulting DTMs might become obsolete by the time their compilation is finished. Furthermore, only one type of sensor, the photogrammetric camera, is suitable for securing images useful for compiling DTMs.

With the availability of satellite-borne sensors, up-to-date images useful for mapping the surface of the earth became available. It seems that the problem of up-to-date maps may finally be nearing solution. As it turns out, because these images are taken from very high altitudes and the angular coverage is usually small, the resulting image geometry is such that the shape of the earth surface cannot be readily recovered from them. So, the primary information that is recovered from these images consists of the planimetric features of the earth's surface. Even

the proper positioning of these features in the ground system, i.e. rectification, requires some knowledge of the shape of the terrain.

A complete description of the terrain requires both its shape and the planimetric features on it. The shape of the terrain can be supplied by DTMs which are compiled through a photogrammetric process which is relatively tedious and time intensive. Since the terrain shape does not change much with time, the resulting DTMs are useful for a variety of applications and for a relatively long time. The planimetric features can be supplied by more modern sensors on-board satellites. Even though planimetric features change rapidly, these sensors are able to provide us with timely images.

The above discussion leads to the necessity of merging or registering DTMs and satellite images in order to produce complete and up-to-date terrain data. In general, two different entities can be merged only if they describe the same phenomenon. This is true for satellite imagery and the DTM covering the same segment of the earth surface. The first step in merging, which is very similar to rectification, is to find the sensor position and angular orientation as a function of time. Presently, these can be provided by satellite tracking observations and by auxiliary sensors on-board the satellite. Unfortunately the accuracy of these observations are not sufficient for merging DTM and

satellite imagery. In the design of the next generation of sensors, effort should be expended to accurately measure the sensor position and its angular orientation.

An alternative is to use common features between the DTM and the imagery to solve for the sensor position and angular orientation. This is quite similar to rectification. At first glance, satellite imagery and DTM cannot be merged because they do not describe the same property of the terrain. The former describes the planimetry of the terrain while the latter describes the shape of the terrain. Fortunately, DTMs may also contain some planimetric features such as roads which do not change as rapidly as other features such as vegetation. The problem of efficiently and accurately finding these common features has to be resolved before any viable merging procedure can be implemented.

Once the sensor exterior orientation parameters are known, the image coordinate of any ground point, hence any DTM point, can be solved for. This procedure is very similar to reverse simulation of satellite image point. The solution is iterative because the sensor angular orientation and position are functions of time. Time, in turn, is a function of image position, which is the unknown quantity. The resulting equations are highly non-linear in terms of the image coordinates. This approach of solving for the image coordinates for DTM points is appropriate if we wish to maintain the point density of DTMs.

The next step in merging is to assign densities to image points corresponding to DTM points. In general, these points will not coincide with pixel centers, hence an interpolation method is needed to assign the proper density to these points. The simplest method is the zero order interpolation also known as the nearest neighbor interpolation. As the name implies, the computed image point is assigned a density equal to that of the nearest pixel center. Higher order interpolation such as bi-linear, bi-cubic, etc. can also be applied. Questions regarding the resampling of satellite imagery need to be addressed.

Instead of solving for the image coordinates of DTM points, we can solve for the ground coordinates of the image pixel centers. The spectral densities of these points will automatically be the densities of the corresponding pixels. This approach makes sense if we want to maintain the point density of the image which corresponds to pixels. This approach is very similar to rectification whereby we are only interested eventually in the horizontal ground position of pixels. The solution for the three ground coordinates of image points given the sensor angular orientation and position is not possible without some knowledge of the shape of the terrain. This is because we are trying to transform a 2-dimensional image into the corresponding ground segment which is 3-dimensional.

The above-mentioned problem can be visualized as that of finding the intersection of a vector and a complex surface in 3-dimension, the surface being represented in digital form. This

problem is complicated because we are interpolating at the same time that we are solving the intersection problem for the discrete surface. We might simplify this problem by representing the terrain as a continuous surface using models such as B-splines. In rectification, if we do not have a DTM, a way around this problem is to assume that the terrain is flat but not necessarily horizontal. For MSS imagery the horizontal error in this assumption is negligible compared to the pixel size except for very mountainous regions where the angular coverage is very small (less than 11 degrees).

Whether we maintain the integrity of the points in the DTM or the pixels in the image depends on the eventual application and on the relative density of the two data sets. If the final end product is a rectified image and if DTM points are denser than the pixel density, then the ground coordinates of pixel centers should be solved for. If the final product is still a DTM and pixels are denser than DTM points, then the image coordinates of DTM points must be solved for. The critical point to consider is the accuracy of interpolation, whether implicit or explicit. The interpolation should preferably be from dense to less dense point distribution.

The possibility of merging DTM data and satellite scanner imagery is based on the premise that the sensor angular orientation and position is available and/or can be computed using features common to both data sets. These common features are usually called control. Therefore, any approach that will make

control selection and measurement faster, more accurate, and will decrease the required number of control features should be investigated. Identifying common features between the DTM and satellite imagery is difficult because these two sets describe inherently different aspects of the terrain. DTM primarily describes the terrain shape with a few selected planimetric features added while satellite imagery describes its planimetric features. These few planimetric features incorporated into DTMs are the only features that the DTMs have in common with the satellite imagery. The situation is worsened by the fact that any planimetric feature in a DTM is represented by lines whereas those in an imagery is continuous.

The problem of dissimilar representation of available common features can be solved by filtering the imagery using differential filters to produce binary images consisting of lines and edges. This binary image is more similar to planimetric features in DTMs than the original continuous image. If the original photographs used in compiling the DTMs are available, these might be digitized and correlated with satellite images to find common points. These photographs are more similar to satellite images, hence more common features can be found. Common features between DTM source photographs and satellite imagery can be used for merging DTM and satellite imagery because these photographs are registered with the DTM.

With the advent of space photography (such as the Large Format Camera on-board the Space Shuttle) merging of DTMs with

satellite imagery is, at least theoretically, made easier. The first step in merging DTMs with satellite imagery using space photographs is to first merge DTMs and the corresponding space photos. The required number of common features in this case is very few (minimum of three control points) because the geometry of space photographs is much stronger compared to satellite scanner imageries. Then the space photo is merged with the corresponding satellite image. Common features between space photos and satellite imagery is much easier to find because both are continuous images of the terrain. Space photographs are more efficient than the DTM source photographs as tools for merging DTM with satellite imagery because of scale. A single space photo, for example, covers almost the same area as a single frame of MSS imagery whereas a large number of DTM source photographs is needed to cover the same area.

In selecting common features between images, the primary tool in matching these features is the use of correlation algorithms. Advanced correlation algorithms are capable of compensating for scale differences, differences in direction of digitization and higher order distortion. Procedures are available also for correlating images with different pixel sizes. Nevertheless, because correlation is central to the measurement of common features, more study and experimentation are needed in this area for our specific application.

Theoretically, the number of common features needed as control for merging or registering DTMs and satellite imageries will

be reduced if overlapping frames of imageries taken from different perspective positions are available. This is because features common to overlapping imageries but not found in the DTM can be used to strengthen the geometry of each individual imagery. These are commonly called pass features. The procedure of using overlapping imageries is called block adjustment. For imageries taken by Landsat MSS or other similar scanners, where the base-height ratio is very small (for overlapping strips if there is any overlap at all), the promise of block adjustment can not be fulfilled. However, this procedure might be advantageous for imageries produced by scanners whose direction can be remotely controlled like those on-board the Spot satellite for example.

6. CONCLUSIONS

Error in the reconstruction of the scanner interior geometry results in image position errors primarily along the column direction which severely limit the obtainable accuracy through rectification. Unless this problem is corrected, highly precise control data sets, even if available, will not be effective. The question of proper weights for image and ground position of control features is easily resolved because we can assume that the image positions are less accurate and the ground positions are more accurate than they really are without adversely affecting rectification accuracy. Regarding the proper parameter combination that should be recovered during rectification, ideally, all six perturbation component parameters should be used. However, unless these parameters are known to within 0.1 pixel equivalent error, fixing the along and across track perturbation components produces more accurate results. Edges are very effective substitutes and/or complements for points as control. Our results show that a pair of edges is equivalent to a point under certain conditions. When correspondence for rectification is established at the same level of accuracy as for registration, then image rectification, for whatever purpose, will always be superior. Given the present capabilities for measuring the positions of common points, single rectification should be used when rectified imageries are primarily the desired results. Direct registration should be used when the registered imageries are of primary interest. Double rectification in general is as accurate as the

registration/rectification sequence. With respect to blunders, post adjustment identification is feasible in rectification of single image, especially for relatively large blunders. Digital Terrain Models (DTMs) can effectively be combined with remotely sensed imagery. This procedure may provide suitable means for rapidly updating maps.

7. ACKNOWLEDGEMENTS

This research was funded by NASA under Contract No. 9-16664 through Texas A&M Research Foundation Subcontract No. L200074. The assistance and guidance of Dr. R. P. Heydorn of NASA Johnson Space Center and Professor L. F. Guseman, Jr. of Texas A&M University are appreciated.

8. REFERENCES

Baarda, W., 1968. A Testing Procedure for Use in Geodetic Networks, Netherlands Geodetic Commission, Volume 2, Number 5.

Bahr, H.P., 1976. Geometrical Models for Satellite Scanner Imagery, 13th ISP Congress, Commission III, Helsinki.

Bahr, H.P., 1978. Geometrical Analysis and Rectification of Landsat MSS Imagery: Comparison of Different Methods, Nachr. Kart. m. Vermess. Wes., Number 36, Institut fur Angewandte Geodasie, Frankfurt, pp. 25-46.

Caron, R. H. and K. W. Simon, 1975. Attitude Time-Series Estimator for Rectification of Spaceborne Imagery, J. Spacecraft, Volume 12, Number 1, pp. 27-32.

Dowman, I. J. and M. A. Mohamed, 1981. Photogrammetric Applications of Landsat MSS Imagery, International J. Remote Sensing, Volume 2, Number 2, pp. 105-113.

Forrest, R. B., 1974. Geometric Correction of ERTS-1 MSS Images, ISP Commission III Symposium, Stuttgart.

Forrest, R. B., 1981. Simulation of Orbital Image Sensor Geometry, Photogrammetric Engineering and Remote Sensing, Volume 47, Number 8, pp. 1187-1193.

Friedman, D. E., J. P. Friedel, K. L. Magnussen, R. Kwok, and S. Richardson, 1983. Multiple Scene Precision Rectification of Spaceborne Imagery With Very Few Ground Control Points, Photogrammetric Engineering and Remote Sensing, Volume 48, Number 12, pp. 1657-1667.

Konecny, G., 1976. Mathematical Models and Procedures for the Geometric Restitution of Remote Sensing Imagery, 13th ISP Congress, Commission III, Helsinki.

Kratky, V., 1972. Photogrammetric Solution for Precision Processing of ERTS Images, 12th ISP Congress, Commission II, Ottawa.

Levine, I., 1981. Computational Aspects of Geometric Correction Data Generation in the Landsat-D Imagery Processing, Sixth Annual Flight Mechanics/Estimation Theory Symposium, NASA Goodard Space Flight Center.

Mikhail, E. M. and F. C. Paderes, 1983. Simulation Aspects in the Study of Rectification of Satellite Scanner Data, Proceedings of the NASA Symposium on Mathematical Pattern Recognition and Image Analysis, Johnson Space Center, Houston, Texas, pp. 413-483.

Mikhail, E. M., 1979. Review and Some Thoughts on the Assessment of Aerial Triangulation Results, Aerial Triangulation Symposium, Department of Surveying, University of Queensland, pp. 15-17.

Paderes, F. C. and E. M. Mikhail, 1983. Photogrammetric Aspects of Satellite Imageries, Proceedings of the ASP Fall Convention, Salt Lake City, Utah.

Paderes, F. C. and E. M. Mikhail, 1984. Rectification of Single and Overlapping Frames of Satellite Scanner Data, 15th ISPRS Congress, Commission III, Rio de Janeiro, Brazil.

Paderes, F. C., E. M. Mikhail, and W. Forstner, 1984. Rectification of Single and Multiple Frames of Satellite Scanner Imagery Using Points and Edges as Control, Proceedings of the Second Annual NASA Symposium on Mathematical Pattern Recognition and Image Analysis, Johnson Space Center, Houston, Texas.

Pope, A. J., 1975. The Statistics of Residuals and the Detection of Outliers, Paper presented at the XVI General Assembly of the International Union of Geodesy and Geophysics, IAG, Grenoble, France.

Sawada, N., M. Kidode, H. Shinoda, H. Asada, M. Iwanaga, S. Watanabe, K. Mori, and M. Akiyama, 1981. An Analytic Correction Method for Satellite MSS Geometric Distortions, Photogrammetric Engineering and Remote Sensing, Volume 47, Number 8, pp. 1195-1203.

Stefanovic, P., 1978. Blunders and Least Squares, The ITC Journal 1978-1.

Synder, J. P., 1982. Geometry of a Mapping Satellite, Photogrammetric Engineering and Remote Sensing, Volume 48, Number 10, pp. 1593-1602.

Trinder, J. C. and S. U. Nasca, 1976. Tests on the Mapping Applications of Landsat Imagery, 13th ISP Congress, Commission III, Helsinki.

Wiesel, J. W., 1984. Image Rectification and Registration, 15th ISPRS Congress, Commission III, Rio de Janeiro, Brazil.

Table 1. Effect of Error in Internal Sensor Geometry

COLUMN STANDARD DEVIATION (pixel)	AVERAGE RMS ERROR IN PLANIMETRY (meters)
0.01	3.55
0.05	3.90
0.1	4.73
0.5	14.70
1.0	28.16
2.0	53.70

Table 2. Effect of Errors in Variances of Image and Ground Coordinates

VARIANCE FACTOR	AVERAGE RMS PLANIMETRIC ERROR IN METERS	
	IMAGE	GROUND
0.0001	*	29.94
0.01	*	29.34
0.1	59.06	29.34
1.	29.34	29.34
10.	29.34	59.06
100.	29.34	*
10000.	29.34	*

* Solution did not converge.

Table 3. Effect of Different Parameter Combinations

PARAMETER ERRORS IN PIXEL	AVERAGE RMS PLANIMETRIC ERRORS IN METERS	
	CASE 1	CASE 2
0.1	11.19	15.36
1	24.83	16.00
2	71.44	16.13
5	170.81	29.34
10		53.21
20		105.53

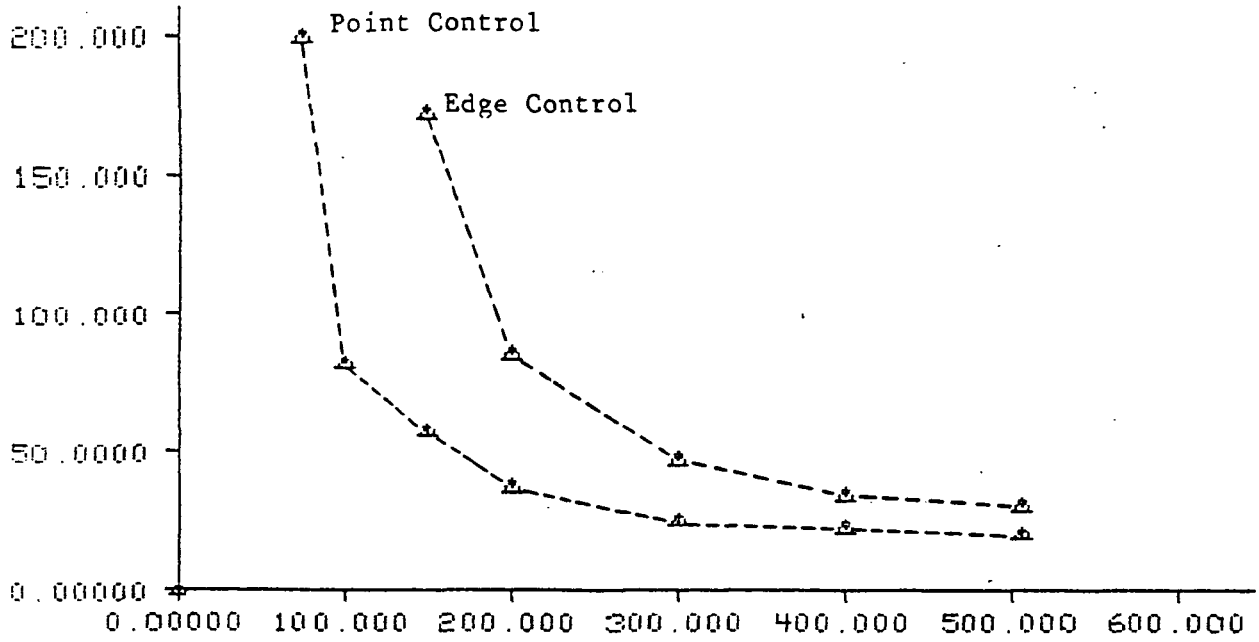
Case 1: True weights for all satellite position deviation, sensor attitude and sensor azimuth component parameters are multiplied by a factor of 125 except the first entry.

Case 2: Along orbit and across orbital plane satellite position deviation components are fixed; radial or elevation component, sensor attitude and sensor azimuth parameters are free.

Table 4. Results of Blunder Identification

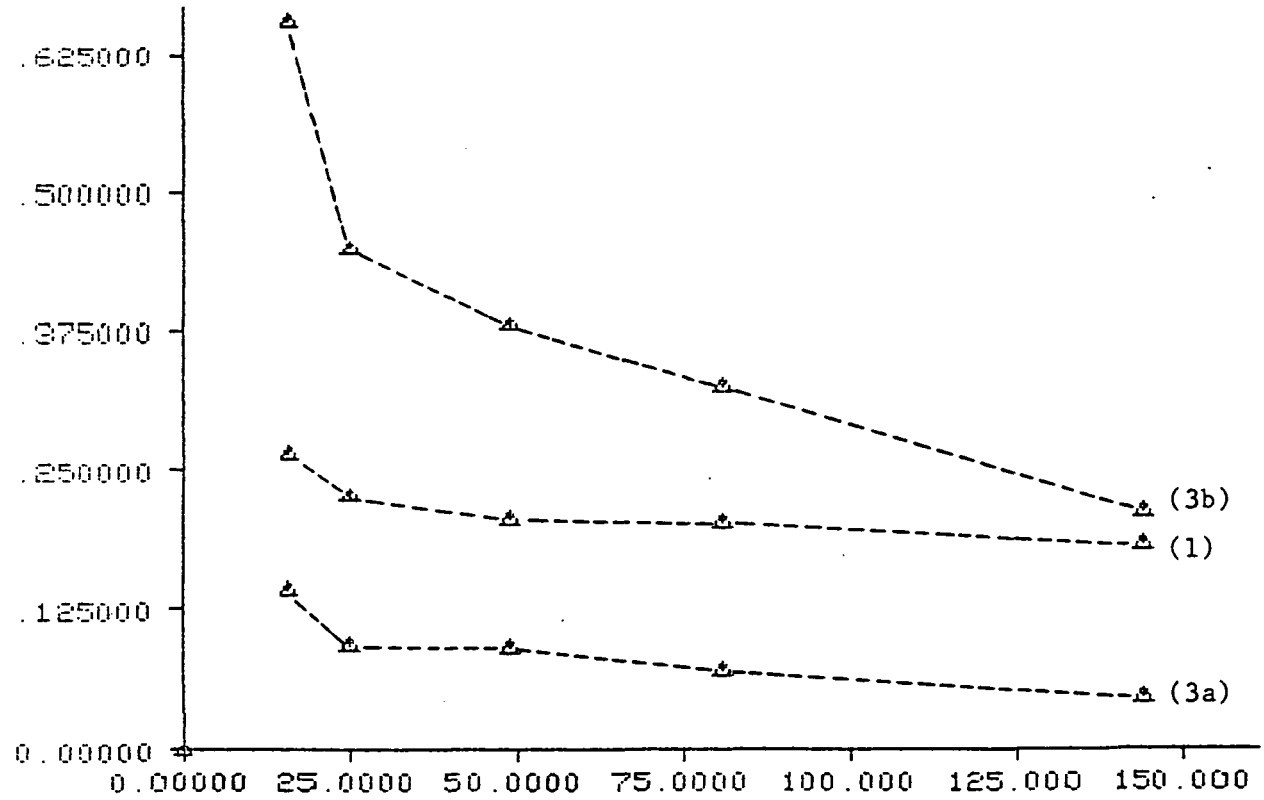
NUMBER OF BLUNDERS	METHODS 1	2	3
1	4σ/4σ	4σ/6σ	4σ/4σ
2	6σ/8σ	8σ/8σ	6σ/8σ
4	6σ/8σ	10σ/8σ	6σ/8σ

Method 1. Data snooping (normal test)
Method 2. Tau test
Method 3. Partial quadratic (chi-squared test)



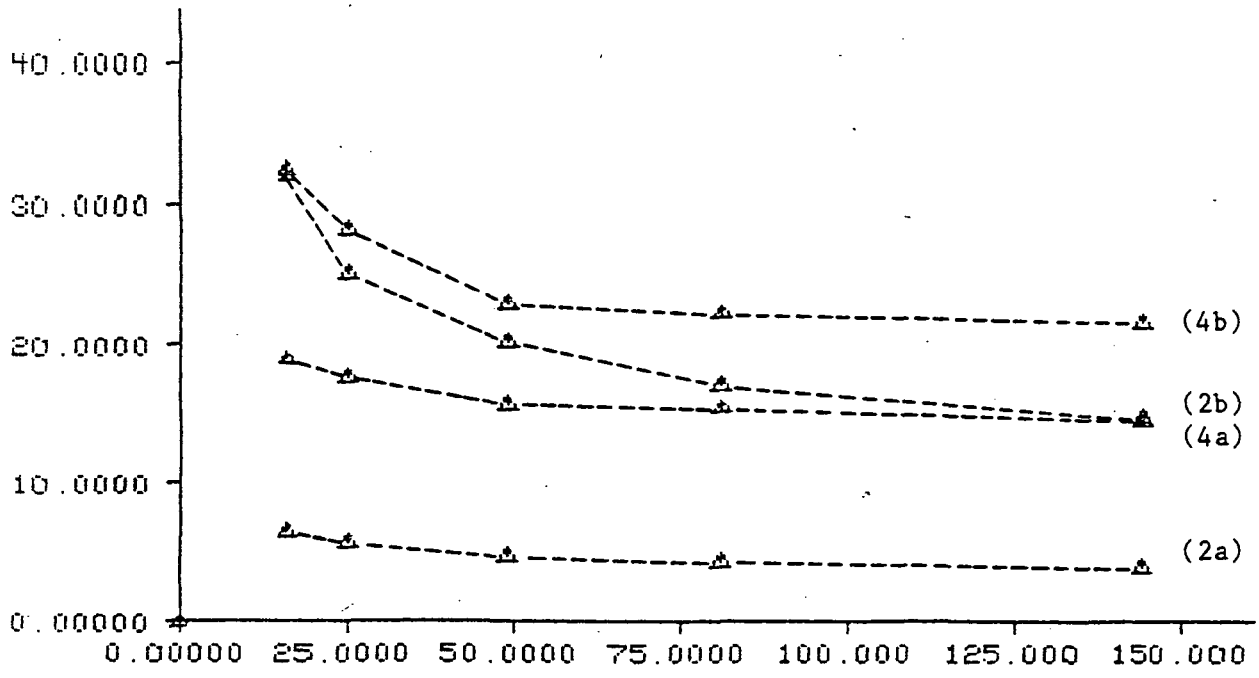
Abscissa: Number of Control Plus Tie Edges/Points
Ordinate: Average RMS Planimetric Error in Meters

Figure 1. Plot of Block Adjustment Results With Edges and Points as Control



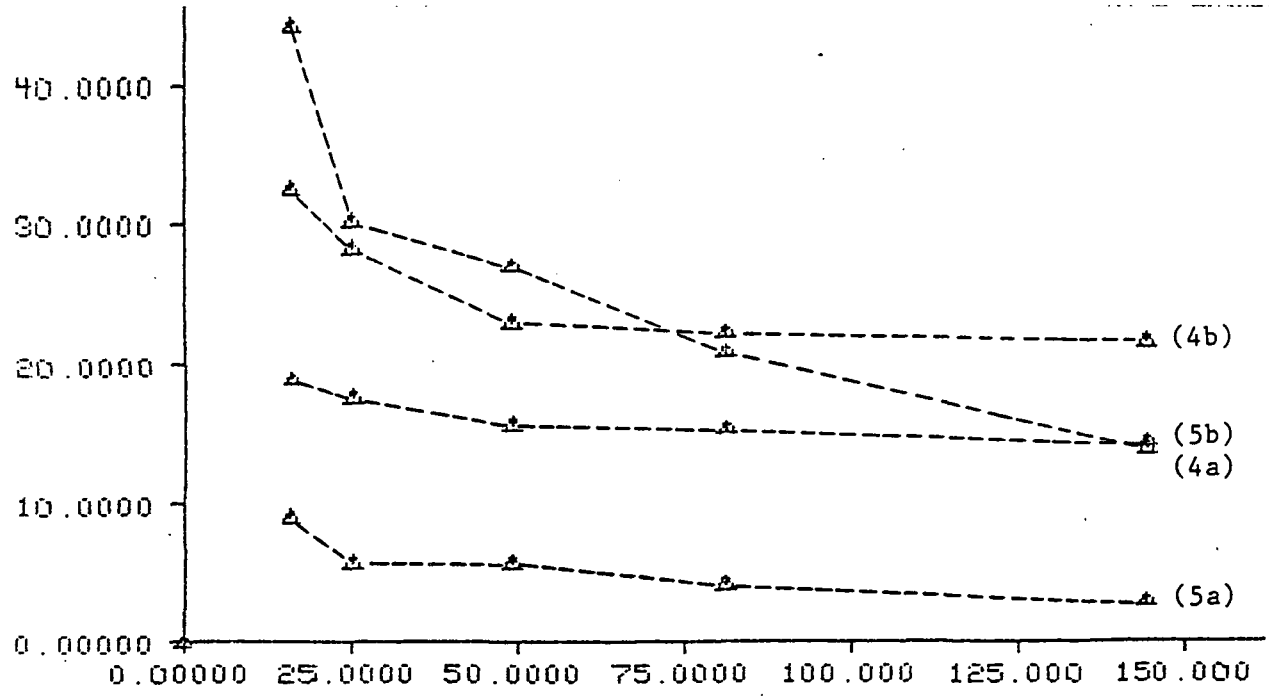
Abscissa: Number of Control Points
Ordinate: Average RMS Image Position Error in Pixel

Figure 2. Accuracy of Registration Via Rectification (Registration of Frame A to Frame B)



Abscissa: Number of Control Points
Ordinate: Average RMS Planimetric Error in Meters

Figure 3. Accuracy of Rectification Via Registration (Rectification of Frame A)



Abscissa: Number of Control Points
Ordinate: Average RMS Planimetric Error in Meters

Figure 4. Rectification/Rectification vs. Registration/Rectification Sequence

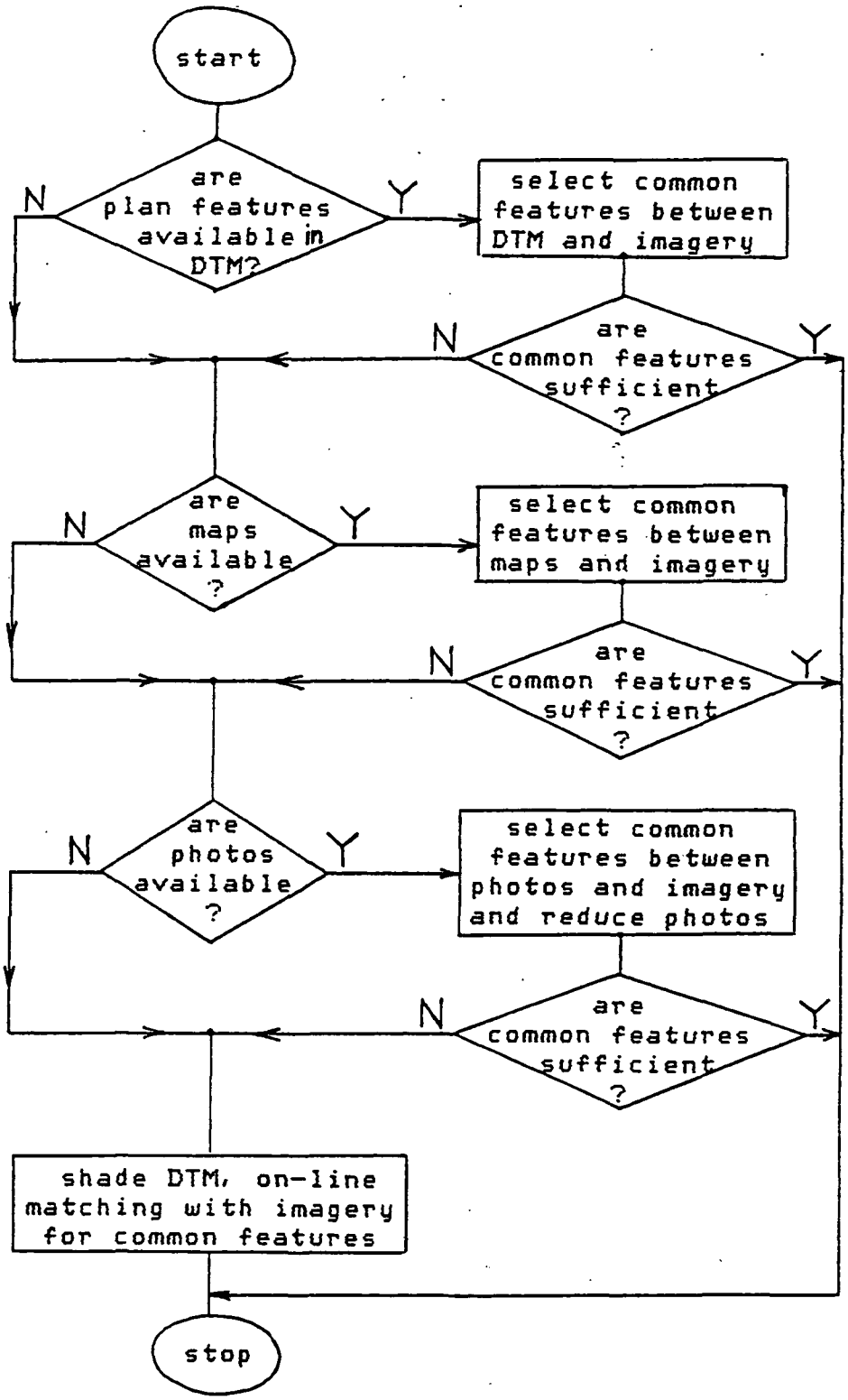


Figure 5. Flow Diagram for Finding Common Points Between Imagery and DTM Data

RELATING GROUND SCENES TO SPATIAL VARIATION IN REMOTELY SENSED IMAGES

Curtis E. Woodcock

Boston University

Alan H. Strahler

Hunter College

Abstract

Variograms are used as a tool to study spatial variation in remotely sensed images from both theoretical and empirical perspectives. The theoretical analysis involves deriving variograms that incorporate the effects of regularization for simple scene models. In addition, variograms are calculated from remotely sensed images from scenes with known characteristics in an empirical portion of the study. The two diverse approaches are linked through the use of simulated images. Several kinds of information about ground scenes can be recovered from analysis of variograms derived from images of the scenes. Also, the effects of changing spatial resolution on the spatial structure of images can be determine through understanding the effects of regularization.

1. Introduction

The long term goal motivating the research presented in this paper is the development of scene inference methods that exploit spatial relationships in remotely sensed imagery. For many years the spatial variation present in images has been a primary information source used in manual interpretation of remotely sensed imagery. However, it has proven a difficult task to quantify the spatial structures that humans recognize in images and incorporate them in computer-assisted scene inference methodologies. Thus, as an intermediate goal an attempt has been made to understand the nature and causes of spatial variation in images as they relate to the characteristics of the ground scene and the spatial resolution of the imagery.

In order to incorporate the characteristics of ground scenes in this investigation, an organized method of describing scenes is necessary. Thus, a *scene model* is defined which specifies the form and nature of the energy and matter in the scene. One characteristic of the scene models used in this research is that they are *discrete* in nature, assuming there are boundaries or discontinuities where the properties of matter change abruptly over space. In this model setting, the scene is perceived as consisting of objects on a background. A scene-model *element* is an abstraction of a real object in the scene which can be regarded as having uniform properties or parameters.

The elements in a scene model can vary widely according to the interests of the interpreter and the scale of the observations, or the spatial resolution. Examples of elements in an agricultural scene could include: leaf, branch, plant, crop row, field. In addition to these discrete elements, a particular type of element, the background, should be recognized. The *background* is usually assumed to be

spatially continuous and is typically partially obscured by other elements in the scene. Soil, snow, rock, and vegetative understory are examples of backgrounds. It is also important to recognize that scene models may be *complex,* or include more than one type of element as well as the background. *Nested* models are also possible in which the properties of larger elements are derived from smaller ones.

In this investigation it is necessary measure spatial variation in images in order to compare them. Variograms were selected for this role in the investigation because they are mathematically quite tractable and are easy to understand. Other choices such as autocorrelation functions or power spectrum density functions are also available. Variograms are approached from both a theoretical and empirical perspective in this investigation. The theoretical phase involves deriving explicit variograms for scene models. The empirical use of variograms consists of calculating observed variograms from images of scenes with known characteristics. These two divergent approaches are linked through the use of simulated images. The variogram, then, becomes the tool linking scene models, simulated images, and real images.

2. Variograms

Variograms measure spatial variation in a *regionalized variable.* Any random variable whose position in space or time is known is a regionalized variable. In this formulation, variables are indexed by their location. Thus, assume $Y(x)$ is a regionalized variable associated with location x. For numerous realizations of the variable Y at different locations, it becomes necessary to index the locations as x_i, where $i=1,...n$ correspond to n observations. If the $Y(x_i)$ are uncorrelated, then the image will consist of random noise. If however, the $Y(x_i)$ are in some way related, then the data will exhibit spatial structure. Perhaps the weakest assumption one can make about this structure is what Matheron [5] refers to as

the "intrinsic" hypothesis, that the increments $Y(x_i+h)-Y(x_i)$ associated with a small distance h are weakly stationary. Under this assumption, the first moment of the increment, its expected value, is constant or at least only slowly varying with spatial position x; and the second moment is also invariant with spatial position. The second moment is called the *variogram:*

$$2\gamma(h) = E[Y(x+h) - Y(x)]^2$$

Just as the variance characterizes the distribution of a nonspatial random variable, so the variogram characterizes the distribution of a regionalized variable. The distance at which samples become independent is often called the range of influence and is denoted as a. The value at which the variogram levels off is denoted c and is called the sill (Clark [1]).

Geostatisticians have used the variogram as a primary tool in many spatial studies. In particular, variograms are used as part of a process called kriging. Kriging is a method of estimating local values from surrounding point samples, a process generically referred to as interpolation. Kriging uses the relationships between point samples established by the variogram to produce the best linear unbiased estimator (Clark [1]). For kriging, a model describing the shape of the variogram is necessary.

One commonly used model for the shape of a variogram is the *spherical model:*

$$\gamma(h) = c[3h / 2a - h^3 / 2a^3] \quad when \quad h \leqslant a$$

and

$$\gamma(h) = c \quad when \quad h > a$$

Figure 1 shows an example of a spherical model of a variogram. As expected, the variogram passes through the origin. If samples are taken exactly zero distance apart then they are the same sample and their variation will also be zero. As h increases within the range of influence, the difference between measurements increases and the variogram rises. Past the distance a, samples from the data are independent and the variogram reaches a stable peak at the value c, the sill. Just as a sample variance is an estimate of the true variance of a variable, the sill is an estimate of the true variance of a regionalized variable. Thus, one can estimate the sill via a sample variance.

The spherical model is often referred to as the "ideal" model for a variogram because there is a well defined sill and the meaning of the range of influence is easily interpreted. Not all models for the shape of a variogram share these characteristics. Figure 2 shows the shape of an exponential model for a variogram compared with a spherical model with the same sill and range of influence. The exponential model is calculated as follows:

$$\hat{\gamma}(h) = c\,[1 - exp\,(-h\,/\,a)]$$

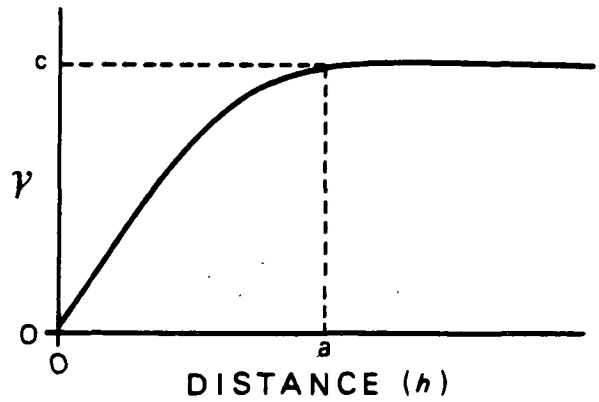


Figure 1. The spherical model of a variogram (modified from Clark [1]).

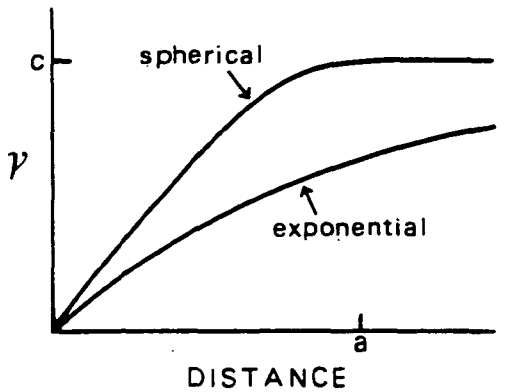


Figure 2. The spherical and exponential models for the same values of a and c (modified from Clark [1]).

The exponential model never reaches its sill, but asymptotically approaches it. In addition, the meaning of a, the range of influence, is not clear. In the spherical model there was a direct physical interpretation of a, but in the exponential model it is a parameter necessary to describe the shape of the model, but has limited interpretive value.

There are models for the shape of variograms which do not have a sill. The simplest form of these is the *linear model:*

$$\gamma(h) = ph$$

where p is the slope of the line. An extension of this model is the *generalized linear model:*

$$\gamma(h) = ph^{\lambda}$$

where $0 \leqslant \lambda < 2$. Figure 3 shows the effect of the exponent, λ, on the shape of the generalized linear model.

While the above models are commonly used in geostatistics, other models could be used. For example, all the above models are monotonic, assuming that

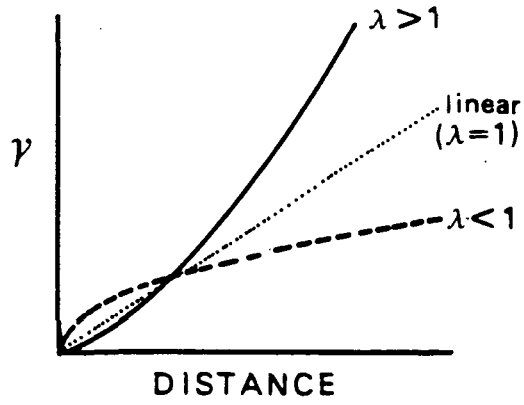


Figure 3. The linear model and generalized linear models of variograms (modified from Clark [1]).

variation will only increase as a function of distance. However, if the data exhibit periodicity models based on trigonometric functions might be appropriate. Also, variograms can be multidimensional. All the examples have shown one-dimensional variograms, but two and three-dimensional variograms are possible. In this situation h becomes a vector and measures both distance and direction (and possibly height). One-dimensional variograms have the advantage of being easy to display and interpret. Two-dimensional variograms are usually displayed as contour plots and can be useful for revealing anisotropy in the data. However, displays using contours can make evaluation of shapes of variograms difficult. As a third dimension is added there is again potential for information on variation in another dimension, but the problems of display and analysis of shape increase. In this paper. one-dimensional variograms are used because of the emphasis on the shape of variograms as influenced by the characteristics of scenes. In a previous paper, two-dimensional variograms of remotely sensed images were presented and interpreted with respect to the degree and causes of anisotropy (Woodcock and Strahler [8]). However, the analysis of shape and determination of the range of influence proved difficult using two-dimensional variograms.

In geostatistics, the models used to describe variograms tend to be combinations of several models. These combinations can include several models of the same type with different parameters or different types of models. The use of combinations of models is reminiscent of fourier analysis where sinusoidal curves with different amplitudes, frequencies and phases are combined to model a function. One difference from fourier analysis is the subjective nature of the methods used to determine the type of models to be combined and their coefficients. Often the nature of the model selected is guided by the specific interests of an application. Criteria which affect model selection are the behavior near the origin, the fit near the sill, and the determination of the range of influence.

2.1 Scene Models and Variograms

The previously described models for the shapes of variograms are necessary for kriging, and as a result have played a significant role in studies involving variograms. However, for the purpose of understanding spatial variation in remotely sensed images, their value is limited. The reason is that there is no apparent way to link these models for the shapes of variograms to scene models. A more useful tool is a variogram whose characteristics can be determined as a function of the parameters describing a scene model. Serra [6] provides a method for calculating explicit variograms for some simple scenes. (The use of Serra's work was made possible by the help of Dr. David L. B. Jupp.)

The derivation of explicit variograms is based on an extension of the binomial. This approach is well suited for a discrete scene model, in which the elements in the scene and the background are considered homogeneous, thus allowing only two states in the image. By approximating the binomial using an exponential, it is possible to determine q, the proportion of the area not covered

by n randomly-distributed objects of area a within a larger area A as:

$$q = \exp(-na / A)$$

The proportion of the area covered by objects is simply 1-q. The variogram for the distance between two points h distance apart is:

$$\gamma(h) = q^2\left[\frac{1}{q} - \exp\left[O(h)n / A\right]\right]$$

where $O(h)$ is the overlap function. The overlap function for the case of randomly-located, overlapping discs of radius r, when $h < 2r$ is:

$$O(h) = 2\cos^{-1}\left(\frac{h}{2r}\right)r^2 - \sqrt{r^2 - h^2/4}$$

If $h \geq 2r$ then no overlap occurs and $\gamma(h) = q(1-q) = qp$, which is the binomial variance.

This formulation of a variogram is slightly different than originally described. In the original description, the variable Y(x) is continuously measured. For this explicit variogram, the variogram is defined as the probability that $Y(x)$ and $Y(x+h)$ will be different, i.e., x is located within the object and $x+h$ is located on the background, or vice-versa. This is equivalent to the probability of crossing a boundary between an object and the background.

Figure 4 shows explicit variograms for scenes of overlapping disks. The variogram is calculated for n=1, 10, 25, 50, 100, and 200 objects of unit radius on an area of size 100 π^2 units. The variogram starts with zero variance and rises to the sill, or maximum variance. The distance to the sill reflects the size of the objects, and the height of the sill is determined by the number of objects. At low values of n variance is low because most of the area is background. As n

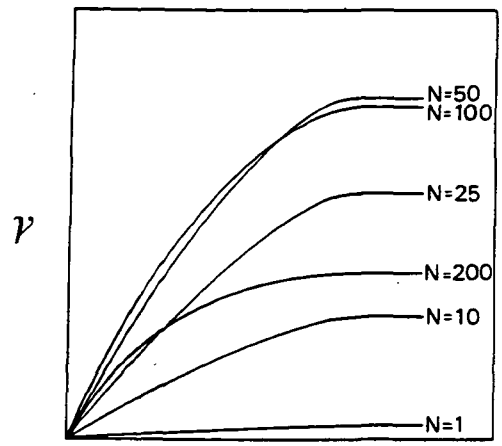


Figure 4. Variograms for scenes with different numbers *(n)* of randomly-located, overlapping discs.

increases, the curves become steeper and the sill successively higher until half the area is covered (p=q=.5, n=69.3). As more than half of the area is covered, the height of the sill decreases because more and more of the area becomes covered by disks. Thus, there will be two different scenes with the same sill, one in which the discs occupy area p, and one in which the background occupies area p. Distinguishing between these two alternatives should not normally present a problem because the general brightness of the scene will be different. The two cases may also be distinguished by their shape. Note that in Figure 4 the variograms for $n > 69.3$ have a more rounded shape than that those for $n < 69.3$. The reason for this may be resolved by studying another of the useful measures of variograms, the slope at the origin. Serra [6] shows that the slope at the origin depends on the amount of boundary between discs and background. This reduces for both high and low n, but in different ways. For higher values of n, the background becomes dissected into a large number of small areas, or slivers between the disks. In this situation the amount of boundary becomes large, and $\gamma(h)$ becomes large at short distances, leading to the more rounded, faster rising shape

of the variogram for large n.

2.2 Variograms and Remotely Sensed Images

Whenever remotely sensed data consist of images, an important new information component is added to the measurement output by the sensor: its spatial position. Since the position of the measurement in the image is usually a quantifiable function of the position in the scene of the resolution cell from which it is derived, each measurement can be associated with a ground location and be positioned relative to other measurements. The sensor's response then becomes a regionalized variable, because its position in space is known. Thus, variograms can be used to characterize the spatial structure in remotely sensed images.

There is an important factor that must be considered when using variograms in conjunction with remotely sensed images. The models presented for the shapes of variograms (spherical, exponential etc.) are for *punctual* variograms, or variograms derived from point measurements. Measurements in remotely sensed images are integrated over areas, and this difference is important. In this instance, when measurements are taken over some length or area, the resulting variogram is referred to as *regularized.* Regionalized variables can be thought of as having a true or underlying punctual variogram based on point measurements, and regularized variograms which are an estimate of the underlying variogram based on measurements taken over an area.

In remotely sensed images, the regularizing area is the instantaneous field of view of the sensor, with the point spread function describing the form of the regularization. For this study, the resolution-cell size of the image is taken as the units of regularization. The effects of regularization are similar to those typically

associated with measurements that represent some form of aggregation. The overall variance of the data is reduced and fine scale variations are blurred. Certainly variation at a scale finer than the scale of regularization can not be detected and variations less than two to three times the scale of regularization can not be reliably characterized.

The effect of regularization on punctual variograms can be determined analytically, but is considerably more straightforward for some models. Geostatisticians have determined the expected results of one-dimensional regularization for several models of variograms for use with core samples. The exponential model for samples of length l is:

$$\gamma_l(h) = C\left\{2a/l + a^2/l^2\left[1-exp(-l/a)\right]\left[exp(-h/a)\right]\left[1-exp(l/a)\right]\right\}$$

where $h \geqslant l$.

Determination of γ_l when $h < l$ is considerably more complex. The linear model is straightforward for all distances:

$$\gamma_l(h) = \frac{ph^2}{3l^2}(3l-h) \quad when \quad h \leqslant l$$

and

$$\gamma_l(h) = p(h - l/3) \quad when \quad h < l$$

The calculation of a regularized spherical model is very complex and tables have been produced to aid in its estimation. The sill for the regularized variogram will be lower than the punctual variogram, as can be seen in Figure 5.

The effect of regularization of disc model variograms can be seen in Figures 6 A-H and Figure 7. These figures show the punctual variogram and the regular-

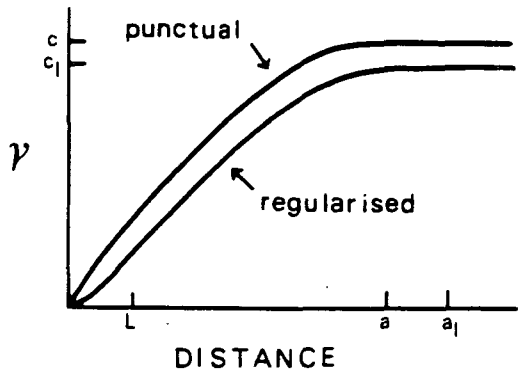
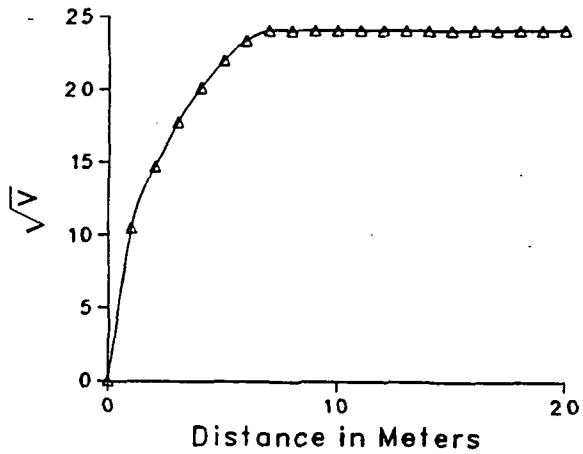


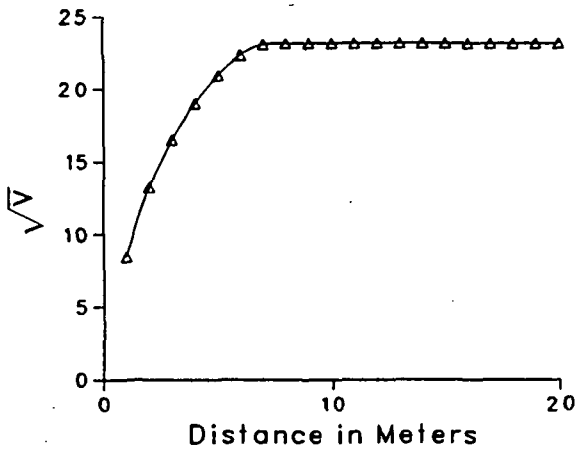
Figure 5. The effect of regularization from samples of length l on the spherical model of a variogram (modified from Clark [1]).

ized variogram for several different units of regularization for the same scene model. The punctual variogram is the same for these Figures, but the units of regularization are increased in size. In essence, increasing the units of regularization is analogous to increasing spatial resolution in remotly sensed data. The scene model used in these tests is randomly distributed discs of radius 3.5 m that cover 10% of the area.

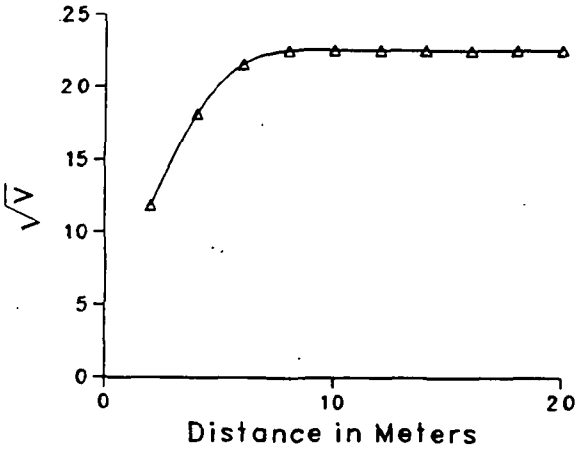
Figures 6 B-H show variograms as they would look if calculated from remotely sensed imagery at various spatial resolutions. In other words, the x axis is in integer multiples of the units of regularization. As a result, the scale of the x axis changes in these graphs. At small units of regularization, the variograms resemble the punctual variogram, with a well developed drop from the sill in the range of influence. At larger units of regularization, the shape of the variogram becomes very simple. In fact, for Figures 6 D-F, or 4, 6, and 8 m, the variogram is essentially one point below the sill. By 12 m and beyond the variogram is essentially flat. Figure 7 is a composite of the graphs in Figure 6 A-F that holds the x axis constant. This composite illustrates several important points about the effect of regularization. As the size of the regularizing units increase, three things



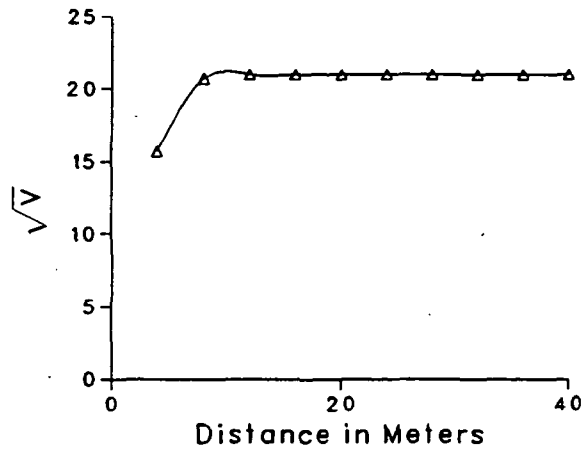
(6A) Punctual Variogram



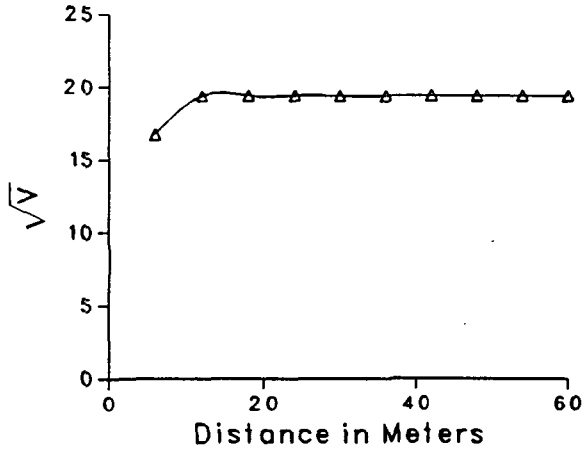
(6B) One Meter Regularization



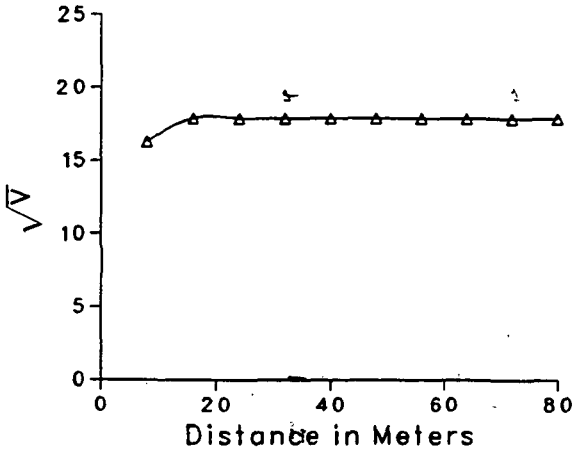
(6C) Two Meter Regularization



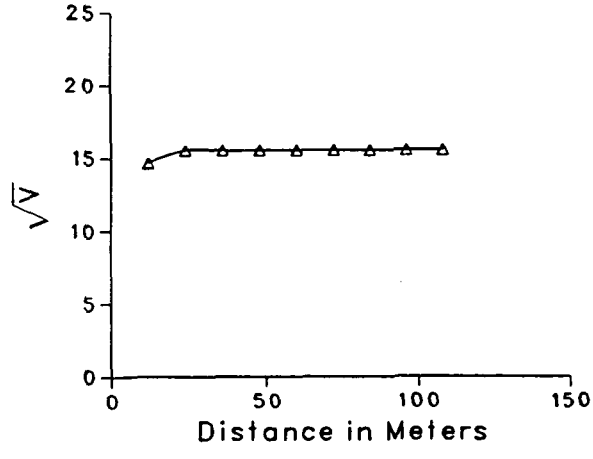
(6D) Four Meter Regularization



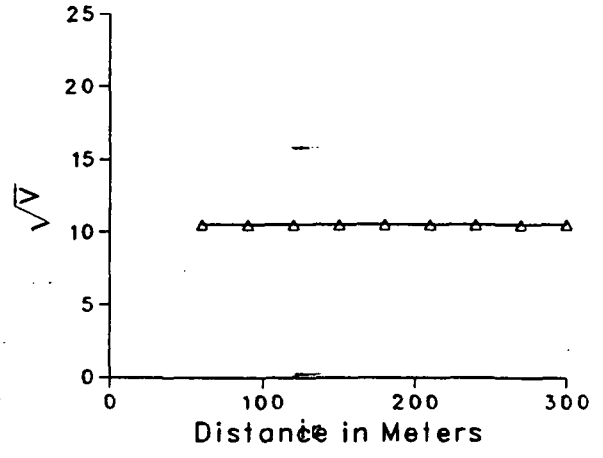
(6E) Six Meter Regularization



(6F) Eight Meter Regularization



(6G) Twelve Meter Regularization



(6H) Thirty Meter Regularization

Figure 6 A-H. The effect of regularization on a disc model variogram. All variograms are for the same scene model but each uses a different level of regularization. B-H are displayed as if measured from a remotely sensed image.

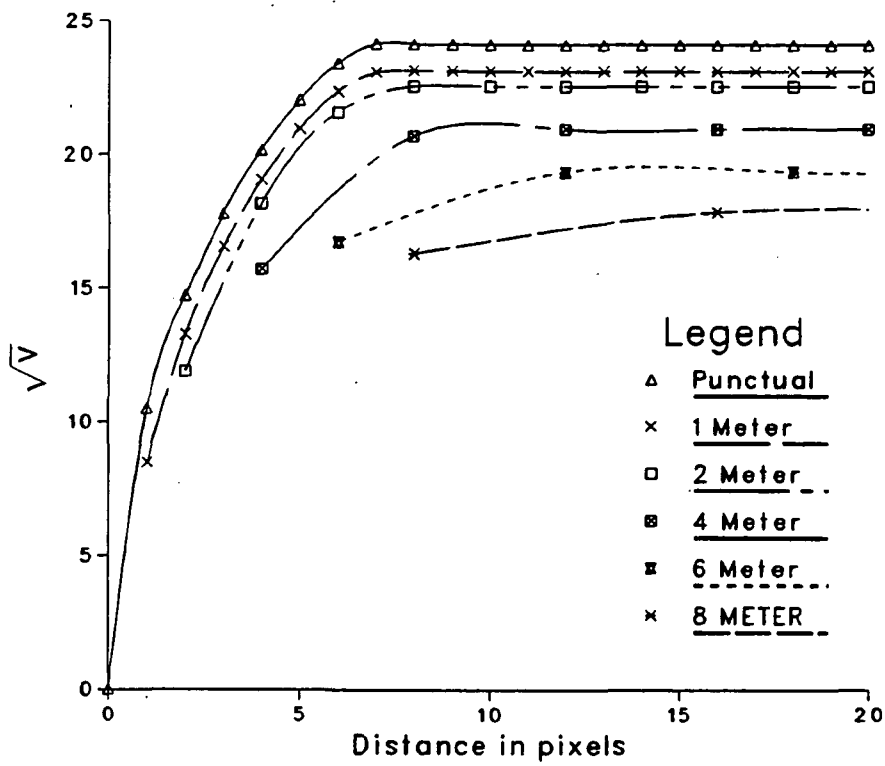


Figure 7. The effect of regularization on a disc model variogram. This graph is a composite of Figures 6 A-F that holds the x axis constant.

should be noted. First, the height of the sill (or the variance of the variable) decreases. Second, the range of influence, or the distance to the sill increases. Third, the height of the variogram at the first measured interval of h increases relative to the sill until they match. While one can determine the regularized variogram from the punctual variogram, in practice, the more common situation is the observed variogram is a regularized variogram and one is interested in the punctual variogram. In this situation, the equation for the regularized variogram is used to estimate a and c, which are then used in the equation for the punctual variogram.

Variograms can be calculated from remotely sensed images as follows:

$$2\gamma(h) = \frac{\sum_{i=1}^{k} \left(Y(x_i) - Y(x_i + h) \right)^2}{k}$$

where k is the number of observations used to estimate γ. A program that estimates both the one-dimensional and two-dimensional variograms of remotely sensed images has been written. Ideally, a variogram should be computed by comparing each point with all others. In a normal application in geostatistics, the number of available samples is limited and an estimate of the variogram is produced in this way. In the remote sensing case, generally the area of interest is entirely sampled, but due to the large sizes of images the comparison of each measurement with all other measurements is computationally unrealistic and constraints need to be imposed. One constraint concerns the distance h over which the variogram is to be measured. This distance can be thought of as a "window size" when using image data and needs to be larger than the range of influence and large enough for any periodicities in the data to be revealed.

A second constraint concerns the number of points in the image to be used as centers of windows. The use of a sample results in an estimate of the true regularized variogram. The actual locations of points to be used are determined randomly from the set of points inside a band of width h around the outside of the image. This restriction is to avoid boundary conditions to assure a constant number of points contributing to the two-dimensional variogram for each vector h. For the one-dimensional variogram, there are not the same number of pixels for each distance h. In fact, the possible combinations of distances between centers of pixels grows large as their distance apart increases. To simplify the resulting variogram, all distances between successive integer multiples of the number of resolution cells are combined to produce a single estimate of γ over that interval. The distance used to index this estimate is the average of the

contributing distances weighted by their frequency of occurrence. For example, there are four pixels one resolution-cell distant from any center point (its nearest neighbors), and four pixels 1.414 resolution cells away (at the diagonals). Thus, for the one-dimensional variogram, the contributions of these eight pixels is used at each center point to estimate the value of γ between 1 and 2 units of distance. The distance used to index their result is 1.212, or the average of the distances of the contributing pixels. As h increases, the combinations become more complicated, and the number of pixels contributing to the estimate of any given interval increases.

3. Image Simulation

In the last section, two diverse approaches to variograms were presented. One approach is empirical, in which the variogram is calculated from observed images. The other is theoretical, with the expected nature of variograms being explicitly defined on the basis of a simple scene model. In an effort to bridge the gap between these two approaches, images were simulated on the basis of known scene models. These simulations served several purposes. First, they confirmed the validity of the explicit variograms through empirical testing. Second, they allowed for testing of the extension of the simple disk model to more complicated scenes. And third, the variograms of simulated images helped lead to a better understanding of the empirically calculated variograms from observed remotely sensed images.

3.1 Simulation Methods

The simulated images are based on a coniferous forest scene model. The basic approach is a modification of a Monte Carlo computer model used by Li and Strahler [4] in their studies of forest canopy reflectance. Monte Carlo

methods are used to locate trees on a plane which are illuminated from a specified angle and azimuth. This approach leads to four elements in the scene: illuminated tree crown and background, and shadowed tree crown and background. The forest simulation represents a general model with several parameters. For this project, these parameters are calibrated primarily by field data collected in the Klamath National Forest in northern California (Li and Strahler [4]).

In the original model of Li and Strahler, many realizations of individual resolution cells were simulated. Their approach specifies two levels of resolution: (1) the scale at which scene elements are differentiated, and (2) the size of the resolution cells. For this project, the simulation program was altered to simulate one larger scene in which the scale at which scene elements are differentiated matches the size of the resolution cells. The size used in the simulations presented is one meter. The distinction between a simulated scene and simulated image is minor in this case. A scene implies different elements and an image implies reflectances (or emittances). The simulation assumes no atmospheric effects and a square wave response on the part of the sensor. As a result, there are only four values for reflectances in the image, one for each type of scene element.

The primary parameters of the simulation concern the characteristics of trees, their number, location, size, and shape. In the Li and Strahler model, the number of trees in a single realization of a resolution cell varies according to a Poisson or Neyman Type A distribution. However, for the single realization of a larger area, a single value, or the mean of a Poisson distribution is used to determine the number of trees for the entire area.

Of more interest is the manner in which the trees are located within the scene. Considerable effort has been devoted to this question, and several alternatives considered. Li [3] measured the spatial patterns of trees using point-pattern techniques based on locations derived from aerial photography and found that a Neyman Type A model fit better than the random model. In a later study in a neighboring area, Franklin et. al. [2] again used locations of trees taken from aerial photography and found that the random model was appropriate except at spacings of about 10-60 m. Evidence for repulsion between trees, or a more regular distribution was found at short distances. Such a result could be easily supported by a competition model of tree growth, in which the likelihood of a tree surviving is reduced if it is very close to an established tree due to competition for resources such as light, water and nutrients. As a result, initial simulations used a "hard-core" model for the location of trees in which trees were randomly located except that a new tree could not be located within the area covered by the crown of a previously located tree. This approach was designed to modify the random assumption to take into consideration competition at short distances. However, it was later realized that Franklin's results may have been due to sampling artifacts resulting from the use of aerial photography to determine the locations of trees.

In an attempt to determine an appropriate model for the location of trees as well as calibrate other parameters for the model, field data was collected in the Goosenest District of the Klamath National Forest. An account of the methods used to collect and process the data is given in Woodcock [7]. The results of the field data indicate that the random model is a reasonable approximation. Thus in the simulations presented, the locations of the center of trees are determined through random coordinates.

The model is based on the use of inverted cones as the shapes of trees. Thus, the model is really limited to coniferous forests. Trees are assumed to have a constant apex angle of 10 degrees, which is based on the field data previously mentioned. A lognormal distribution of the sizes of trees is used. This decision is based on the results of other published studies, and the parameters of the distribution were calibrated from the field data collected in the Klamath. For a more complete description of the model and its parameters see Li and Strahler [4].

3.2 Validation of the Explicit Variograms

One use of the simulated images was to validate the explicit variograms. Due to the nature of the forest simulation model it was easily generalized to correspond to the disc model used for explicit variograms. By reducing the variance of the heights of trees to a number close to zero, and eliminating shadows

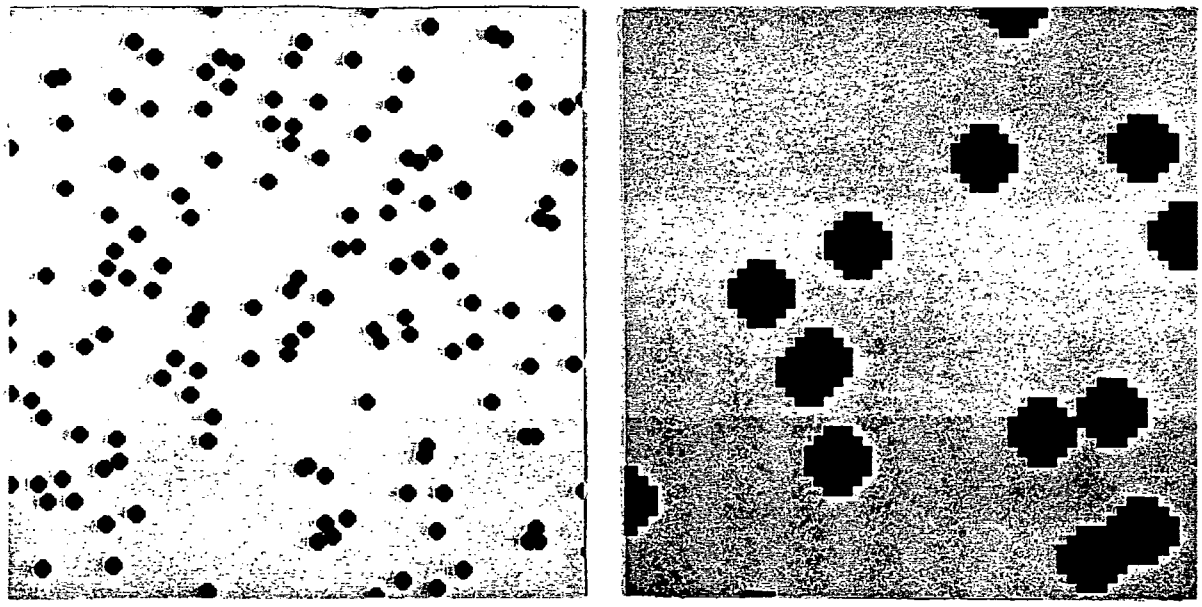


Figure 8. A portion of the simulated disc image (A), and an enlargement (B).

through the use of a solar zenith angle of zero, an image corresponding to discs on a background at one meter regularization was simulated. Figure 8 shows the simulated disc image, which has discs of 7-m diameter covering 9.92% of the background. In order to test the validity of the explicit variograms, an empirical variogram was calculated from the simulated disc image, and an explicit variogram for the corresponding scene model was calculated at one-meter regularization. Figure 9 shows these two variograms plotted together for comparison. These two variograms do not match exactly, but are very close.

There are several possible reasons why the observed and expected variograms do not match exactly. The empirical variogram is derived from one realization of a simulation process based on randomization. Thus, it is likely that this one realization will depart from the model to some extent. Also, the empirical variogram is estimated, in this case from a sample of 600 points in the image.

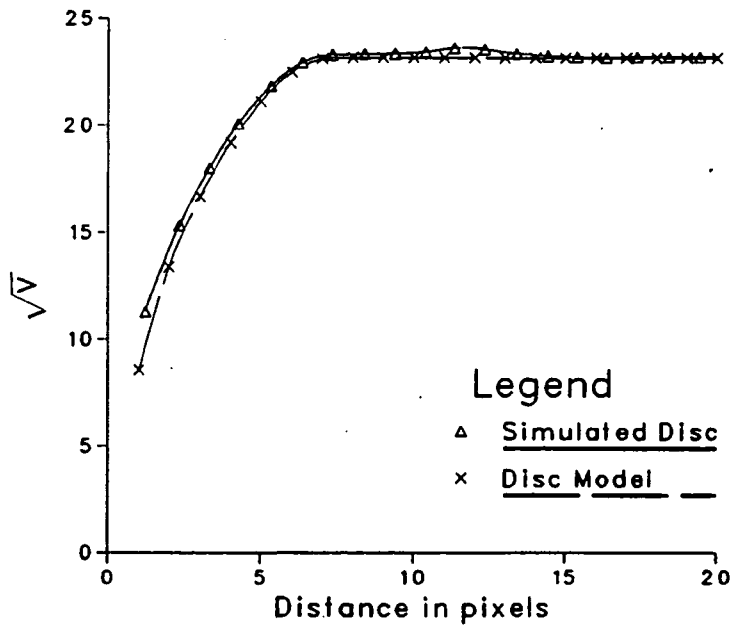


Figure 9. Comparison of an explicit variogram and an empirically calculated variogram for the same scene model. The empirical variogram was calculated from the simulated image in Figure 8.

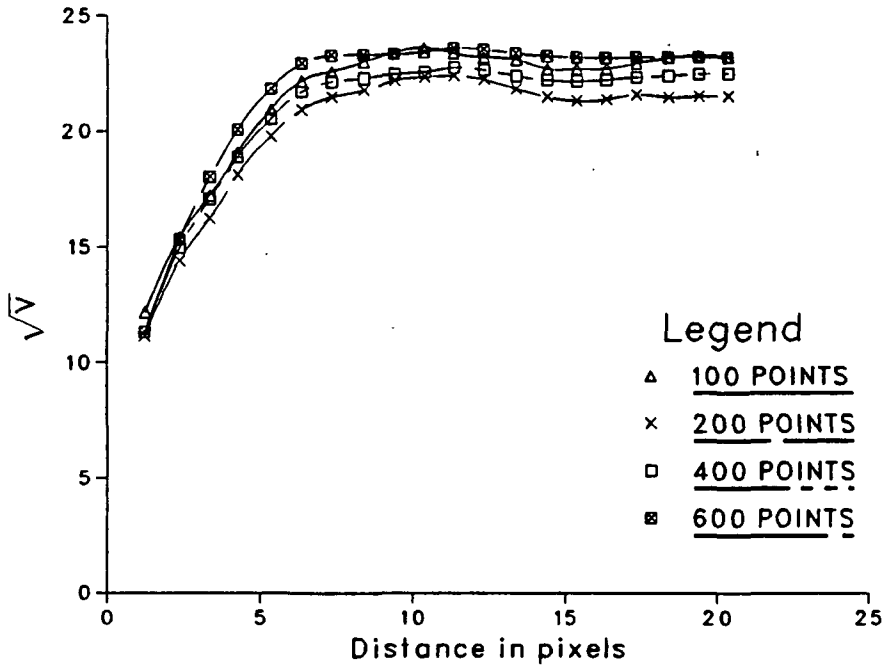


Figure 10. The effect of sampling density on empirically-estimated variograms.

As the number of points is changed, the variogram changes slightly. Clearly, the more points that are used, the more stable and accurate the estimate is likely to be. Figure 10 shows four estimates of the variogram for the simulated disc image using four different sampling densities. Their variation is large relative to the difference between the explicit and empirical variograms shown in Figure 9.

The ability to reproduce empirically through image simulation the results for a disc model expected by theoretical formulation is a significant step in the use of variograms to study spatial structure in images. This "closing of the loop" validated the theory as well as the software used to estimate variograms from observed images and the image simulation procedures.

3.3 Extension of the Disc Model

Having demonstrated the connection between observed variograms and theoretical variograms using a simple disc model, it is possible to test the effect of variations in that model on variograms. Obviously, real scenes are not composed of randomly located discs of the same size on a uniform background. However, it may be possible to use the characteristics of explicit variograms from this simple model to help explain the nature of variograms derived from real images.

3.3.1 Shape. To test the effect on observed variograms of the shape of objects, a forest image was simulated using the previously described methods. The same parameter settings that were used for the simulated disc image (Figure 8) were used with one exception; the angle of illumination was changed from zero to 20 degrees in order to produce shadows. The resulting image (Figure 11) exhibits all four components of the forest model: illuminated canopy, shadowed canopy, illuminated background, and shadowed background. In order to compare the observed variogram from this image with the disc model, it was necessary to convert the image to only two values, or tones. In this instance, trees and shadows were stretched to black and the background was left white. The resulting image (Figure 12) looks like cones on their sides. These cones do not strictly match the disc model due to their shape, but the ability to extend the disc model to this case is interesting.

A variogram was calculated from the observed black and white image for comparison with the result of the explicit variograms for the disc model. However, it was not clear what values should be used for the disc model in the calculation of the explicit variogram. In particular, it was not obvious what should be used as the size parameter. For the forest cone image, the radius changes as a

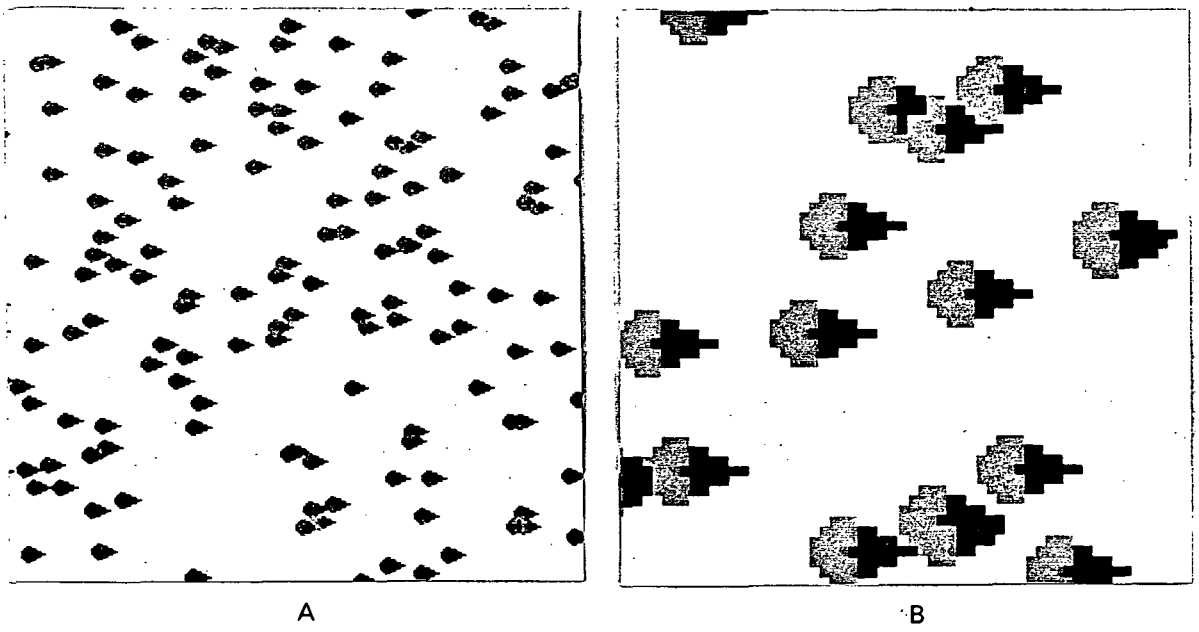


Figure 11. A portion of the simulated forest image (A), and an enlargement (B).

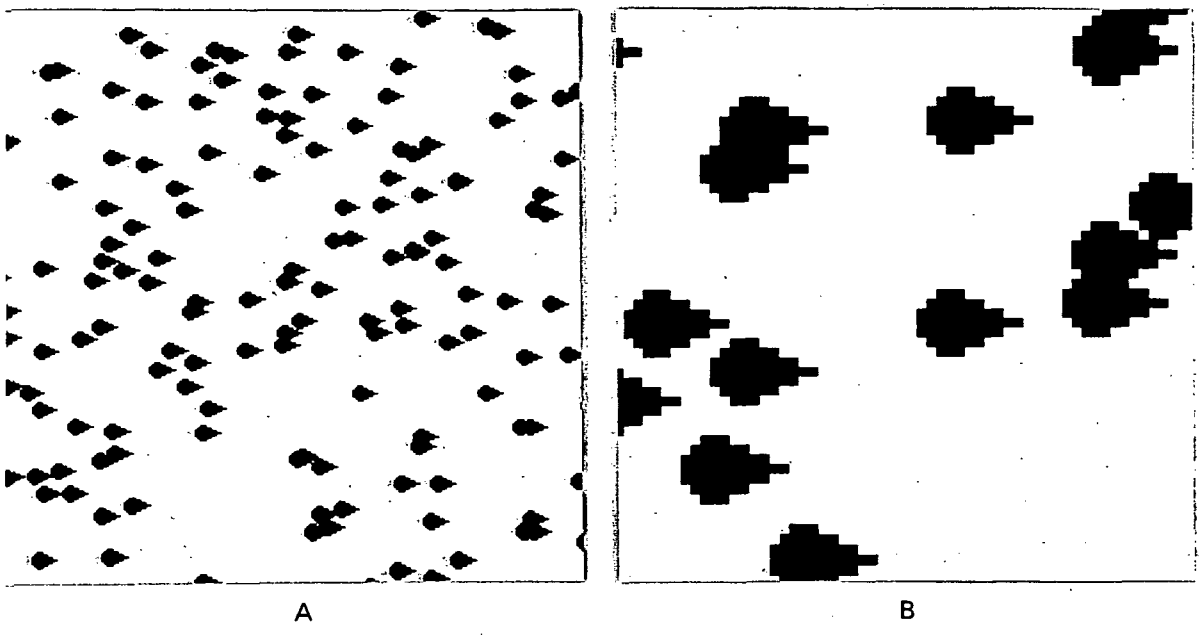


Figure 12. A portion of the simulated forest image stretched to two tones (A) for comparison with the disc model, and an enlargement (B).

function of orientation from 3.5 meters across the tree to 5.5 meters from the far

edge of the tree to the tip of the shadow. Figure 13 shows the observed variogram calculated from the image in Figure 12 compared with three explicit variograms for the disc model using 3.5, 4.5, and 5.5 meters for the radii of the discs. Interestingly, the 3.5 meter radius is the best approximation of the forest model, which is the same size as the trees before the addition of their shadows. The shadows markedly affect their shape but do not significantly influence their effective size. Figure 14 presents a comparison of the observed variogram with a variogram for discs with area equal to the area of the forest cone. While these two variograms are not a perfect match, they demonstrate that shape is a relatively minor factor in this case. Using just the area covered by individual objects it was possible to produce a reasonable fit with the disc model. This result is

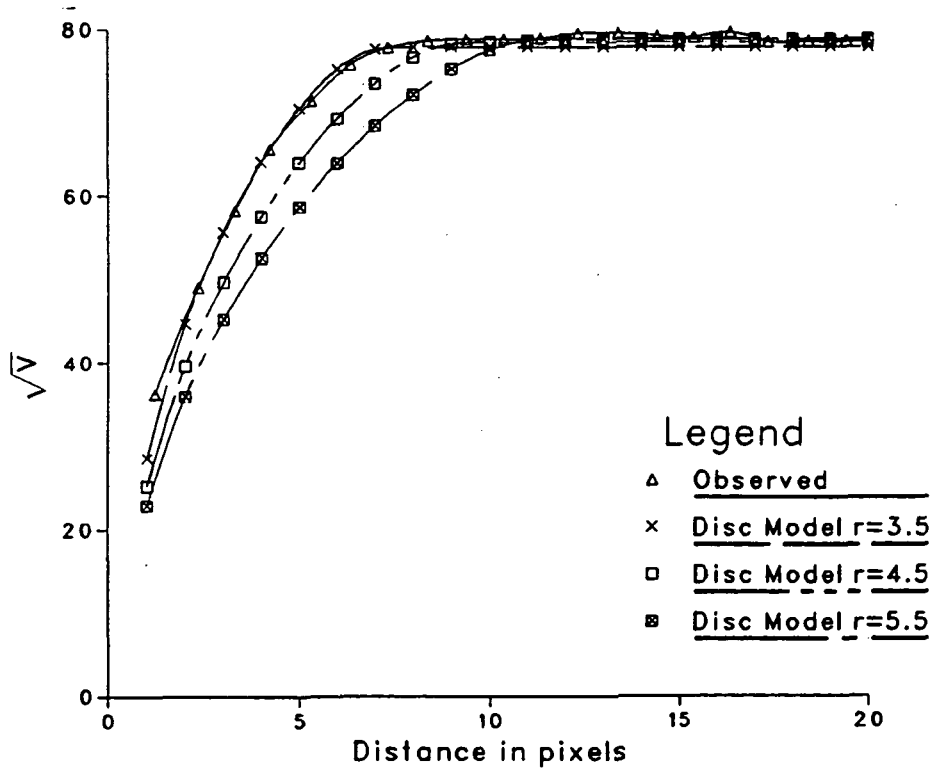


Figure 13. Comparison of the observed variogram from the simulated forest image with three disc model variograms for different size discs.

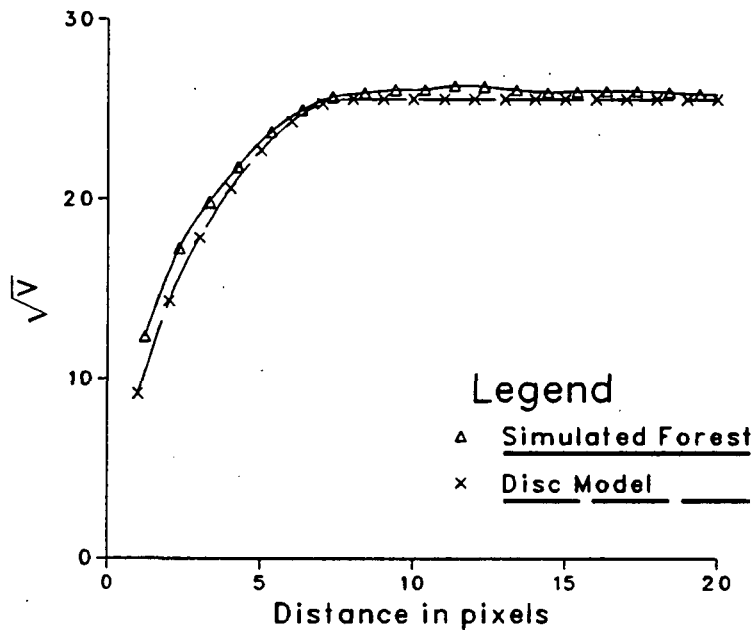


Figure 14. Comparison of the observed variogram from the simulated forest image and the disc model. The size of the discs used in calculation of the explicit variogram match the area of the cones in the image.

important because it indicates that the disc model might be used as a reasonable approximation of scenes with elements of other shapes.

3.3.2 Size Variance. The derivation of the explicit variograms assumes that all the discs are the same size, which is unlikely for real scenes. To test the influence of variance in the size of discs, an image was simulated using the same parameters of the initial simulation of the disc image (Figure 8) with the exception of the variance in disc size. As mentioned earlier, a lognormal distribution is used to describe the size distribution and its standard deviation was set intentionally high at 3.168. The resulting image is shown in Figure 15. To calculate an explicit variogram for comparison it was again necessary to determine the appropriate size to be used for the discs. The mean radius is not a good approximation as the area covered is related to the square of the radius, not the radius. Instead, a value for the radius that produces the same area covered by discs as

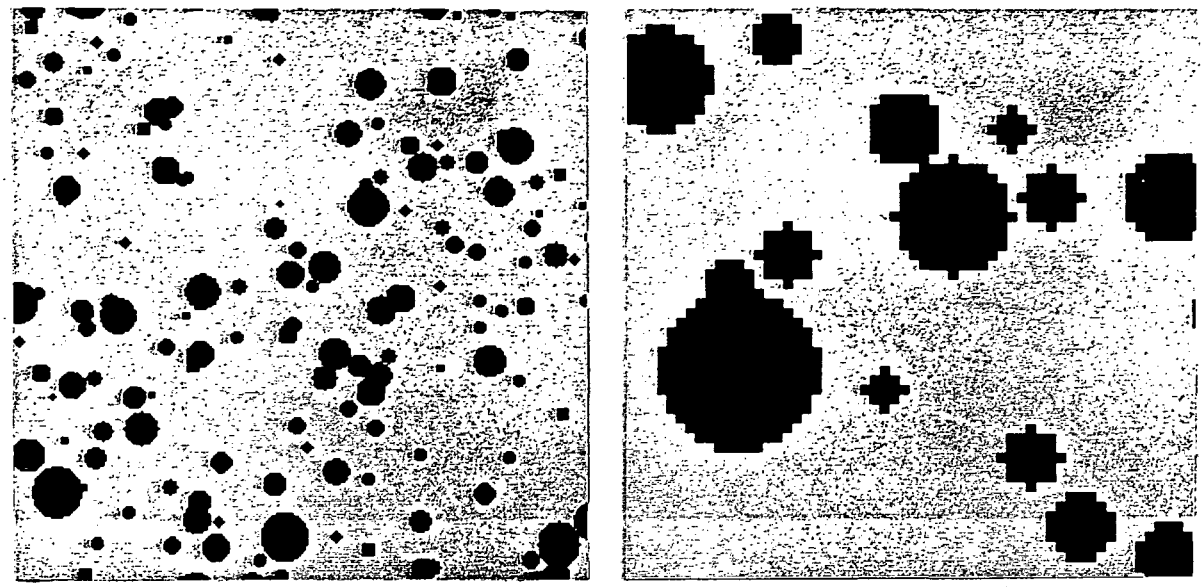


Figure 15. A portion of the simulated image in which the sizes of the discs are lognormally distributed (A), and an enlargement (B).

the lognormally distributed discs would be appropriate. This radius can be calculated using the mean *(m)* and variance (s^2) of the lognormal distribution:

$$r = \sqrt{m^2+s^2}$$

For the simulated image shown in Figure 8, the appropriate radius for use in the disc model is 4.72 meters.

Figure 16 is a comparison of the observed variogram from the simulated image with a lognormal distribution of disc sizes and the equivalent explicit variogram for fixed size discs. The two variograms agree closely with one interesting difference. The observed variogram exhibits a more rounded shape than the explicit variogram for fixed disc size. This rounded shape can be understood by examining the effect of the distribution of sizes on the variogram. At small distances, the variogram is a little higher than expected and at distances near the range of influence it is lower than expected. At short distances the

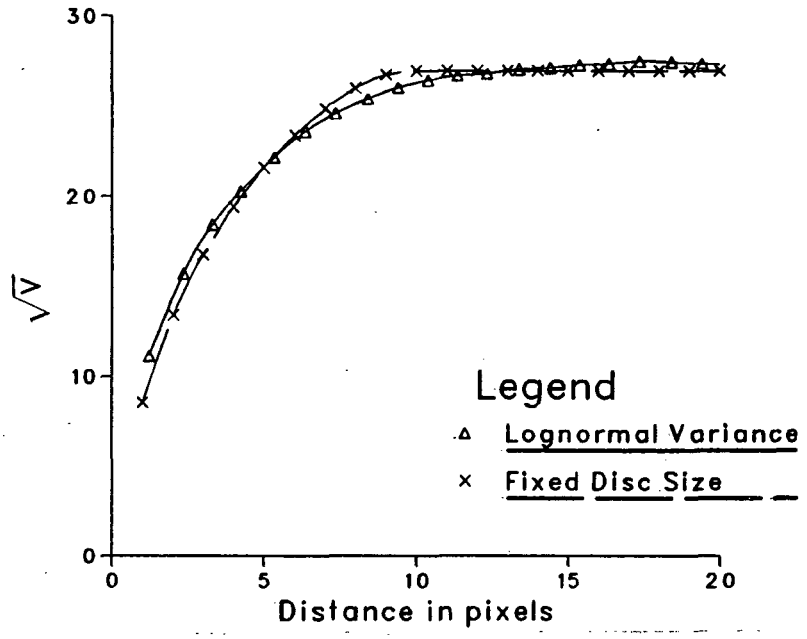


Figure 16. Comparison of the observed variogram from the simulated image with lognormal variance of disc sizes with an explicit variogram for a fixed size disc model.

existence of small discs causes an increased amount of perimeter for the same area covered, increasing the likelihood that movements of short distances will result in crossing a boundary. At distances near the range of influence, an opposite effect occurs. One result of the lognormal distribution is discs larger than the size of the fixed discs of the the explicit variogram. These discs reduce the likelihood of crossing a boundary at distances smaller than their diameter, which can still be larger than the zone of influence of the fixed disc model. This accounts for the difference between the two graphs in the 7- to 11-m range.

4. Remotely Sensed Images

The long range goal of this research is to be able to determine directly the characteristics of a scene using variograms derived from images of the scene. It has become apparent that extracting information from images is dependent on having a model for the scene and being able to determine explicit regularized

variograms for those scene models. To date, the ability to move directly between a scene model and an observed variogram has been demonstrated only for a simple disc model of scenes. This limited model is not sufficient to directly recover scene characteristics. However, through the use of the disc model and simulated images a considerable amount has been learned about the behavior of variograms in response to scene parameters. In this section, variograms from real images will be interpreted on the basis of the experience of the last sections. A brief summary of the major points learned through the disc model, explicit variograms, and image simulations that relate to interpretation of variograms from real images would emphasize the following:

- The height of the variogram, the sill, is related to the proportion of the area covered by objects, which is a function of their number or density.

- The distance to the sill, or the range of influence is related to the size of the objects in the scene. The shape of the variogram and the range of influence are more closely related to the area of objects than to their shape, at least for shapes not highly dissimilar from discs.

- The shape of variograms is related to the variance of the size of objects in the scene. A more rounded or gradual shape is characteristic of higher variance in the size of objects.

- Increasing the size of the units of regularization (which is analogous to increasing the spatial resolution of remotely sensed imagery) has the following effects on variograms: (1) the height of the sill is reduced, (2) the range of influence is increased, and (3) the height of the variogram at the distance equal to one unit of regularization increases relative to the sill.

In evaluating the variograms derived from real images, there are three things to be determined. The first has been mentioned and concerns the characteristics of the scene that can be determined on the basis of the variograms derived from images of the scene. The second issue to be addressed concerns the

applicability of the disc model to individual scenes at the resolution of the images. The third issue is to assess if another model for the shape of a variogram is more appropriate than the disc model. In particular, the exponential model holds interest because of its resemblance to the shape of the variogram from the simulated image with variance in the size of discs.

The approach used to compare variograms from observed images with the disc model requires calculation of an explicit variogram for a disc model with characteristics derived from the observed images. If the explicit variogram matches the observed variogram for the image, then the disc model can be assumed an appropriate scene model. To determine the necessary parameters for the disc model several steps are required. Objects in the image that represent "discs" must be identified. In order to match the assumptions of the disc model, the image must be stretched so that the "discs" are assigned one value (black for example), and the rest of the image to a different value (white). This black and white image will be used in the comparison with the disc model. From this image the percent cover of "discs," their approximate size, and the brightness of the discs and background are determined. These parameters are used to calculate an explicit variogram corresponding to the observed image. For comparison, an observed variogram is calculated from the black and white image. The quality of the match, and thus the appropriateness of the disc model for the image in question is evaluated visually. It is worth noting that this procedure can not be done or is not appropriate for all images. For example, objects occurring in the scene man not be well represented by discs. Also, it is critical that the objects can be separated spectrally from the background when converting the images to two tones.

The comparison of observed variograms with fitted exponential variograms is only a comparison of shape, as there is not a known scene model that is tied to the exponential model for the shape of a variogram. As such, the value of this comparison is limited and is done as an exploratory exercise. The actual comparison involves an empirical fit of the exponential model to the observed variogram. Ideally the form of the exponential model that should be fit is the regularized form given earlier. However, this form is considerably more complicated than the equation for the punctual variogram and would prove tedious to use. Instead a simple approach is used that is based on the model for the punctual variogram:

$$\gamma = c\,(1-\exp(h\,/\,a))$$

In this equation c and a are the unknown variables. In order to fit this model, the variance of the image is used as an estimate of c, the sill, and a is estimated using linear least squares of a natural logarithm transform:

$$\gamma' = -h\,/\,a$$

where

$$\gamma' = ln\,(\frac{c-\gamma}{c})$$

This approach forces the variogram through the origin, which is a requirement of all variograms. However, this form does not take into account regularization which can affect the behavior of the model near the origin. Figure 17 shows the effect of regularization on the exponential variogram, which is to reduce γ slightly at each h, resulting in a graph that is shifted to the right near the origin. Thus, an exponential model forced through the origin might be expected to be shifted to the left of the observed regularized variogram at short distances. This inconvenience is considered minor compared to the problems involved in fitting

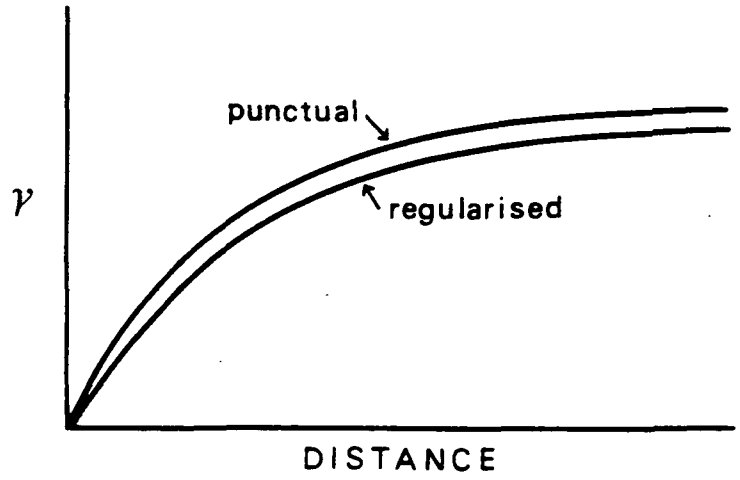


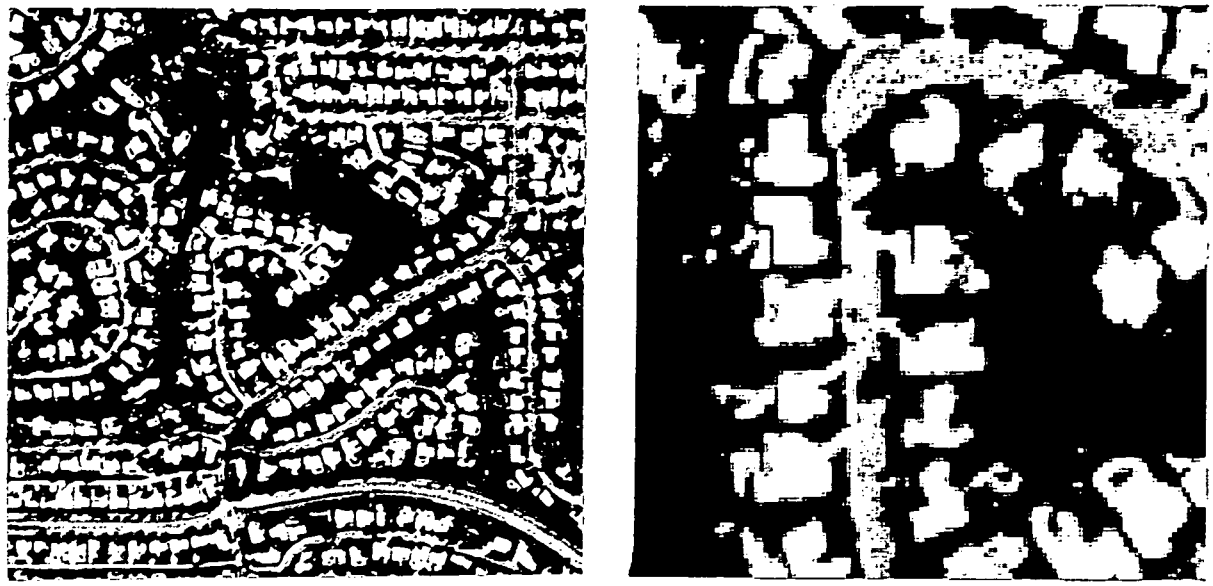
Figure 17. The effect of regularization on the exponential model.

the equation for the regularized variogram.

The examination of variograms from remotely sensed images involves three kinds of environments: forested, agricultural, and urban/suburban. For each environment there are images at two resolutions; very fine resolution (between 0.15 m and 2.5 m) designed to reveal the inherent structure of the scene, and 30-m resolution from the Thematic Mapper (TM) or Thematic Mapper Simulator (TMS).

4.1 Canoga Park Residential Image

An image of a residential portion of Canoga Park, California was obtained through NASA Ames Research Center (Figure 18). The image is from the red portion of the spectrum and has approximately 2.5-m resolution. This scene is complex in nature, having several kinds of elements arranged in a mosaic. The most obvious elements are houses (or roofs from the aerial perspective), trees, streets, lawns, cars, and a vegetated canyon that runs through the area. Close



A B

Figure 18. The Canoga Park residential image (A) at 2.5-m resolution, with an enlargement of a portion of the image shown in B.

examination of an enlargement of a portion of the image indicates that there are three distinctive tones in the image: bright tones which are houses, intermediate tones which are mostly streets, and dark areas which include vegetation of all kinds and shadows (Figure 18B). Vegetation covers most of the spaces between the houses and streets and is undoubtably composed of many types of plants, but in the observed image they all appear dark and can not be differentiated. In addition, these areas are sufficiently dark that they can not be differentiated from shadows.

The variogram calculated from this image is shown in Figure 19 and exhibits similar structure to the theoretical and observed variograms previously discussed. The variogram begins at a relatively low value and gradually rises to a level plateau. The distance at which the variogram levels is approximately twelve pixels, about equal to the diameter of the larger houses in the scene. The

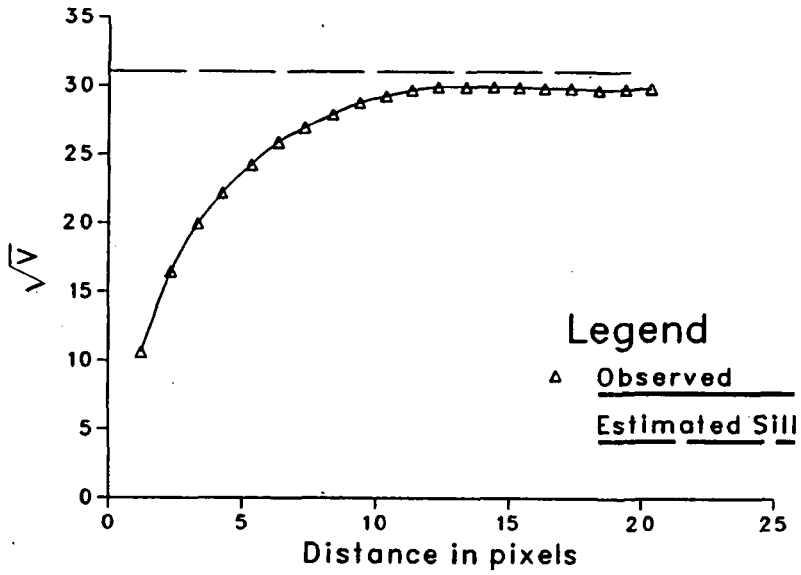


Figure 19. Observed variogram from the Canoga Park image.

strong influence of houses on the shape of the variogram is not surprising as they are the most distinctive and common elements in the scene.

The dashed line on the graph is the standard deviation of the image and serves as an estimate of the sill against which the observed variogram can be compared. This variogram approaches but does not reach the estimate of the sill over the 20-pixel distance for which the variogram was calculated. One reason may be that there are homogeneous areas in the image, such as the canyon, that are wider than 20 pixels. Because of these large, homogeneous areas, the difference between measurements for pixels a distance less than 20 pixels apart on average will be less than if they were selected at random. Under these circumstances the variogram would not quite reach the sill. The existence of these large areas in the image illustrates a point that will be important throughout this discussion, that remotely sensed images commonly exhibit several scales of variation. The ability to detect and understand multiple scales of variation in images will be important for interpreting variograms. In the long run, the ability to

derive information about multiple scales of variation in images from variograms may prove to be one of the attractive features of variograms.

To compare the disc model with the Canoga Park image, houses were used as "discs" and stretched to white, and everything else became background and was stretched to black. The resulting image (Figure 20) was compared to the disc model using several different fixed sizes of discs. The shape of the observed variogram from the black and white image generally resembles the disc model but does not match any of the sizes that were used (Figure 21). In general, the observed variogram is more rounded or gradual, not rising as sharply to the sill. This deviation from the disc model recalls the effect of variance in the size of discs, which may explain the observed situation because there is substantial variance in the size of houses in this scene. In addition, the observed image does not match two of the assumptions of the disc model. First, the houses are clearly not

Figure 20. The two-toned version of the Canoga Park image used for comparison with the disc model.

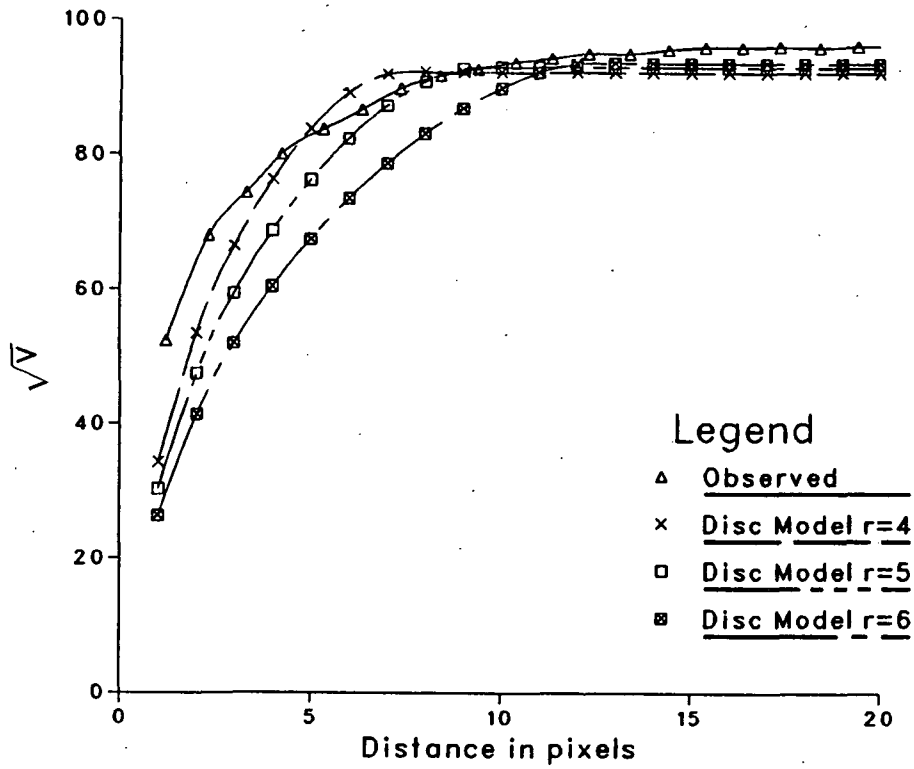


Figure 21. Comparison of the observed variogram from Figure 20 with three explicit variograms of the disc model for different size discs.

shaped like discs. The significance of this difference, however, may not be great since the forest simulations using elongated shapes showed a good fit to the disc model. A second factor that may be important is the regular location pattern of the houses, which violates the random assumption of the disc model. In particular, houses do not overlap, which was an important feature of the disc model.

The exponential shape fit to the Canoga Park variogram is compared with the original in Figure 22. Initially, the shape of the fitted model appears promising, but the quality of the fit is adversely affected by being forced through the origin. However, it is interesting to note that the direction of the deviation from the exponential model of the observed variogram is opposite of the expected influence of regularization. As mentioned earlier, the form of the exponential model that is fit to the observed does not take into account regularization, which

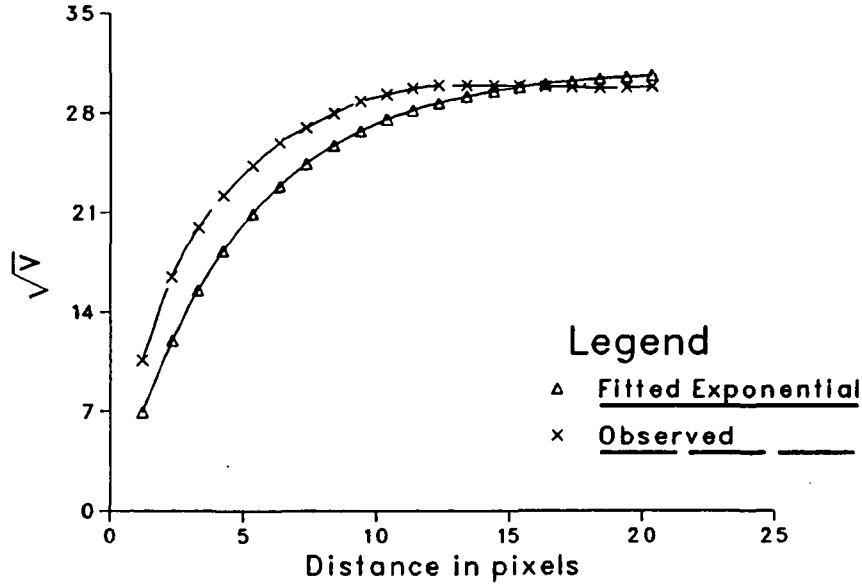


Figure 22. Comparison of the observed variogram with the exponential model.

would cause the exponential model to overestimate γ at short distances. The exponential model fit to the observed variogram underestimates γ at short distances. In addition, the Canoga Park variogram has a well-developed sill that is not present in the exponential shape. These factors combine to indicate that the exponential model is probably not a good approximation for this variogram.

4.2 Washington D.C. Thematic Mapper Image

A TM image of Washington D.C. was used as an example of an urban/suburban environment. The image is the red band (Band 3, .63 - .69 μm) on November 2, 1982 (Figure 23). Due to the diversity of the scene, variograms were calculated from two subareas of the image. One area includes the area around the Capitol, including numerous government buildings, the Mall, the Smithsonian, and several memorials and museums (Figure 24A). In this small area there are several types of elements: large buildings, lawns, roads, trees, and

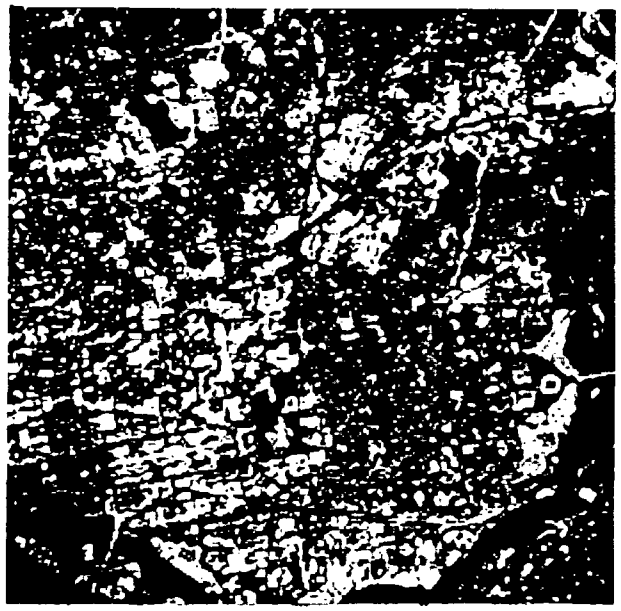
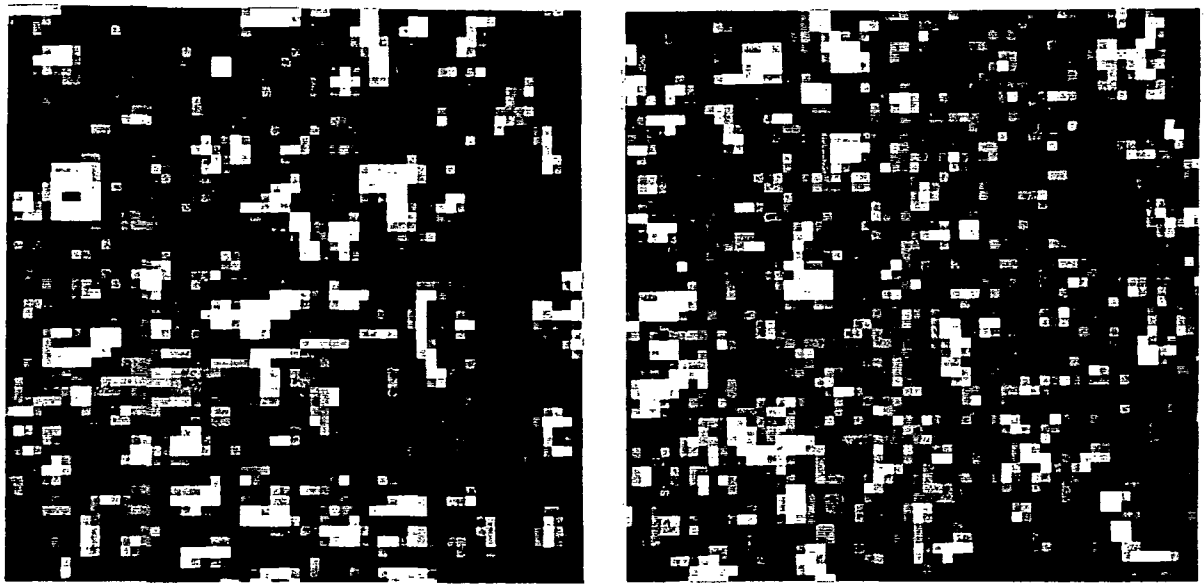


Figure 23. A TM image of Washington D.C..



A B

Figure 24. Enlargements of portions of the subareas of the Washington D.C. image used to calculate variograms. (A) is the "Capitol" area and (B) is the "city" area. The general contrast of these two subareas appear similar in this Figure, but this is an artifact of the preparation of the photographs. See Figure 23.

ponds. The variogram from this subimage looks considerably different from those

previously described. The variogram starts relatively high and rises abruptly in just 2 to 3 pixels to a gently sloping plateau (Figure 25). There are multipixel elements in the image but on average there is a high degree of difference associated with short movements in the image. The gently sloping plateau that does not reach the estimate of the sill indicates that there are homogeneous objects in the image of a wide variety of sizes.

The second subarea is a portion of the city that is directly east of the Capitol area and extends to Kennedy Stadium and the Anascotia River. This area of the city is primarily residential and commercial, with considerably smaller buildings and narrower streets. On the image of the entire Washington area (Figure 23) it appears as a fairly homogeneous region, medium grey in tone. However, considerable variation is visible within the area in the enlargement shown in Figure 24B. The variogram for this area is essentially flat, exhibiting behavior similar to the expectation for random data (Figure 26). There is a small drop from

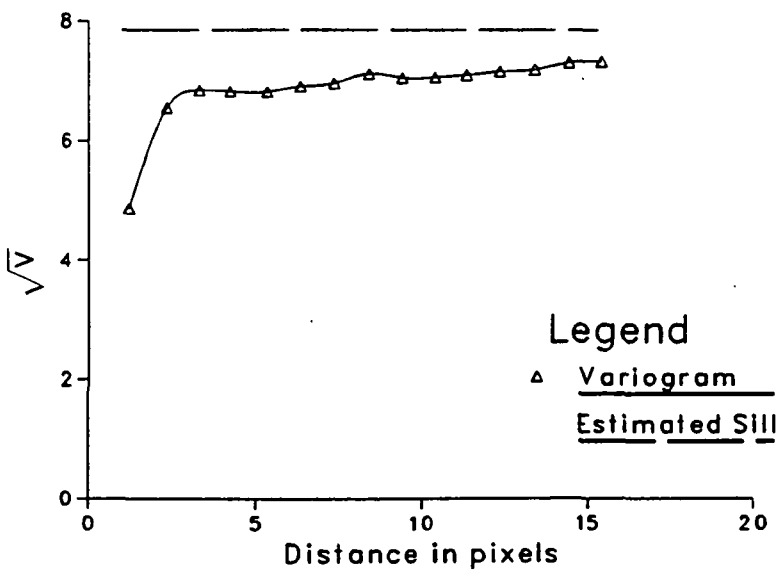


Figure 25. Variogram of the Capitol area in the Washington D.C. image.

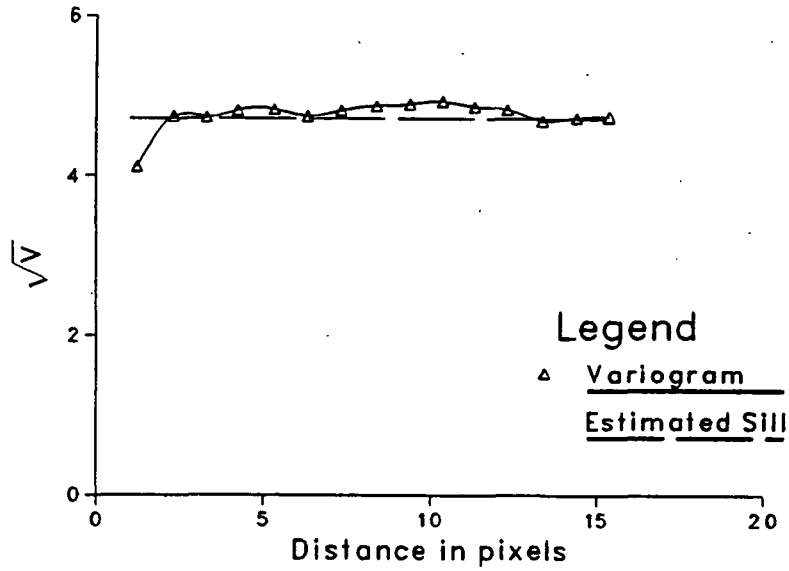


Figure 26. Variogram of the city subarea of the Washington D.C. image.

the random expectation at the distance of one pixel, but for greater distances the variogram has only minor fluctuations around the expected sill. This result is dramatic, as the relationship between neighboring pixels would be expected to be stronger solely on the basis of the overlap in the IFOV of the sensor. Close examination of the enlargement does show a general lack of multipixel elements in the area and a random appearance.

Figure 27 shows the variograms from both subareas of the Washington D.C. image plotted together for comparison. The variogram from the Capitol area is higher than the neighboring city area due to the higher overall variance or contrast between elements in that portion of the image. This graph also highlights the flat nature of both graphs indicating little spatial structure in this scene at the observed resolution.

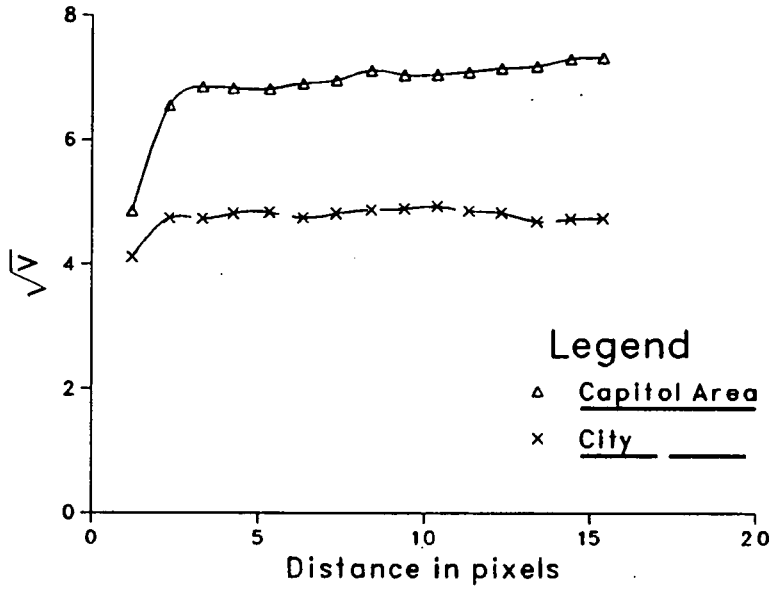


Figure 27. Composite of the variograms from both subareas of the TM image of Washington D.C..

Comparisons of these variograms with disc and exponential models were not done as they seemed inappropriate. There were not any definable groups of objects to serve as "discs" in either image. Also, the shape of the exponential model did not hold much promise for these variograms.

4.3 Agricultural Fields Image

To produce an image of an agricultural environment at very fine resolution (0.15 m), an aerial photograph of agricultural fields in Oklahoma was scanned using a microdensitometer at the Johnson Space Center. The image reveals the structure within fields (Figure 28). The crops, corn and soybeans, exhibit a distinct row structure and are near maturity as the canopy is almost closed. This image is relatively simple in structure, with crop rows, shadows, and an almost entirely obscured soil background as the only elements.

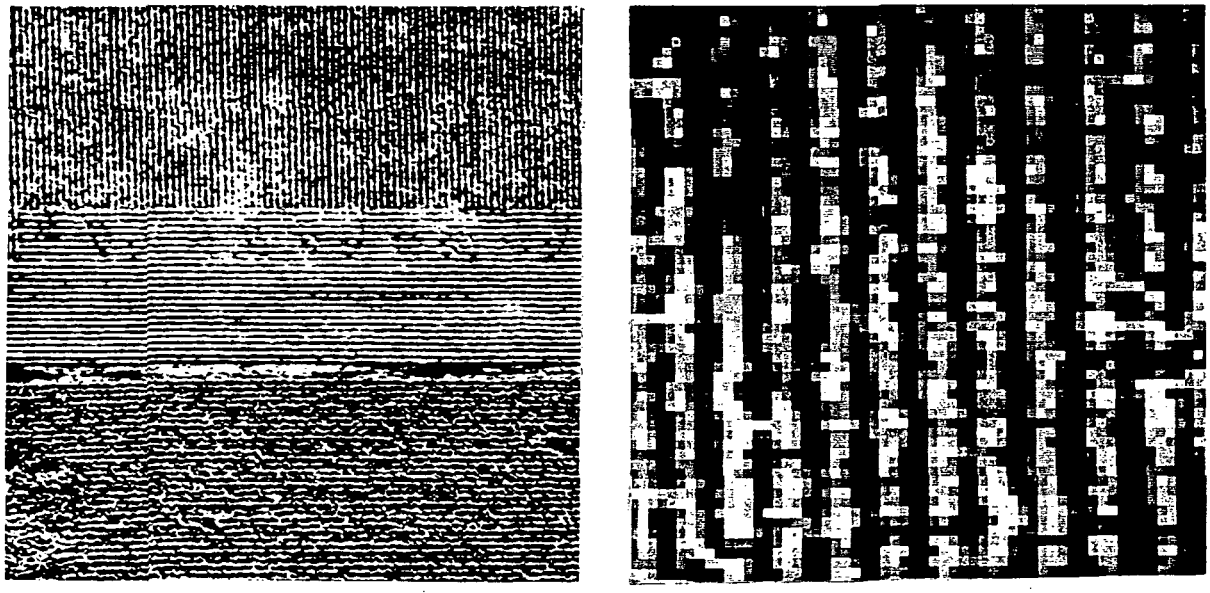


Figure 28. A portion of the image of agricultural fields (A) and an enlargement (B).

The shape of the variogram calculated from this image is wavelike, with repeating crests and troughs (Figure 29). The shape indicates the periodicity in

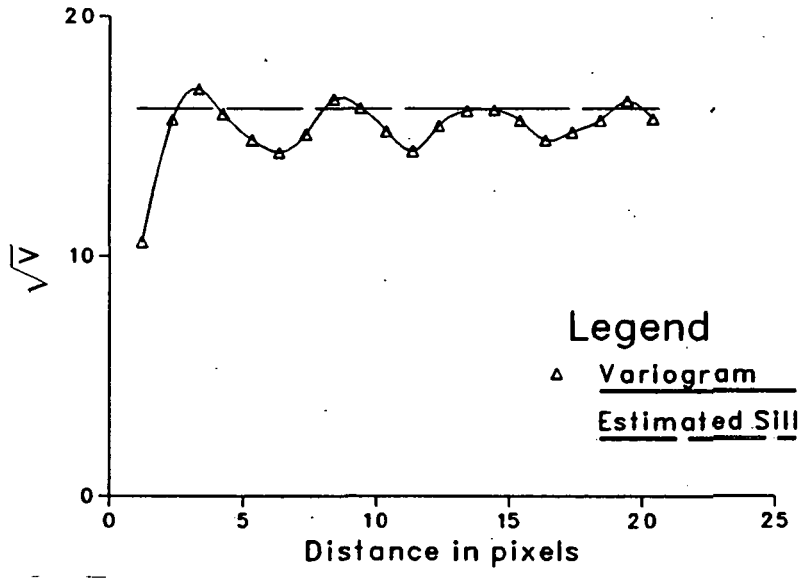


Figure 29. Variogram for the agricultural fields image.

the image, as the spacing of the rows remains constant throughout the image. The fact that the one-dimensional variograms are integrated over all directions has a profound impact on this variogram due to the strong anisotropy in the image. The variogram calculated over a single direction would look significantly different, and the observed variogram is best interpreted as the average of many variograms. First, consider the variogram calculated only in the direction along the rows. This variogram would be essentially flat and low relative to the estimated sill, as low variation is associated with movements of even large distances as long as the measurements are in the same position relative to the crop row.

A variogram calculated normal to the crop rows would look very different, with high peaks and low troughs. The troughs would be well below the estimated sill and would correspond to movements to the same relative position on a different row. The peaks would be well above the sill and correspond to movements to different parts of the rows, for example from the illuminated side to the shadow between rows. In addition, the variograms from all diagonal directions would contribute to the final observed result. The combined result still illustrates the periodicity of the rows, but the integration over all directions suppresses the magnitude of the effect. An interesting effect of this integration is the relatively high value of γ at a distance of one pixel, which is caused by the large amount of boundary associated with the rows.

No attempt was made to compare this variogram with either the disc or exponential models due to their obvious inappropriateness. If an attempt were made to fit a model, sinusoidal functions such as the sine or cosine would be more appropriate.

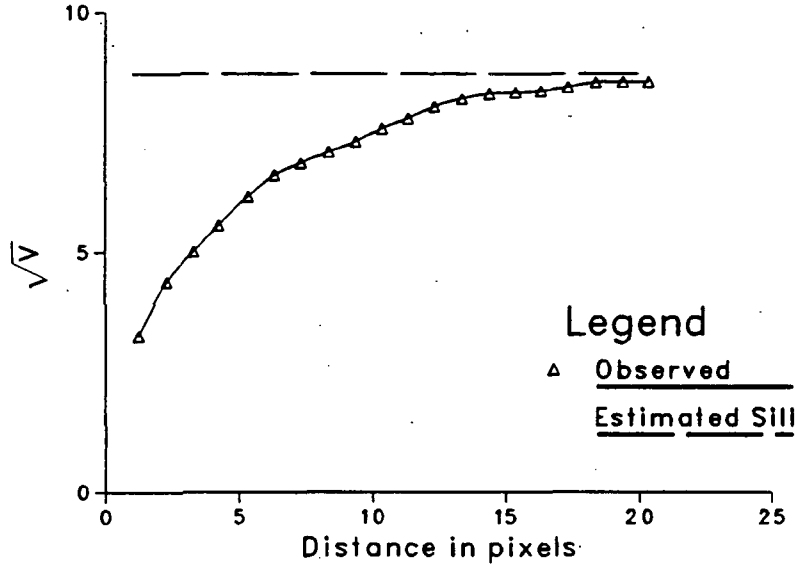


Figure 31. Variogram of the Thematic Mapper image of agricultural fields.

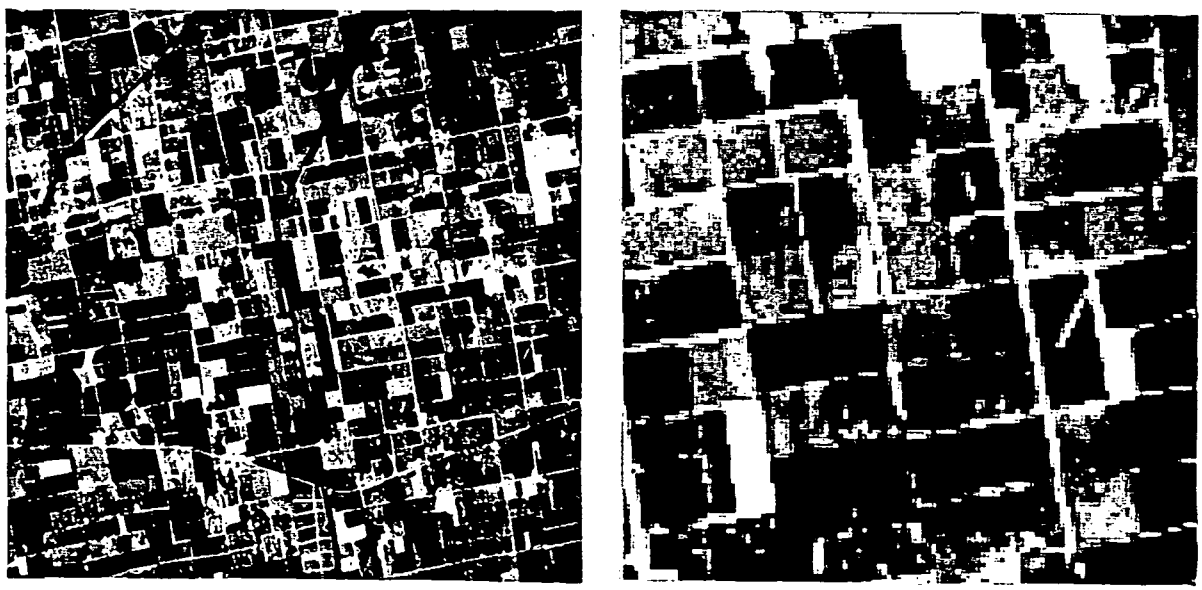
indicates that the fields in the image are relatively large. The most common field size in this scene is a quarter-section, which at 30-m resolution is 14 pixels in diameter. The variogram exhibits a break in slope at the 14 pixel distance, becoming considerably flatter. Although the variogram approaches the estimate of the sill, it does not quite reach it. This difference may be attributed to the fields that are two or more quarter-sections in size.

Due to the existence of only one kind of element in the image, fields, the comparison with the disc model is a little unusual. Instead of elements on a contrasting background, the image was stretched into bright fields and dark fields (Figure 32). Figure 33 shows the comparison of the variogram calculated from the black and white image with disc models using three different sizes of discs. The disc model does not seem appropriate for this image as the shapes are dissimilar. The disc model produces variograms that rise too sharply to a well-developed sill, where the observed variogram is more gradual and still gently

4.4 Thematic Mapper Agricultural Image

A TM Band 3 image was obtained from an area near Dyersburg Tennessee, which also includes the corners of Kentucky, Missouri, and Arkansas. The subimage used in this project covers an agricultural area west of the Mississippi River (Figure 30). The area looks like a patchwork of homogeneous blocks. With a change in resolution there is a change is the elements that describe the scene. The elements are now entire fields rather than the crop rows that comprise the fields.

The variogram calculated from this image begins at a low value and rises gradually to a value very close to the estimate of the sill at a distance of 18 pixels (Figure 31). The low values of γ at short distances indicate two features of the image: the relatively small amount of boundary in the image, and the homogeneity within the fields (Figure 30B). The gradual rise in the variogram



A B

Figure 30. The TM image of agricultural fields (A) and an enlargement (B).

Figure 32. Stretched version of the Thematic Mapper agricultural image for comparison with the disc model.

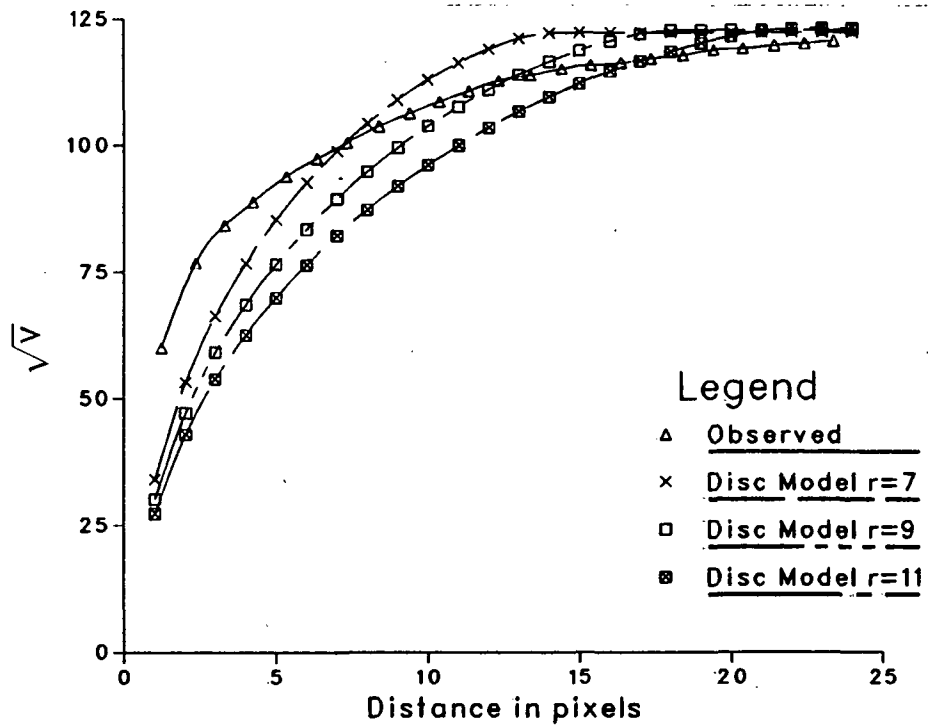


Figure 33. Comparison of the disc model with the observed variogram from the TM agricultural image.

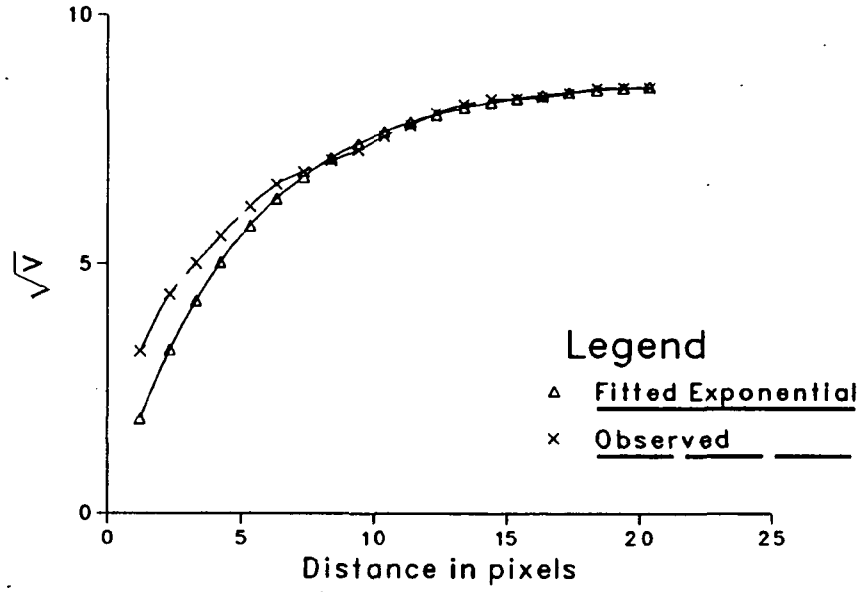


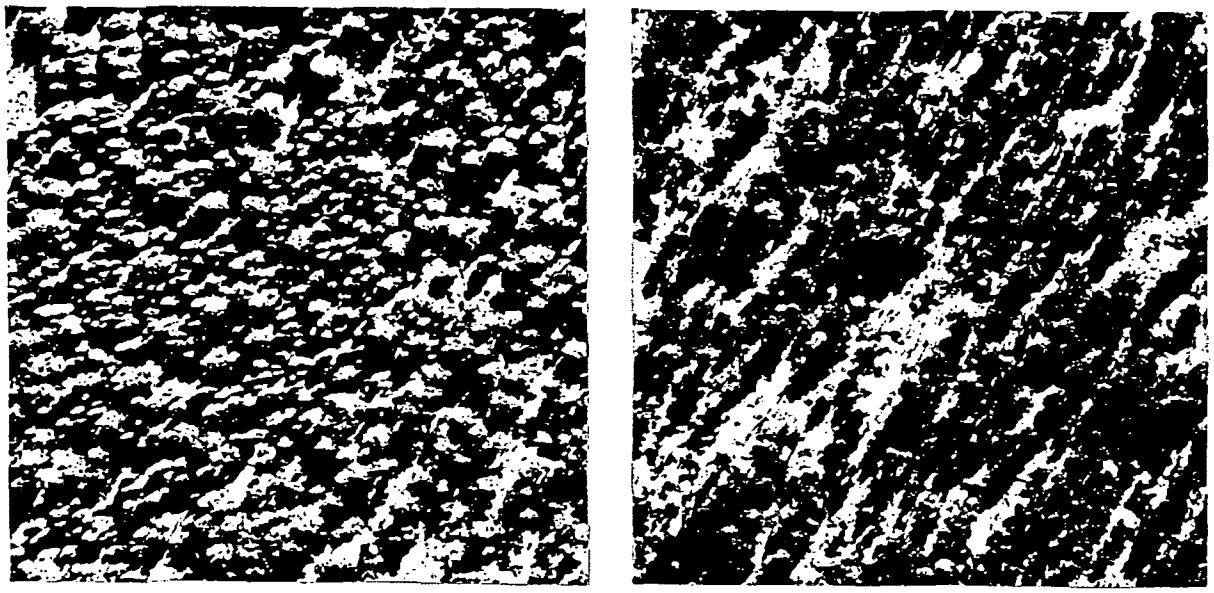
Figure 34. Comparison of the exponential model with the observed variogram for the TM agricultural image.

sloping at large distances. The inappropriateness of the disc model is not surprising as the elements in the scene dramatically violate the assumption of random, overlapping discs.

The fit of the exponential model to the observed variogram is close except at short distances (Figure 34). The poor fit near the origin is caused by the restriction forcing it through the origin, not by the shape of the exponential model which seems to match this variogram well. Again, the direction of deviation near the origin is opposite of the expected due to the lack of consideration for regularization.

4.5 South Dakota Forest Image

This image of a forest area in South Dakota (Figure 35) was created by scanning an aerial photograph using a microdensitometer at Johnson Space Center. The exact location of the area covered in South Dakota is unknown, but it serves as a good example of a simple forest environment composed of trees on a



A B

Figure 35. The two subareas of the South Dakota forest image: Dense (A), and Sparse (B).

relatively smooth background. The spatial resolution is approximately 0.75 m and a red filter was used in scanning the image.

Variograms were calculated from two subareas of this image due to the variation in the density and size of trees in the image. One subarea is more densely stocked and the trees are somewhat smaller (Figure 35A). The variogram from this area rises gradually but does not quite reach the sill (Figure 36). It is difficult to determine the distance to the sill, which should correspond to the tree diameter. By counting pixels in the image, an estimate for the diameters of trees of 8 m (10 or 11 pixels) is obtained. By this point γ is close to the estimated sill, but it still continues to rise slightly at distances past that point.

The second subarea (Figure 35B) is more sparse than the last site and the trees are a little larger. The variogram has a very similar shape but does not come as close to the estimate of the sill (Figure 37). As can be seen in Figure

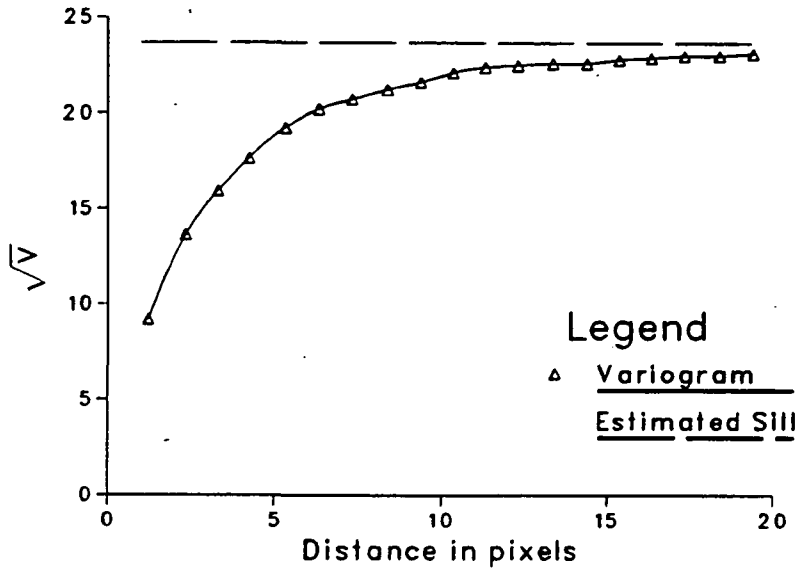


Figure 36. Variogram of the densely stocked subarea of the South Dakota forest image.

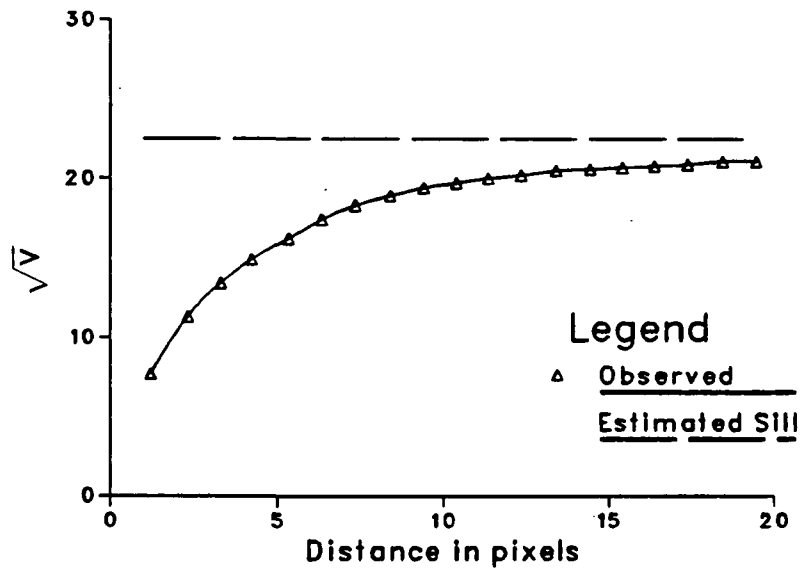


Figure 37. Variogram of the more sparsely stocked subarea of the South Dakota forest image.

35B, there are larger areas of background in the sparse subarea, explaining the difference between the estimate of the sill and the variogram at distances larger than the size of trees. In this variogram it is also difficult to determine a well-

defined break in the variogram that might reflect the size of trees. Counting produces an estimate of 10m or approximately 13 pixels for the diameter. Again, γ at this value has risen to a high level and is increasing at a very slow rate.

The composite of both forest variograms (Figure 38) illustrates the effect of density, or percent cover on variograms. The variogram from the more dense area is higher than the variogram from the sparse area, empirically demonstrating the effect shown in Figure 4.

A comparison of the South Dakota forest image with the disc model was attempted, but proved impossible because the trees and shadows could not be reliably separated from the background on the basis of the one spectral band available. The problem was that the well-illuminated portions of many of the tree crowns were of the same tone as the background.

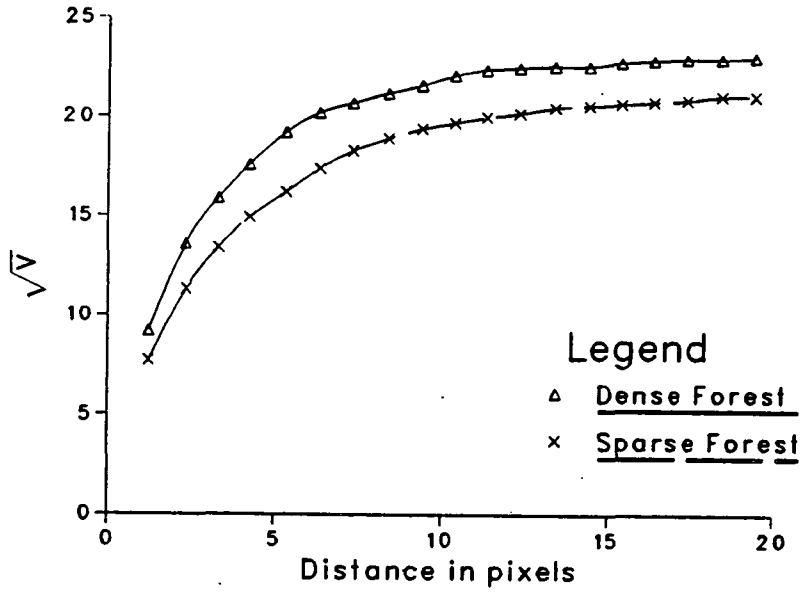


Figure 38. Composite of the variograms from both subareas of the South Dakota forest image.

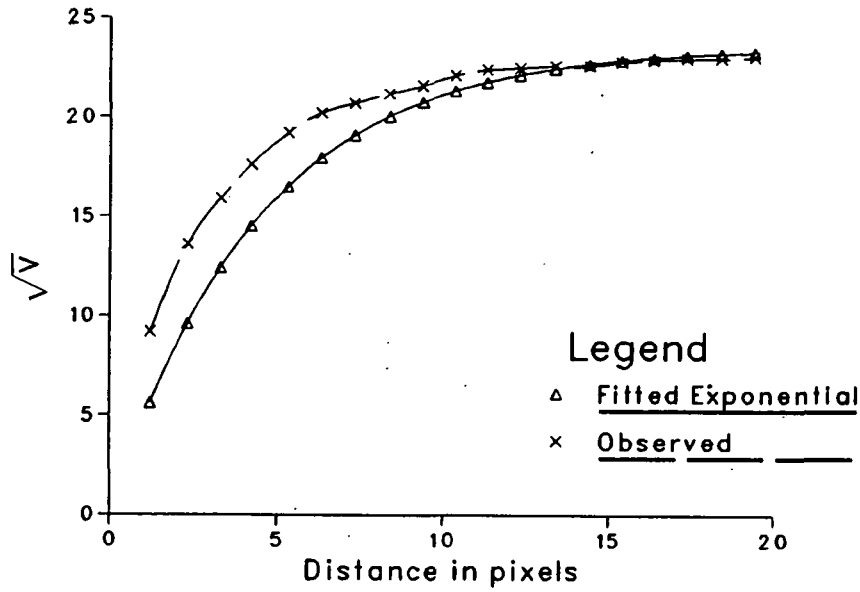
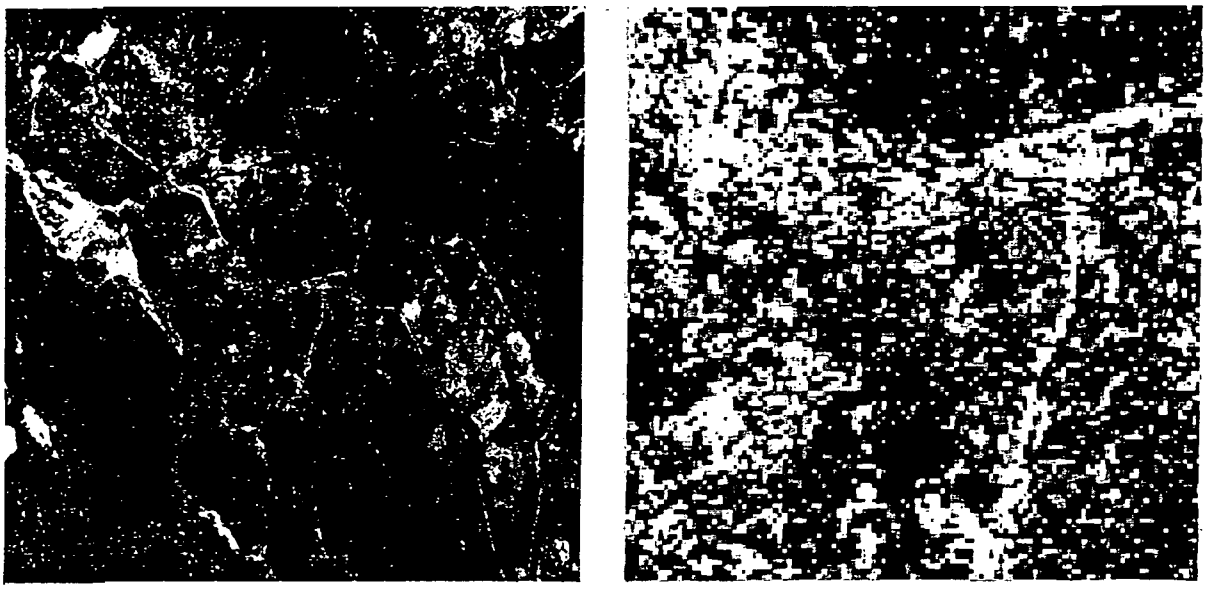


Figure 39. Comparison of the exponential model with the observed variogram from the densely stocked subarea of the South Dakota forest image.

An exponential model was fitted to the variogram from the dense subarea with results similar to those for the Canoga Park image and the TM agricultural image. The shape seems promising, but the deviation from the observed near the origin is opposite of that expected (Figure 39).

4.6 Thematic Mapper Simulator Forest Image

This Band 3 (.63 to .69 μm) image (Figure 40) from the TMS was obtained from NASA Ames Research Center and serves as an example of a forest environment at 30-m resolution. The image is from an area in northern California near Mt. Shasta that is close to the area where the field data were collected to calibrate the simulations. The area is reasonably flat and is primarily eastside pine, a vegetation association that runs along the east slopes of the Sierra Nevada and continues in extensive stands on many dry, flat areas of northeastern California. *Pinus Jeffreyi* and *P. ponderosa* are the dominant tree species in stands that tend



A B

Figure 40. A portion of the TMS image (A) of a forested area in northern California, and an enlargement (B).

to be sparse with a broken understory of shrubs and grasses.

The elements in this scene model have different characteristics than those previously discussed. At 30-m resolution in a forest environment the trees are considerably smaller than the resolution cells, and thus are not useful as elements in the scene model. Instead, stands of trees, or areas within which the characteristics of the trees are similar, become the elements. The use of stands as scene-model elements is different from those previously discussed because of the high internal variance of the forest stands (Figure 40B). In all other cases the elements have corresponded to objects that were spatially homogeneous, with low internal variance. The result of the high internal variance associated with forest stands is the relatively high level of γ at short distances (Figure 41). In general, the variogram exhibits a gently sloping, almost linear shape. This shape is attributable to the wide variety of sizes and shapes of the forest stands in the scene.

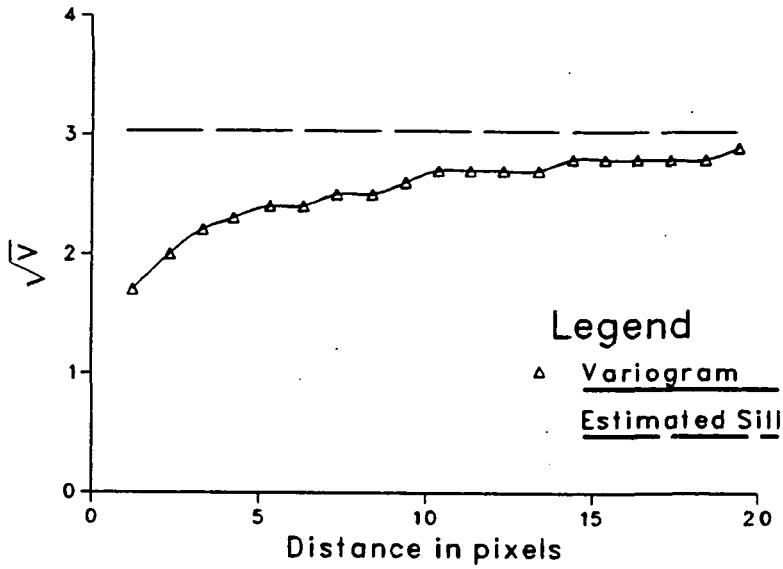


Figure 41. Variogram of the TMS forest image.

It is difficult to find anything approaching a common size for forest stands (Figure 40A). The variogram does not reach the estimated sill at a distance of 20 pixels which is attributable to the large stands in the scene. This image serves as a good example of the importance of scale, as variance can occur both within elements in the scene and between elements and both factors will influence the shape of the variogram.

5. Conclusions

Variograms are a useful tool for studying spatial variation in remotely sensed images. Theoretically derived variograms for simple scene models illustrated two features of the relationship between the characteristics of scenes and variograms. First, the range of influence in a variogram is related to the size of the objects in the scene. Second, the height of the sill is determined by the percent cover of the objects. In addition, the theoretically derived variograms were used to investigate the effect of regularization on variograms. The concept of regularization is critical in the use of regionalized variables in conjunction with

remotely sensed images as individual measurements are integrated over areas and are not point measurements. The units of regularization in a regionalized variable are analogous to the spatial resolution of a sensor in remote sensing. The effects of increasing the size of the regularizing units were shown to be: (1) decreasing the height of the sill, (2) increasing the range of influence, and (3) increasing the height of the first observed value of the variogram relative to the sill.

The simulated images served as a bridge between theoretical variograms for simple scene models and observed variograms calculated from remotely sensed imagery. The image simulations were done using a modification of a computer model of a coniferous forest. One result of the images simulations was the demonstration of the link between theoretical and observed variograms via a matching of these two types of variograms for a "disc model" of a scene. In addition, the area covered by objects was found to have more effect on one-dimensional variograms than their shape, at least for shapes not highly dissimilar from discs. Also, variance in the size of objects produces a more rounded shape in variograms than the fixed-size disc model.

The analysis of variograms calculated from remotely sensed images proved informative and served to: (1) empirically demonstrate many of the effects observed through the use of theoretical variograms and image simulation, (2) suggest that information about a ground scene can be recovered from variograms of images of the scene, and (3) show the importance of understanding multiple scales of effects in the interpretation of variograms derived from real images.

6. References

[1] Clark, I., *Practical Geostatistics.* (Applied Science Publishers Ltd., Essex, England, 1979).

[2] Franklin, J., Michaelson, J., and A.H. Strahler, Spatial Analysis of Density Dependent Patterns in Coniferous Forest Stands, *Vegetatio,* in press.

[3] Li, X., An Invertible Coniferous Forest Canopy Reflectance Model, M.A. Thesis, Department of Geography, University of California, Santa Barbara (May 1981).

[4] Li, X., and A.H. Strahler, Geometric-Optical Modeling of a Conifer Forest Canopy, *I.E.E.E. Transactions on Geoscience and Remote Sensing,* in press.

[5] Matheron, G., *The Theory of Regionalized Variables and Its Applications.* (Les Cashiers du Centre de Morphologie Mathematique de Fontainebleau, 1971).

[6] Serra, J., *Image Analysis and Mathematical Morphology.* (Academic Press, 1982).

[7] Woodcock, C.E., Understanding Spatial Variation in Remotely Sensed Imagery, Ph.D. Thesis, Department of Geography, Univeristy of California, Santa Barbara (1985).

[8] Woodcock, C.E., and A.H. Strahler, Image Variance and Spatial Structure in Remotely Sensed Images, Proceedings of the NASA Symposium on Mathematical Pattern Recognition and Image Analysis (1984).

Influence of Ground Control Point Selection on Landsat MSS and TM Rectification Accuracy: Whole Scene vs. Portions of a Scene

David D. Dow

National Aeronautics and Space Administration
Earth Resources Laboratory
National Space Technology Laboratories
HA10; Bldg. 1100
NSTL, Mississippi 39529

Abstract

An investigation of the influence of ground control point selection on the rectification accuracy of Landsat MSS was conducted on data from southeastern Louisiana/coastal Mississippi and eastern Kansas. The analysis investigated areas ranging from a full Landsat scene to a quarter of a scene in area. The optimum nmber of ground control points required to rectify a full or partial Landsat MSS scene is 24. An investigation of the spatial arrangement of ground control points showed that a random and regular pattern gave comparable rectification accuracy which was much better than that obtained when the ground control points were clustered. Excellent rectification accuracy for the random and regular spatial distribution cases was indicated by a row bias of 0.11 pixels and a column bias of 0.26 pixels for the Louisiana scene, while for the Kansas data the row bias was 0.15 pixels and the column bias was 0.27 pixels. A quarter of a TM scene from Louisiana with a random and a regular spatial distribution of ground control points was analyzed with a row bias 0.07 pixels and a column bias of 0.08 pixels. These results are discussed in light of other data from the scientific literature.

Introduction

This investigation focuses on the influence of ground control point (GCP) selection on the scene-to-map registration accuracy of Landsat Multispectral Scanner (MSS) and Thematic Maper (TM) data. The rectification of Landsat MSS data to a Universal Transverse Mercator (UTM) or other map base is an important pre-processing step in the analysis of earth resources science data. This study will investigate the influence of the number and spatial distribution of GCPs on the rectification accuracy.

The accuracy with which GCPs can be selected is an important source of error in the rectification of Landsat MSS and TM data. The construction of a mapping equation relates the Landsat scene coordinates of a GCP (element and scan line) to the map coordinates of the GCP (eastings and northings in the UTM system). Investigations of GCP selection accuracy revealed the following (Mikhail and Paderes [9]; Steiner and Kirby [13], and Welch and Usery [17]):

1. GCPs can be selected more accurately on maps than on Landsat images (GCPs on images can be determined to an accuracy of +0.5 data pixels if refinements are employed in choosing the GCPs).
2. GCPs can be measured more accurately on man-made features (road intersection) than on natural features (land-water interface).
3. Better rectification accuracy in the mapping equation is obtained if higher degree polynomials are employed as well as more GCPs are used.
4. The rectification process compensates better for errors in the ground position of control points than it does for errors in the image position.
5. Sub-pixel rectification accuracy can be accomplished only if points on the image can be identified to a sub-pixel level.

The affine transformation and higher degree polynomials are an example of interpolative or surface fitting models in which a least squares approach is used to generate residuals which measures how well the data (GCPs location in the map and Landsat image) fits the mapping equation. The root mean square (RMS) value is a measure of the degree of fit. The residuals stem from non-linear distortions in satellite orbit and attitude, errors attributable to the curvature of lines resulting from earth rotation and map projection, scanner mirror velocity nonlinearity, and random variation. Wong [18] reported an RMS value of $\pm$57 meters for a 20 term polynomial, while the RMS value for a first degree polynomial applied to the same Landsat data was $\pm$115 meters. There is a trade off involved, however, in that up to 30 GCPs must be used per Landsat frame to provide a least squares solution to a 20 term polynomial, which is many more GCPs than is required for the least squares solution of a lower degree polynomial. Also, a higher degree polynomial requires that the GCPs must be well distributed near the edge and corners of the frame (Van Wie and Stein [15]; Walker et al. [16]).

The P-format Landsat MSS tapes (spatially and radiometrically corrected) have associated with them a quality assessment number, which is truncated integer of the form (N+7)/8 (where "N" is the number of GCPs employed to rectify a scene of Landsat MSS data). The quality assessment numbers range from zero (machine corrected without utilizing GCPs) to 5 (33-40 GCPs employed by Master Data Processor). In practice there is not a straight forward relationship between increasing quality assessment number and better rectification accuracy (Graham and Luebbe [6]). In theory if 25 to 50 GCPs are used the rectification accuracy should be within 1 pixel more than 99% of the time (Nelson and Grebowsky [10]). A number of investigators (USGS [14]; Colwell et

al. [2]; Graham and Luebbe [6]; Dow [4]) have reported multiple pixel rectification inaccuracies in P-format Landsat MSS data.

The other type of model used in rectification is the parametric model which incorporates information on satellite position and sensor attitude (Horn and Woodham [8]; Sawada et al. [12], Mikhail and Paderes [9]; Paderes, Mikhail, and Forstner [11]). Mikhail and Paderes [9] developed a satellite collinearity equation to combine the sensor and platform parametric models. In this case the GCPs were employed to estimate the unknown parameters in the collinearity equations (there were 19 unknown parameters in the 1983 version of the parametric model of Mikhail and Paderes [9]). Some of the conclusions of the research by the Purdue group are (Mikhail and Paderes [9]; Paderes, Mikhail, and Forstner [11]):

1. The maximum rectification accuracy for a polynomial model is about half a pixel.
2. Rectification accuracy is not significantly improved when the number of GCPs utilized exceeds 25.
3. Rectification accuracy is better if the GCPs are regularly distributed in space, rather than being randomly distributed.
4. The collinearity model gives a lower RMS value for the same number of GCPs than does the polynomial model. (the difference being more pronounced for 10 GCPs than for greater than 40 GCPs).

Methods

The Landsat 2 MSS frames used in this study were acquired over path: 23 and row: 39 of the world wide reference system (southeastern Louisiana-coastal Mississippi) and over path: 29 and row: 33 (western Missouri-eastern Kansas). The Kansas data was collected on November 11, 1981, while the Louisiana data

was collected on November 21, 1981. Both Landsat MSS scenes had 10% cloud cover. The Landsat 5 TM frame employed in this investigation was acquired over path: 22 and row: 39 of the world wide reference system. The TM quadrant utilized covered parts of southeastern Louisiana and south-central Mississippi. The TM quadrant utilized was basically cloud free and was collected on September 13, 1984.

Figure 1 illustrates the differences between the Kansas and Louisiana MSS data sets. The pictures represent a band 7 density slice, to separate the water in black from the gray-toned land. In order to display the whole Landsat MSS scene on the image display device, only every sixth line and every sixth element is displayed. The Louisiana data set features the New Orleans metropolitan area with Lake Pontchartrain in the left center of the frame and has the Gulf of Mexico at the right of the scene. The Kansas scene features the Kansas City metropolitan area in the upper right hand corner of the photograph with the Topeka metropolitan area a little bit left and north of center. The Kansas scene was hilly (elevation 730 to 1450 feet above sea level) with only small amounts of open water (mostly as reservoirs). The Louisiana scene was relatively flat (elevation: 0 to 362 feet above sea level) and contained up to 35% open water. The extensive amount of open water and wetlands in the Louisiana scene present a significant challenge for accurate scene-to-map registration when compared to the Kansas Landsat frame.

The points to be utilized as ground control points (GCPs) and ground reference points (GRPs) were chosen on 1:24,000 scale, 7.5 minute quadrangle sheets produced by the U.S. Geological Survey (USGS). The GCPs are used to generate the mapping equations used in the rectification procedure, while the GRPs were employed as test points to independently evaluate the accuracy of

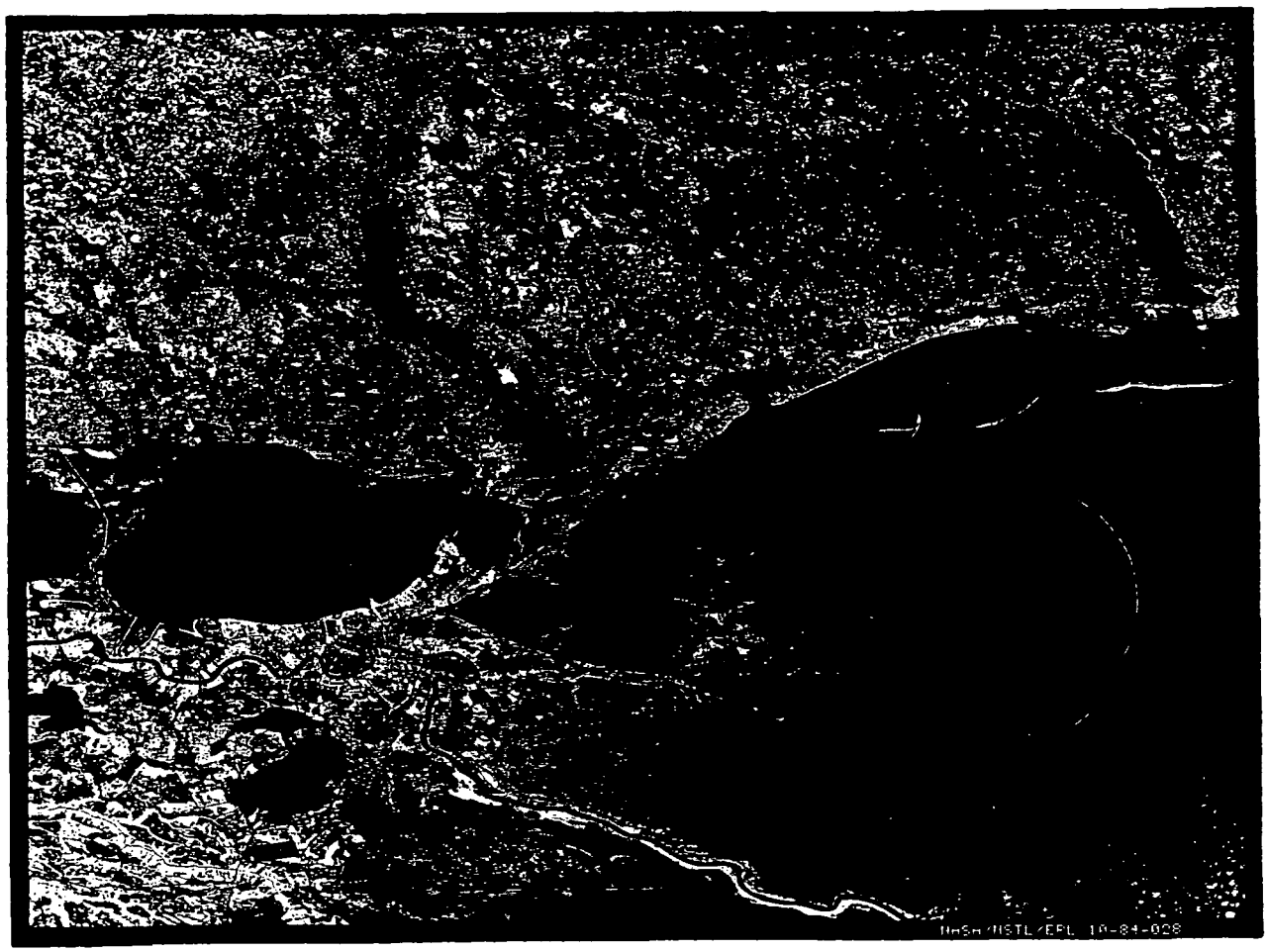


Figure 1a. Landsat MSS frame of the Louisiana-Mississippi area. Band 7 density slice with water in black and land as gray tones. The scene is reduced six fold with every sixth line and element being displayed.

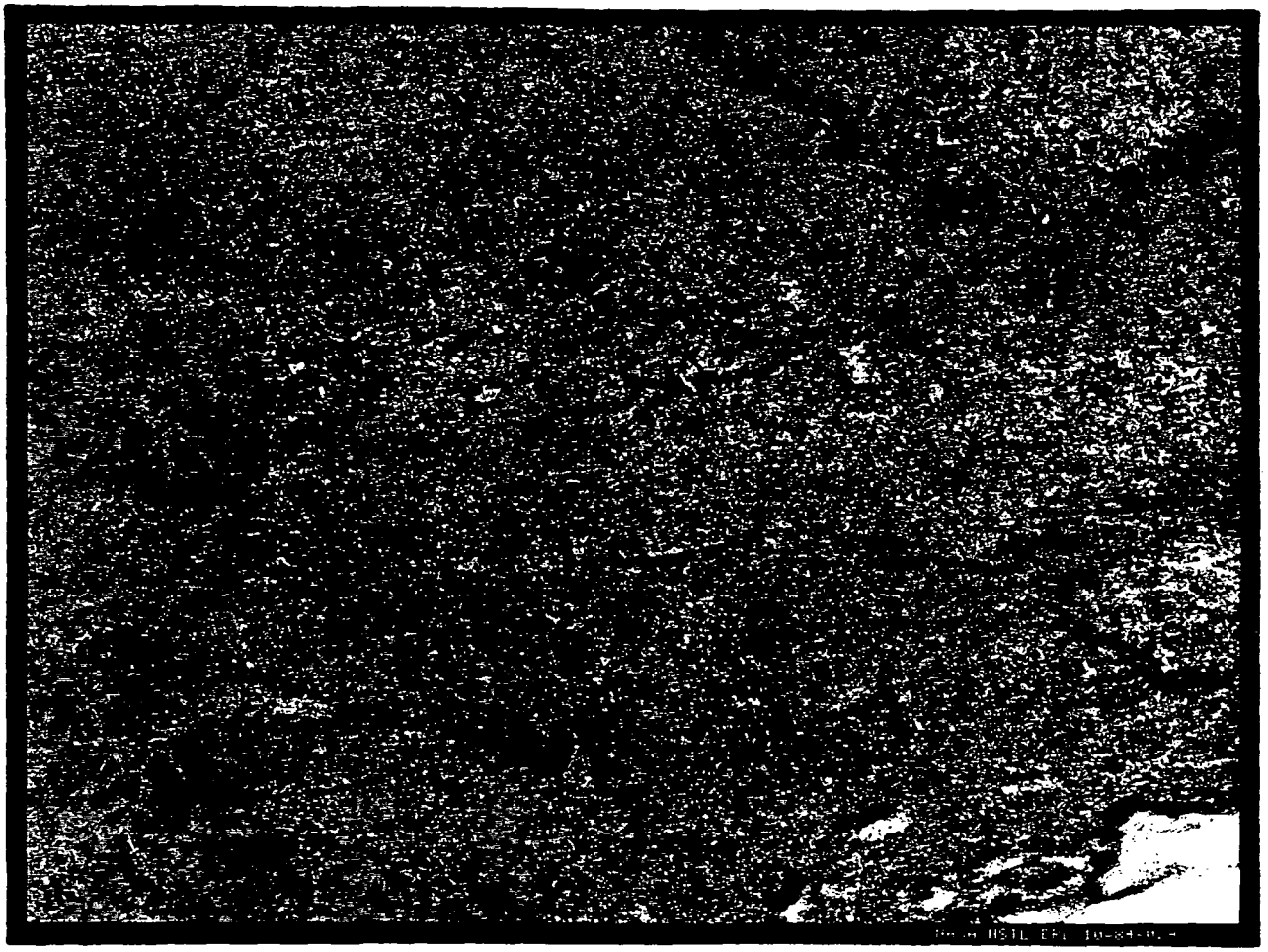


Figure 1b. Landsat MSS frames of the eastern Kansas region. Band 7 density slice with water in black and land as gray tones. The scene is reduced six fold with every sixth line and element being displayed.

the georegistration procedure. The ground point map coordinates were recorded in the UTM system as northings and eastings, while the Landsat scene coordinates were recorded as scan lines and elements. The same points were identified on the 7.5 minute USGS quadrangle sheet and the Landsat A-format MSS frame. Man made (road intersections) and natural (river intersections) features were used as ground points. For the whole scene analysis 356 ground points were selected for the Louisiana data set and 359 ground points were used in the Kansas data set. The TM quadrant utilized 361 ground points in the rectification accuracy experiment. The ground points available were divided into GCPs and GRPs.

The Earth Resources Laboratory Applications Software (ELAS) package developed at the National Space Technology Laboratories was used in all the subsequently described analysis (Graham et al. [7]). The mapping equation utilized was a linear polynomial and the fit of the GCPs to the mapping equation was quantified by the computation of the RMS value through the ELAS module BMGC. To evaluate the rectification accuracy of the Landsat MSS and TM products, the procedure of Graham and Luebbe [6] was utilized. This procedure quantifies the rectification accuracy in terms of RBIAS (row offset), CBIAS (column offset), RSD (row standard deviation) and CSD (column standard deviation). Good georegistration accuracy would be characterized by sub-pixel offsets and standard deviation values.

The equations for computing bias and standard deviation are:

$$(1) \qquad RBIAS = \frac{\sum_{i=1}^{NP} (ROW1i - ROW2i)}{NP}$$

$$(2) \qquad RSD = \sqrt{\frac{\sum_{i=1}^{NP} (ROW1i - ROW2i - RBIAS)^2}{NP-1}}$$

where NP is the number of GRPs utilized, Row 1 is the Landsat row predicted from the mapping equation, and ROW 2 is the Landsat row read from the MSS or TM imagery. The units of RBIAS and RSD are in pixels. The ELAS module BMGC is used to compute the bias and standard deviation values.

The module SSPA was utilized to compute "R" values which give a measure of the spatial distribution of ground control point (Dow [4]). The "R" value compares the mean observed nearest neighbor distance (irrespective of direction) to the mean nearest neighbor distance if the population was distributed at random (Clark and Evans [1]). The "R" values can range from 0 (maximum aggregation or a clustering of points) to 2.15 (maximum spacing or a regular/uniform distribution of points). For the purposes of this paper "R" values of between 0.7 and 1.3 are indicative of a random spatial distribution, while values less than 0.7 indicate a clustered distribution and values greater than 1.3 denote a regular distribution. Another feature of the module SSPA is that given a file of ground points, it allows the operator to choose a subset of GCPs that are distributed randomly, regularly, or in a clustered format. The clustered distribution of GCPs was conducted around four independent locii spread throughout the scene for whole frame analysis. For the half scene analysis for a clustered distribution three independent locii were utilized, while two independent locii were used in the quarter frame analysis.

The GCPs were chosen in intervals of eight in order to coincide with the quality assessment numbering system used to indicate how many GCPs were utilized to rectify a scene of Landsat P-format MSS and TM data. The number of GCPs used in the MSS data analysis is 8, 16, 24, 32, and 40 (Tables 1 through 9, while the number of GCPs utilized in the TM data analysis is 8, 16, 24, 32, 40, 48, and 56 (Table 10, 11, and 12). In this paper the first 8 points are used in common with all other other combinations (16, 24, 32, and 40) and the 16 and 24 combination numbers share 16 points in common. This process extends to 32 and 40 GCPs used which share 32 ground points in common.

Most of the statistical analysis utilized in this report was generated using the BMDP Statistical Package (Dixon et al. [3]). The descriptive statistics (mean, standard deviation, standard error of mean) and analysis of variance were run using program BMDP7D. The analysis of variance model was tested for equality of variances using Levene's test and if the Levene's test results were statistically significant at the 5% level, then the Brown-Forsythe procedure was used for the analysis of variance computations (Dixon et al. [3]).

Results and Discussion

Tables 1, 4, and 7 present the results of the analysis of a whole, half and quarter of a Landsat MSS scene with randomly distributed GCPs and the evaluation of the rectification accuracy using GRPs analyzed by the procedure of Graham and Luebbe [6]. Dow [5] pointed out that 24 GCPs appears to be more than adequate to rectify a whole or partial scene of Landsat MSS data with randomly distributed GCPs. The RBIAS and CBIAS values, in conjunction with the RSD and CSD values, of the randomly distributed GCPs will be used as a baseline to evaluate the rectification accuracy of the regular (Tables 2, 5,

and 8) and clustered (Tables 3 6, and 9) GCP distribution cases. The random and regular GCP distribution experiments with a quadrant TM data are presented in Tables 10 and 11, while the clustered data is shown in Table 12.

The columns represent the same parameters in all of the tables. The "N" column gives the number of GCPs used to develop the mapping equation. The "R" column gives an indication of the type of spatial distribution that the GCPs exhibit across the Landsat scene. The "RMS" column is a measure of how well the GCPs utilized fit the mapping equation (measured in meters). The accuracy of the georegistration procedure is measured by the RBIAS, RSD, CBIAS, and CSD values (measured as fractions of a pixel). The bias and standard deviation values are computed from the GRPs. The row and column bias values were averaged as absolute numbers, so that the sign of the bias values was ignored between replicates and the magnitude of the bias number was exphasized. Some authors have used the root mean square error values in place of the bias computations as an independent measure of rectification accuracy (Welch and Usery [17]. The individual bias value within a replicate will be lower in magnitude than the root mean square error number because of the fact that positive and negative values cancel one another in the bias computation.

The significance row indicates whether the analysis of variance (ANOVA) is statistically significant at the 5 percent level. The values in the last row of each column represents the mean and 95 percent confidence interval about the mean. This row is presented as a general descriptive overview of the results, but should not be interpreted literally in those cases where the ANOVA results are statistically significant (indicated by *). The results presented represent the outcome of 40 replicates for each of the "N" equals 8 thorugh 40 (MSS) or 56 (TM).

In Table 1 through 9 the RMS column shows what appears to be a counter-intuitive result in that the RMS value goes up as the number of GCPs utilized increases from 8 to 40. The reason for this appears to be that as the number of GCPs increases, it is more likely to encounter outlier GCPs which distort the overall RMS value. The RBIAS and CBIAS values decrease in magnitude as the number of GCPs used increases from 8 to 40. In this case outliers do not distort the results because there are many more GRPs used to check the rectification accuracy than the GCPs employed to generate the mapping equation (GRPs = ground point file - GCPs). The RSD and CSD values are fairly constant in magnitude with increasing N values. This being the case it was decided to concentrate on the RBIAS and CBIAS values in order to decide what the optimum number of GCPs required to register a whole scene of Landsat MSS data was. The rationale for choosing the optimum number of GCPs required to rectify a full or partial scene of Landsat MSS data for a random spatial distribution of GCPs is described in Dow [5]. This work (Dow [5]) agreed with the results of Mikhail and Paderes [9] that 24 GCPs is more than adequate to rectify a whole scene of Landsat MSS data. Mikhail and Paderes [9] analyzed a parametric model, while Dow [5] utilized an empirical appraoch with a polynomial model. It can be seen in Table 2 that 24 GCPs is all that is necessary to rectify a whole scene of regularly distributed GCPs data (it was not possible to produce a regular distribution for the Louisiana scene because of the large amount of water in this frame), while the clustered distribution case has much larger variation between replicates which results in a non-significant between replicate effect in three out of four cases for the RBIAS and CBIAS results. For a whole scene of clustered data, four independent locii were chosen to cluster around throughout the frame. This gave lower R values than the half

scene (3 locii) or quarter scene (2 locii) cases, as can be seen by comparing the R values in Tables 3, 6, and 9.

There appears to be no consistent differences between the Louisiana and Kansas frames regarding the magnitude of the RBIAS, CBIAS, RSD, or CSD values, so that both data sets yield the same conclusions. In both data sets the RBIAS and RSD numbers were less than the CBIAS and CSD values, as can be seen by comparing Tables 1 through 9. Thus, registration accuracy is more accurate in the row direction than in the column direction. A similar result was reported by Colwell et al. [2], when evaluating the georegistration accuracy of a P-format Landsat MSS tape. One would expect this result from the variation in MSS sensor attitude between scans as the satellite moves along its track. However, the TM data (Tables 10, 11, and 12) does not exhibit a consistent difference between RBIAS and RSD numbers and CBIAS and CSD values. This can be attributed to the backward and forward scanning mode of the TM sensor.

In the Graham and Luebbe [6] method of assessing rectification accuracy, our results indicate excellent georegistration as evidenced by sub-pixel bias and standard deviation values for both Kansas and Louisiana. However, the TM data (Tables 10, 11, and 12) appears to have achieved better rectification accuracy than the MSS data has (Tables 7, 8, and 9). In addition, the RMS values of the TM data (less than 24 meters) is much better than the RMS numbers for the MSS data (greater than 69 meters). This suggests that GCPs can be picked with greater precision for TM data than it can with MSS data. The RMS value is a measure of how well the GCPs fit the mapping equation and is not a measure of rectification accuracy (Dow [5]).

Tables 1, 4, and 7 show the results obtained with a random distribution of GCPs, while Tables 2, 5, and 8 exhibit the rectification accuracy (as measured by the bias and standard deviation values) of a regular distribution of GCPs. It can be seen that the RBIAS, CBIAS, RSD, and CSD numbers are of comparable magnitude for the random and regular spatial distribution of GCPs cases, whether one is dealing with a whole or partial frame of Landsat MSS data. Tables 10 and 11 show that similar results are obtained with the TM data for a quarter of a Landsat frame. This finding is at odds with the results reported by Paderes et al. [11] which found better rectification accuracy with a regular distribution of GCPs than with a random distribution of GCPs. Part of the reason for this difference between the results of the present study and that of Paderes et al. [11] is that our study used the distribution of actual GCPs with a maximum "R" value of 1.56, while the investigation of Paderes et al. [11] employed simulated data where the "R" value would be 2.15 (maximum spacing case).

An examination of the clustered spatial distribution of GCPs (Tables 3, 6, and 9) shows much poorer rectification accuracy (higher bias and standard deviation values) for both a whole and a partial frame of Landsat MSS data. A similar result is found with the clustered case for TM data (Table 12). In many undeveloped regions of the world it will only be possible to choose GCPs (around regional centers of anthropogenic activities or visible regions of natural features) in a clustered fashion. These results should be borne in mind when choosing the number and spatial distribution of GCPs required to georegister a whole or partial frame of Landsat MSS or TM data.

Conclusions

For the regular and random distribution of GCPs it appears that 24 GCPs is more than adequate to rectify a whole or portion of a Landsat MSS frame. Analysis of a quadrant of TM data supports this conclusion. TM data can be rectified with greater accuracy than MSS data, especially in the column direction. The RBIAS and RSD numbers are less for Landsat MSS data than are the CBIAS and CSD values, while they are all roughly equal in Landsat TM data. A clustered distribution of GCPs gives much poorer rectification accuracy than does the random or regular spatial distribution of GCP cases. A clustered distribution of GCPs, though less costly to implement, should be avoided where possible, when good scene-to-map registration accuracy is desired. A comparison of the Louisiana and Kansas Landsat MSS frame results suggests that these conclusions are not data set specific.

TABLE 1

Louisiana - Whole Scene: Random Distribution of Ground Control Points

N	R	RMS	RBIAS	RSD	CBIAS	CSD
8	0.77	94.58	0.38	0.06	0.82	0.14
16	0.77	119.18	0.20	0.06	0.39	0.12
24	0.73	129.02	0.17	0.06	0.42	0.12
32	0.71	132.72	0.16	0.06	0.36	0.12
40	0.71	133.95	0.14	0.06	0.37	0.12
Signif:	*	*	*	*	*	*
All	0.74±0.02	121.89±3.94	0.21±0.03	0.06	0.47±0.06	0.12±0.002

Kansas - Whole Scene: Random Distribution of Ground Control Points

N	R	RMS	RBIAS	RSD	CBIAS	CSD
8	0.86	112.60	0.27	0.07	0.70	0.16
16	0.85	140.80	0.21	0.06	0.44	0.14
24	0.83	144.88	0.17	0.06	0.39	0.14
32	0.82	148.72	0.16	0.06	0.30	0.14
40	0.83	146.30	0.15	0.06	0.27	0.14
Signif:	N.S.	*	*	*	*	*
All	0.84±0.02	138.66±5.59	0.19±0.02	0.06±0.002	0.42±0.06	0.14±0.002

N.S.: ANOVA not significant at the 5% level

* : ANOVA significant at 5% level

Mean ± 95% Confidence Interval

No. Replicates: 40

TABLE 2

Kansas Whole Scene: Regular Distribution of Ground Control Points

N	R	RMS	RBIAS	RSD	CBIAS	CSD
8	1.51	117.90	0.24	0.06	0.64	0.14
16	1.38	147.10	0.23	0.06	0.40	0.14
24	1.35	151.25	0.21	0.06	0.34	0.13
32	1.34	152.02	0.18	0.06	0.34	0.13
40	1.33	154.25	0.15	0.06	0.30	0.14
Signif:	*	*	N.S.	N.S.	*	*
All	1.38±0.014	144.50±5.32	0.20±0.03	0.06±0.001	0.41±0.05	0.14±0.001

N.S.: ANOVA not significant at 5% level

* : ANOVA significant at 5% level

Means ± 95% Confidence Interval

No. Replicates: 40

TABLE 3

Louisiana Whole Scene: Clustered Distibution of Ground Control Points

N	R	RMS	RBIAS	RSD	CBIAS	CSD
8	0.14	86.48	1.85	0.14	3.18	0.25
16	0.22	95.40	1.15	0.10	2.38	0.22
24	0.27	104.25	0.76	0.08	1.98	0.20
32	0.30	106.22	0.58	0.08	1.89	0.20
40	0.33	108.80	0.56	0.08	1.75	0.19
Signif:	N.S.	N.S.	N.S.	N.S.	N.S.	N.S.
All	0.25±0.01	100.23±6.71	0.98±0.41	0.10±0.02	2.23±0.57	0.21±0.02

Kansas Whole Scene: Clustered Distribution of Ground Control Points

N	R	RMS	RBIAS	RSD	CBIAS	CSD
8	0.11	85.20	0.61	0.12	1.15	0.20
16	0.17	95.08	0.53	0.11	1.05	0.19
24	0.20	100.45	0.46	0.09	1.00	0.19
32	0.23	105.92	0.38	0.08	0.97	0.19
40	0.25	106.28	0.30	0.08	0.94	0.19
Signif:	*	*	*	*	N.S.	N.S.
All	0.19±0.01	98.58±3.78	0.46±0.06	0.10±0.01	1.02±0.08	0.19±0.01

N.S. : ANOVA not significant at 5% level

* : ANOVA significant at 5% level

Mean ± 95% Confidence Interval

No. Replicates: 40

TABLE 4

Louisiana - Half Scene: Random Distribution of Ground Control Points

N	R	RMS	RBIAS	RSD	CBIAS	CSD
8	0.75	84.18	0.19	0.06	0.54	0.14
16	0.70	100.55	0.14	0.05	0.34	0.13
24	0.67	108.65	0.12	0.05	0.31	0.12
32	0.66	110.45	0.10	0.05	0.28	0.12
40	0.66	111.50	0.10	0.05	0.27	0.13
Signif.	N.S.	*	*	*	*	*
All:	0.69±0.02	103.07±2.64	0.13±0.02	0.05±0.001	0.35±0.04	0.13±0.002

Kansas - Half Scene: Random Distribution of Ground Control Points

N	R	RMS	RBIAS	RSD	CBIAS	CSD
8	0.90	111.55	0.43	0.08	0.70	0.16
16	0.81	133.88	0.26	0.08	0.58	0.16
24	0.79	140.38	0.22	0.08	0.44	0.16
32	0.80	143.75	0.19	0.08	0.38	0.16
40	0.79	146.30	0.16	0.08	0.33	0.16
Signif.	*	*	*	N.S.	*	N.S.
All	0.82±0.02	135.17±6.59	0.25±0.05	0.08±0.002	0.48±0.05	0.16±0.002

N.S.: ANOVA not significant at 5% level

*: Significant at 5% level in ANOVA

Mean ± 95% confidence interval

No. Replicates: 40

TABLE 5

Louisiana Half Scene: Regular Distribution of Ground Control Points

N	R	RMS	RBIAS	RSD	CBIAS	CSD
8	1.54	99.20	0.26	0.06	0.46	0.15
16	1.37	111.85	0.21	0.06	0.32	0.15
24	1.34	118.25	0.16	0.06	0.32	0.14
32	1.32	119.55	0.12	0.06	0.29	0.15
40	1.32	119.80	0.10	0.06	0.25	0.15
Signif.	*	*	*	N.S.	*	*
All:	1.38±0.02	113.73±1.95	0.17±0.02	0.06±0.001	0.33±0.03	0.15±0.001

Kansas Half Scene: Regular Distribution of Ground Control Points

N	R	RMS	RBIAS	RSD	CBIAS	CSD
8	1.56	98.08	0.30	0.09	0.64	0.19
16	1.36	124.98	0.28	0.09	0.52	0.18
24	1.34	134.90	0.21	0.09	0.42	0.18
32	1.34	140.48	0.20	0.10	0.45	0.19
40	1.33	146.20	0.23	0.09	0.35	0.19
Signif.	*	*	N.S.	N.S.	*	*
All	1.39±0.02	128.92±5.98	0.24±0.03	0.09±0.001	0.48±0.06	0.19±0.002

N.S.: ANOVA not significant at 5% level

*: ANOVA significant at 5% level

Means ± 95% confidence interval

No. Replicates: 40

TABLE 6

Louisiana Half Scene: Clustered Distribution of Ground Control Points

N	R	RMS	RBIAS	RSD	CBIAS	CSD
8	0.16	54.78	0.38	0.06	1.18	0.18
16	0.25	72.68	0.33	0.06	1.05	0.17
24	0.30	77.22	0.28	0.06	1.05	0.16
32	0.33	83.10	0.27	0.05	1.05	0.16
40	0.37	88.32	0.27	0.05	0.98	0.16
Signif.	*	*	N.S.	*	N.S.	*
All:	0.28±0.01	75.22±2.27	0.31±0.04	0.06±0.002	1.06±0.10	0.16±0.004

Kansas Half Scene: Clustered Distribution of Ground Control Points

N	R	RMS	RBIAS	RSD	CBIAS	CSD
8	0.20	97.95	1.19	0.13	1.56	0.23
16	0.30	119.15	0.60	0.10	1.10	0.18
24	0.36	118.12	0.46	0.09	1.07	0.18
32	0.41	121.18	0.42	0.09	0.96	0.18
40	0.45	125.98	0.46	0.09	0.84	0.17
Signif.	*	N.S.	*	*	*	*
All:	0.34±0.01	116.48±8.42	0.62±0.14	0.10±0.008	1.11±0.15	0.19±0.010

N.S. : ANOVA not significant at the 5% level

* : ANOVA significant at the 5% level

Mean ± 95% confidence interval

No. Replicates: 40

TABLE 7

Louisiana Quarter Scene - Area B: Random Distribution of Ground Control Points

N	R	RMS	RBIAS	RSD	CBIAS	CSD
8	1.06	60.68	0.26	0.06	0.36	0.08
16	1.07	71.70	0.17	0.06	0.22	0.07
24	1.05	75.35	0.10	0.06	0.21	0.07
32	1.00	76.00	0.12	0.06	0.21	0.07
40	1.00	76.58	0.11	0.06	0.17	0.07
Signif.	N.S.	*	*	*	*	*
All:	1.03±0.02	72.06±2.01	0.15±0.02	0.06±0.001	0.24±0.02	0.07±0.002

Kansas Quarter Scene - Area B: Random Distribution of Ground Control Points

N	R	RMS	RBIAS	RSD	CBIAS	CSD
8	1.20	53.78	0.26	0.08	0.27	0.13
16	1.15	78.48	0.15	0.07	0.25	0.13
24	1.14	85.38	0.13	0.07	0.27	0.13
32	1.14	87.23	0.12	0.07	0.24	0.13
40	1.12	88.08	0.12	0.07	0.21	0.14
Signif.	N.S.	*	*	N.S.	N.S.	N.S.
All:	1.15±0.02	78.58±5.21	0.16±0.02	0.07±0.002	0.25±0.03	0.13±0.002

N.S.: ANOVA not significant at 5% level

*: ANOVA Significant at 5% Level

Mean ± 95% Confidence Interval

No. Replicates: 40

TABLE 8

Louisiana Quarter Scene: Regular Distribution of Ground Control Points

N	R	RMS	RBIAS	RSD	CBIAS	CSD
8	1.49	81.35	0.18	0.06	0.43	0.17
16	1.38	102.50	0.14	0.06	0.33	0.16
24	1.34	106.25	0.11	0.06	0.25	0.16
32	1.32	105.55	0.09	0.07	0.25	0.17
40	1.32	105.88	0.08	0.07	0.23	0.17
Signif:	*	*	*	*	*	*
All:	1.37±0.01	100.30±2.23	0.12±0.02	0.06±0.002	0.30±0.03	0.17±0.002

Kansas Quarter Scene: Regular Distribution of Ground Control Points

N	R	RMS	RBIAS	RSD	CBIAS	CSD
8	1.56	57.70	0.19	0.07	0.28	0.13
16	1.39	68.25	0.14	0.07	0.22	0.13
24	1.35	70.68	0.14	0.07	0.16	0.13
32	1.34	73.40	0.12	0.07	0.16	0.14
40	1.34	78.10	0.11	0.07	0.17	0.15
Signif:	*	*	*	*	N.S.	*
All:	1.40±0.02	69.62±2.43	0.14±0.02	0.07±0.001	0.20±0.03	0.14±0.002

N.S. : ANOVA not significant at 5% level

* : ANOVA significant at 5% level

Mean ± 95% Confidence Interval

No. Replicates: 40

TABLE 9

Louisiana Quarter Scene: Clustered Distribution of Ground Control Points

N	R	RMS	RBIAS	RSD	CBIAS	CSD
8	0.24	48.92	0.94	0.13	1.11	0.16
16	0.33	58.65	0.58	0.09	0.59	0.10
24	0.40	63.70	0.41	0.08	0.52	0.08
32	0.45	67.35	0.32	0.07	0.46	0.08
40	0.50	68.52	0.34	0.07	0.48	0.08
Signif:	*	*	*	*	*	*
All:	0.38±0.02	61.43±1.62	0.52±0.09	0.09±0.006	0.63±0.09	0.10±0.010

Kansas Quarter Scene: Clustered Distribution of Ground Control Points

N	R	RMS	RBIAS	RSD	CBIAS	CSD
8	0.30	68.15	1.10	0.16	1.72	0.31
16	0.42	90.32	0.39	0.09	0.83	0.18
24	0.49	94.15	0.35	0.08	0.52	0.15
32	0.57	95.32	0.31	0.08	0.55	0.16
40	0.63	92.82	0.28	0.08	0.48	0.15
Signif:	*	N.S.	*	*	N.S.	*
All:	0.48±0.02	88.16±7.79	0.49±0.11	0.10±0.008	0.82±0.31	0.19±0.027

N.S. : ANOVA not significant at 5% level

* : ANOVA significant at 5% level

Mean ± 95% Confidence Interval

No. Replicates: 40

TABLE 10

Louisiana Quarter TM Scene: Random Distribution of Ground Control Points

N	R	RMS	RBIAS	RSD	CBIAS	CSD
8	0.86	19.38	0.24	0.04	0.21	0.04
16	0.82	22.80	0.14	0.04	0.13	0.03
24	0.80	24.00	0.10	0.03	0.11	0.03
32	0.78	23.90	0.09	0.04	0.09	0.03
40	0.78	23.75	0.09	0.04	0.09	0.03
48	0.77	24.05	0.08	0.04	0.08	0.03
56	0.76	23.85	0.07	0.04	0.08	0.03
Signif.	*	*	*	*	*	*
All:	0.79±0.01	23.10±0.41	0.11±0.01	0.04±0.001	0.11±0.01	0.03±0.001

N.S.: ANOVA not significant at 5% level

*: ANOVA significant at 5% level

Mean ± 95% Confidence Interval

No. Replicates: 40

TABLE 11

Louisiana Quarter TM Scene: Regular Distribution of Ground Control Points

N	R	RMS	RBIAS	RSD	CBIAS	CSD
8	1.54	18.32	0.15	0.04	0.18	0.03
16	1.39	21.85	0.10	0.03	0.11	0.03
24	1.36	22.30	0.09	0.03	0.10	0.03
32	1.34	22.73	0.07	0.04	0.08	0.03
40	1.32	23.25	0.07	0.04	0.09	0.03
48	1.32	23.10	0.07	0.04	0.08	0.03
56	1.32	23.23	0.07	0.04	0.08	0.03
Signif.	*	*	*	*	*	N.S.
All	1.37±0.01	22.11±0.42	0.09±0.01	0.04±0.000	0.10±0.01	0.03±0.000

N.S.: ANOVA not significant at 5% level

*: ANOVA significant at 5% level

Mean ± 95% Confidence Interval

No. Replicates: 40

TABLE 12

Louisiana Quarter TM Scene: Clustered Distribution of Ground Control Points

N	R	RMS	RBIAS	RSD	CBIAS	CSD
8	0.22	19.28	1.07	0.09	1.23	0.09
16	0.30	21.78	0.79	0.08	0.78	0.06
24	0.36	22.78	0.39	0.05	0.38	0.04
32	0.42	22.55	0.32	0.04	0.27	0.04
40	0.46	22.90	0.33	0.04	0.21	0.04
48	0.51	23.30	0.29	0.04	0.21	0.04
56	0.54	23.38	0.24	0.04	0.20	0.04
Signif.	*	*	*	*	*	*
All	0.40±0.01	22.28±0.49	0.49±0.08	0.06±0.004	0.47±0.10	0.04±0.004

N.S.: ANOVA not significant at 5% level

*: ANOVA significant at 5% level

Mean ± 95% Confidence Interval

No. Replicates: 40

References

[1] Clark, P.J., and Evans, F.C. 1954, Distance to nearest neighbor as a measure of spatial relationships in populations, Ecology 35: 445-453.

[2] Colwell, J., David, G. and Thompson, F. 1980, Detection and measurement of changes in the production and quality of renewable resources, Environ. Research Inst. Michigan, Report No. 145300-4-F: 73-85.

[3] Dixon, W.J., Brown, M.B., Engelman, L., Frane, J.W., Hill, M.A., Jennrich, R.I., and Toporek, J.D. 1983, BMDP Statistical Software, Univ. of California Press, Berkley, California, 733 p.

[4] Dow. D.D. 1983, Progress in the scene-to-map registration investigation, In Proceedings of the NASA Symposium on Mathematical Pattern Recognition and Image Analysis (L.F. Guseman, Jr., Ed.) NASA Johnson Space Flight Center, Houston, TX, pp. 485-506.

[5] Dow, D.D. 1984, The Influence of the number of ground control points on the scene-to-map registration accuracy, In Proceedings of the Second Annual NASA Symposium on Mathematical Pattern Recognition and Image Analysis (L. F. Guseman, Jr., Ed.) NASA Johnson Space Flight Center, Houston, TX pp. 401-426.

[6] Graham, M.H. and Luebbe, R. 1981, An evaluation of MSS P-format data registration, NSTL/ERL-197; AgRISTARS Report No. DC-Y1-04069: 51 p.

[7] Graham, M.H., B.G. Junkin, M.T. Kalcic, R.W. Pearson, and B.R. Seyfarth. 1985. ELAS: Earth Resources Laboratory Applications Software, NSTL/ERL Report No. 183.

[8] Horn, B.K.P. and Woodham, R.J. 1979, Landsat MSS coordinate transformations, In 1979 Machine Processing of Remotely Sensed Data Symposium (I.M. Pendam and D.B. Morrison, Ed.), I.E.E.E., New York, NY, pp. 59-68.

[9] Mikhail, E.M. and Paderes, F.C. 1983, Simulation aspects in the study of rectification of satellite scanner data, In Proceedings of the NASA Symposium on Mathematical Pattern Recognition and Image Analysis (L.F. Guseman, Jr., Ed.), NASA Johnson Space Flight Center, Houston, TX, pp. 413-483.

[10] Nelson, R. and Grebowsky, G. 1982, Evaluation of temporal registration of Landsat scenes, Int. J. Remote Sensing 3:45-50.

[11] Paderes, F.C., Jr., Mikhail, E.M., and Forstner, W. 1984, Rectification of single and multiple frames of satellite scanner imagery using points and edges as control, In Proceedings of the Second Annual NASA Symposium on Mathematical Pattern Recognition and Image Analysis (L.F. Guseman, Jr., Ed.), NASA Johnson Space Flight Center, Houston, TX, pp. 309-400.

[12] Sawada, N.,Kidode, M., Shinoda, H., Asada, H., Iwanaga, M., Watanabe, S., Mori, K-I, and Akiyama, M. 1981, An analytic correction method for satellite MSS geometric distortions, Photogramm. Eng. Remote Sens. 47: 1195-1203.

[13] Steiner, D. and Kirby, M.E. 1977, Geometrical referencing of Landsat images by affine transformation and overlaying of map data, Photogrammetria 33:41-75.

[14] U.S. Geologicl Survey. 1979, EDIPS image accuracy tests, U.S. Department of Interior, EROS Data Center, Sioux Falls, S.D., Landsat Data User Note 9:4.

[15] Van Wie, P. and Stein, M. 1977, A Landsat digital image rectification system, IEEE Trans. Geosci. Electron, GE-15: 130-137.

[16] Walker, R.E., Zobrist, A.L., Bryant, N.A., Gokhman, B., Friedman, S.Z., and Logan, T.L. 1984. An analysis of Landsat-4 Thematic Mapper geometric properties, IEEE Trans. Geosci. Remote Sensing GE 22 (3): 288-293.

[17] Welch, R. and Usery, E.L. 1984, Cartographic accuracy of Landsat-4 MSS and TM image data, IEEE Trans. Geosci. Remote Sensing GE-22:281-288.

[18] Wong, K.W. 1975, Geometric and cartographic accuracy of ERTS-1 imagery, Photogramm. Eng. Remote Sens. 41: 621-635.

Smooth Multidimensional Interpolation

W. Tobler and S. Kennedy

Department of Geography

University of California

Santa Barbara, CA 93106

Partial support was provided by the Fundamental Remote Sensing Research Program in Mathematical Pattern Recognition and Image Analysis of the National Aeronautics and Space Administration.

Abstract

A simple method of assigning values to missing data in a geographic context is to use an average of adjacent observations. The value thus obtained is a linear combination of neighboring values with appropriately chosen weights. The same general method can be used when the observations consist of regular pixels, of irregularly arranged resels, or scattered point observations. Smooth assignments are made by this method; iterations are required when adjacent values are missing.

Our primary interest is in geographical problems and the discussion focuses on examples in which the interpolation estimates are to be made in two dimensions. We believe that the simplest and most sensible method of geographic interpolation consists of the assignment of an average value to the location or locations for which data are required. The set over which the average is taken is obviously important, and, as weighted averages are almost invariably used, the choice of weights is also critical. For spatial variables the relevant set usually consists of values in the vicinity of the locations for which the estimates are desired. Observe here that we implicitly assume that the variable of interest is numerical, and not categorical, so that averages have meaning. Suggestions as to how to proceed when this is not the case may be found in Guptill (1975), Switzer (1975), and Tobler (1979a). We also restrict our attention to arithmetical averages, ignoring geometrical and harmonic averages and medians which may be appropriate in some cases. It should be recognized that no interpolation scheme can overcome the problem of insufficient resolution in the original observations.

We consciously avoid explicit distance weighted averages as being computationally too cumbersome, but recognize that they are common in the literature. A rather thorough treatment of this subject is that of Gandin (1965), which includes coverage of covariance and varigram estimation approaches more recently popularized as Kriging, optimal interpolation, objective analysis, collocation, and regionalized variable techniques. Additional literature is referenced in Akima (1975), Barnhill and Nielson (1984), Besag (1974), Brady (1982), Brodlie (1980), Duchon (1975), Franke (1982), Grimson (1982), Harder (1972), Hardy (1971), Hessing (1972), Journel (1978), Kraus (1972), Lawson (1978), Matheron (1971), Moritz (1970), Ripley

(1981), Schumaker (1976), Swain (1976), Tobler (1979c), and Wahba (1980), to give only a short selection. It is here assumed that the observations are without error so that filtering of the values is not included; see the foregoing references if this is of interest.

We present three simple cases in which spatial averages can be used for interpolation. The first case involves pixels, or data on a regular mesh; in the second and third cases the known data are irregularly arranged on the plane either as resels or as point locations.

Consider first data given as square pixels (picture elements) with the value for one interior pixel missing (Figure One). Then (using an obvious row-column notation) the value at the missing i,j location is estimated as an average from its neighbors by

$$\hat{Z}_{ij} = \frac{1}{4} (Z_{i+1j} + Z_{i-1j} + Z_{ij-1} + Z_{ij+1}).$$

This works equally well when several interior values are missing, as shown in Figure Two, by an iteration equivalent to solving Laplace's equation by finite difference methods (Birkhoff 1972). How the missing values are initialized for the iterations is not critical but a good guess saves computational effort. In order to terminate the iterations one invokes the usual stopping rules. This of course is just the classical Dirichlet problem in two dimensions and the interpolated value has the harmonic property (Courant and Hilbert 1937) by the construction method. Now it is well known (Kantorovitch and Krylov 1958) that Laplace's equation arises from the least

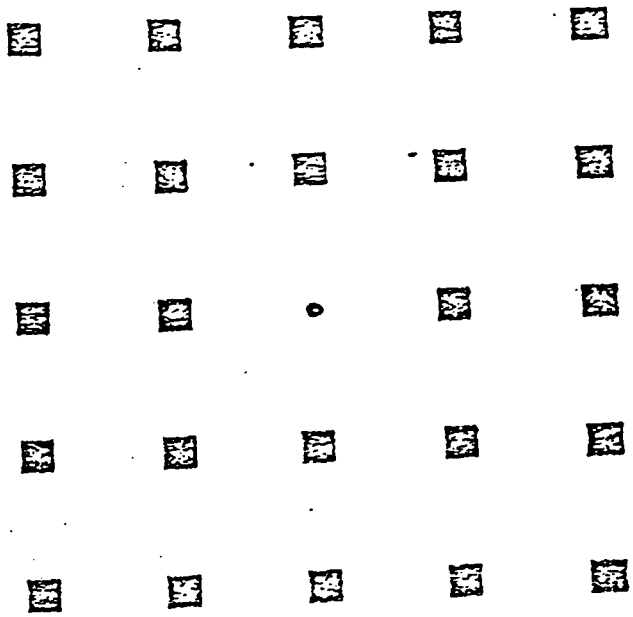


Figure 1. Small boxes denote known values. Small dot indicates location for which an estimate is desired.

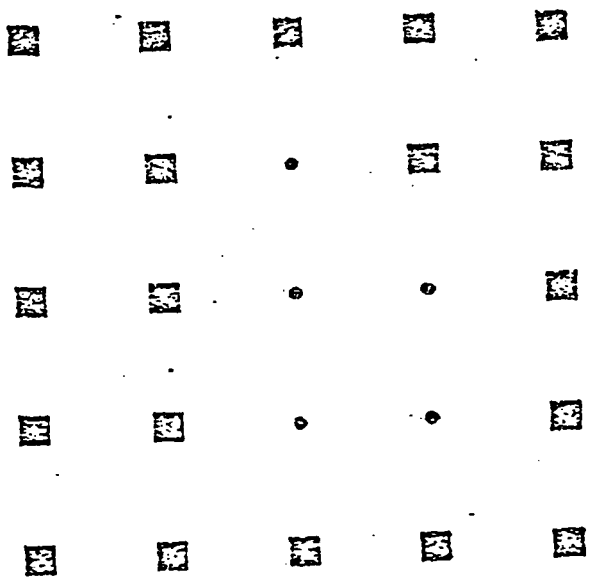


Figure 2. Small boxes denote known values. Small dots indicate locations for which estimates are desired.

squares problem:

$$\min: \int_R \int \frac{\partial Z^2}{\partial X} + \frac{\partial Z^2}{\partial Y} \, dxdy$$

with, in the present instance, Dirichlet boundary conditions. Thus the interpolation is spatially smooth, the squared variation of the derivatives, which is minimized, providing a measure of roughness.

The foregoing simple solution has several disadvantages. One of these is that we have provided only a point estimate, without any statement of the standard error of the estimate. An obvious way around this is to sample from a distribution having the mean of the neighbors as its expectation with a variance also estimated from these neighbors. A second shortcoming of the harmonic interpolation is that the estimated value can never rise above, nor fall below, its neighbors in magnitude. This restriction can be overcome by enlarging the neighborhood and by requiring that the partial derivatives of the estimate be smooth, that is, by solving the biharmonic equation. In finite difference form this leads to

$$\begin{aligned}\hat{Z}_{ij} = \frac{1}{20} [& 8(Z_{i+1j} + Z_{i-1j} + Z_{ij+1} + Z_{ij-1}) \\ & -2(Z_{i-1j-1} + Z_{i+1j-1} + Z_{i-1j+1} + Z_{i-1j-1}) \\ & -(Z_{i-2j} + Z_{ij-2} + Z_{ij+2} + Z_{i+2j})],\end{aligned}$$

and iterative procedures are again used when several adjacent values are missing.

Now suppose that the data are given in the form of irregularly arranged resels (resolution elements); census tracts or counties in the United States, with one or more values missing. A generalization of the above results, using first order neighbors, can be written as

$$\hat{Z}_i = \sum_{j=1}^{n} L_{ij} Z_j$$

where n is the number of neighbors of region i and L_{ij} are normalized neighbor weights. First order neighbors are areas having direct contact along borders of non-zero length, second order neighbors are the first order neighbors of the initial neighbors, and so on. As an example Figure Three shows first and second order neighbors for Kansas, with the numerical values given in Table One. For the population density of Kansas, using only first order neighbors, and with normalized boundary lengths as weights we obtain 36.05 persons per square kilometer, whereas the observed value is 27.50. Taking each individual state in turn yields an average success rate of 72%, which may be considered impressive in light of the simplicity of the technique (Figure Four). The method has been extended to the case in which several interior values are estimated (Kennedy and Tobler 1983). Table II illustrates the comparable biharmonic density estimate for Kansas. We believe this method of adjacency weighting to be far superior to the use of arbitrary points ("centroids") to represent geographic areas.

As a final example consider the problem of interpolating a continuous scalar field from irregularly arranged point observations in two dimensions. As the first step, to reduce extrapolation, we rotate to principle axes. Thus observations which, for example, fit within an oblique rectangle are readily accomodated. We next pass one coordinate line through each observation (Figure Five). The result is an irregular orthogonal mesh, with observations at N of the nodes and up to N*N-N nodes at which we need to make an estimate. The obvious procedure is to let the mesh define the adjacencies and then to use neighbor averaging as before. In this example we solve Laplace's equation by using

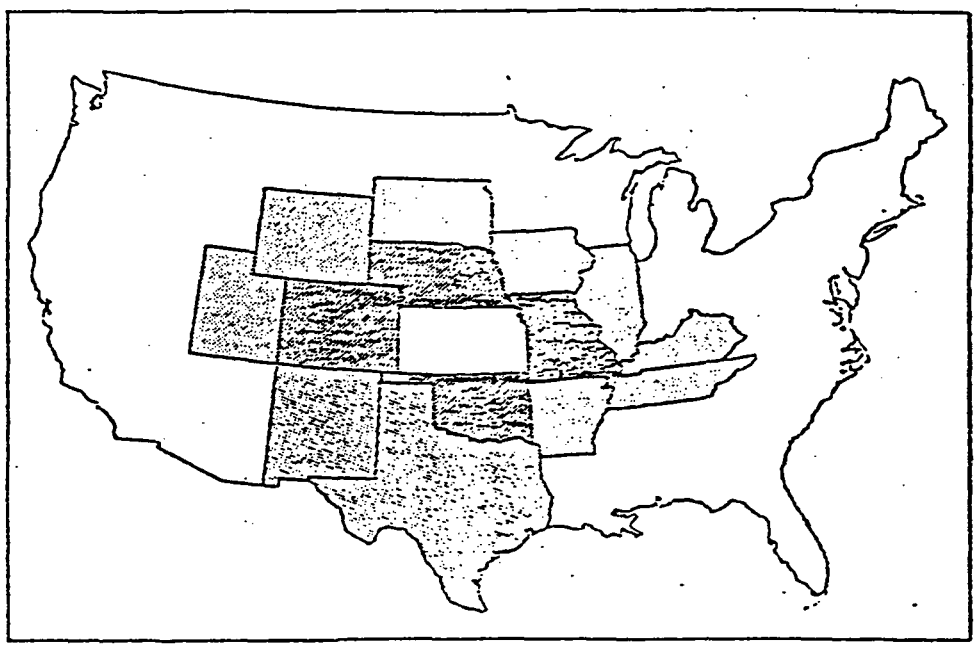


Figure 3. First and second order neighbors of the state of Kansas.

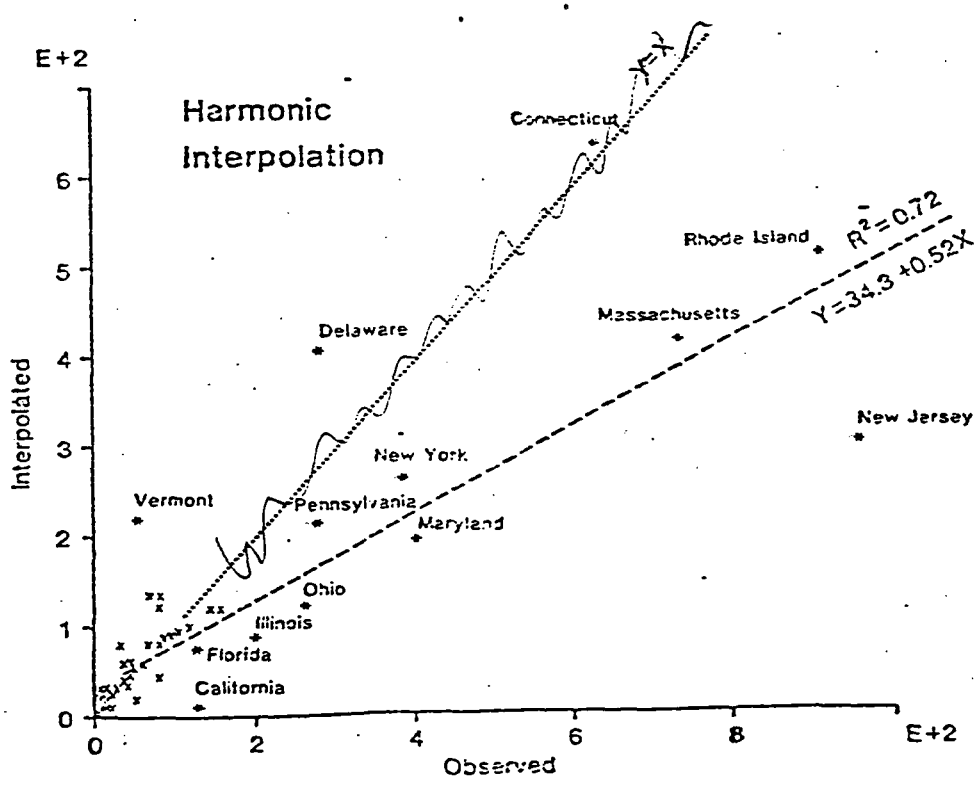


Figure 4. Scatter diagram comparing actual and estimated population densities for States. Data from Kennedy and Tobler (1983), Table II.

Table I

FIRST ORDER DENSITY ESTIMATE FOR KANSAS

Length of Border of Kansas with neighboring states, and their population densities

Neighbor	km border	Density
Colorado	338	21.3
Oklahoma	667	37.2
Missouri	433	67.8
Nebraska	572	19.4

Sum of border lengths = 2010

Sum of border * population densities = 72466

72466/2010 = 36.05

which yields the density estimate for Kansas

Table II

SECOND ORDER DENSITY ESTIMATE FOR KANSAS

Length of Border of	km border	Density
Nebraska		
with		
South Dakota	641	8.8
Wyoming	222	3.4
Iowa	192	50.5
Colorado		
with		
Wyoming	419	3.4
Utah	444	12.9
New Mexico	542	8.4
Oklahoma		
with		
New Mexico	58	8.4
Texas	1534	42.7
Arkansas	319	37.0
Missouri		
with		
Arkansas	548	37.0
Tennessee	156	94.9
Kentucky	111	81.2
Illinois	613	199.4
Iowa	378	50.5
	6177	

Density estimate from second order neighbors = 291003/6177 = 47.11.

Density estimate for Kansas = density estimate from first order neighbors plus difference of first order estimate and second order estimate = 36.05 + (36.05-47.11) = 24.99 persons per square kilometer.

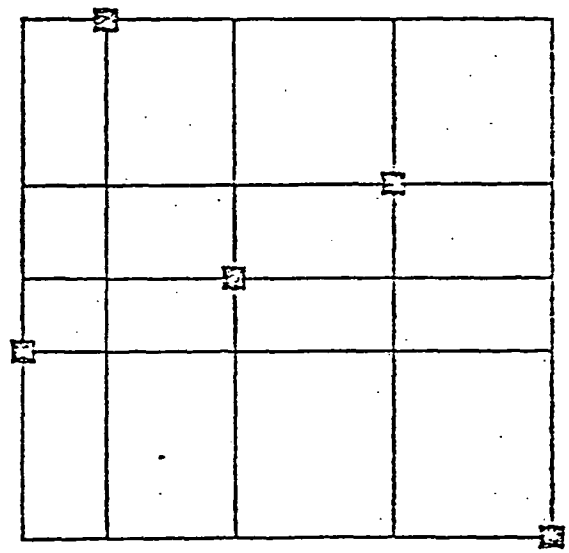


Figure 5. Small boxes denote known values. Remaining intersections of the mesh indicate locations for which estimates are desired.

$$\hat{Z}_{ij} = W_1 Z_{i+1j} + W_2 Z_{i-1j} + W_3 Z_{ij-1} + W_4 Z_{ij+1}$$

with weights chosen from simple geometric considerations. These weights are essentially normalized inverse distances but only to immediately adjacent locations on this mesh. The grid is orthogonal so that only 4(N-1) distances (instead of N(N-1)/2) are required and they can be computed in advance for the entire mesh. With more neighbors, different weights, and additional boundary conditions, the method is easily extendable to the biharmonic case to obtain an interpolation with smooth derivatives. An iteration is used since most of the mesh points do not have observations at the adjacent mesh positions. Points which are neighbors on the mesh may not be spatially nearest points, but the influence of all points is felt by each point, through the coupling via the mesh. The iterations start from an initial guess and end when an error tolerance is satisfied. Convergence accelerating techniques are available to speed the iterations (Graham 1983). The result is a set of smoothly varying values at the corners of the rectangles defining the mesh and the original observations are exactly satisfied. Interpolation within the rectangles is then easily effected using conventional bilinear or splining techniques. The method of course bears a resemblance to the "lattice tuning" described earlier by Tobler (1979b) except that the observational values are everywhere retained which was not the case in that procedure. An advantage of the rectangular mesh over a triangulation is that it can be used directly in other computations or for display purposes. Computational experience with several extensive sets of data has reinforced our belief in the efficacy of spatial averaging for interpolation. Any interpolation scheme of course requires hypotheses about the phenomena under investigation and cannot be applied uncritically.

The smooth interpolation-by-averaging techniques described here can all be extended rather easily to higher dimensional cases and to the interpolation of vector or tensor field components. An example application would be for non-parametric "rubber sheeting" in order to fit satellite images to conventional maps. It has also not escaped our notice that the methods may be reversed, in order to parse large data sets.

References

H. Akima, 1975, "A Method of Bivariate Interpolation and Smooth Surface fitting for Values given at Irregularly Distributed Points", Washington DC, Office of Telecommunications Report No. 75-70.

R. Barnhill, and G. Nielson, eds., 1983, "Surfaces", Rocky Mountain Journal of Mathematics, 14, 1:1-299.

J. Besag, 1974, "Spatial Interaction and the Statistical Analysis of Lattice Systems," J. Roy. Stat. Soc., B, 36, 192-236.

Birkhoff, G., 1972, The Numerical Solution of Elliptic Equations, Philadelphia, SIAM.

J. Brady and B. Horn, 1982, "Rotationally Symmetric Operators for Surface Interpolation", MIT Artificial Intelligence Lab Memo no. 654.

K. Brodlie, 1980, Mathematical Methods in computer Graphics and Design, London, Academic Press.

Courant, R., and Hilbert, D., 1937, Methoden der Mathematischen Physik, Berlin, Springer.

J. Duchon, 1975, "Functions-Spline du type plaque mince en dimension 2", Grenoble, University Technical Report No. 231.

R. Franke, 1982, "Scattered Data Interpolation: Tests of Some Methods", Mathematics of Computation, 38(157):181-199.

Gandin, L., 1963, Objective Analysis of Meteorological Fields, Leningrad, GIMIZ (US Department of Commerce translation, TT 65-50007).

Graham, N., 1983, A Combined Algorithm for Sample Design and Interpolation, Discussion Paper #5, Santa Barbara, Department of Geography, UCSB.

W. Grimson, 1982, "A Computational Theory of Visual Surface Interpolation", Phil. Trans. Royal Soc. of London, B298:395-427.

Guptill, S., 1975, Spatial Filtering of Nominal Data, Ph.D. thesis, Ann Arbor, University of Michigan.

R. Harder, and R. Desmaris, 1971, "Interpolation Using Surface Splines", J. Aircraft 9,2:189-191; 9,12:869-871.

R. Hardy, 1971, "Multiquadric Equations of Topography and Other Irregular Surfaces", J. Geophysical Research, 76:1905-1915.

R. Hessing, et al., 1972, "Automatic Contouring using Bicubic Functions", Geophysics, 37,4:669-674.

A. Journel, and C. Huijbregts, 1978, Mining Geostatics, New York, Academic Press.

Kantorovitch, L., and Krylov, V., 1958, Approximate Methods of Higher Analysis, The Hague, Noordhoff.

Kennedy, S., and Tobler, W., 1983, "Geographic Interpolation", Geographical Analysis, 15,2:151-156.

K. Kraus, and E. Mikhail, 1972, "Linear Least Squares Interpolation", Photogrammetric Engineering, 40:1016-1029.

C. Lawson, 1978, "Software for C^1 Surface interpolation", Mathematical Software III, New York, Academic Press.

G. Matheron, 1971, The Theory of Regionalized Variables and its Applications, Fontainbleau, Ecole Nationale.

H. Moritz, 1970, "Eine Allgemeine Theorie der Verarbeitung von Schwermessungen mach Kleinsten Quadraten", Heft Nr. 67A, Munich, Deutsche Geodaetische Kommission.

B. Ripley, 1981, Spatial Statistics, New York, J. Wiley.

L. Schumaker, 1976, "Fitting Surfaces to Scattered Data", Approximation Theory II, New York, Academic Press.

C. Swain, 1976, "A Fortran IV Program for Interpolating Irregularly Spaced Data using the Difference Equations for Minimum Curvature", Computers and Geosciences, 1:231-240.

Switzer, P., 1975, "Estimation of the Accuracy of Qualitative Maps", pp. 1-13 of J. Davis and M. McCullagh, Display and Analysis of Spatial Data, New York, J. Wiley.

Tobler, W., 1979a, "Cellular Geography", pp. 379-386 of S. Gale and G. Olsson, Philosophy in Geography, Dordrecht, Reidel.

Tobler, W., 1979b, "Lattice Tuning", Geographical Analysis, 11,1:36-44.

W. Tobler, 1979c, "Smooth Pycnophlyactic Interpolation for Geographical Regions", J. Am. Statistical Assn., 74(367):519-530.

G. Wahba and J. Wendelberger, 1980, "Some New Mathematical Methods for Variational Objective Analysis Using Splines and Cross Validation", Monthly Weather Review, 108(8):1122-1143.

An Optimal Frequency Domain Textural Edge Detection Filter

J. Keith Townsend, K. Sam Shanmugan and Victor S. Frost

Telecommunications and Information Sciences Laboratory
University of Kansas
Space Technology Center
Nichols Hall
2291 Irving Hill Road
Lawrence, Kansas 66045
(913) 864-4832

ABSTRACT

An optimal frequency domain textural edge detection filter is developed and its performance evaluated. For the given model and filter bandwidth, the filter maximizes the amount of output image energy placed within a specified resolution interval centered on the textural edge. Filter derivation is based on relating textural edge detection to tonal edge detection via the complex lowpass equivalent representation of narrowband bandpass signals and systems. The filter is specified in terms of the prolate spheriodal wave functions translated in frequency. Performance is evaluated using the asymptotic approximation version of the filter. This evaluation demonstrates satisfactory filter performance for ideal and non-ideal textures. In addition, the filter can be adjusted to detect textural edges in noisy images at the expense of edge resolution.

This work was supported by NASA under Contracts NAS-9-16664 and NAGW-381.

LIST OF FIGURES

Figure 1. Block diagram of the optimum textural edge detection filter for two textures.

Figure 2. Single sided transfer function of the optimum textural edge detection filter. The bandwidths of $H_1(\omega)$ and $H_2(\omega$ are narrow enough that response at ω_1 and ω_2 is zero.

Figure 3. (a) Input image consisting of two ideal textures.
(b) Magnitude of the optimum textural edge detector response (in the spatial domain).

Figure 4. Magnitude of the response of the textural edge detection filter due to an input image with four ideal textures and three textural edges. The normalized spatial frequencies of the four textures are $.04\pi$, $.06\pi$, $.08\pi$, and $.1\pi$.

Figure 5. (a) Spectrum of an arbitrary input image.
(b) Spectrum of optimum textural edge detection filter with bandwidth shown in terms of ω_n and Ω.

Figure 6. (a) Input image with both amplitude and frequency varying in proportion to a bandlimited Gaussian noise process (horizontal axis magnified two times around each textural edge).

(b) Magnitude of the optimum textural edge detector response due to (a).

I. INTRODUCTION AND OVERVIEW

Edge detection is an important first step in extracting information from an image. Many edge detection schemes have been employed to enhance the boundaries between regions of different average gray tone. These tonal edge detectors are inadequate when regions in an image are characterized by similar average gray tone, but different textural features.

A textural edge detection filter is presented in this paper which is optimal in the sense that, for the given model, a maximum amount of output image energy is placed within a given resolution interval width and a given filter bandwidth. The resolution interval is centered on the textural edge in the input image. The filter is derived in the frequency domain, and is easily implemented on a digital computer using Fast Fourier Transform (FFT) techniques.

The optimum textural edge detection filter is developed by treating the textural edge as a bandpass extension of a tonal edge. Hence, the optimum tonal edge detector derived by Shanmugan, Dickey and Green [1] (correspondence by Lunscher [2]), is related to the textural edge detection case via the complex lowpass equivalent representation of signals and systems. It should be pointed out that the development is carried out in one-dimension. However, symmetries required for extension to two-dimensions are retained.

Section II presents a brief review of the optimum tonal edge detector. The textural model used in the development of the optimum textural edge detector is then introduced in Section III. The mathematical form of the optimum textural edge detection filter and some one-dimensional examples are presented in Section IV. Concluding remarks are given in Section V.

II. REVIEW OF THE OPTIMUM TONAL EDGE DETECTOR

Ths purpose of this section is to briefly review the optimum tonal edge detector derived by Shanmugan, et al., [1]. For a given filter bandwidth, the optimum tonal edge detector places a maximum amount of output image energy within a given resolution interval length in the vicinity of tonal edges. The tonal edge detector is insensitive to textural edges where the average gray levels of the different textural regions are equal.

The derivation of the optimum tonal edge detector is based on representing the filter output (for a step edge input) in terms of prolate spheriodal wave functions (for the derivation, see [1], [2]). The exact one-dimensional form of the filter transfer function is given in Shanmugan, et al., [1] as

$$H_{STEP,E}(\omega) = \begin{cases} B_1 \omega \, \psi_1(c, \omega I/2\Omega) & |\omega| < \Omega \\ 0 & \text{elsewhere} \end{cases} \qquad (1)$$

where $c = \frac{\Omega I}{2}$ and ψ_1 is the first order prolate spheriodal wave function. (The subscript STEP,E in Equation (1) denotes the Exact form of the STEP edge detector). For any given values of spatial bandwidth, Ω, and resolution interval length, I, the transfer function in Equation (1) places the maximum amount of energy in I. The filter is difficult to implement in this form, because the values of ψ_1 cannot be easily calculated. Application of approximations by Slepian and Streifer [1], yield an asymptotic approxi-

mation of the filter which is in closed form, hence easy to implement. The resulting expression is

$$H_{STEP,E}(\omega) \tilde{=} H_{STEP}(\omega) = K_1 \omega^2 \exp(-\frac{c\omega^2}{2\Omega^2}) \qquad (2)$$

Combining the constants that appear in the argument of the exponent, and dropping the gain factor, K_1, yields

$$H_{STEP}(\omega) = \omega(\omega e^{-K\omega^2}) = \omega^2 e^{-K\omega^2} \qquad (3)$$

It should be noted that the parameters I and Ω can no longer be independently specified.

Choice of K sets the bandwidth of the filter, and also the resolution interval length. As K increases, resolution interval size increases, and filter bandwidth decreases. Note that even though the asymptotic approximation to the optimum transfer function is not strictly bandlimited, $H_{STEP}(\omega)$ is effectively zero for spatial frequencies above a certain value, depending on the choice of K. The asymptotic approximation will be used in the remainder of the development.

III. TEXTURAL MODEL

One inherent difficulty with textural processing is the fact that no single "best" model exists for characterizing texture in images. The model used here in the development of the optimum textural edge detector capitalizes on the relationship between texture and spatial frequency by representing each texture as a sinusoid of different spatial frequency (i.e., fine textures contain greater concentrations of energy at higher spatial frequencies than coarser textures do) [3], [4], [5], [6], [7], [8], [9].

In general, a class of one-dimensional images with n textures can be defined as

$$q(x) = A(x) \cos(\omega_i x + \Theta(x)) \qquad i = 1, 2, \cdots, n \quad (4)$$

where

$$A(x) = a(1 + \alpha(x)) \qquad |\alpha(x)| < 1 \quad (5a)$$

and

$$\Theta(x) = b \int_{-\infty}^{x} \beta(\lambda)\, d\lambda \quad (5b)$$

The functions $\alpha(x)$ and $\beta(x)$ are random processes, ω_1 represents the i^{th} texture, a and b are constants, and x is the spatial variable. Note that q(x) is allowed to be negative. This can be

viewed as subtracting off the mean level from an image, thus allowing negative brightness or gray level. In this model, $\alpha(x)$ represents average gray level, and $\beta(x)$ represents the variation of spatial frequency within a texture. In other words, the envelope of $q(x)$ can be thought of as the average gray level variation, while the underlying texture is represented by each different ω_i, where the random change of texture for a given ω_i is controlled by $\beta(x)$. Note that if time were the independent variable, $q(x)$ would be a double sideband plus large carrier modulated waveform, with simultaneous frequency modulation.

An ideal texture is represented in this model by a sinusoid with constant spatial frequency and constant amplitude. Hence, a transition between two ideal textures can be represented by a pure sinusoid at one spatial frequency followed by a pure sinusoid at another spatial frequency. For the ideal two texture case let

$A(x) = 1$

$\Theta(x) = 0$

$-\infty < x < \infty$ (infinite size)

Thus, an image with two ideal textures and a textural edge at $x = 0$ is represented mathematically as

$$f(x) = \cos(\omega_i x), \qquad -\infty < x < \infty \tag{6}$$

where

$i = 1$ for $x < 0$ and

$i = 2$ for $x \geq 0$.

The optimum textural edge detector is derived using the ideal, two texture image, f(x).

IV. OPTIMUM TEXTURAL EDGE DETECTOR RESULTS AND PERFORMANCE

This section presents the mathematical form of the optimum textural edge detection filter and discusses the performance of the filter for several different classes of input images. The derivation is only briefly sketched here, the details are given in Townsend [10].

For a two texture input image with one texture represented by a sinusoid with frequency ω_1, and the other texture represented by a sinusoid with frequency ω_2, the transfer function of the optimum tonal edge detector is given by

$$H_{OPT}(\omega) = H_1(\omega) + H_2(\omega) \tag{7}$$

where

$$H_1(\omega) = H_{STEP}(\omega - \omega_1) + H_{STEP}(\omega + \omega_1) \tag{8a}$$

$$H_2(\omega) = H_{STEP}(\omega - \omega_2) + H_{STEP}(\omega + \omega_2) \tag{8b}$$

and

$$H_{STEP}(\omega) = \omega^2 \, e^{-K\omega^2} \tag{3}$$

It is clear from Equations (7), (8), and (3), that the optimum textural edge detector is the sum of the responses of two bandpass "sub" filters, $H_1(\omega)$ and $H_2(\omega)$. Each "sub" filter is a trans-

lated-in-frequency version of the optimum tonal edge detector, $H_{STEP}(\omega)$, discussed in Section II. Note that $H_{STEP}(\omega)$ is translated to each of the two textural frequencies.

The optimum textural edge detector is derived by recognizing that the two-ideal-texture input image, f(x), given in Section III can be expressed as the sum of two truncated sinusoids, one at frequency ω_1, defined for $-\infty < x < 0$ and the other at frequency ω_2, defined for $0 \leq x < +\infty$. But each of these two truncated sinusoids are bandpass at frequencies ω_1 and ω_2 respectively. Each truncated sinusoid has a step function for its complex low-pass equivalent [11]. Because $H_{STEP}(\omega)$ is optimized for detecting step type edges, a bandpass version of $H_{STEP}(\omega)$ centered on frequency ω_1 is optimum for detecting the discontinuity (modulated step function), in the truncated sinusoid at frequency ω_1 [10]. Similarly, a bandpass version of $H_{STEP}(\omega)$ translated in frequency to ω_2 is optimum for detecting the discontinuity in the truncated sinusoid at frequency ω_2. The sum of the outputs of these two bandpass filters produces the optimized output. A block diagram of the filter structure for the two texture case is shown in Figure 1.

A qualitative discussion is presented here to gain insight into how the filter works. Figure 2 presents an example of the optimum textural edge detector in the frequency domain. Note from the figure that the response at ω_1 and ω_2 (the spatial frequencies representing the two ideal textures) is zero. Hence, $H_{OPT}(\omega)$ does not respond to any input which has spectral energy only at these

two frequencies. Therefore, the response to an input representing either pure texture (in steady state) is zero. The textural edge is characterized by a transition from one texture to the other. The Fourier transform of this boundary contains spectral energy at frequencies other than ω_1 and ω_2. In particular, there is energy in the passband portions of $H_{OPT}(\omega)$, therefore filter response near the textural edge is non-zero resulting in a large amount of output image energy in the vicinity of the textural edge.

The Fourier transform of the entire input image is given by

$$F(\omega) = F_1(\omega) + F_2(\omega) \tag{9}$$

where $F_1(\omega)$ amd $F_2(\omega)$ are the Fourier transforms of the truncated textures represented by sinusoids at ω_1 and ω_2 respectively. Multiplication of $F(\omega)$ with $H_{OPT}(\omega)$ yields the transform of the output, $G(\omega)$, i.e.,

$$G(\omega) = F(\omega)\ H_{OPT}(\omega) \tag{10}$$

but this is equivalent to

$$\begin{aligned} G(\omega) &= [F_1(\omega) + F_2(\omega)]\ [H_1(\omega) + H_2(\omega)] \\ &= F_1(\omega)\ H_1(\omega) + F_1(\omega)\ H_2(\omega) \\ &\quad + F_2(\omega)\ H_1(\omega) + F_2(\omega)\ H_2(\omega) \end{aligned} \tag{11}$$

but

$$F_1(\omega)\ H_2(\omega) \tilde{=} 0 \tag{12}$$

and

$$F_2(\omega)\ H_1(\omega) \tilde{=} 0 \tag{13}$$

Substitution of Equations (12) and (13) into Equation (11) yields

$$\begin{aligned} G(\omega) &= F_1(\omega)\ H_1(\omega) + F_2(\omega)\ H_2(\omega) \\ &= G_1(\omega) + G_2(\omega) \end{aligned} \tag{14}$$

Hence,

$$g(x) = g_1(x) + g_2(x) \tag{15}$$

Equations (12) and (13) are true because of the spectral separation between the two sets of bandpass inputs and systems. In non-ideal texture cases, there can be considerable spectral overlap between the Fourier transforms of the textures. The spectral overlap can cause non-zero response of a system, $H_1(\omega)$, for example, to a texture not centered at ω_1, $F_2(\omega)$ for example. This could also occur if the bandpass bandwidth of $H_1(\omega)$ is wide enough to pass a significant amount of energy due to $F_2(\omega)$.

Choosing the exponential parameter, K, such that the bandpass bandwidths of $H_1(\omega)$ and $H_2(\omega)$ are wider than the spatial frequency separation between ω_1 and ω_2 results in non-zero response to the two textures. There is improved resolution at the expense of an increase in the "background" level in the output image, thus decreasing edge visibility. The "background" refers to the out-of-resolution-interval gray level. Edge visibility describes the difference in gray level between the in-resolution-interval and out-of-resolution-interval (background) portions of the output image. The spatial frequency separation of the textures affects the performance of the filter, i.e., the greater the separation, the better the performance.

It was shown in Shanmugan, et al., [1] that the optimum tonal edge detector could be used to enhance tonal edges in images corrupted by additive white Gaussian noise. The same theory applies to the optimum textural edge detector. The exponential parameter, K, can be chosen to decrease the bandwidth of the "sub" filters to decrease the effects of the noise. The price paid for this is an increase in the resolution interval length [10]. The benefits of increased edge visibility may more than offset the decrease in resolution.

Figure 3 shows the result of implementing the filter on a digital computer. Displayed are the input and output images (one-dimensional) of the optimum textural edge detection filter for an input with two ideal textures (one textural edge). The textural edge is clearly marked in the output image.

The transfer function, $H_{OPT}(\omega)$, can be generalized to n textures by simply adding more translated-in-frequency versions of $H_{STEP}(\omega)$. Denote the generalized, n texture transfer function as $H_{OPT,n}(\omega)$, defined as

$$H_{OPT,n}(\omega) = \sum_{i=1}^{n} H_i(\omega) \tag{16}$$

where

$$H_i(\omega) = H_{STEP}(\omega - \omega_i) + H_{STEP}(\omega + \omega_i) \tag{17}$$

and ω_i represents the frequency of the i^{th} texture. Each of the n filters respond to transient energy where textural transitions occur but null out response to the i^{th} texture in steady state. An example of a one-dimensional output image for an input image containing four ideal textures with three textural edges is shown in Figure 4. The normalized frequencies of the four different textures in the figure are $.04\pi$, $.06\pi$, $.08\pi$, and $.1\pi$, with each texture occurring once in the input image.

It should be pointed out that although each of the "sub" filters (i.e., $H_1(\omega)$, $H_2(\omega)$, $\cdots$) are narrowband bandpass about the respective textural frequencies, the overall system bandwidth and image bandwidth are about equal, as shown in Figure 5. The total textural edge detector bandwidth, BW, is written in terms of the tonal edge detector bandwidth as follows:

$$BW = \omega_n + \Omega \tag{11}$$

where ω_n represents the highest-frequency texture, and 2Ω is the bandpass bandwidth of the filter centered on ω_n.

The most general case of the model used in this development is one in which each of the spatial frequencies representing the different textures in the image are allowed to randomly deviate about some average frequency. This complication is introduced to allow for some of the irregularity of a real texture. A one-dimensional example in which both the amplitude and spatial frequency vary in proportion to independent random processes is shown in Figure 6. In this example, the average normalized spatial frequencies representing the two textures are $.04\pi$ and $.1\pi$ respectively. In terms of the general model presented in Section III, $\alpha(x)$ and $\beta(x)$ are independent Gaussian noise processes, with unit variance. The bandwidths of the amplitude noise and frequency noise processes are $.008\pi$ and $.006\pi$ respectively. Note that the filter adequately marks the two textural edges in the image, but also responds to regions within each texture where the spatial frequency changes. Decreasing the bandwidth of the noise modulating the frequency causes the spectral separation of the textures in the input image to increase. This results in improved performance of the filter at distinguishing textural edges from frequency deviations within a texture.

V. CONCLUSIONS

A frequency domain textural edge detection filter has been developed which, for the given model and filter bandwidth, places a maximum amount of image energy within a specified resolution interval near the textural edge. The textural edge detector was derived by relating textural edge detection to tonal edge detection via complex lowpass equivalent transformation. Hence, the optimum textural edge detector was found to be a sum of translated-in-frequency versions of the optimum tonal edge detector. This form allows the filter to be adapted to multitextural images. In addition, examples were presented which show the filter's insensitivity to tonal features in an image. The filter is adjustable; resolution can be traded for edge visibility in the case where the input image has been corrupted by noise.

The qualitative and complex nature of texture suggests that a totally general approach to modeling and classifying texture may never be found. It has been an objective in this investigation to develop a filter which optimizes a certain criteria relating to textural edge detection. But, as always, simplifications and assumptions were made indicating the need for further research. The model used in this development represented texture in terms of spatial frequency, and gray tone in terms of amplitude. One example of further research might be to base the development on a more complex model which incorporates a statistical description of texture. In addition, further work is needed in extension of the one-dimensional filter to two-dimensions.

This work has provided an approach to textural edge detection which can be implemented on digital hardware using the FFT. With the increased size and availability of digital computing facilities at a decreased cost, digital image processing methods will become more popular in the future.

REFERENCES

[1] K. S. Shanmugan, F. M. Dickey, J. A. Green, "An Optimal Frequency Domain Filter for Edge Detection in Digital Pictures," IEEE Trans. Pattern Anal. Machine Intell., Vol. PAMI-1, Jan. 1979.

[2] W. H. H. J. Lunscher, "The Asymptotic Optimal Frequency Domain Filter for Edge Detection," IEEE Trans. Pattern Anal. Machine Intell., Vol. PAMI-5, Nov. 1983.

[3] L. Kirvida, "Texture Measurements for the Automatic Classification of Imagery, IEEE Trans. Electromagnetic Compat., Vol. 18, pp. 38-42, Feb. 1976.

[4] L. Kirvida and G. Johnson, "Automatic Interpretation of ERTS Data for Forest Management," Symp. on Significant Results Obtained from the Earth Res. Technol. Satellite, NASA SP-327, March 1973.

[5] R. P. Kruger, W. B. Thompson, and A. F. Turner, "Computer Diagnosis of Pneumoconiosis," IEEE Trans. Systems, Man,

and Cybernetics, Vol. SMC-4, No. 1, pp. 40-49, January 1974.

[6] R. N. Sutton and E. L. Hall, "Texture Measures for Automatic Classification of Pulmonary Disease," IEEE Trans. Computers, Vol. C-21, pp. 667-676, July 1972.

[7] H. Maurer, "Texture Analysis with Fourier Series," Proc. Ninth Int. Symp. on Remote Sensing of Environment (Environmental Research Institute of Michigan, Ann Arbor, MI), pp. 1411-1420, April 1974.

[8] R. Bajcsy and L. Lieberman, "Computer Description of Real Outdoor Scenes," in Proc. Second Int. Joint Conf. on Pattern Recognition (Copenhagen, Denmark), pp. 174-179, August 1974.

[9] R. Bajcsy and L. Lieberman, "Texture Gradient as a Depth Cue," Comput. Graph. Image Processing, Vol. 5, No. 1, pp. 52-67, 1976.

[10] J. K. Townsend, "An Optimal Frequency Domain Textural Edge Detection Filter," Master's Thesis, University of Kansas, December 1984.

[11] J. G. Proakis, Digital Communications, McGraw-Hill, Inc., 1983.

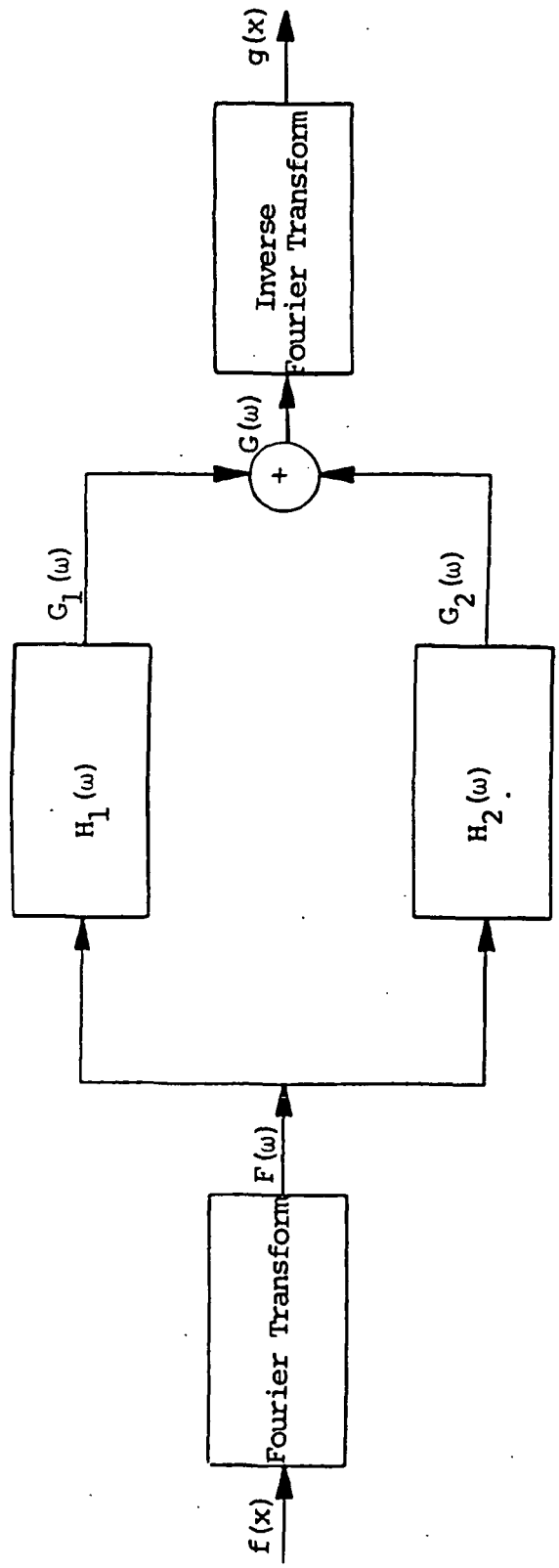


Figure 1 Block diagram of the optimum textural edge detection filter for two textures.

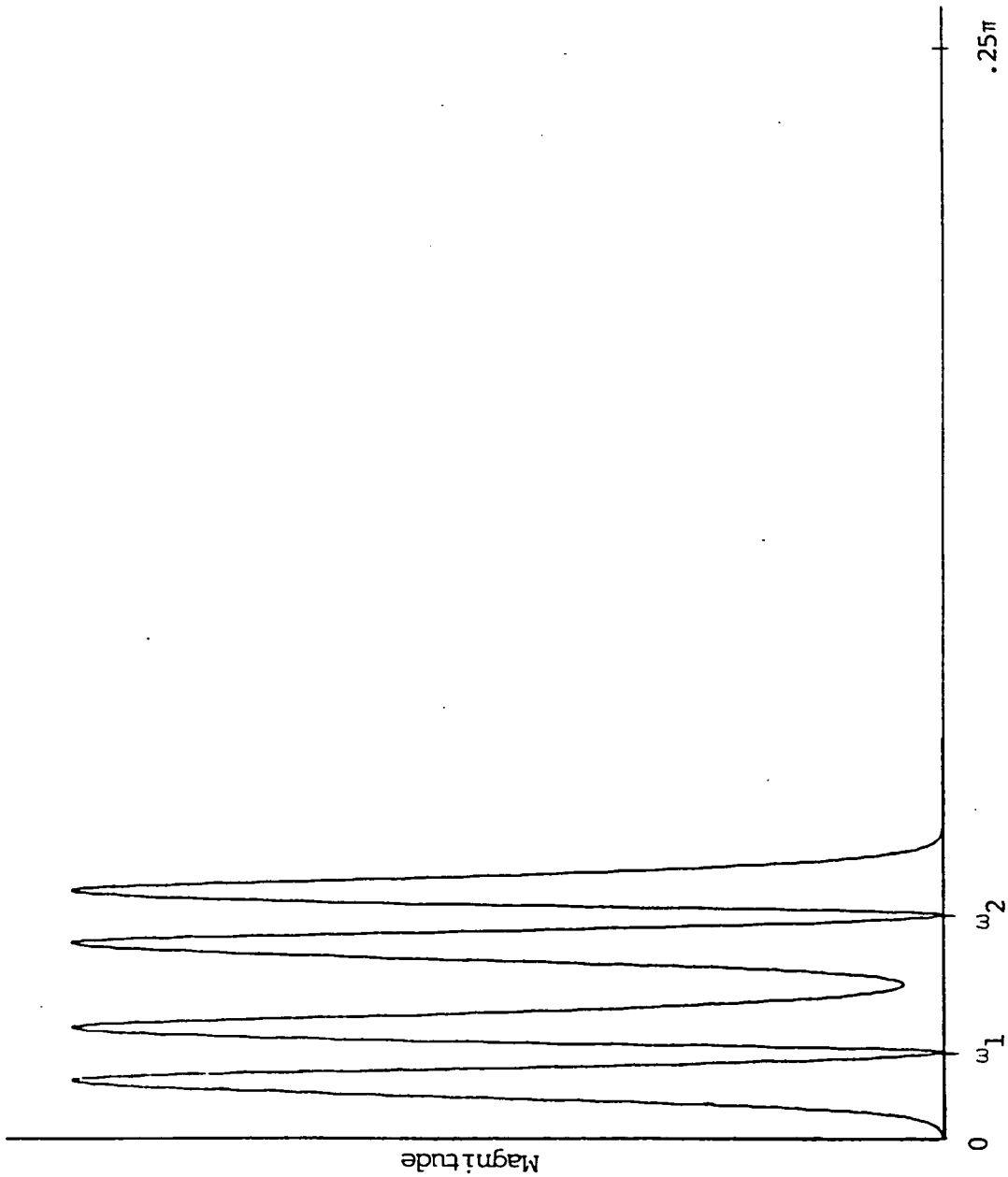
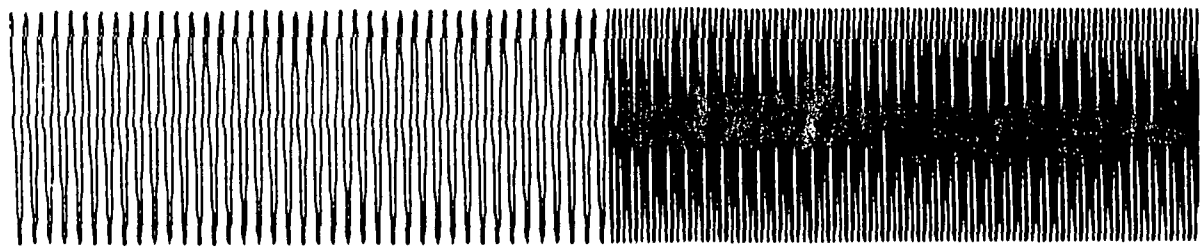
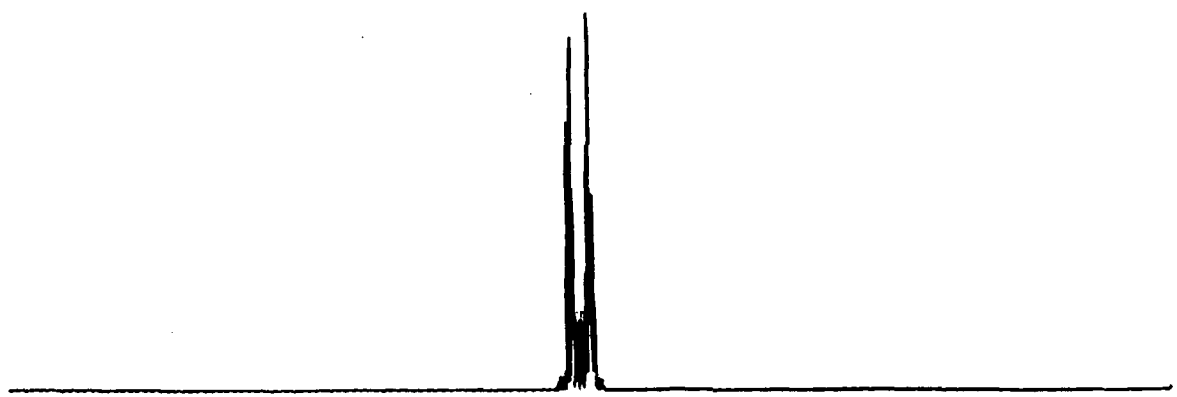


Figure 2 Single sided transfer function of the optimum textural edge detection filter. The bandwidths of $H_1(\omega)$ and $H_2(\omega)$ are narrow enough that response at ω_1 and ω_2 is zero.



(a)



(b)

Figure 3 (a) Input image consisting of two ideal textures.
(b) Magnitude of the optimum textural edge detector response (in the spatial domain).

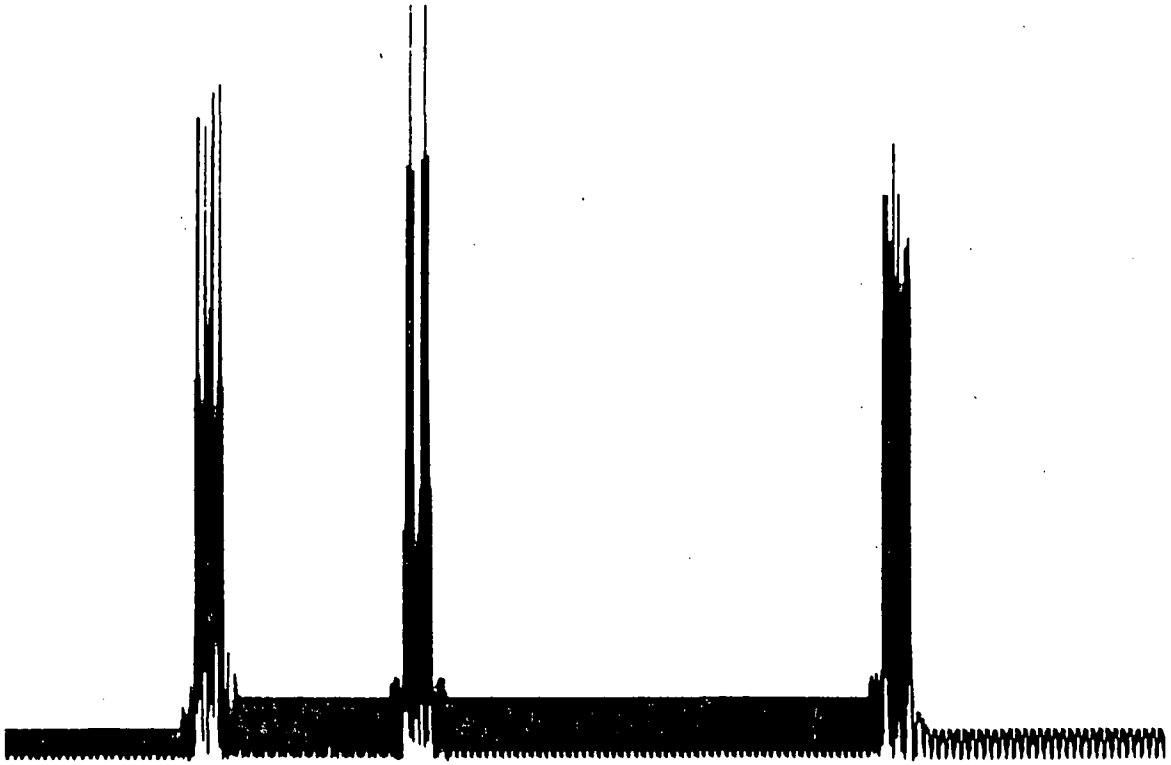


Figure 4 Magnitude of the response of the textural edge detection filter due to an input image with four ideal textures and three textural edges. The normalized spatial frequencies of the four textures are .04π, .06π, .08π, and .1π.

APPENDIX

AGENDA

Monday, June 10:

8:00 - 8:30	Coffee, tea and kolaches
8:30 - 9:00	Program Overview
	Dr. Diane Wickland, Program Manager for Terrestrial Ecosystems, NASA Headquarters, Washington, D.C. "An Overview of NASA Land Processes Program"
	R. P. Heydorn, Science Manager, Fundamental Research Program: MPRIA, NASA/Johnson Space Center, Houston, Texas
	Math/Stat: Session I
9:00 - 9:45	L. F. Guseman, Jr. and L. Schumaker, Texas A&M University "Multivariate Spline Methods and Their Use in Classification Procedures"
9:45 - 10:30	Charles Peters, University of Houston "Methods of Normal Mixture Analysis Applied to Remote Sensing"
10:30 - 10:45	**Break**
10:45 - 11:30	E. Parzen, Texas A&M University "Quantile Data Analysis Methods and Edge Detection for Noisy Images"
11:30 - 1:00	**Lunch**
	Math/Stat: Session II
1:00 - 1:45	C. Morris, D. V. Hinkley, and W. Johnston, University of Texas at Austin "Classification in a Spatially Correlated Environment"
1:45 - 2:30	R. P. Heydorn, NASA/JSC "Estimating Parameters in a Mixture of Probability Densities"
2:30 - 3:15	David Scott, Rice University "Experiences with Examining Large Multivariate Data Sets with Graphical Nonparametric Methods"
3:15 - 4:00	**Discussion**

Pattern Recognition: Session I

4:00 - 4:45 Carlos Berenstein, Laveen N. Kanal, and David Lavine, LNK Corporation
"Further Analysis of Subpixel Registration Accuracy: Geometrical and Statistical Results"

4:45 - 5:30 Grahame Smith, SRI International
"Recovery of Surface Shapes from Multiple Images"

Tuesday, June 11:

8:30 - 9:00 Coffee, tea, and kolaches

Pattern Recognition: Session II

9:00 - 9:45 Vincent Hwang, Larry Davis, University of Maryland and Takashi Matsuyana, Kyoto University, Japan
"Integration of Evidence in Image Understanding Systems"

9:45 - 10:30 E. Mikhail and F. C. Paderes, Purdue University
"Investigation of Critical Issues in Rectification and Registration of Satellite Scanner Imagery"

10:30 - 10:45 **Break**

10:45 - 11:30 Curtis E. Woodcock, Boston University and Alan H. Strahler, Hunter College
"Relating Ground Scenes to Spatial Variation in Remotely Sensed Images"

11:30 - 1:00 **Lunch**

Pattern Recognition: Session III

1:00 - 1:45 David Dow, National Space Technology Labs.
"Influence of Ground Control Point Selection on Landsat MSS Rectification Accuracy: Whole Scene vs. Portions of the Scene"

1:45 - 2:30 W. Tobler and S. Kennedy, University of California--Santa Barbara
"Smooth Multidimensional Interpolation"

2:30 - 3:00 Discussion

LIST OF ATTENDEES

David Dow, NASA Earth Resources Laboratory

L. F. Guseman, Jr., Texas A&M University

R. P. Heydorn, NASA/JSC
J. Hill, UT-Austin
D. V. Hinkley, UT-Austin
Vincent Hwang, University of Maryland

Johnny Johnston, UT-Austin

Laveen Kanal, LNK Corp.

R. B. MacDonald, NASA/JSC
E. Mikhail, Purdue University
Robert Murphy, NASA Headquarters

H. J. Newton, TAMU

F. Paderes, Purdue University
Emanuel Parzen, TAMU
Charles Peters, University of Houston
Robert Price, NASA Headquarters

L. Schumaker, TAMU
David Scott, Rice University
Grahame B. Smith, SRI
W. B. Smith, TAMU

W. R. Tobler, University of California-Santa Barbara

Diane Wickland, NASA Headquarters
Curtis Woodcock, Boston University