USCIPI REPORT #620

Semiannual Technical Report
Covering Research Activity During the Period
1 March 1975 to 31 August 1975

by

William K. Pratt

September 1975

Signal and Image Processing Institute
UNIVERSITY OF SOUTHERN CALIFORNIA
Department of Electrical Engineering-Systems
3740 McClintock Avenue, Room 404
Los Angeles, CA 90089-2564 U.S.A.

ABSTRACT

This technical report summarizes the image processing research activities performed by the University of Southern California during the period of 1 March 1975 to 31 August 1975 under Contract No. F08606-72-C-0008 with the Advanced Research Projects Agency, Information Processing Techniques Office.

The research program, entitled, "Image Processing Research," has as its primary purpose the analysis and development of techniques and systems for efficiently generating, processing, transmitting, and displaying visual images and two dimensional data arrays. Research is oriented toward digital processing and transmission systems. Five task areas are reported on: (1) Image Coding Projects: the investigation of digital bandwidth reduction coding methods; (2) Image Restoration and Enhancement Projects: the improvement of image fidelity and presentation format; (3) Image Data Extraction Projects: the recognition of objects within pictures and quantitative measurement of image features; (4) Image Analysis Projects: the development of quantitative measures of image quality and analytic representation; (5) Image Processing Systems Projects: the development of image processing hardware and software support systems.

PROJECT PARTICIPANTS

Project Director

William K. Pratt

Research Staff

Harry C. Andrews
Werner Frei
Ali Habibi
Ernest L. Hall
Richard P. Kruger
Nasser E. Nahi
Ram Nevatia
Guner Robinson
Alexander A. Sawchuk
William Thompson
Lloyd R. Welch

Support Staff

Toyone Mayeda
James M. Pepin
Ray Schmidt
Joyce Seguy
Dennis Smith
George Soen
Florence B. Tebbets
Donna Whiteneck

Students

Peter Alfvin
Behnam Ashjari
Evelyn Boka
Ben Britt
Marilyn Chan
Steve Dashiell
Faramarz Davarian
Farshid Farshad
Gary Graham
Michael Huhns
Steve Hou
Mohammad Jahanshahi
Scott Johnson
Mohsen Khalil
Alan Larson
Paul Liles
Dennis McCaughey
Simon Lopez-Mora
Clanton Mancill
Lee Martin
Dave Merle
David Miller
Hideo Murakami
Firouz Naderi
David Nagai
Clay Olmstead
Javad Peyrovian
Jin Soh
Richard Stephens
Robert Wallis
James Wiedel

Table of Contents

1. Research Project Overview

This report describes the progress and results of the University of Southern California image processing research study for the period of 1 March 1975 to 31 August 1975. The image processing research study has been subdivided into five projects:

Image Coding Projects
Image Restoration and Enhancement Projects
Image Data Extraction Projects
Image Analysis Projects
Image Processing Systems Projects

In image coding the orientation of the research is toward the development of digital image coding systems that represent monochrome and color images with a minimal number of code bits. Image restoration is the task of improving the fidelity of an image in the sense of compensating for image degradation. In image enhancement, picture manipulation processes are performed to provide a more subjectively pleasing image, or to convert the image to a form more amenable to human or machine analysis. The objectives of the image data extraction projects are the registration of images, detection of objects within pictures and measurements of image features. The image analysis projects comprise the background research effort into the basic structure of images in order to develop meaningful quantitative characterizations of an image. Finally, the image processing systems projects include research on image processing computer languages and the development of experimental equipment for the sensing, processing, and display of images.

The next section of this raport summarizes some of the research project activities during the past six months. Section 2 is a list of putlications by prcject members. Sections 3 to 6 describe the research effort on the projects listed above during tha reporting period.

2. Publications

The following is a list of papers, articles, and reports of research staff members of the USC Image Processing Institute which have been published or accepted for publication during the past six months, and which have resulted from ARPA sponsored research.

H.C. Andrews, "Numerical Analysis Techniques in Digital Imge Restoration," Proceedings 1975 Symposium on Circuits and Systems, April, 1975.

H.C. Andrews, "MTF Restoration by Pseudoinversion," Proceedings of the International Optical Computing Conference, April, 1975, Washington, D.C.

H.C. Andrews, Chapter 4, "Two Dimensional Transforms," Picture Processing and Digital Filtering, F.S. Huang, ed., Springer Verlag, May, 1975.

H. C. Andrews and C. L. Patterson "Outer Product Expansions and Their Uses in Digital Image Processing," IEEE Transactions on Computers, (accepted for publication).

H. C. Andrews and C. L. Patterson, "Digital Interpolation of Discrete Images," IEEE Transactions on Computers (accepted for publication).

H. C. Andrews and C. L. Patterson, "Singular Value Decompositions and Digital Image Processing," IEEE Transactions on Audio, Speech, and

Signal Processing, (accepted for publication).

H. C. Andrews and B. R. Hunt, Digital Image Restoration, Prentice Hall (accepted for publicaton).

S. R. Dashiell and A. A. Sawchuk, "Optical Synthesis of Nonlinear Nonmonotonic Functions," accepted for publication in Optics Communicatons.

W. Frei, "The Need for a Minimum Picture Data Basis," presented at 1975 IEEE Computer Society Workshop on Machine Pattern Analysis, March 3-5, 1975.

W. Frei, "Accuracy Considerations for Digitized Images and Hardcopy Output," presented at 1975 IEEE Computer Society Workshop on Machine Pattern Analysis, March 3-5, 1975.

E. L. Hall, W. O. Crawford, And F. E. Roberts, "Moment Measurements for Computer Texture Discrimination in Chest X-Rays," IEEE Transactions Biomedical Engineering, November, 1975.

E. L. Hall, Z. H. Cho, J. K. Chan, R. P. Kruger and D. G. McCaughey, "A Comparative Study of 3-D Image Reconstruction Algorithms," IEEE Trans. on Nuclear Science, March, 1975.

E. L. Hall, R. P. Kruger and A. F. Turner, "Automated Measurements from Chest X-Rays for Lung Disease Classification," USA-Japan Computer Conference, August, 1975.

E. L. Hall, R. P. Kruger and A. F. Turner, "Automated

Measurements from Chest X-Rays," Proceedings of the Computer Applicatons in Radiclogy Cenference, March, 1975.

E. L. Hall, W. B. Thcmpson, G. Varsi and R. Gaulden, "Computer Measurement of Particle Sizes in Electron Microscope Images," IEEE Trans on Systems, Man and Cybernetics, to be published, 1975.

G. C. Huth and E. L. Hall, "Computer Tomography and its Application to Nuclear Medical Imaging," Computers in Nuclear Medicine, to be published.

N. D. A. Mascarenhas and W. K. Pratt, "Digital Image Restoration Under a Regression Model," IEEE Transactions on Circuits and Systems, March, 1975.

N. E. Nahi and M. Naraghi, "A General Image Estimation Algorithm Applicable to Multiplicative and Non-Gaussian Noise," Proceedings of 18th Midwest Symposium on Circuits and Systems, August 11-12, 1975, Concorshia Univ., Montreal P.Q., Canada.

N. E. Nahi and A. Habibi, "Nonlinear Recursive Image Enhancement," IEEE Transactions on Circuits and Systems, March, 1975.

R. Nevatia, T. O. Binford, "Recognition and Description of Complex Curved Objects," Fifth Annual Symposium on Imagery Pattern Recognition, University of Maryland, April 17-18, 1975.

R. Nevatia and T. O. Binford, "Recognition and Description of Complex Curved Objects", Fifth Annual Symposium on Automatic Imagery Pattern Recognition, Univ. of Maryland, April 1975.

R. Nevatia, "Object Boundary Determination in a Textured Environment," (To be presented) Annual ACM Conference, Minneapolis, October 1975.

R. Nevatia, "Depth Measurement by Motion Stereo," Accepted for publication in Computer Graphics and Image Processing.

R. Nevatia, "Structured Descriptions of Complex Curved Objects for Recognition and Visual Memory," Accepted for publication as a book by Birkhauser-Verlag, Basle, Switzerland.

W. K. Pratt and M. Huhns, "DPCM Quantization Error Reduction for Image Coding," Society of Photo-Optical Instrumentation Engineers, 19th Annual Technical Symposium, San Diego, August, 1975.

W. K. Pratt and C. E. Mancill, "Spectral Estimation Techniques for the Spectral Calibration of a Color Image Scanner," Applied Optics, November, 1975.

W. K. Pratt, "Vector Space Formulation of Two Dimensional Signal Processing Operations, Journal Computer Graphics and Image Processing, Academic Press, March, 1975.

J. A. Roese, W. K. Pratt, G. S. Robinson, A. Habibi, "Interframe Transform Coding and Predictive Coding Metods," IEEE Internatonal Communicatons Conference, San Francisco, June, 1975.

J. A. Roese, G. S. Robinson, "Combined Spatial and Temporal Coding of Image Sequences", 19th Annual SPEI Symposium on Efficient Transmission of Pictorial Information, San Diego, Calif., August

21-22, 1975.

A. A. Sawchuk and M. J. Peyrovian, "Restoration of Astigmatism and Curvature of Field", Journal of the Optical Society of America, vol. 65, 1975.

A. A. Sawchuk and S. R. Dashiell, "Nonmonotonic Nonlinearities in Optical Processing", Proc. IEEE International Optical Computing Conference, Washington, D.C., April 23-25, 1975.

W. B. Thompson, A. F. Turner, and R. P. Kruger, "Automated Chest Radographic Diagnosis," accepted for publication, Investigative Radiology.

W. E. Thompson, A. L. Zobrist, "Building a Distance Function for Gestalt Grouping," accepted for publication IEEE Transactions on Computers.

3. Image Coding Prcjects

The effort in image coding is directed toward the research and development of image coding systems providing a transmission bit rate reduction and tolerance to channel errors. Coding systems are under investigation for: monochrome and color imagery; slow scan and real time television; and information preserving and controlled fidelity operation. Results of this research study during the past six months are summarized here and presented in detail in subsequent sections.

3.1 Singular Value Decomposition Image Coding

Harry C. Andrews

The singular value decomposition algorithm (also referred to as SVD) is a computational algorithm developed for a variety of applications including matrix diagonalization, regression, and pseudoinversion [1,2]. The algorithm has also been suggested for image coding [3,4]. By approaching the image coding task from a viewpoint of numerical analysis, it is possible to formulate a solution in terms of least squares methods which results in deterministically best truncation errors over all other unitary transforms [6]. A discrete image may be considered to be an array of nonnegative scalar entries formed into a matrix. Let such an image matrix be designated as G. Without loss of generality, let G be square with a quadratic form given by

$$\underline{G} = \underline{A}\,\underline{\alpha}\,\underline{B}^T \qquad (1)$$

where the $\underline{A}$ and $\underline{B}$ matrices are assumed unitary. Solving for $\underline{\alpha}$ it is observed that

$$\underline{\alpha} = \underline{A}^T \underline{G}\ \underline{B} \qquad (2)$$

The $\underline{\alpha}$ matrix is seen to be the "transform" of the image matrix where $\underline{A}$ transforms the columns of the image and $\underline{B}$ transforms the rows of the image. A list of traditional transform techniques is presented in Table 1 indicating some of the properties of the individual transform methods. The entries are listed in terms of general decreasing usefulness as decorrelation devices as well as decreasing complexity.

The first entry in the table is the one of interest here and has decidedly different transform properties from the remaining. The singular value decomposition (SVD) method has the unique property that the coefficient matrix $\underline{\Lambda}$, is diagonal with at most only N nonzero entries. The definition of this transform is given by

$$\underline{\alpha} = \underline{\Lambda}^{\frac{1}{2}} = \underline{U}^T \underline{G}\ \underline{V} \qquad (3)$$

where

Table 1: Transform Domains for Image Representations

Entry	Transform Name	Representation	Unknown Parameters to be computed)	Algorithmic Implementation (on the order of:)	Reference
1	Singular Value Decomposition	$[G][G]^t = [U][\Lambda][U]^t$ $[G]^t[G] = [V][\Lambda][V]^t$ $[\alpha] = [\Lambda^{1/2}] = [U]^t[G][V]$	basis vectors and singular values	$\left.\begin{matrix} N^3 \\ N^3 \\ 2N^3 \end{matrix}\right\}$ computed for each image	Golub (1970) Albert (1972)
2	Karhunen-Loève Hotelling, Principal Components, Factor Analysis	$[\phi_x] = [E_x][\Lambda_x][E_x]^t$ $[\phi_y] = [E_y][\Lambda_y][E_y]^t$ $[\alpha] = [E_x]^t[G][E_y]$	transform coefficients	$\left.\begin{matrix} N^3 \\ N^3 \end{matrix}\right\|$ computed only once $2N^3$	Wintz (1972) Huang (1971)
3	Cosine	$[\alpha] = [\cos]^t[G][\cos]$	transform coefficients	$4N^2\log_2 2N$	Ahmed (1974)
4	Fourier	$[\alpha] = [\mathfrak{F}][G][\mathfrak{F}]$	transform coefficients	$2N^2\log_2 N$ (complex)	Andrews (1970)
5	Slant	$[\alpha]=[\text{Slant}]^t[G][\text{Slant}]$	transform coefficients	$2N^2\log_2 N$	Pratt (1974)
6	DLB	$[\alpha] = [\text{DLB}]^t[G][\text{DLB}]$	transform coefficients	$2N^2\log_2 N$ (integer. arith.)	Haralick (1974)
7	Walsh	$[\alpha] = [W][G][W]$	transform coefficients	$2N^2\log_2 N$ (additions)	Pratt (1969)
8	Haar	$[\alpha]=[\text{HAAR}]^t[G][\text{HAAR}]$	transform coefficients	$2(N-1)$	Andrews (1970)

$$\underline{G}\,\underline{G}^T = \underline{U}\,\underline{\Lambda}\,\underline{U}^T \qquad (4a)$$

and

$$\underline{G}^T\underline{G} = \underline{V}\,\underline{\Lambda}\,\underline{V}^T \qquad (4b)$$

The $\underline{\Lambda}$ matrix is diagonal and comprises the singular values of the picture matrix $\underline{G}$, while $\underline{U}$, and $\underline{V}$ are the respective singular vector matrices of $\underline{GG}^T$ and $\underline{G}^T\underline{G}$, and are orthogonal as a result of the nonnegative definiteness of $\underline{GG}^T$ and $\underline{G}^T\underline{G}$. Because of these properties of $\underline{U}$ and $\underline{V}$ it is possible to solve for the image matrix such that

$$\underline{G} = \underline{U}\,\underline{\Lambda}^{\frac{1}{2}}\underline{V}^T \qquad (5a)$$

or equivalently

$$\underline{G} = \sum_{i=1}^{R} \lambda_i^{\frac{1}{2}}\, \underline{u}_i\underline{v}_i^T \qquad (5b)$$

Where R ≤ N and represents the rank or number of nonzero singular values λ_i . The coding implications are that one must transmit the 2N singular vectors as well as N singular values for image reconstruction at the receiver. Figure 1 presents a pictorial representation of the singular value decomposition. Traditional image transform methods

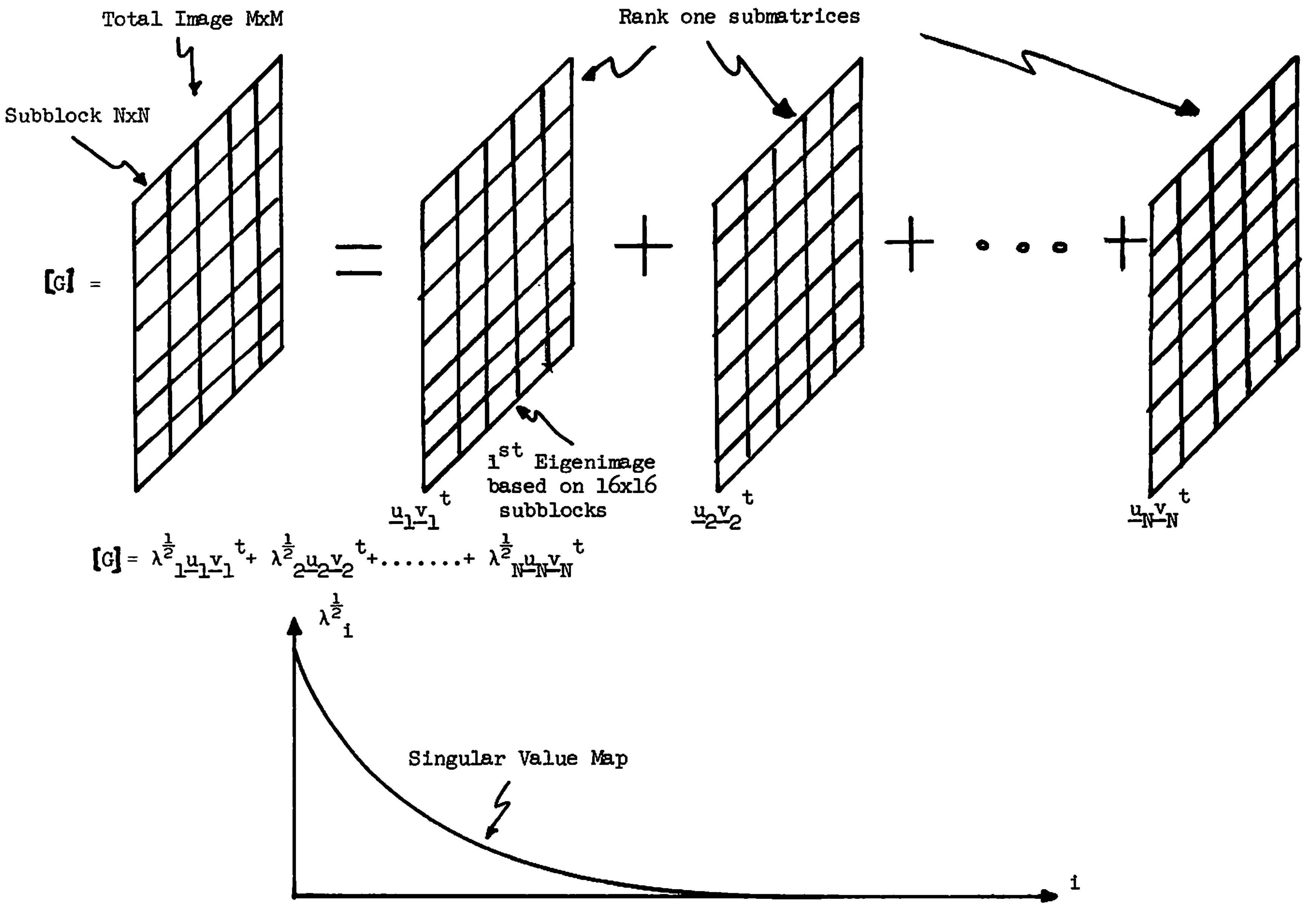

Total Image MxM
Subblock NxN
Rank one submatrices
[G] =
$\underline{u}_1\underline{v}_1^t$
1st Eigenimage based on 16x16 subblocks
$\underline{u}_2\underline{v}_2^t$
$\underline{u}_N\underline{v}_N^t$
$[G] = \lambda^{\frac{1}{2}}_1\underline{u}_1\underline{v}_1^t + \lambda^{\frac{1}{2}}_2\underline{u}_2\underline{v}_2^t + \ldots\ldots + \lambda^{\frac{1}{2}}_N\underline{u}_N\underline{v}_N^t$
$\lambda^{\frac{1}{2}}_i$
Singular Value Map
i

Figure 3.1-1. Subblock SVD Algorithm

usually break an image up into subblocks for ease in hardware implementations. This technique is developed here because the computational expenses for large image singular value decompositions is great. Specifically, if an $M \times M$ image is broken into $N \times N$ subblocks, then each subblock takes on the order of N^3 computations to get to the SVD domain. Since there are $(M/N)^2$ such subblocks, a total of M^2N computations are required as compared to M^3 computations if the entire image were decomposed. A similar comparison exists for fast computational transforms which require $M^2\log N$ total subblock operations for the $M \times M$ image. Thus the number of computations for SVD compared to fast transforms is M^2N vs $M^2\log N$. The ratio of N/log N increase to implement the SVD transform on 16 x 16 subblock sizes is only a factor of 4 for SVD versus Fourier, cosine, Walsh, or the like.

Figure 2 contains a block diagram of the SVD coding scheme. The major components at the transmitter consist of the SVD domain transformer, a possible truncator, and parallel singular value and singular vector coders. The SVD transformer, as discussed above, would require on the order of four times the number of real computations compared to a real $N^2\log N$ transform algorithm. The truncator is included in the diagram to emphasize the tremendously large energy compaction property of the SVD technique. From eq. (6) the truncated image $\underline{G}_K$ may contain an extremely large amount of original image energy in a very few number of singular values.

The two remaining blocks in the coder concern themselves with the singular value coding and singular vector coding. In the former the

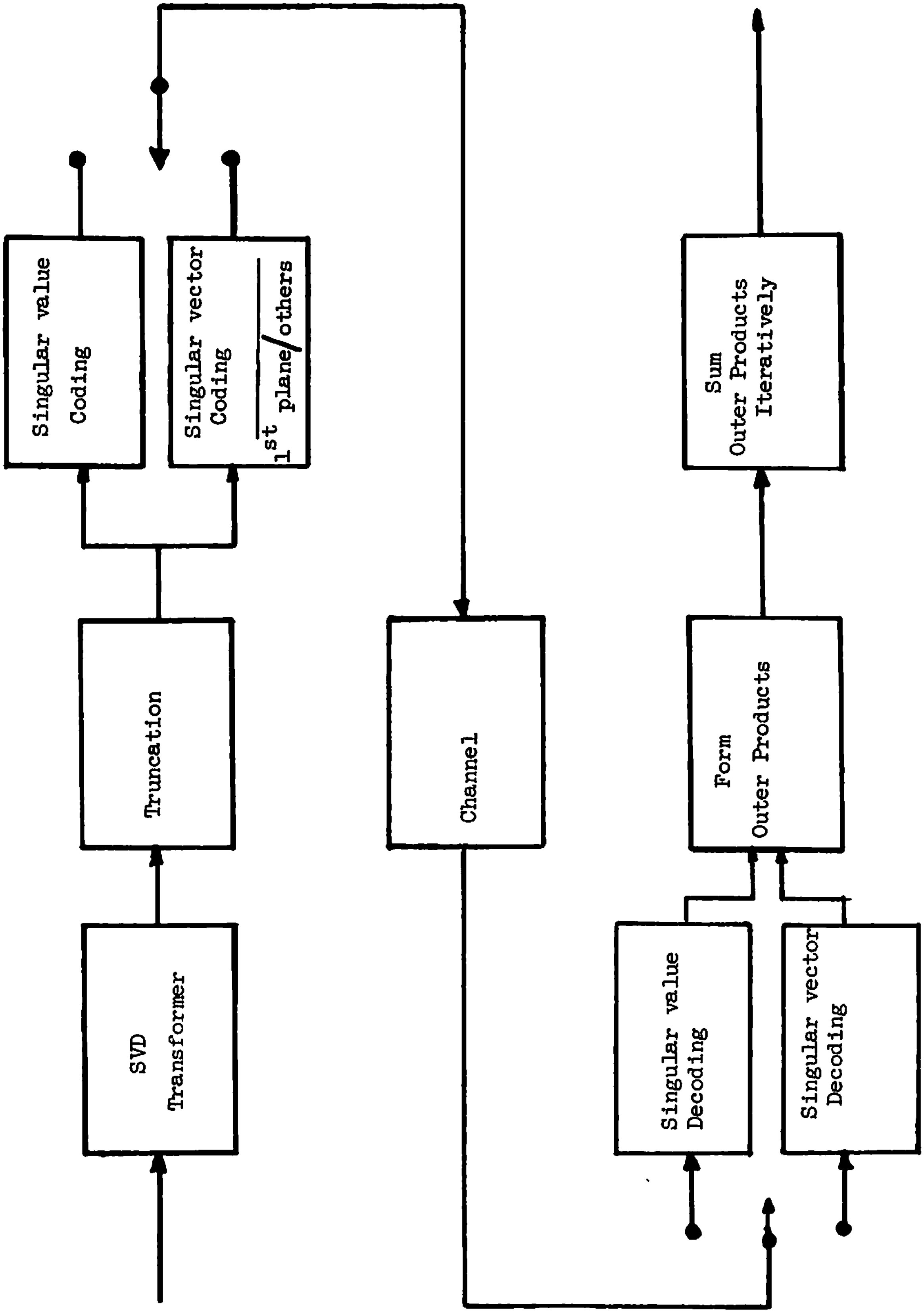


Figure 3.1-2. SVD Coding Scheme

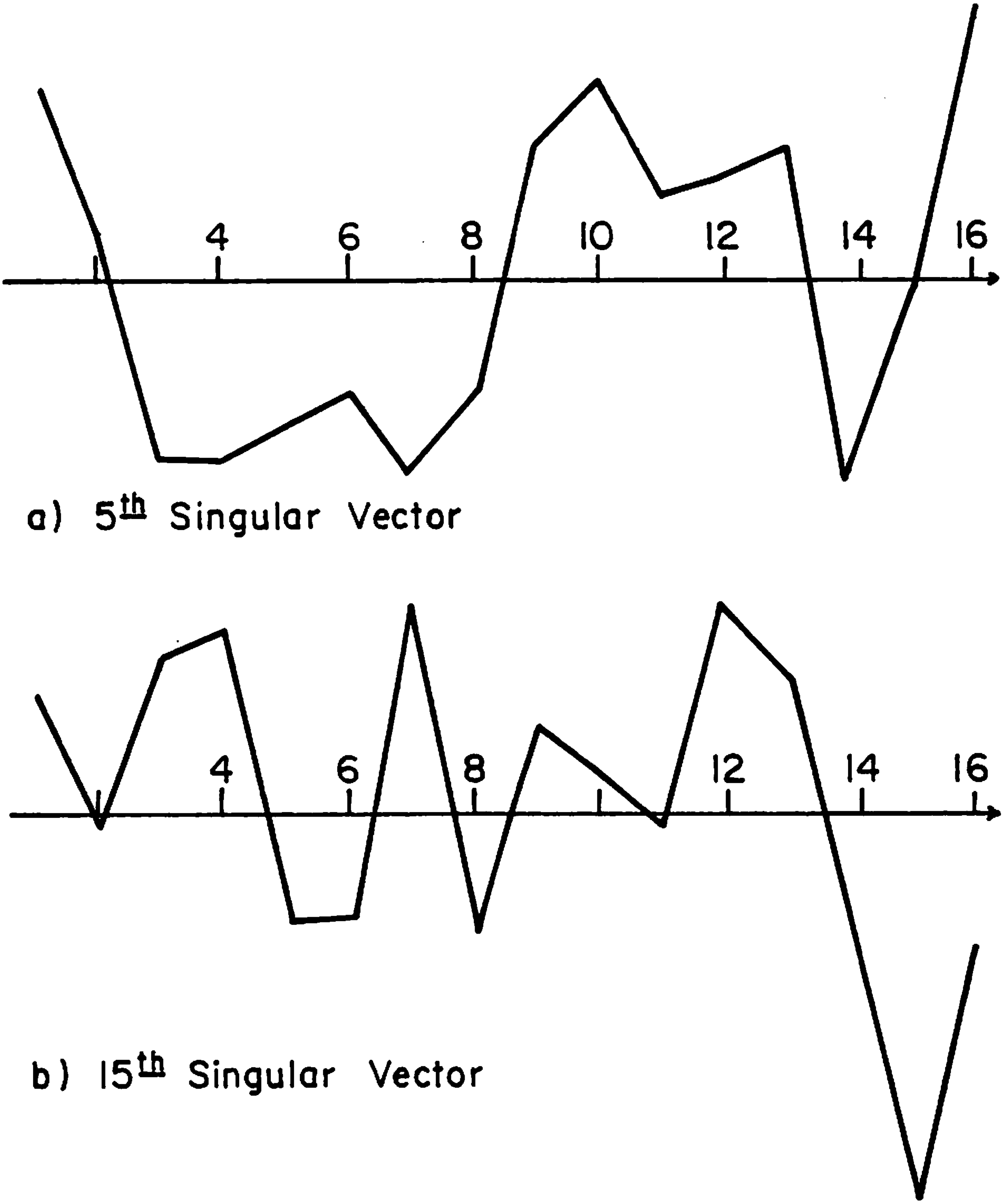
large dynamic range of the singular values indicates a certain amount of care needed for coding, but because there are so few large dynamic ranged coefficients, (N vs.N^2) the total bit requirement still remains low. The singular vector coding algorithm is broken into two components, that associated with the first singular plane (or eigenimage) and that associated with the remaining eigenimage planes. Because each of these planes (actually two vectors which when, outer producted, produce a plane) is orthonormal, the scalar entries in the singular vectors are quite well behaved, and lend themselves to easy requantization.

Using the basic configuration of figure 2, the number of bits necessary for transmission of a subblock then becomes a function of the truncation, if any, the bit distribution over the singular values, and the bit distribution over the singular vectors. Typical distributions on the singular values track the variance of these values, and, in fact, tend to be proportional to the value itself. For the singular vectors, two more bits are provided for scalar entries of the first eigenimage than for subsequent eigenimages. In addition, because the singular vectors are orthonormal, one need not transmit N scalar values per vector but only N-i-1 such values for the i-th vector. (Orthonormality reduces the degrees of freedom on the vectors such that the vectors can be completed at the receiver.)

In order to develop efficient quantizers and coders for the SVD domain, a test image (512 x 512) was broken into 16 x 16 subblocks and data was gathered over each subblock of the entire frame. Statistics describing the singular values and vectors were then gathered and are described here. For 16 x 16 subblocks, one obtains 16 singular values and monotonic decreasingly ordered means and variances of each singular value can be computed. The exceedingly large dynamic range of between 4 and 5 orders cf magnitude indicates the need for variable bit coding as a functicn of the singular value index. The distribution of the singular values naturally is one sided (no negative entries) and appears as a curve intermediate between a broad Gaussian and uniform density.

The statistics describing the singular vectors are much better behaved. Figure 3 presents two specific singular vectors from a particular sutblock as an illustration of the shape of these parameters. The singular vectors tend to be well behaved in their range and tend tc have an incraasing number of zero crossings as a function of increasing index. In fact it is known that the first singular vector never exhitits zeio crossings when the subblock is ncnnegative (as it always is for imagery) [7]. Thus the lower indexed vectors tend to have a great deal of adjacent sample correlaticn.

Since the first vectoi for both $\underline{U}$ and $\underline{V}$ are guaranteed to have no zero crossings (similar to the dc vectors of Fourier, Walsh, cosine, etc. transforms), these vectors form a separate set of statistical parameters from the remaining set. The mean vector over all subblocks in the test image becomes a constant value of 0.25 with a very tight variation provided by a variance of 10^{-3}. Naturally a given first singular vector will not, in fact, be a constant of value 0.25 (the



a) 5th Singular Vector

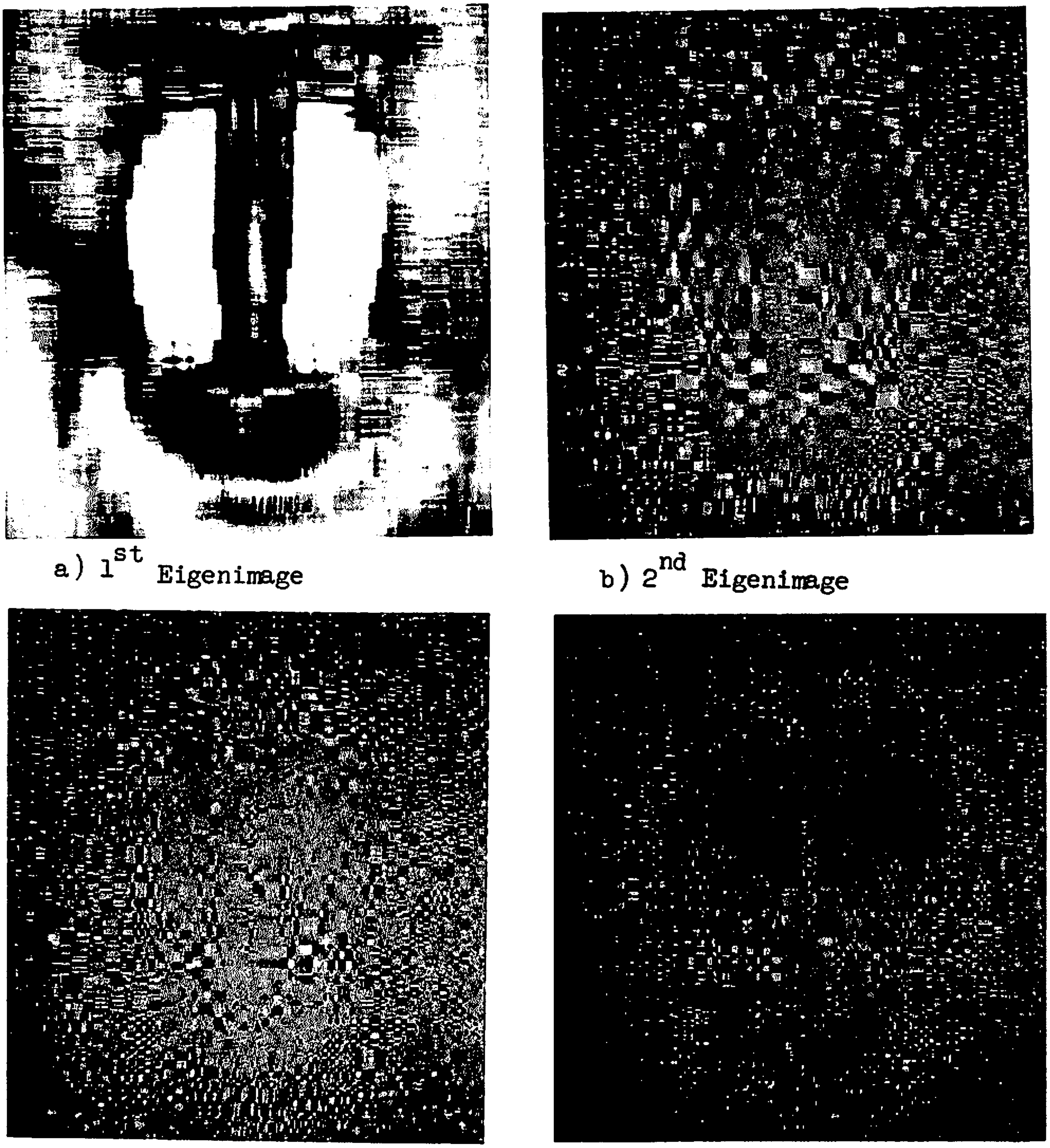
b) 15th Singular Vector

Figure 3.1-3. Typical singular vectors.

appropriate normalized value to guarantee orthonormality), but will have variations which when weighted by the corresponding large singular value will appear quite different from a constant. In fact the distribution of the scalar values defining the entries in the first singular vector are very close to Gaussian with parameters $N(0.25, 10^{-3})$.

The remaining eigenvectors are also quite well behaved with the average or mean of each cf these vectors being the zero vector. The variance of these singular vectors is on the order of 10^{-1} and the scalar entries which comprise these vectors are also close to normally distributed with $N(0, 10^{-1})$. Because of the difference in the statistics of the first singular vector with those of the remaining singular vectors, they are coded separately as indicated in the block diagram of figure 2.

One image is used fcr experimental purposes here. Its SVD structure is revealed in figure 4. In figure 4 the image is broken into 16 x 16 subblocks and each subblock is decomposed into its 16 singular values and associated 32 singular vectors. The first, second, third, and fourth eigenimages cbtained frcm the corresponding singular vectors are displayed in the figure. The first plane has no zero crossings and consequently the display of negative data is not necessary. In the three remaining planes, a linear dislay is presented with negative values being dark and positive values being light. Notice that considerable recognition of the original scene is available in the first eigenimage and subsequent eigenimages tend to



a) 1st Eigenimage

b) 2nd Eigenimage

c) 3rd Eigenimage

d) 4th Eigenimage

Figure 3.1-4. "Baboon" image with SVD on 16x16 subblocks provide more and more zero crossing information which can be likened to higher frequency information.

A complete SVD coding system has been simulated according to the block diagram cf figure 2. The ccding strategy used a linear PCM quantizer with variable bit assignment on the singular values and a Max quantizer [5] with variable bit assignment on the PCM values of the entries in the singular vectors. A variety of bit assignments were investigated, and an optimization routine in terms of mean square error measured the best bit assignment. Figure 5 provides some performance curves developed during the optimization process. The two lower curves indicate the truncation effects as the number of equivalent bits per pixel are increased. The uppermost curve illustrates the mean square error using a linear quantizer on the singular values. The Max quantizer curve indicates about a 0.20% mean square error improvement over the linear curve and is only about 0.20% worse (or introduces 0.20% more mean square error) than the truncated but uncoded curves. Pictorial results, from which the upper two curves are derived, are presented in figure 6. Here the percentage mean square errors and bit rates per pixel are listed under the respective coded images for both linear and Max quantization on the singular vectors.

In concluding this section it is important to emphasize a few points. First, the work is incomplete, and it is premature to base any conclusions cn the viability of SVD coding in competition with cther existing techniques. It is fair to say that if as much effort

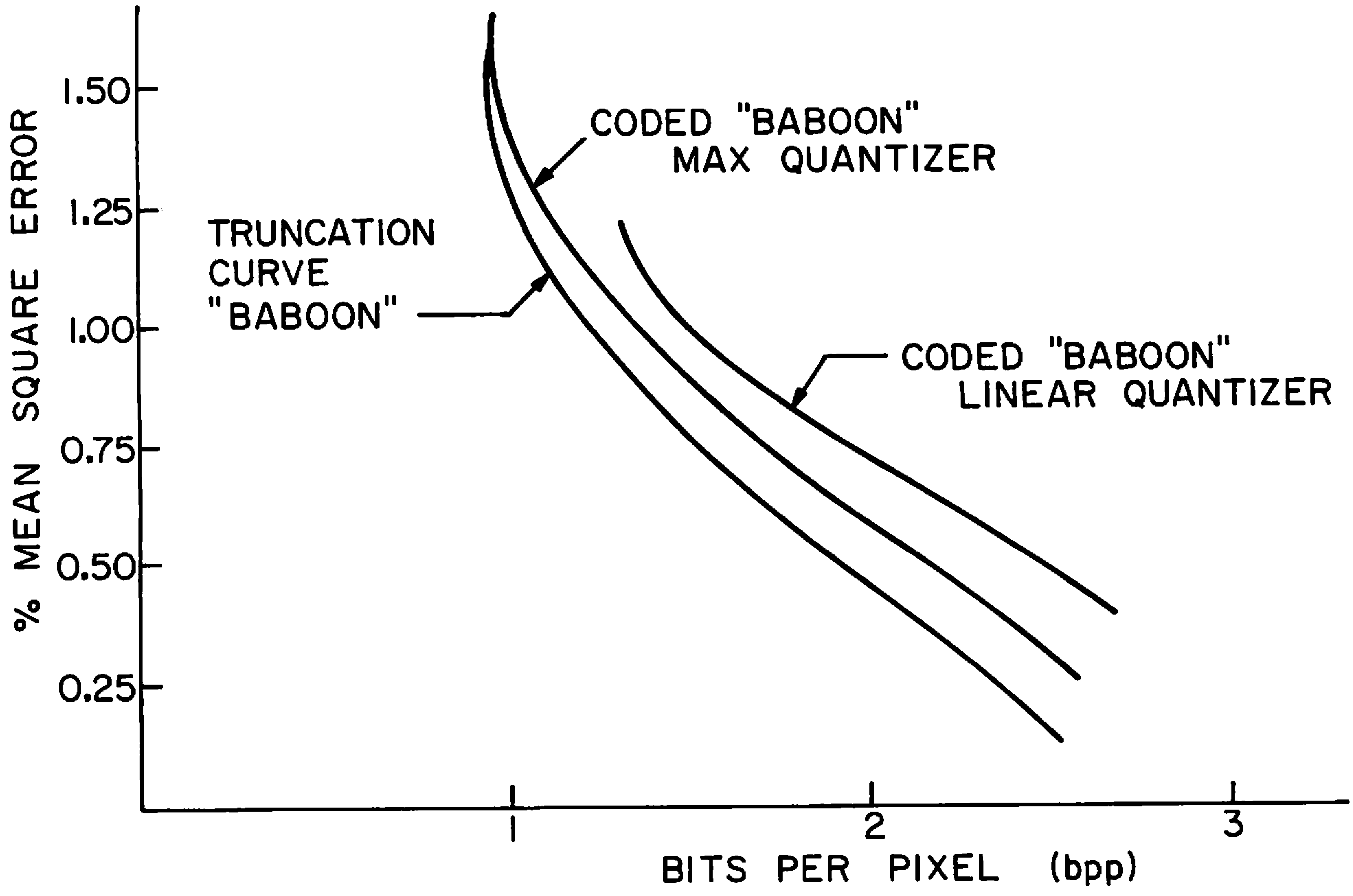
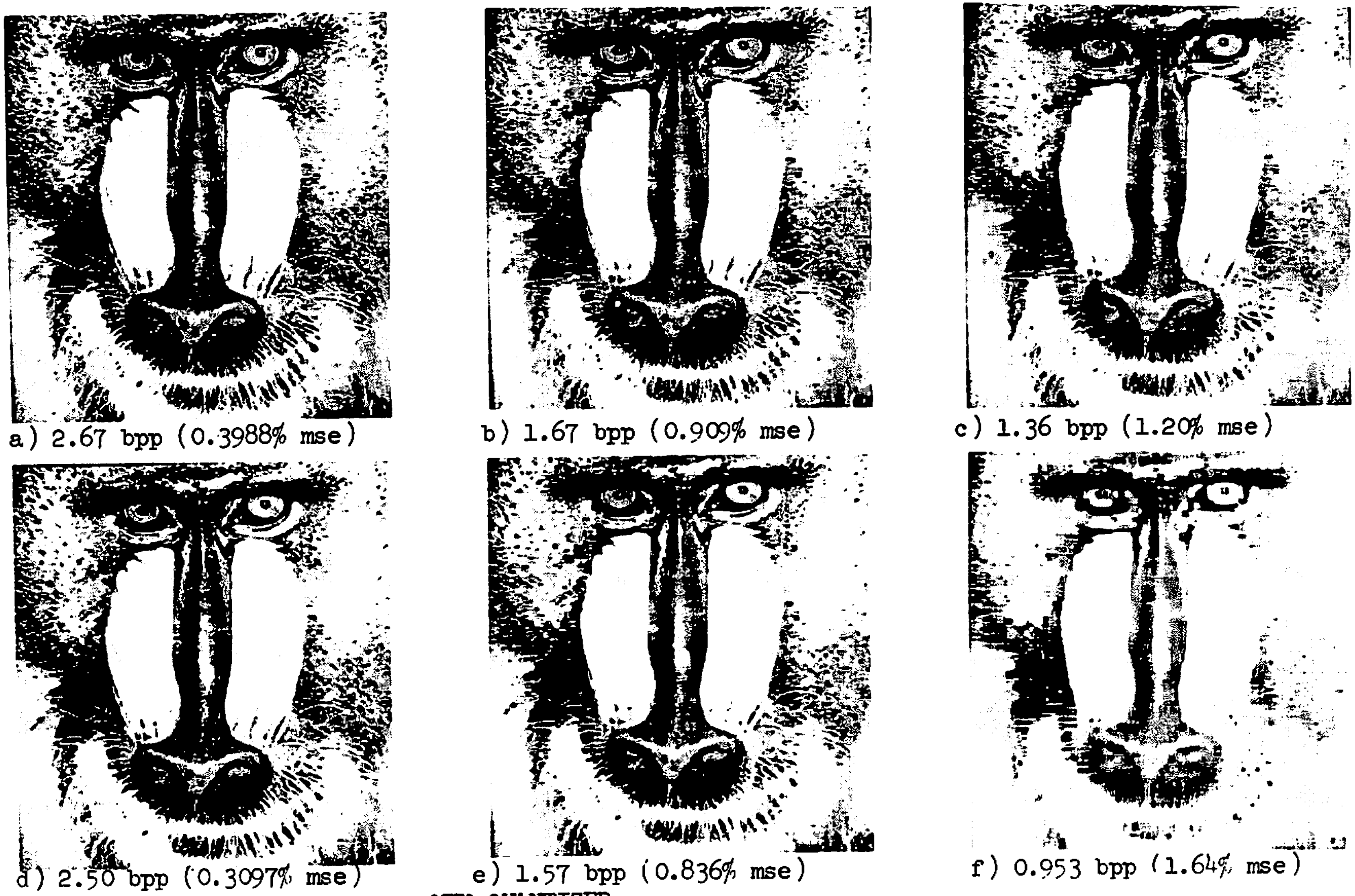


Figure 3.1-5. Percentage mean square error versus bits per pixel.



a) 2.67 bpp (0.3988% mse)

b) 1.67 bpp (0.909% mse)

c) 1.36 bpp (1.20% mse)

d) 2.50 bpp (0.3097% mse)

e) 1.57 bpp (0.836% mse)
MAX QUANTIZER

f) 0.953 bpp (1.64% mse)

Figure 3.1-6. "Baboon" Image with SVD coding using "Baboon" statistics is put into investigating the potential for SVD coding as has been put into traditional transform methods, then considerable improvement over the results presented here can be expected. However, algorithmic implementation might become quite complex. On the other hand five years ago realtime (video bandwidth rate) FFT transform coders were thought to be too complex, and yet they exist today. Consequently only time and future study will tell whether SVD coding becomes a practical reality.

References

1. G.H. Golub and C. Reinsch, "Singular Value Decomposition and Least Squares Solutions," Numer. Math, Vol. 14, 1970, pp. 403-420.

2. A. Albert, Regression and the Moore-Penrose Pseudoinverse, Academic Press, New York, 1972.

3. J.D. Kennedy, S.J. Clark, and W.A. Parkyn, Jr., "Digital Imagery Data Compression Techniques," McDonnell Douglas Corporation Report No. MDC-G0-402, January 1970.

4. H.C. Andrews and C.L. Patterson, "Outer Product Expansions and their Uses in Digital Image Processing," American Mathematical Monthly, Vol. 1, No. 82, January, 1975, pp. 1-13.

5. T. Max, "Quantizing for Minimum Distortion," IRE Transactions on Information Theory, Vol. IT-16, March, 1960, pp. 7-12.

6. P.A. Wintz, "Transform Picture Coding," Proceedings of the IEEE,

Vol. 60, No. 7, July, 1972, pp. 809-820.

7. J. Todd, Survey of Numerical Analysis, Chapter 8, McGraw-Hill, New York, 1958.

3.2 Restoration for Binary Symmetric Channel Errors

Michael N. Huhns

A previous report [1] has presented and analyzed a technique for restoring the output of a quantizer so that the result more accurately matches the quantizer's input with respect to a mean-square error criterion. The restoration is obtained by the use of

$$E\{x \mid x \in R\} = \frac{\int_R \underline{x}\, p(\underline{x}) d\underline{x}}{\int_R p(\underline{x}) d\underline{x}} \tag{1}$$

where R is a region in N-space to which an N x 1 vector x is assigned during quantization, and $p(\underline{x})$ is the multidimensional probability density function of $\underline{x}$. The restoraticn is based essentially upon exact knowledge of the quantizer output. A similar, but more difficult problem results when the quantizer output is not known exactly. This could occur, for example, when the quantizer output is transmitted over a noisy channel. The first section in this report explores the effect of channel errors on the restorations obtained using eq.(1). The next section examines a technique that statistically compensates for the effect of channel errors.

Effects of Channel Errors on Quantized Signals: In this analysis, channel errors are assumed to be modelled by a binary symmetric channel (BSC) [2]. The characteristics of this type of channel are shown in figure 1. The channel is discrete and memoryless and can be specified by a transition probability assignment P(j|k), for j,k=0,1, as

$$p = \begin{bmatrix} 1-p & p \\ p & 1-p \end{bmatrix} \qquad (2)$$

Since the channel is memoryless, the probability of an output sequence $\underline{z}=(z_1,z_2,\ldots,z_N)$, given an input sequence $\underline{x}=(x_1,x_2,\ldots,x_N)$, is given by

$$p(z|x) = \prod_{i=1}^{N} p(z_i|x_i) \qquad (3)$$

Based on this definition, a BSC was computer simulated with the channel error probability, p, chosen to be 0.01. The simulated channel was then applied tc transform coded images. Three images were zonal transfcrm coded in 16 x 16 blocks and their quantized transform domain components were encoded by assigning each a binary code word. The resulting sequence cf binary digits was operated on by the

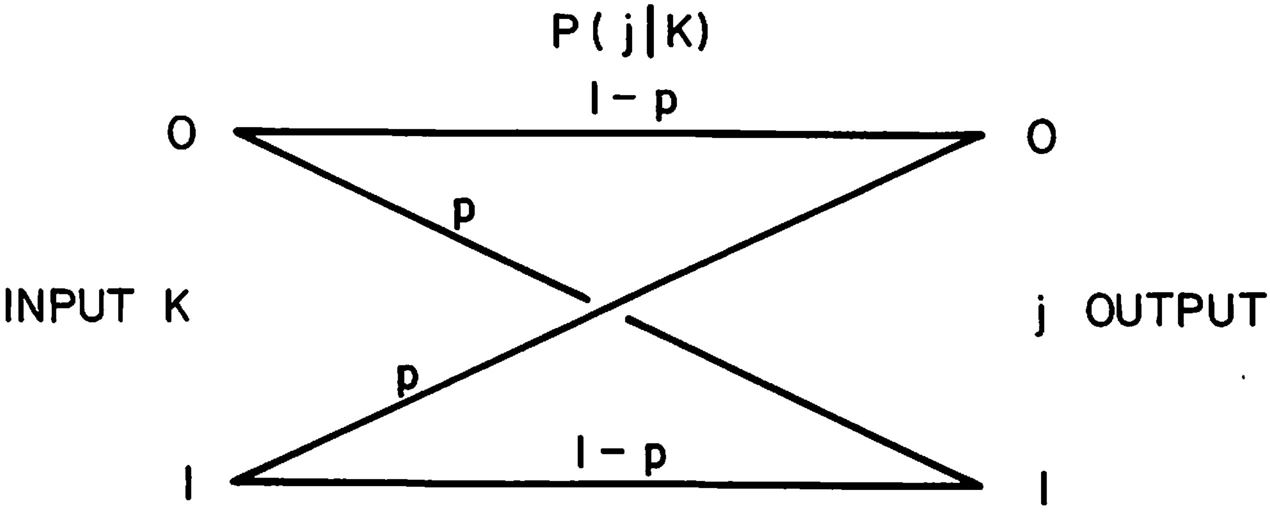
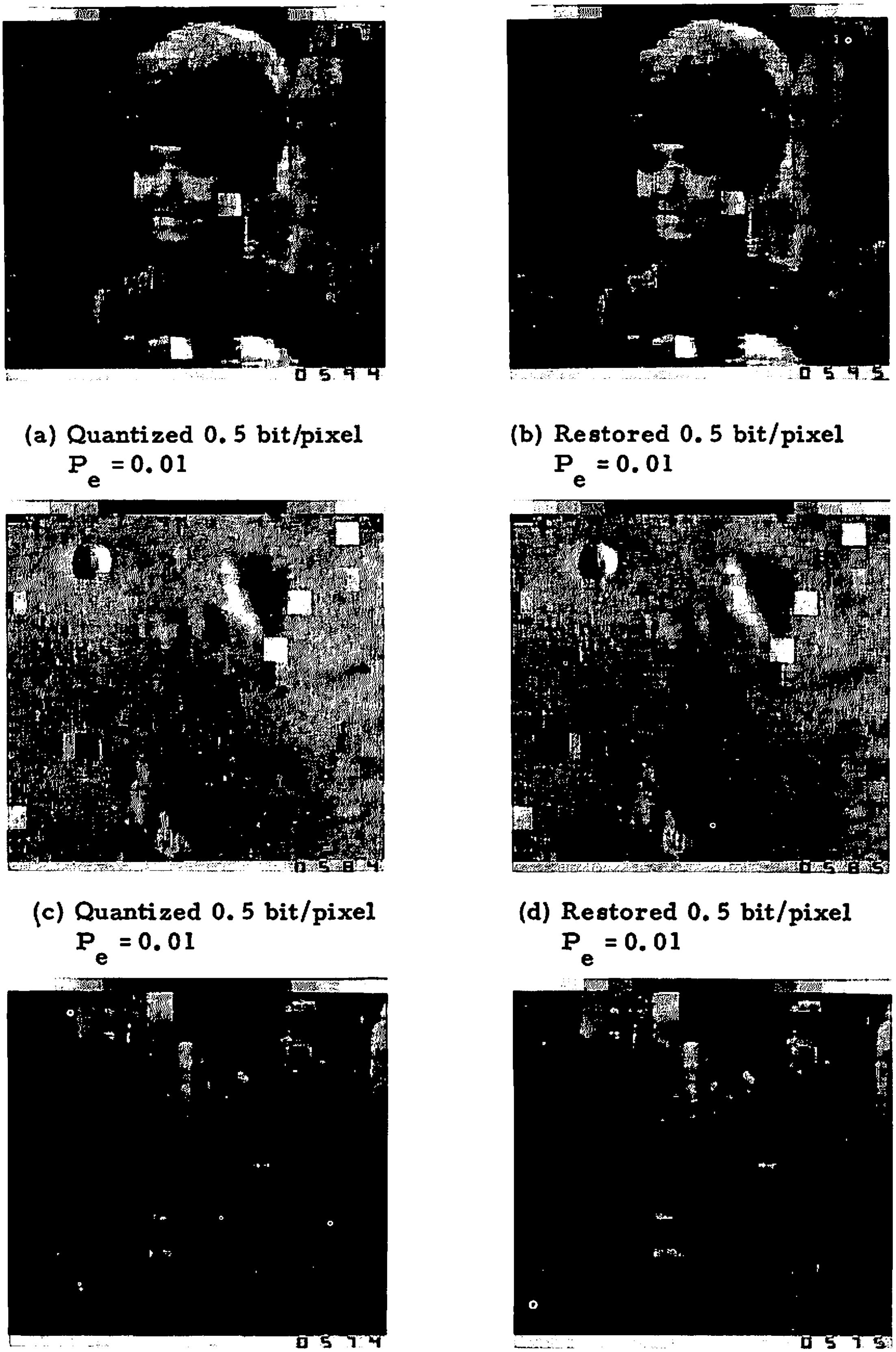


Figure 3.2-1. Transition probabilities for a binary symmetric channel.

simulated channel. The error-corrupted bit stream was then either decoded directly, as shown in figures 2a, 2c, and 2e, or restored by the use of eq.(1) to reduce the effects of the quantization process. Figure 3 contains a schematic of this procedure. The decoded images with the quantization effects reduced are shown in figures 2b, 2d, and 2f.

Bit errors in transform coding that arise due to a binary symmetric channel are seen to result in an emphasis of the block structure and a subjective error that extends over the entire block. This latter effect occurs because inverse transforming a block containing an error distributes this error over all the resultant image domain components. The reconstruction technique implied by eq.(1) is thus insensitive to channel errors. Since it provides visual and mean-square error improvements in noise-free cases, it can be utilized equally well in noisy environments.

Reconstruction of Quantized and Transmitted Signals: The previous section demonstrated that channel errors do not adversely affect the performance of the restoration technique derived previously. However, this technique does nothing to ameliorate the effects of the channel errors. This is because the fundamental restoration formula presented in eq.(1) was derived without any consideration of channel structure. By including the channel structure in the derivation, the resultant restoration technique can simultaneously reduce the effects of the quantization process and mitigate the effects of channel errors.



(a) Quantized 0.5 bit/pixel
$P_e = 0.01$

(b) Restored 0.5 bit/pixel
$P_e = 0.01$

(c) Quantized 0.5 bit/pixel
$P_e = 0.01$

(d) Restored 0.5 bit/pixel
$P_e = 0.01$

(e) Quantized 0.5 bit/pixel
$P_e = 0.01$

(f) Restored 0.5 bit/pixel
$P_e = 0.01$

Figure 3.2-2. Minimum mean square error restoration of Hadamard transformed zonal quantized images.

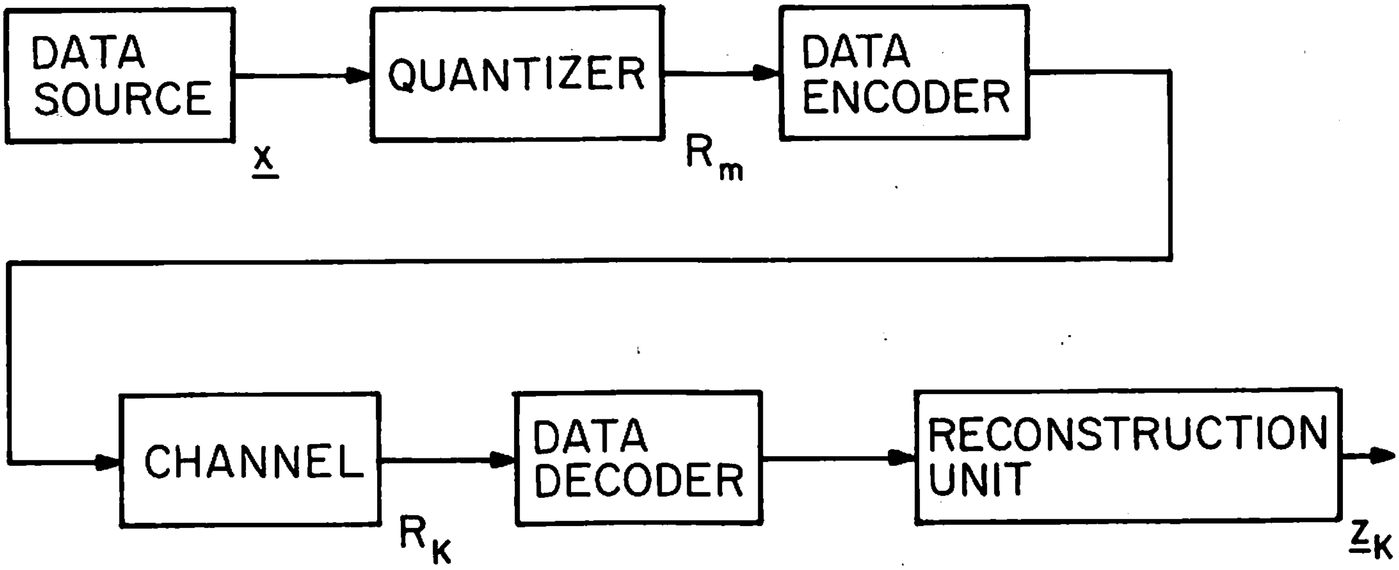


Figure 3.2-3. Data system used to model the effects of channel errors on the quantization restoration process.

Let the output cf a data source (this output could consist of DPCM samples, PCM samples, or transform domain samples) be denoted by $\underline{x}=(x_1,x_2,\ldots,x_N)$ and described by a probability density function $p(\underline{x})$. The reconstruction of $\underline{x}$, after $\underline{x}$ has been quantized to one of M regions and channel-error corrupted, is denoted by $\underline{z}=(z_1,z_2,\ldots,z_N)$ for k=1,2,...,M (refer to figure 3). The mean-square error that results from this process is

$$\delta = \sum_{k=1}^{M}\sum_{m=1}^{M} p(m|k)\int_{R_m} (\underline{x}-\underline{z}_k)\,(\underline{x}-\underline{z}_k)^T\, p(\underline{x})d\underline{x} \qquad (4)$$

This error can be minimized by proper choice of the restoration points, z_k. Setting the partial derivatives of this error with respect to z_k equal to zero yields

$$z_k = \frac{\sum_{m=1}^{M} p(m|k)\int_{R_m} \underline{x}p(\underline{x})d\underline{x}}{\sum_{m=1}^{M} p(m|k)\int_{R_m} p(\underline{x})d\underline{x}} \qquad (5)$$

for k=1,2,...,M. This expression is the noisy channel version of eq.(1) and provides a minimum mean-square error estimate of the input to a quantizer based on the output of a noisy channel, the characteristics of the quantizer, and the a priori statistics of the input. This equation is also a multidimensional version of a result first derived in [3]. For a noiseless channel, the channel matrix P becomes the identity matrix and eq. (5) reduces to eq.(1). When the probability vclume integrals in the denominator of eq.(5) are all equal, which is approximately true for Max quantization, the restoration equation simplifies to

$$z_k = \sum_{m=1}^{M} p(m|k) \frac{\int_{R_m} \underline{x}p(\underline{x})d\underline{x}}{\int_{R_m} p(\underline{x})d\underline{x}} \qquad (6)$$

or

$$z_k = \sum_{m=1}^{M} p(m|k)y_m \qquad (7)$$

where y_m is given by eq.(1). This result holds for maximum output entropy quantizers and two-level symmetrical quantizers, and is approximately correct for many other types.

A signal that has been quantized and then transmitted over a noisy channel can thus be cptimally restored by utilizing eq.(5). The restoration sclutions found earlier for Gaussian and Laplacian probability density functions (see [4] and [5], respectively) can be substituted directly into eq.(5) once the transition matrix for the channel has been determined. The resultant estimator can then be used to restore the cutputs of transform and DPCM coders that have been degraded by channel errors.

References

1. M.N. Huhns, "Transform Domain Spectrum Interpolation," University of Southern California Image Processing Institute Technical Report, USCIPI Report 530, March 1974, pp. 28-38.

2. R.G. Gallager, Information Theory and Reliable Communications, John Wiley and Sons, New York, 1968, p. 73.

3. A.J. Kurtenbach and P.A. Wintz, "Quantizing for Noisy Channels," IEEE Transactions on Communication Technology, Vol. COM-17, April, 1969, pp. 291-302.

4. M.N. Huhns, "Quantization Error Reduction for Image Coding," USC Image Processing Institute Technical Report, USCIPI Report 540, September, 1974, pp. 16-26.

5. M.N. Huhns, "Optimum Image Reconstruction from DPCM Samples," USC Image Processing Institute Technical Report, USCIPI Report 560, March, 1975, pp. 15-18.

3.3 Interframe Image Coding

Guner S. Robinson and John A. Roese

Interframe coding of digital image sequences encompasses those techniques which make use of the high correlation that exists between pixel amplitudes in successive frames. Intraframe coding techniques that exploit spatial correlations can, in principle, be extended to

*This research is partially supported by the Naval Undersea Center, San Diego, California.

include correlations in the temporal domain. Previous research in the area of three-dimensional Fourier and Hadamard transformations has indicated that bit rates can be reduced by a factor of five by incorporating correlations in the temporal direction [1]. However, three-dimensional transform systems are unattractive since they require large amounts of data storage and excessive computation.

To alleviate the problems associated with three-dimensional transform systems, new hybrid (two-dimensional transform)/DPCM image coding systems have been developed [2]. These systems utilize both spatial and temporal correlations while greatly reducing memory storage and computational requirements. A block diagram for a hybrid (two-dimensional transform)/DPCM system is shown as figure 1. In present implementations of this system, either a two-dimensional cosine or Fourier transformation is performed on 16 x 16 subblocks. DPCM linear predictive coding in the temporal domain is then applied to the transform coefficients of each subblock. For notational convenience, the hybrid interframe coders employing two-dimensional Fourier transforms will be denoted as FFD and those using two-dimensional cosine transforms as CCD. The FFD and CCD coders are adaptive in the sense that statistics of the transform coefficient differences of each subblock are computed prior to encoding the transform coefficients in the temporal direction by parallel DPCM coders. At the receiver, the transmitted transform coefficients are decoded and a replica of each frame is reconstructed by the appropriate inverse two-dimensional transformation. These systems require only a single frame of storage and involve significantly less memory and fewer computations than three-dimensional transform coding techniques.

Operational Modes: At least three operational modes have been identified for the hybrid interframe coding systems. These operational modes depend on the initial conditions assumed for the previous coder. The initial conditions are:

a. No apriori information available at the receiver;

b. Limited information (such as mean, variance and temporal correlations based on a statistical model) available at the receiver; and

c. First frame available at the receiver.

In the no apriori information available case, several frames are required for the hybrid coder to settle. However, it has been experimentally verified that in the remaining two cases, nearly stable coder performance is achieved within the first 4 to 6 frames. From operational considerations, the third set of initial conditions is the most realistic as periodic full frame updating will be required to eliminate the cumulative effects of channel noise.

Mathematical Formulation: Let $f(x,y)$ denote a two-dimensional array of intensity values on an $N \times N$ subblock of a digital television image of size $M \times M$. Typical values for M and N are 256 and 16, respectively. Let $F(u,v)$ be the two-dimensional array obtained by taking the two-dimensional transform of $f(x,y)$. In the case of the two-dimensional discrete Fourier transform, the expressions relating $f(x,y)$ and $F(u,v)$ are

$$F(u,v) = \frac{1}{N^2} \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(x,y) \exp\left[-\frac{2\pi i}{N}(ux + vy)\right] \quad (1)$$

and

$$f(x,y) = \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} F(u,v) \exp\left[+\frac{2\pi i}{N}(ux + vy)\right] \quad (2)$$

for $u,v,x,y=0,1,\ldots,N-1$. For image processing applications, $f(x,y)$ is a positive real function representing brightness of the spatial sample. The two-dimensional Fourier transform of a real-valued function has the conjugate symmetry property. Also, the Fpurier transform consists of $2N^2$ components, i.e., the real and imaginary or magnitude and phase components of each spatial frequency. However, as a result of the conjugate symmetry properties mentioned above, only N^2 components are required to completely define the Fourier transform [3].

In the case cf the Fourier transform, a shift in the spatial-domain variables results in a multiplication of the Fourier transform of the un-shifted image by a phase factor. If the input imge $f(x,y,t_1)$ is shifted by the amount x_0 in the x-direction and y_0 in the y-direction between times t_1 and t_2, then the Fourier transform of the shifted image is given by

$$F(u,v,t_2) = F(u,v,t_1) \exp\left[\frac{2\pi i}{N} (ux_0 + vy_0)\right] \quad (3)$$

This shifting property is expected to be useful in detecting and compensating for effects of motion between frames since many types of motion, such as panned motion, produce significant changes in phase components but small changes in amplitude components. Thus, compensation for camera platform motion could be implemented directly in the array of phase components by application of appropriate phase correction factors.

The two-dimensional Fourier transform F(u,v) of a spatial signal function f(x,y) is separable, i.e., it can be computed as two sequential one-dimensional transforms since the Fourier kernel, is separable and symmetric. Thus, the basic one-dimensional discrete Fourier kernel transform that must be performed is

$$F(u) = \frac{1}{N} \cdot \sum_{x=0}^{N-1} f(x) \exp\left(-\frac{2\pi i}{N} ux\right) \qquad (4)$$

for u=0,1,...,N-1.

In the case of the discrete Cosine transform, the one-dimensional transform is

$$F(u) = \frac{1}{N} \sum_{x=0}^{N-1} f(x) \cos\left(\frac{(2x+1)u\pi}{2N}\right) \qquad (5)$$

for u=0,1,...,N-1. The cosine transform is also separable and a two-dimensional discrete cosine transform of an N x N subblock results in N^2 real coefficients.

Experimental evidence derived from transmission of a typical "head and shoulders" picture telephone scene has shown that the frame difference signal has a probability density closely approximated by a double sided exponential function [4]. The optimum minimum mean square error quantizer for this distribution has been found to be a uniform quantizer combined with a ccmpanding of the frame difference signal [5].

Since the variances of the transform domain coefficient differences are different, it is necessary to use different quantizer parameters for each coefficient.. Each coefficient difference signal is allocated a number of bits proportional to the estimated variance in accordance with an optimum bit assignment algorithm.

Fidelity Criteria: In figure 1, differences between the input signal $f(x,y,t)$ and output signal $\hat{f}(x,y,t)$ are due to two sources: quantization errors and channel noise errors. To evaluate coding efficiency of the hybrid encoders, two objective criteria were used. The first criterion, NMSE, is a measure of the mean square error between $f(x,y,t)$ and $\hat{f}(x,y,t)$ averaged over an entire frame of size M x M. Normalization is achieved by dividing the mean square error by the mean signal energy within the frame to give

$$\mathrm{NMSE} = \frac{\frac{1}{M^2}\sum_{x=0}^{M-1}\sum_{y=0}^{M-1}\left[f(x,y,t) - \hat{f}(x,y,t)\right]^2}{\frac{1}{M^2}\sum_{x=0}^{M-1}\sum_{x=0}^{M-1}\left[f(x,y,t)\right]^2} \qquad (6)$$

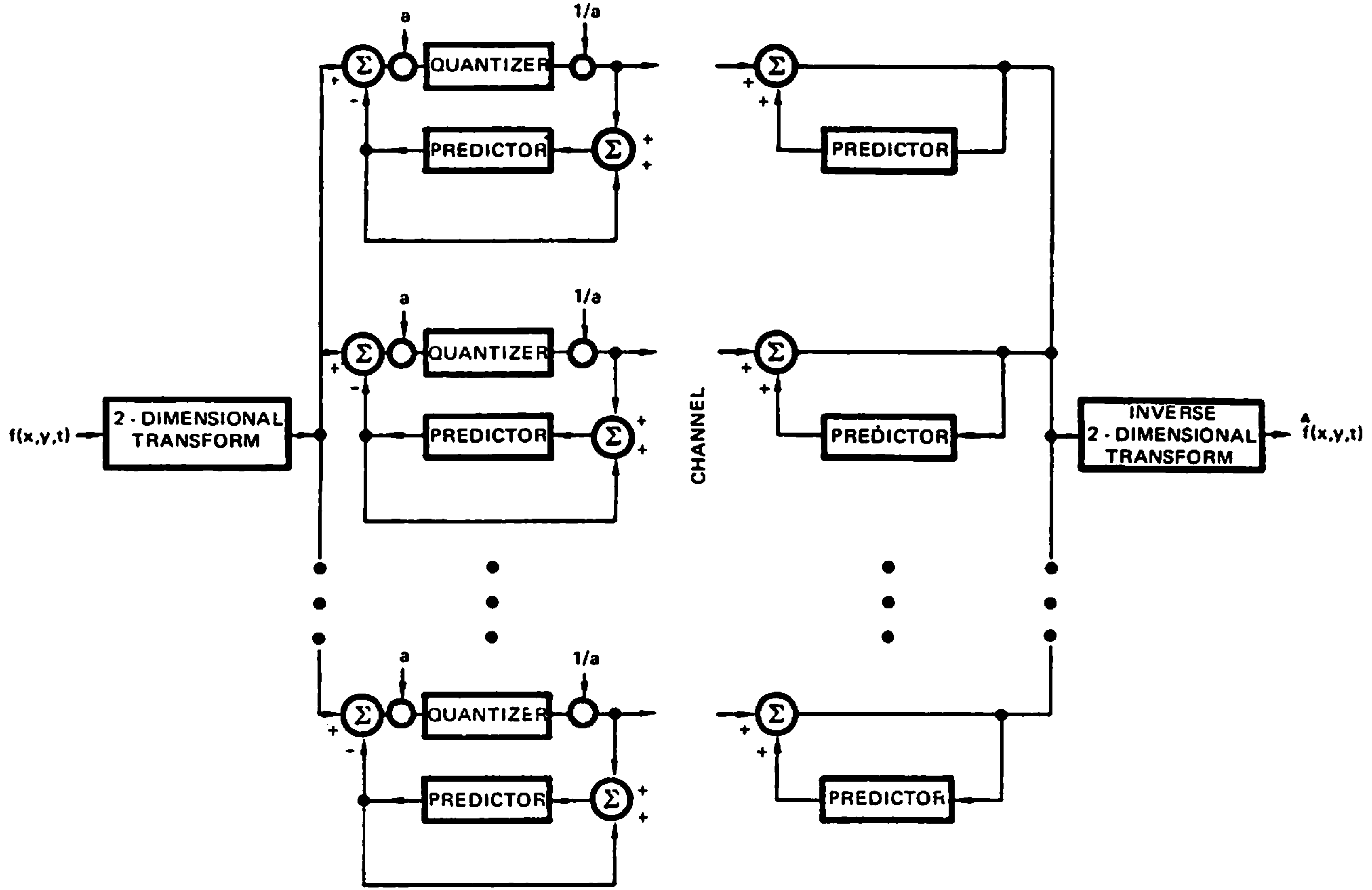


Figure 3. 3-1. HYBRID (TWO-DIMENSIONAL TRANSFORM)/DPCM CODER

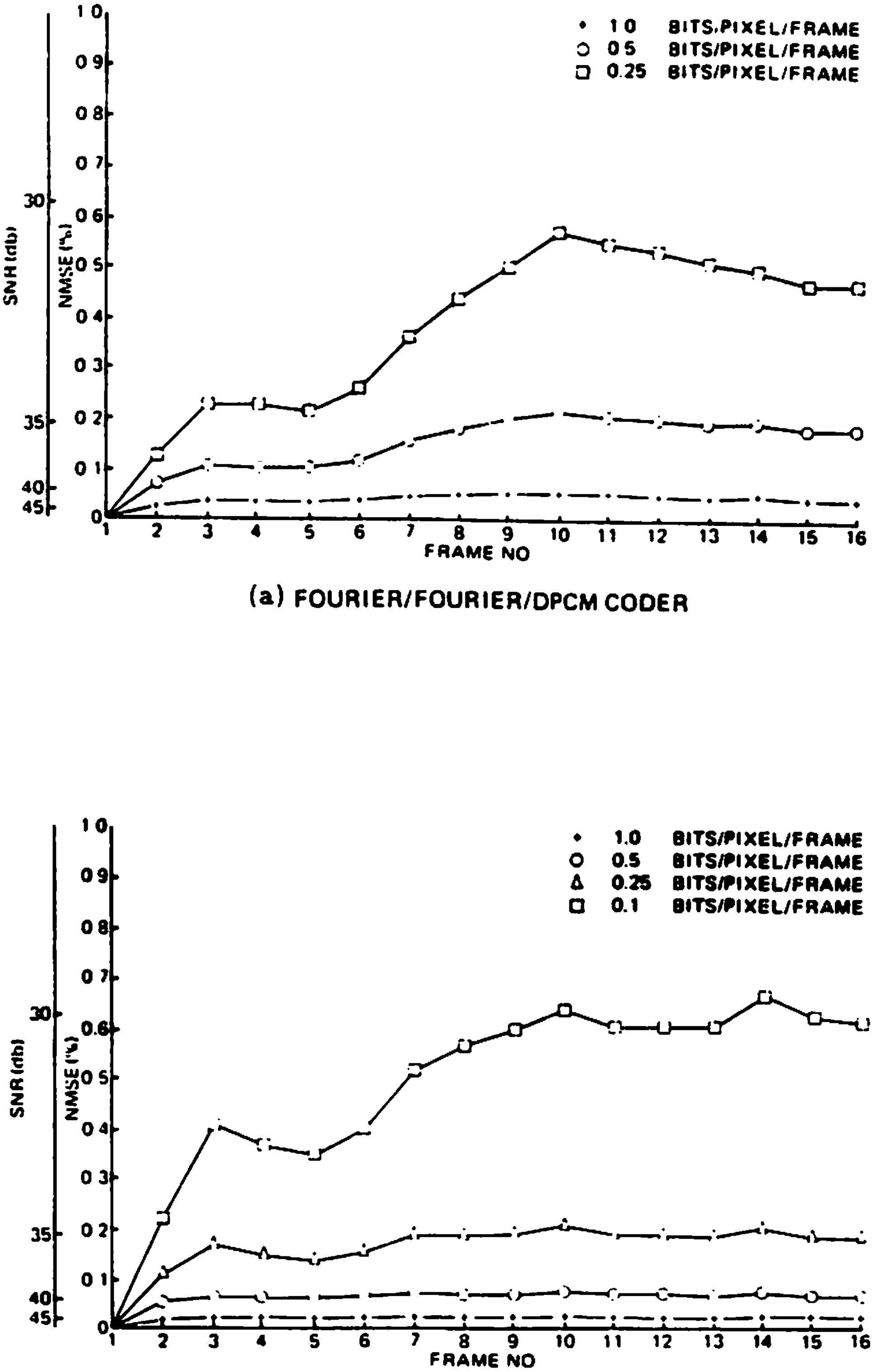
The second criterion, SNR, measures the ratio of peak-to-peak signal to RMS noise as defined by

$$\mathrm{SNR} = -10 \log_{10} \frac{\frac{1}{M^2} \sum_{x=0}^{M-1} \sum_{x=0}^{M-1} \left[f(x,y,t) - \hat{f}(x,y,t) \right]^2}{255^2} \tag{7}$$

Figures 2a and 2b are graphs illustrating the coding efficiency of the hybrid FFD and CCD coders at various bit rates in the interval 0.1 to 1.0 bits/pixel/frame. To perform this series of experiments, a 256 x 256 resolution data base consisting of 16 consecutive frames of a 24 frames per second (fps) motion picture was digitized. /Initial conditions assumed were that the first frame was available at the receiver.

Photographs of frame number 16 after coding by the FFD and CCD coders at average pixel bit rates of 1.0, 0.5, 0.25, and 0.1 are shown as figures 3 and 4. The results shown in figure 3 for the FFD coder were obtained by coding the real and imaginary components of the Fourier coefficients by assigning half of the available bits to each component.

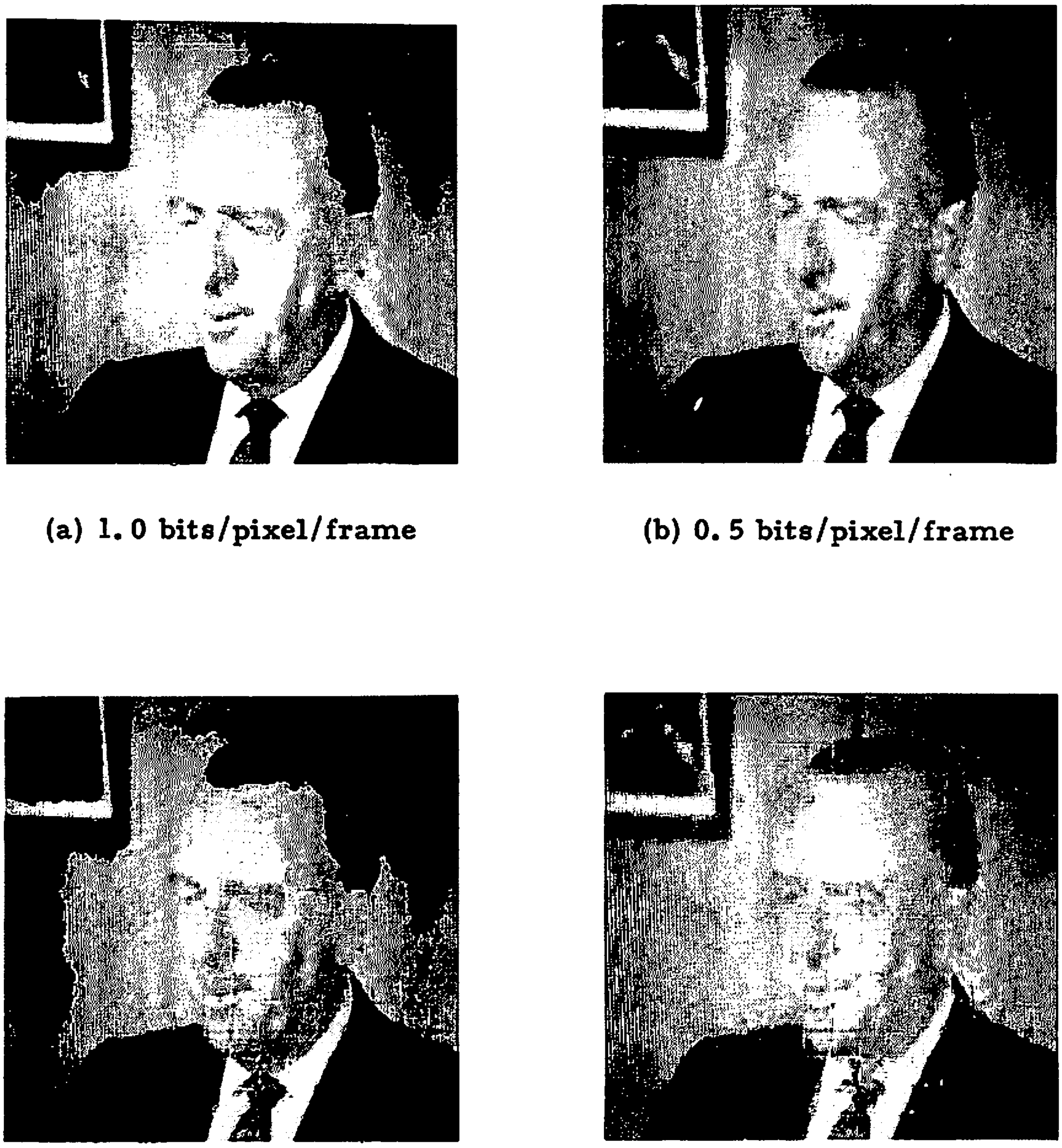
<u>Noise Immunity</u>: Performance of the FFD and CCD hybrid interframe coders was investigated in the presence of channel noise. In order to study the effect of channel noise, a binary symmetric channel was simulated. The channel is assumed to operate on each binary digit independently, changing each digit from 0 to 1 or from 1 to 0 with



(a) FOURIER/FOURIER/DPCM CODER

(b) COSINE/COSINE/DPCM CODER

Figure 3.3-2. Error performance of hybrid coders



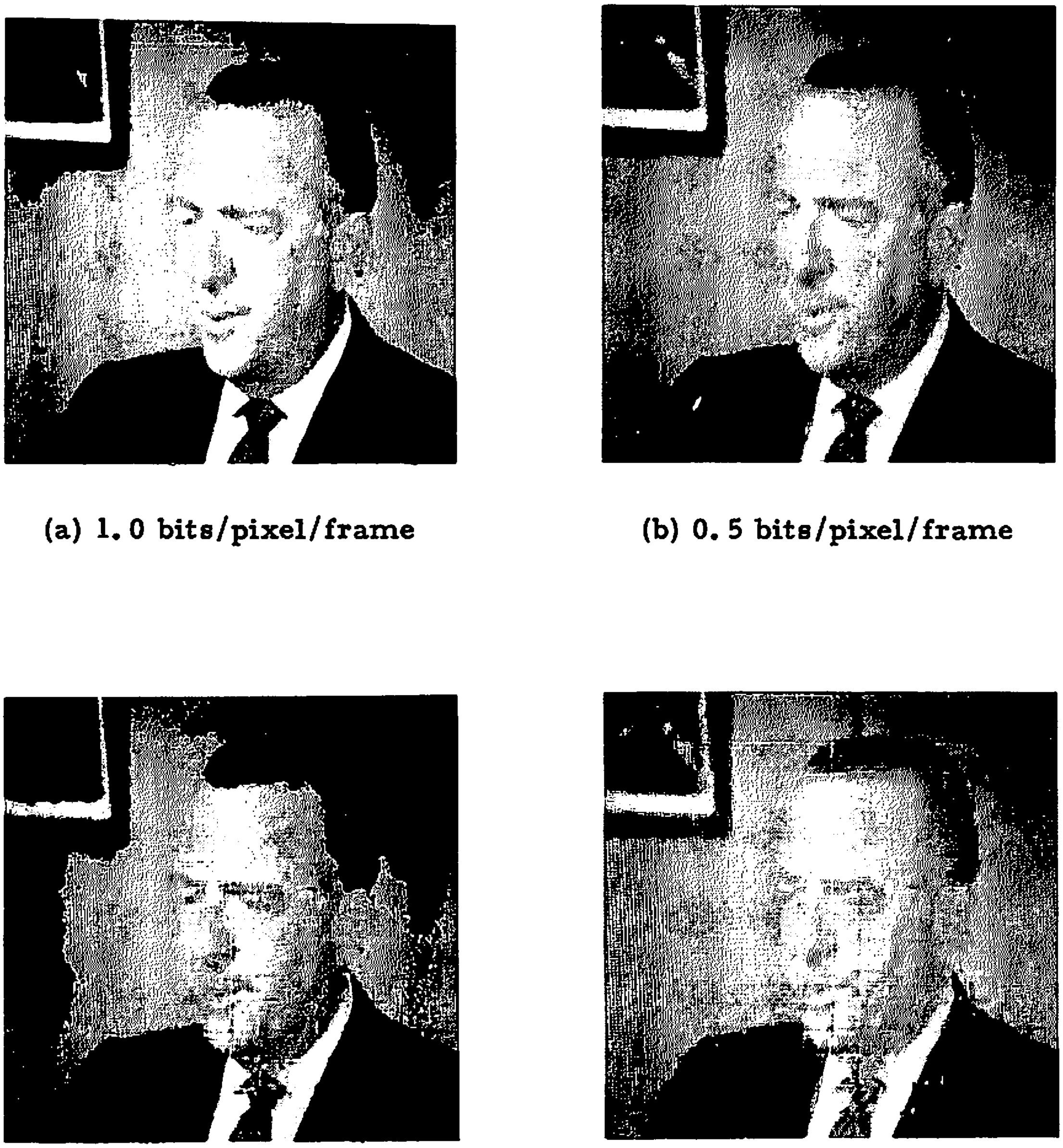
(a) 1.0 bits/pixel/frame

(b) 0.5 bits/pixel/frame

(c) 0.25 bits/pixel/frame

(d) 0.1 bits/pixel/frame

Figure 3.3-3. FFD coder for frame 16.



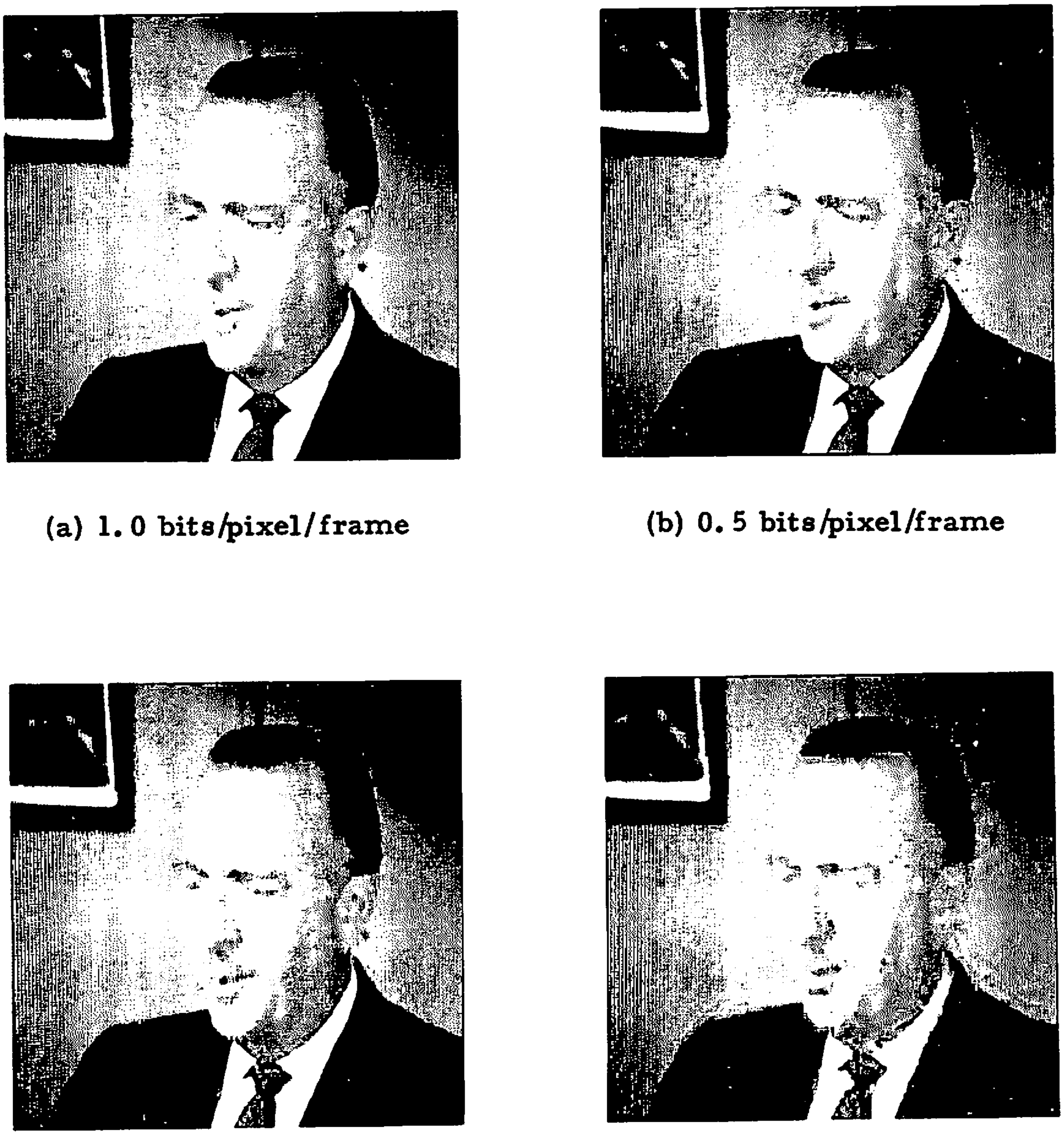
(a) 1.0 bits/pixel/frame

(b) 0.5 bits/pixel/frame

(c) 0.25 bits/pixel/frame

(d) 0.1 bits/pixel/frame

Figure 3.3-3. FFD coder for frame 16.



(a) 1.0 bits/pixel/frame

(b) 0.5 bits/pixel/frame

(c) 0.25 bits/pixel/frame

(d) 0.1 bits/pixel/frame

Figure 3.3-4. CCD coder for frame 16.

probability p and leaving the digit unchanged with probability 1-p. At the receiver, the encoded picture is reconstructed from the string of binary digits, including errors, transmitted across the channel.

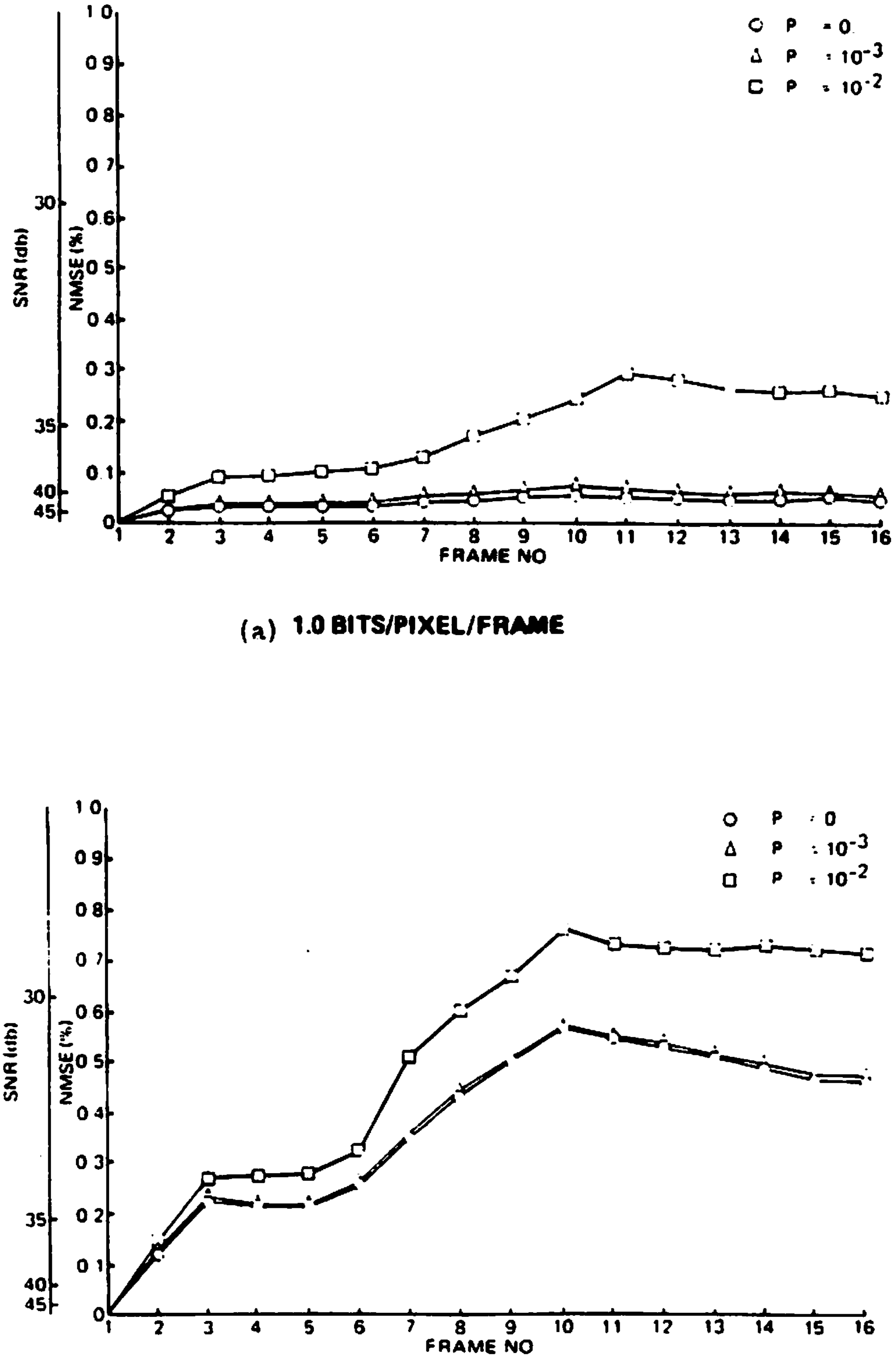
Degradations due to channel noise probabilities, p of zero, 10^{-3} and 10^{-2} for the FFD and CCD coders at average bit rates of 1.0 and 0.25 bits/pixel/frame are shown in figures 5 and 6. The generally monotonically increasing character cf these curves illustrates the fact that once an error has occurred, it tends to propagate in the temporal direction until corrected by a frame refresh.

Resulting pictures shcw that, fcr both coder implementations studied, minimal image degradaticn occurred for channel error probability cf 10^{-3} cr less.

Photographs corresponding to average bit rates of 1.0 and 0.25 bits/pixel/frame for the FFD and CCD coders with channel error probabilities of 10^{-3} and $1C^{-2}$ are shown in figures 7 and 8.

Bit Transfer Rate: In keeping with the previously mentioned objective of minimizing the number of bits transmitted while retaining image fidelity, a series of experiments was performed in which certain bit transfer rates (BTR) across the channel were fixed. The BTR is defined as the prcduct of average pixel bit rate per frame and frame rate and has units of bits/pixel/sec.

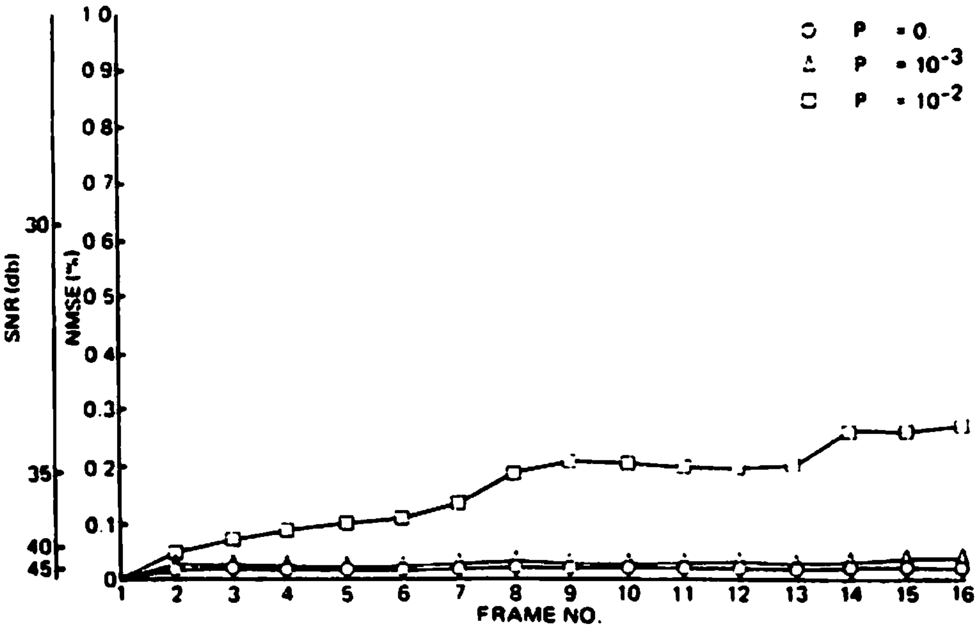
The available 16 frame test data base was extracted frcm a 24 fps motion picture sequence. By emplcying frame skipping techniques,



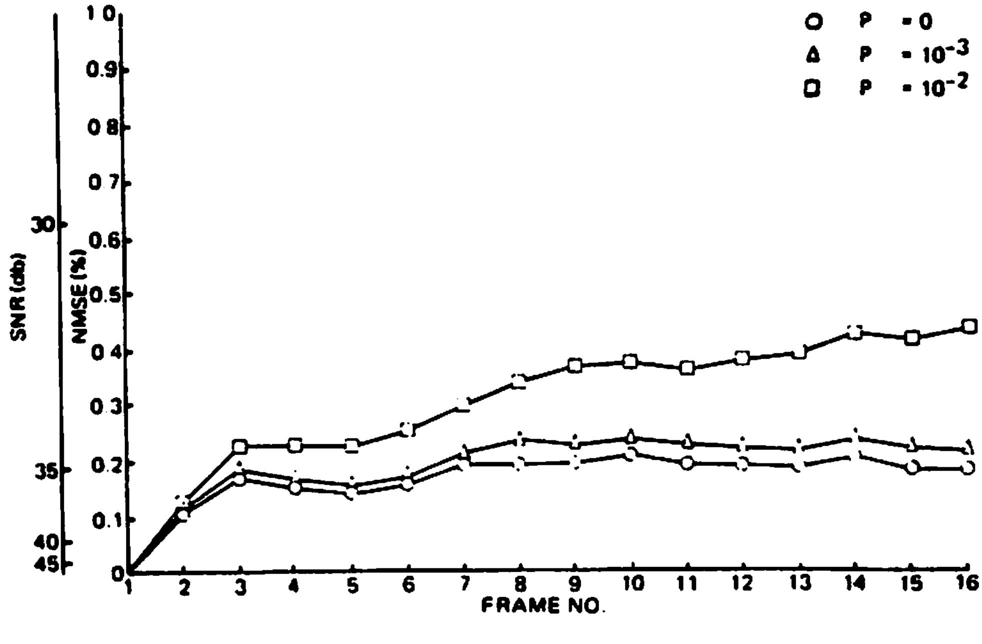
(a) 1.0 BITS/PIXEL/FRAME

(b) 0.25 BITS/PIXEL/FRAME

Figure 3.3-5. Effects of Channel Noise For Fourier/Fourier/DPCM Coder

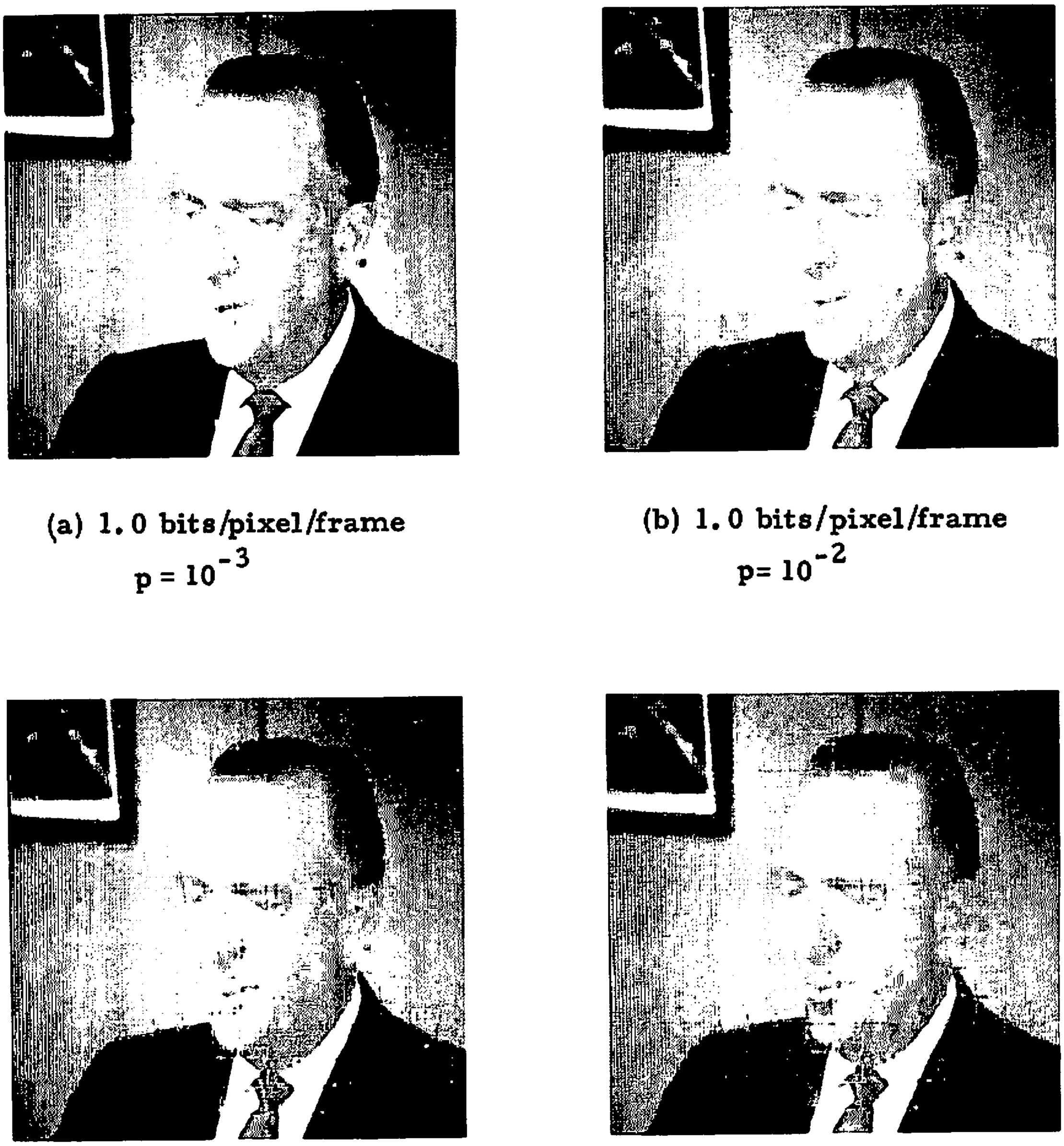


(a) 1.0 BITS/PIXEL/FRAME



(b) 0.25 BITS/PIXEL/FRAME

Figure 3.3-6. Effects of Channel Noise For Cosine/Cosine/DPCM Coder



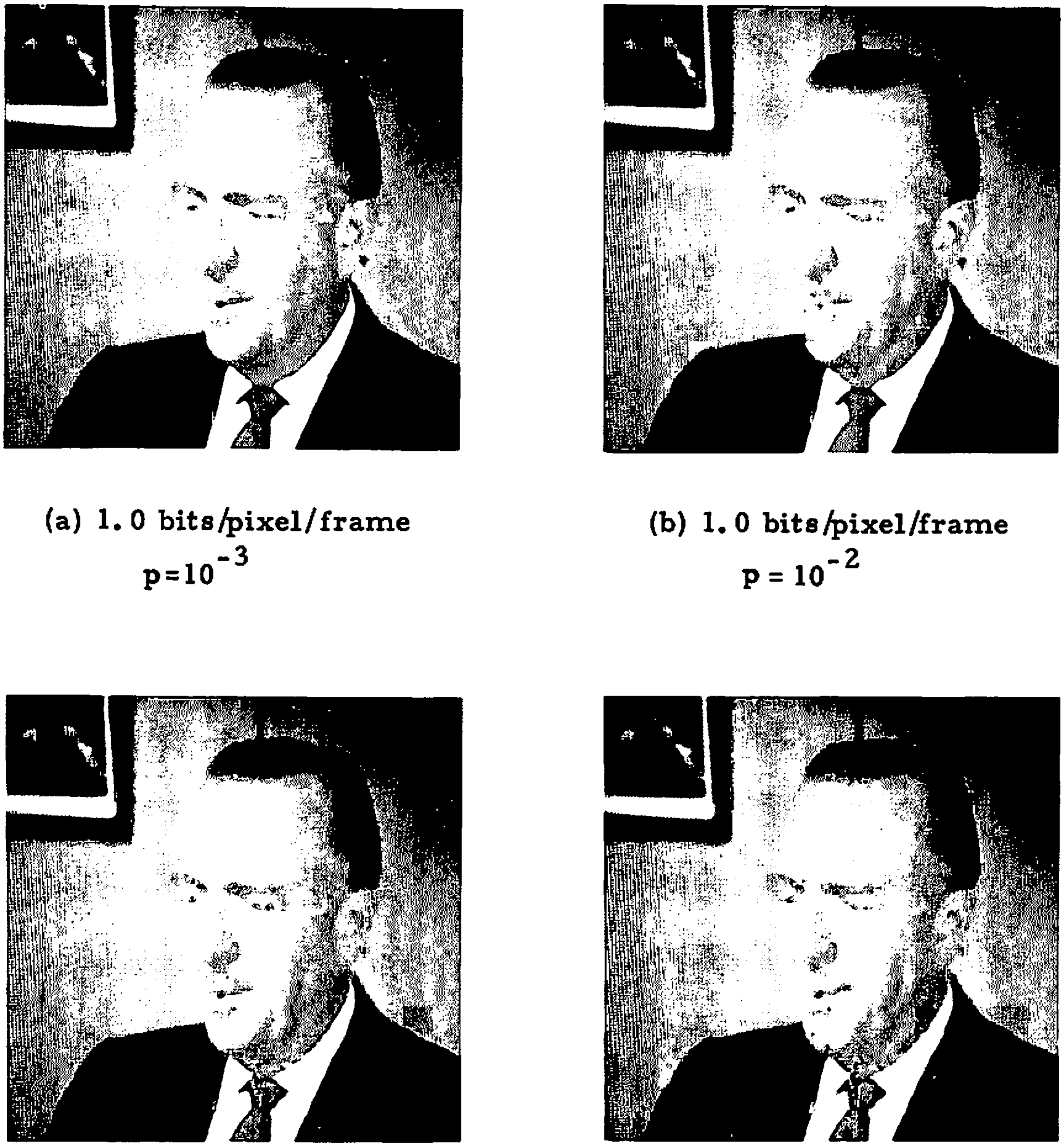
(a) 1.0 bits/pixel/frame
$p = 10^{-3}$

(b) 1.0 bits/pixel/frame
$p= 10^{-2}$

(c) 0.25 bits/pixel/frame
$p = 10^{-3}$

(d) 0.25 bits/pixel/frame
$p = 10^{-2}$

Figure 3.3-7. FFD coder with channel noise.



(a) 1.0 bits/pixel/frame $p=10^{-3}$

(b) 1.0 bits/pixel/frame $p = 10^{-2}$

(c) 0.25 bits/pixel/frame $p = 10^{-3}$

(d) 0.25 bits/pixel/frame $p= 10^{-2}$

Figure 3.3-8. CCD coder with channel noise.

temporal subsampling was used to simulate short 12, 8 and 6 fps sequences from the 16 frame test data base.

Average bit rates in the interval 0.083 to 1.333 bits/pixel/frame were used in conjunction with the four frame rates mentioned above to perform simulations with BTR values of 8, 6, 4 and 2 bits/pixel/sec. Results of these experiments for 4 bits/pixel/sec are shown in figure 9. For all cases examined, the graphs show that reduced frame rates produce smaller NMSE values for the individual frames coded. This indicates that reductions experienced in frame-to-frame correlations due to temporal subsampling are completely compensated for by the increased number of bits available for coding. However, subjectively, reduced frame rates tend to result in jerky subject motion. This is most apparent for rapidly moving objects in the field of view and is of lesser consequence for slowly changing scenes.

Conclusions: Based on theoretical and experimental results obtained to date, two main conclusions have been reached. The first is that exploitation of temporal correlations in addition to spatial correlations has been demonstrated to be a viable technique for coding sequences of digital images. This fact is demonstrated by a comparison of the average bit rates required for the interframe cosine/cosine/DPCM and the existing intraframe cosine/DPCM coders to achieve the same level of NMSE performance. The sixteenth frame of the test data base was chosen for comparison and was coded at an average 0.25 bits/pixel by the interframe cosine/cosine/DPCM coder. When using the intraframe cosine/DPCM coder, it was necessary to code this frame at a bit rate of more than 2 bits/pixel to achieve the same NMSE.

The second conclusion is that the perfcrmance of the hybrid interframe coders investigated are heavily dependent upon the type of motion. In the case of the 16 frame head and shoulders test data base, good coding performance was achieved since subject movement was restricted tc a relatively small portion of the image. However, coding performance with a different aerial data base was degraded from the previous case due principally to camera platform motion which caused frame-to-frame pixel amplitude variations across the entire image. Since the performance of the hybrid interframe coders is dependent on temporal correlation, a reduced level of performance is to be anticipated for image sequences distorted by motion.

References

1. A.G. Tescher, "The Role of Phase in Adaptive Image Coding," Ph.D. Thesis, University of Southern California, Electrical Engineering Departmen, January 1974. Published as Report 510, University of Southern California, Image Processing Institute.

2. J.A. Roese, W.K. Pratt, G.S. Robinson and A. Habibi, "Interframe Transform Coding and Predictive Coding Methods," Proceedings of 1975 International Conference on Communications (ICC 75), Vol. II, Paper 23, pp. 17-21, June 16-18, 1975.

3. G.S. Robinscn, "Orthogcnal Transform Feasibility Study," NASA

Final Report NASA-CR-115314, N72-13143 (176 pages) (submitted by COMSAT Laboratories to NASA Manned Spacecraft Center, Houston, Texas) November 1971.

4. A.J. Seyler, "Probability distributions of television frame differences," Proceedings IREE, Australia, pp. 355-366, November 1965.

5. B. Smith, "Instantaneous companding of quantized signals," Bell System Technical Journal, Vol. 36, pp. 653-709, May 1957.

4. Image Restoration and Enhancement Projects

Image restoration and image enhancement are two classifications of image improvement methods. Image restoration techniques seek to reconstruct or recreate an image to the form it would have had if it had not been degraded by some physical imaging system. Image enhancement techniques have two major purposes: improvement in the visual quality of a picture to a human viewer: and manipulation of a picture for more efficient processing and data extraction by a machine. Research in both areas during the past six months is described below.

4.1 Eigenvectors of Space-Variant Point Spread Function Systems

Harry C. Andrews

In image restoration systems a linear model results in an object $\underline{f}$ being mapped into an image $\underline{g}$ by a point spread function matrix $\underline{H}$. Thus with noise

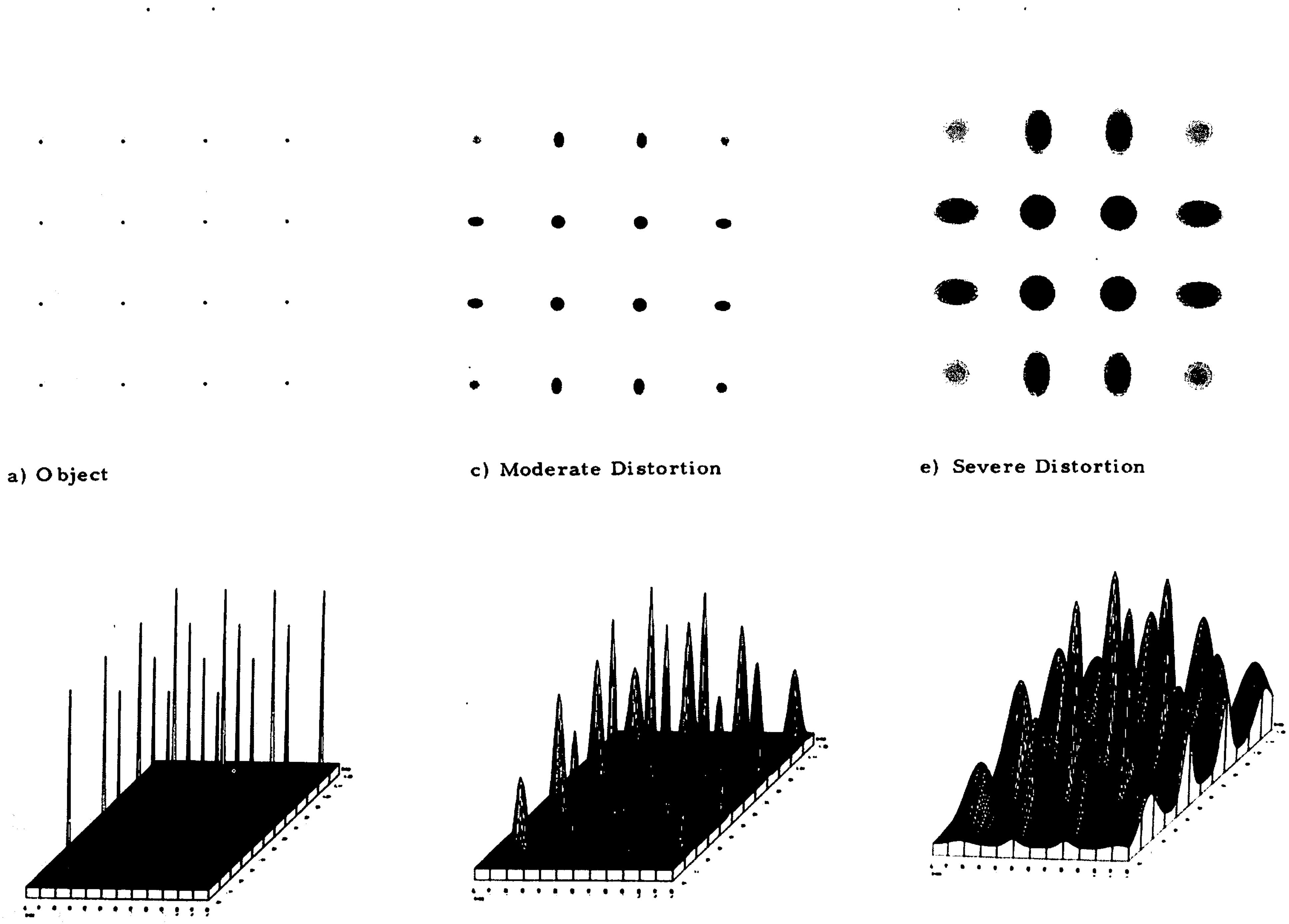
$$\underline{g} + \underline{H}\,\underline{f} + \underline{n} \qquad (1)$$

The simplest linear models for imaging systems are given by space invariant point spread functions (SIPSF) in which case $\underline{H}$ is block circulant. If the linear model is not space invariant, $\underline{H}$ then represents a space variant point spread function (SVPSF). In the case of separable systems eq.(1) becomes

$$\underline{G} = \underline{A}\,\underline{F}\,\underline{B} + \underline{N} \qquad (2)$$

where $\underline{A}$ represents the column blur and $\underline{B}$ represents row blur on the object $\underline{F}$. In the SIPSF case $\underline{A}$ and $\underline{B}$ are circulants, but for the SVPSF case $\underline{A}$ and $\underline{B}$ may have very general structure. It is interesting to investigate the eigenvectors of such systems to get a better feel for the underlying eigenspace of the distortions representing such systems. In the case of SIPSF systems, the eigenvectors are sine and cosine waveforms and the eigenspace of such distortions are given by the Fourier transform. In the SVPSF situation, the eigenvectors often turn out to be variations on sines and cosines depending on how variant the blur actually is.

To illustrate this point a separable (SVPSF) system has been simulated for two degrees of blur (moderate and severe). Figure 1 illustrates this situation in which 16 point sources experience spatially variant degradations. The imaging systems are separable and are in better focus in the center and get more blurred toward the edges. Figures 2 and 3 present selected eigenvectors for both the moderate and severe distortion cases. As the eigenvector index increases, the eigenvectors experience an increasing number of zero crossings similar to sine and cosine functions. Also note that the first eigenvector has no zero crossings and is not a constant. These SVPSF eigenvectors appear to be FM modulated trigonometric waveforms. It is interesting to conjecture that as a function of the decreasing variant nature of the blur involved, these eigenvectors will converge to unmodulated trigonometric functions. In examining figures 2 and 3,



a) Object

c) Moderate Distortion

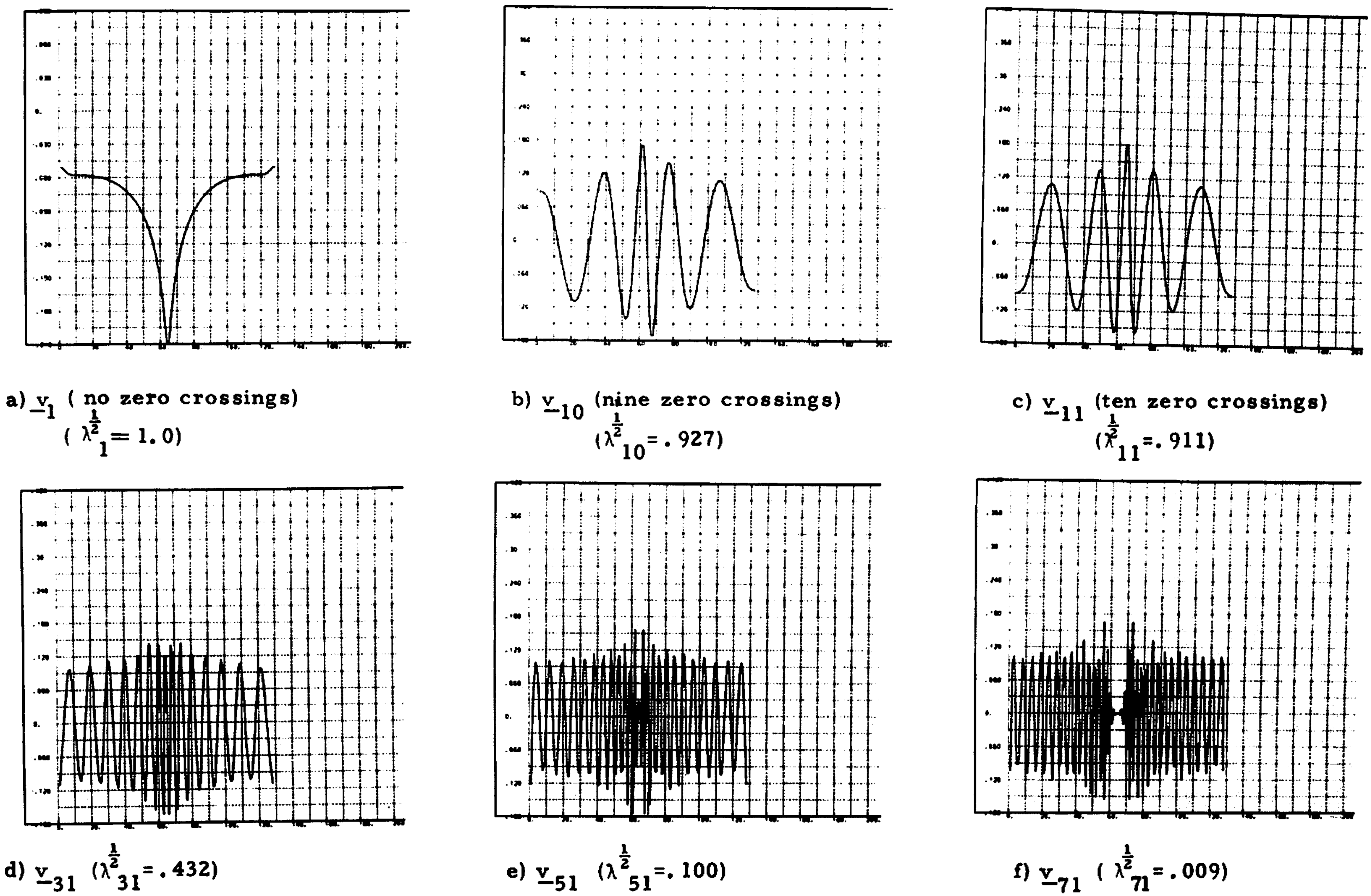
e) Severe Distortion

b) Object Perspective

d) Moderate Distortion Perspective

f) Severe Distortion Perspective

Figure 4.1-1. Point source simulations



a) $\underline{v}_1$ (no zero crossings) $(\lambda^{\frac{1}{2}}_1 = 1.0)$

b) $\underline{v}_{10}$ (nine zero crossings) $(\lambda^{\frac{1}{2}}_{10} = .927)$

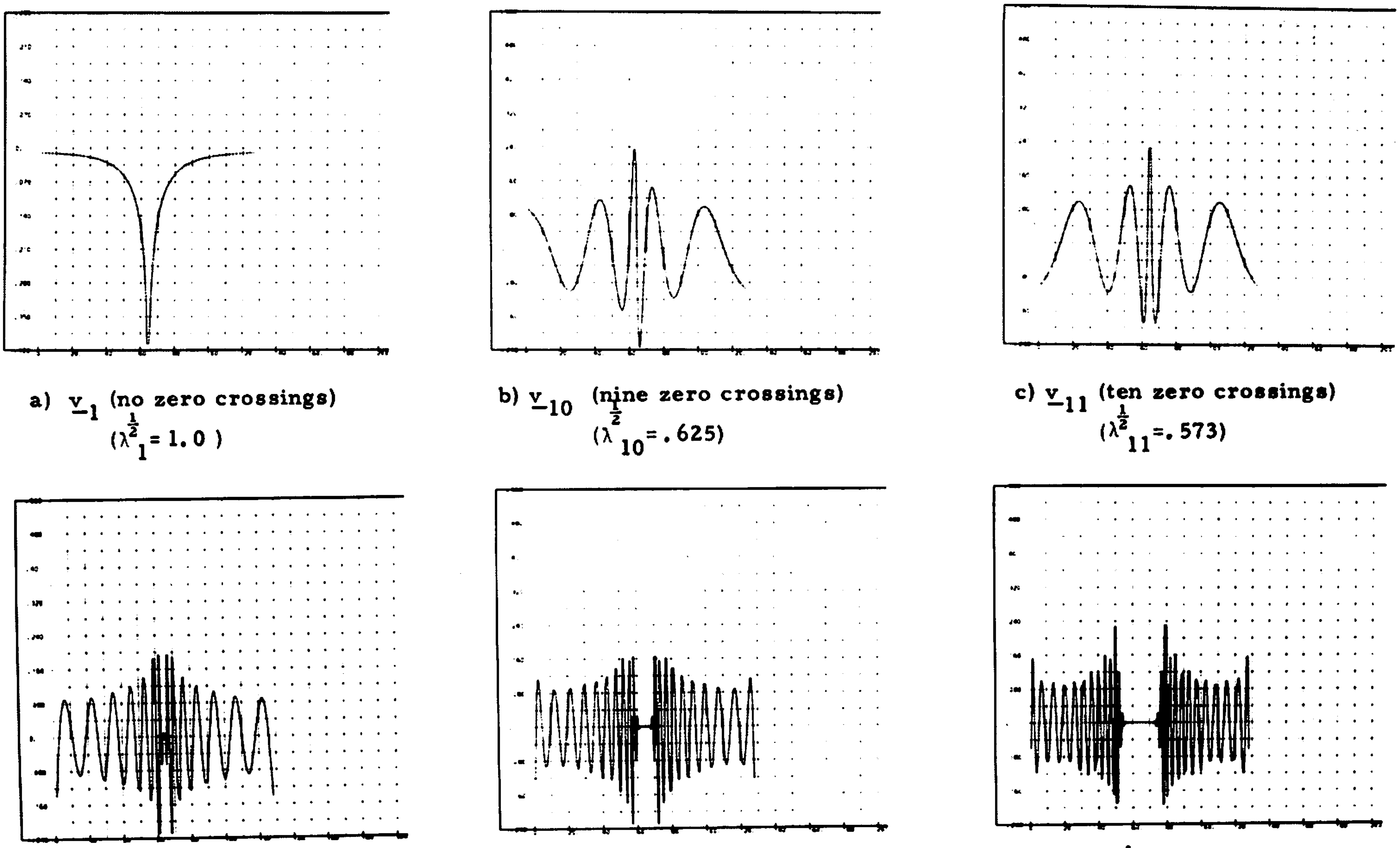
c) $\underline{v}_{11}$ (ten zero crossings) $(\lambda^{\frac{1}{2}}_{11} = .911)$

d) $\underline{v}_{31}$ $(\lambda^{\frac{1}{2}}_{31} = .432)$

e) $\underline{v}_{51}$ $(\lambda^{\frac{1}{2}}_{51} = .100)$

f) $\underline{v}_{71}$ $(\lambda^{\frac{1}{2}}_{71} = .009)$

Figure 4.1-2. Selected eigenvectors for moderate distortion



a) $\underline{v}_1$ (no zero crossings) ($\lambda_1^{\frac{1}{2}} = 1.0$)

b) $\underline{v}_{10}$ (nine zero crossings) ($\lambda_{10}^{\frac{1}{2}} = .625$)

c) $\underline{v}_{11}$ (ten zero crossings) ($\lambda_{11}^{\frac{1}{2}} = .573$)

d) $\underline{v}_{31}$ ($\lambda_{31}^{\frac{1}{2}} = .014$)

e) $\underline{v}_{51}$ ($\lambda_{51}^{\frac{1}{2}} = 1.0\mathrm{x}10^{-4}$)

f) $\underline{v}_{71}$ ($\lambda_{71}^{\frac{1}{2}} = 1.0\mathrm{x}10^{-9}$)

Figure 4.1-3. Selected eigenvectors for severe distortion

it is interesting to nota the effort each eigenvector goes to in order to resolve finer detail at certain points along the axis compared to other positions. Also note the eigenvectors effectively go to zero at higher indices in the center of the axis indicating they have no effect on the restoration here.

4.2 Least Squares Restoration for the Continuous-Discrete Model

Steve Hou

For image restoration purposes, a realistic model is that given by the continuous-discrete model defined by

$$\underline{g} = \iint \underline{h}(\epsilon,\eta)f(\epsilon,\eta)d\epsilon d\eta \tag{1}$$

where a discrete image $\underline{g}$ is obtained from a possibly space variant imaging system, described by $\underline{h}(\epsilon,\eta)$, observing a continuous object $f(\epsilon,\eta)$. In digitally restoring such a model only a finite number of samples are available for description of the estimate $\hat{f}(\epsilon,\eta)$ of the object. Using cubic spline interpolation

$$\hat{f}(\epsilon,\eta) = \sum_i \sum_j C_{ij} S_i(\epsilon) S_j(\eta) \tag{2}$$

where $S_i(\lambda)$ is the i th cubic spline centered at ϵ_i. An objective function for restoration with a smoothness constraint is given by

$$W(\hat{f}) = \|\underline{g}-\hat{\underline{g}}\| + \gamma \iint_{-\infty}^{\infty} [\hat{f}''(\epsilon,\eta)]^2 d\epsilon\, d\eta \tag{3}$$

where

$$\hat{\underline{g}} = \iint \underline{h}(\epsilon,\eta)\hat{f}(\epsilon,\eta)\, d\epsilon\, d\eta \tag{4}$$

By differentiaticn and subsequent manipulation, the systemization equation result is

$$[\underline{P}^T\underline{P}+\gamma\underline{B}]\underline{C} = \underline{P}^T\underline{g} \tag{5}$$

Here

$$\underline{P} = \iint \underline{h}(\epsilon,\eta)\underline{s}^T(\epsilon,\eta)\, d\epsilon\, d\eta \tag{6a}$$

$$\underline{s}^T(\epsilon,\eta) = \underline{s}^T(\epsilon) \otimes \underline{s}^T(\eta) \tag{6b}$$

$$\underline{C} = \underline{C}_i \otimes \underline{C}_j \tag{6c}$$

$$\underline{B} = \int_{-\infty}^{\infty} \underline{s}''(\epsilon,\eta)\underline{s}''^T(\epsilon,\eta)\, d\epsilon\, d\eta \tag{6d}$$

Equation (5) is known as the normal equation.

The method of ccnjugate gradients has been used to iteratively search for the solution in eq.(5). Because of computer limitations, a separable point spread function has been assumed, for both space variant and invariant systems. For the separable formulation, the normal equation becomes

$$[\underline{A}^T\underline{A}+\gamma\,\underline{B}_x\otimes\underline{B}_y]\,\underline{C}=\underline{A}^T\underline{g} \qquad (7)$$

where

$$\underline{A}=\iint \underline{h}_{ij}(\epsilon,\eta)s_k(\epsilon)s_l(\eta)\,d\epsilon\,d\eta \qquad (8a)$$

and

$$\underline{B}_x=\int_{-\infty}^{\infty}\underline{s}_k''(\epsilon)\,\underline{s}_m''^{T}(\epsilon)\,d\epsilon \qquad (8b)$$

$$\underline{B}_y=\int_{-\infty}^{\infty}\underline{s}_l''(\eta)\,\underline{s}_n''^{T}(\eta)\,d\eta \qquad (8c)$$

Using this formulation, the generalized extrapolated Jacobi iterative formulation is given by

$$\underline{C}^{(i+1)}=\underline{C}^{1}+\alpha^{-1}\left[\overset{0}{B}_x^{-1}\otimes\overset{0}{\underline{B}}_y^{-1}\right]\left[\underline{A}^T\underline{g}-\underline{A}^T\underline{A}+\gamma(\underline{B}_x\otimes\underline{B}_y)\,\underline{C}^{(i)}\right] \qquad (9)$$

where $\overset{0}{B}_x$ is defined as B_x except that no derivatives are taken of the spline functions in eqs. (8b) and (8c). The advantage of the formulation in eq.(9) is that no large matrix inverses need be taken.

A conjugate gradient algorithm has been implemented for both space variant and invariant cases. The blur impulse response is given by

$$\eta_{ij}(\epsilon,\eta)=h_i(\epsilon)h_j(\eta) \tag{10a}$$

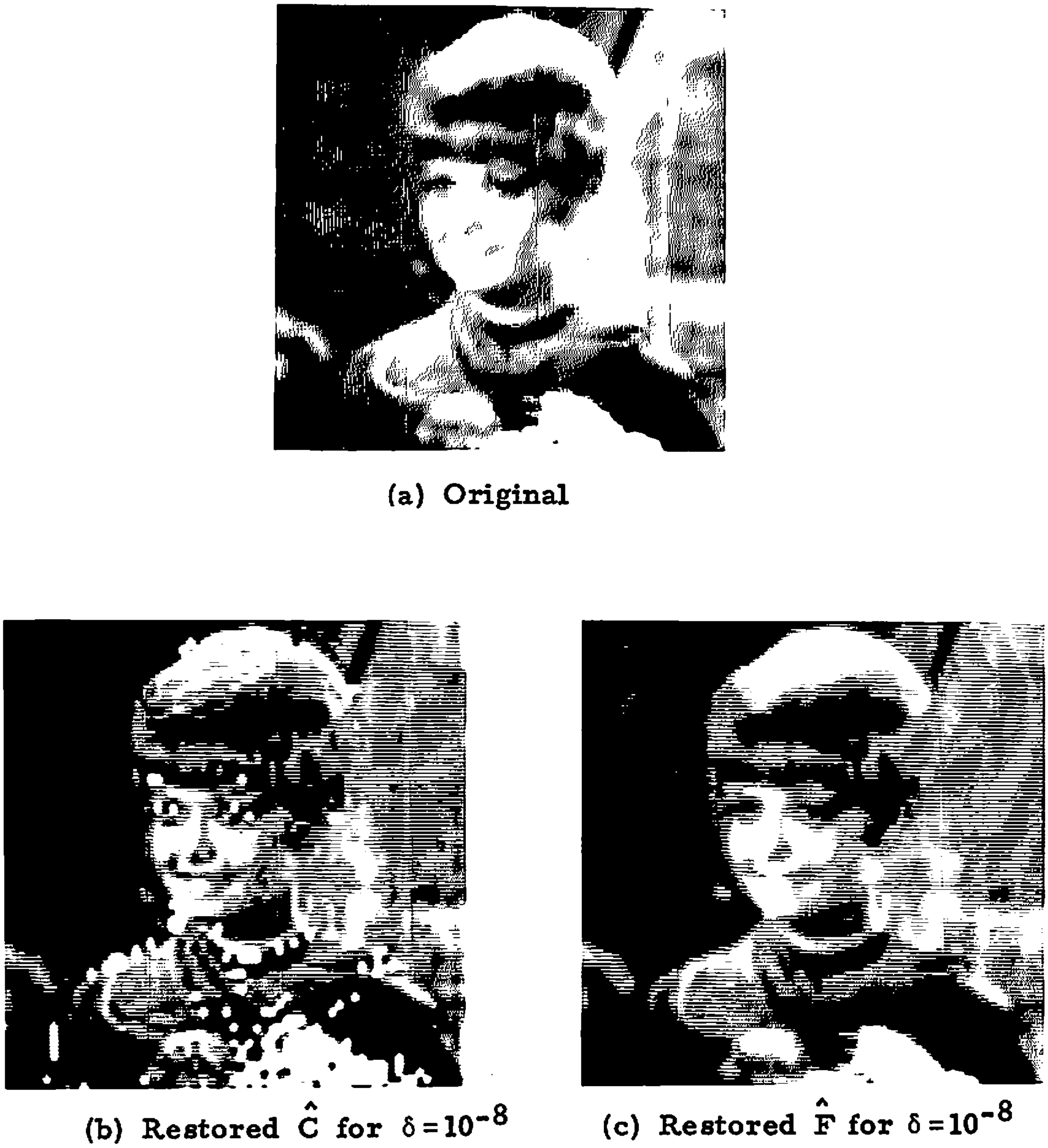
where

$$h_i(\epsilon) = k \exp\left[\frac{(\epsilon - x_i)^2}{\sigma_i}\right] \tag{10b}$$

and $\sigma_i = |kx_i|$ such that k gcverns the amount of blur or spread as a function of pcsition (x_i) in the imaging plane. A similar equation results for $h_j(\eta)$. For the space invariant case σ_i was set equal to k without x_i contributing to the spread of σ_i .

The simulated results by using the conjugate gradient algorithm are shown in figures 1 through 6. For both restorations from moderate SIPSF blur (figure 1) and from moderate SVPSF blur (figure 2), the results are strikingly good for γ =0. The justification for such results is that the PSF is fairly localized (i.e. narrow), and thus, the matrix $\underline{A}$ is well conditioned. In other words, the eigenvalues of $\underline{A}$ are clustered together sc that $\underline{A}$ is far from singularity.

On the other hand, as the PSF spreads out and the image becomes more blurred, the restored objects for both SIPSF and SVPSF are far from perfect. For x=0 ringing in separable form shows up in the SVPSF

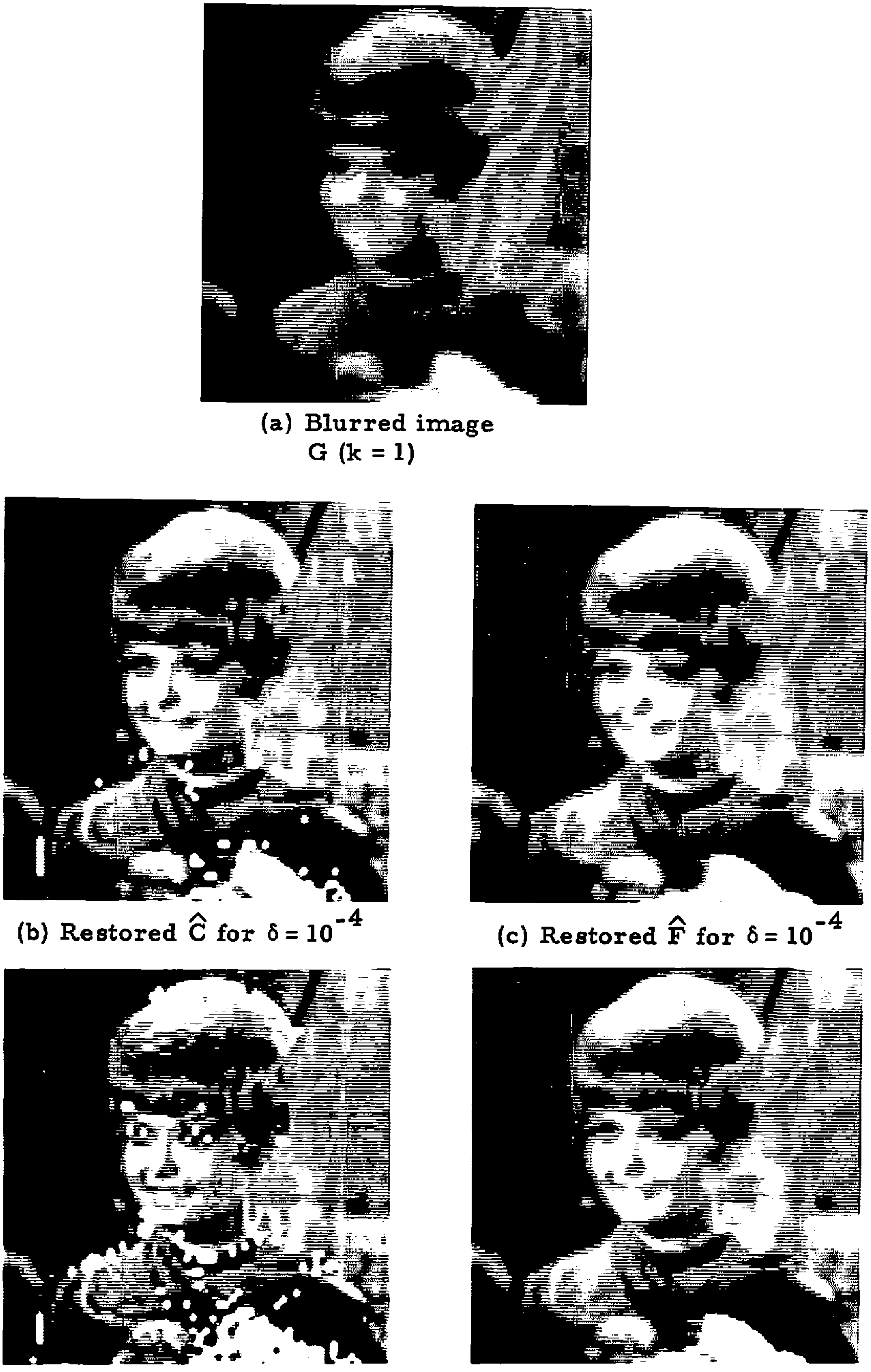


(a) Original

(b) Restored $\hat{C}$ for $\delta = 10^{-8}$

(c) Restored $\hat{F}$ for $\delta = 10^{-8}$

Figure 4.2-1. Restoration from moderate SIPSF blur (k = 1).



(a) Blurred image
G (k = 1)

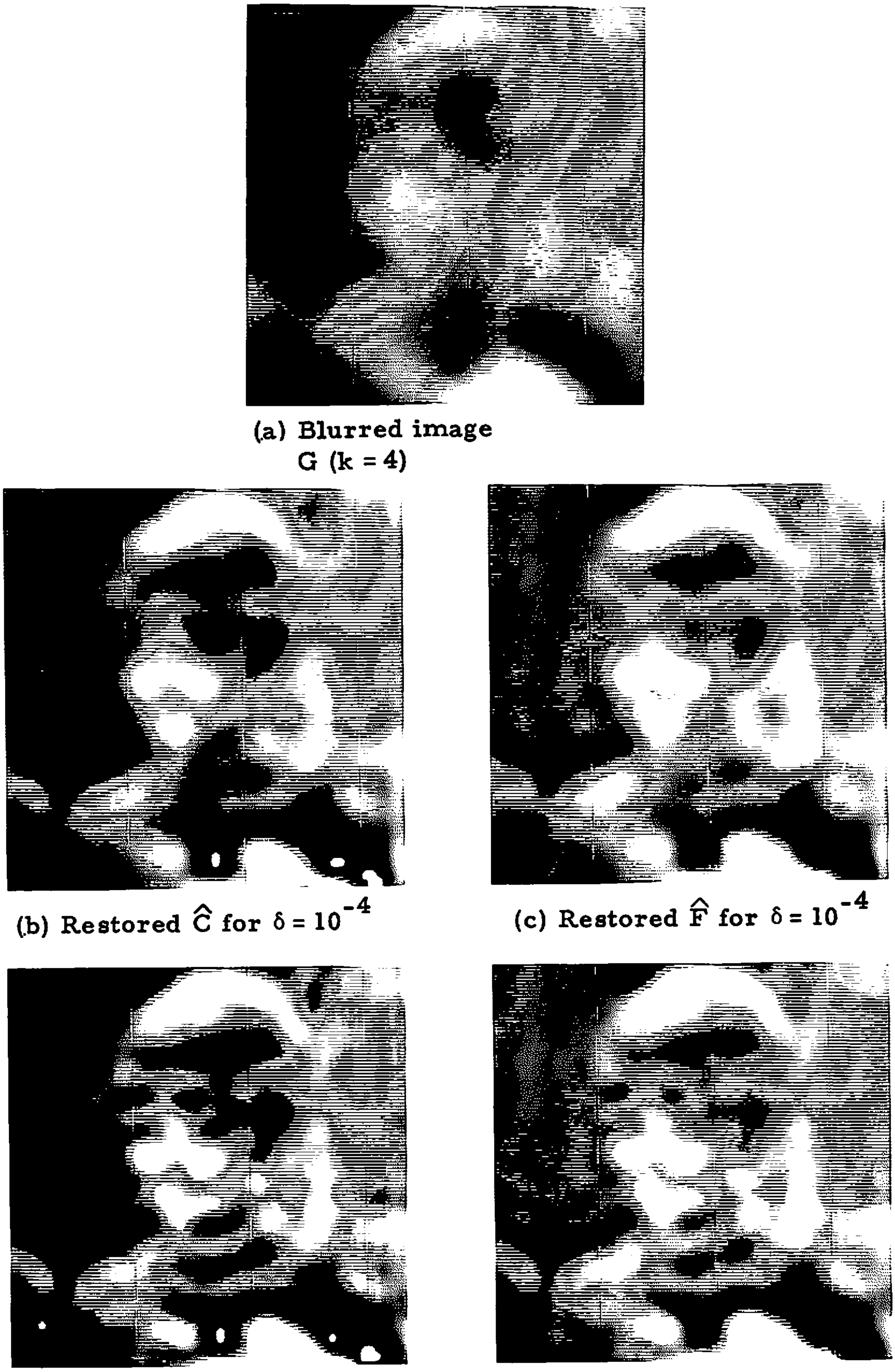
(b) Restored $\hat{C}$ for $\delta = 10^{-4}$

(c) Restored $\hat{F}$ for $\delta = 10^{-4}$

(d) Restored $\hat{C}$ for $\delta = 0$

(e) Restored $\hat{F}$ for $\delta = 0$

Figure 4.2-2. Restoration from moderate SIPSF blur (k = 1).



(a) Blurred image G (k = 4)

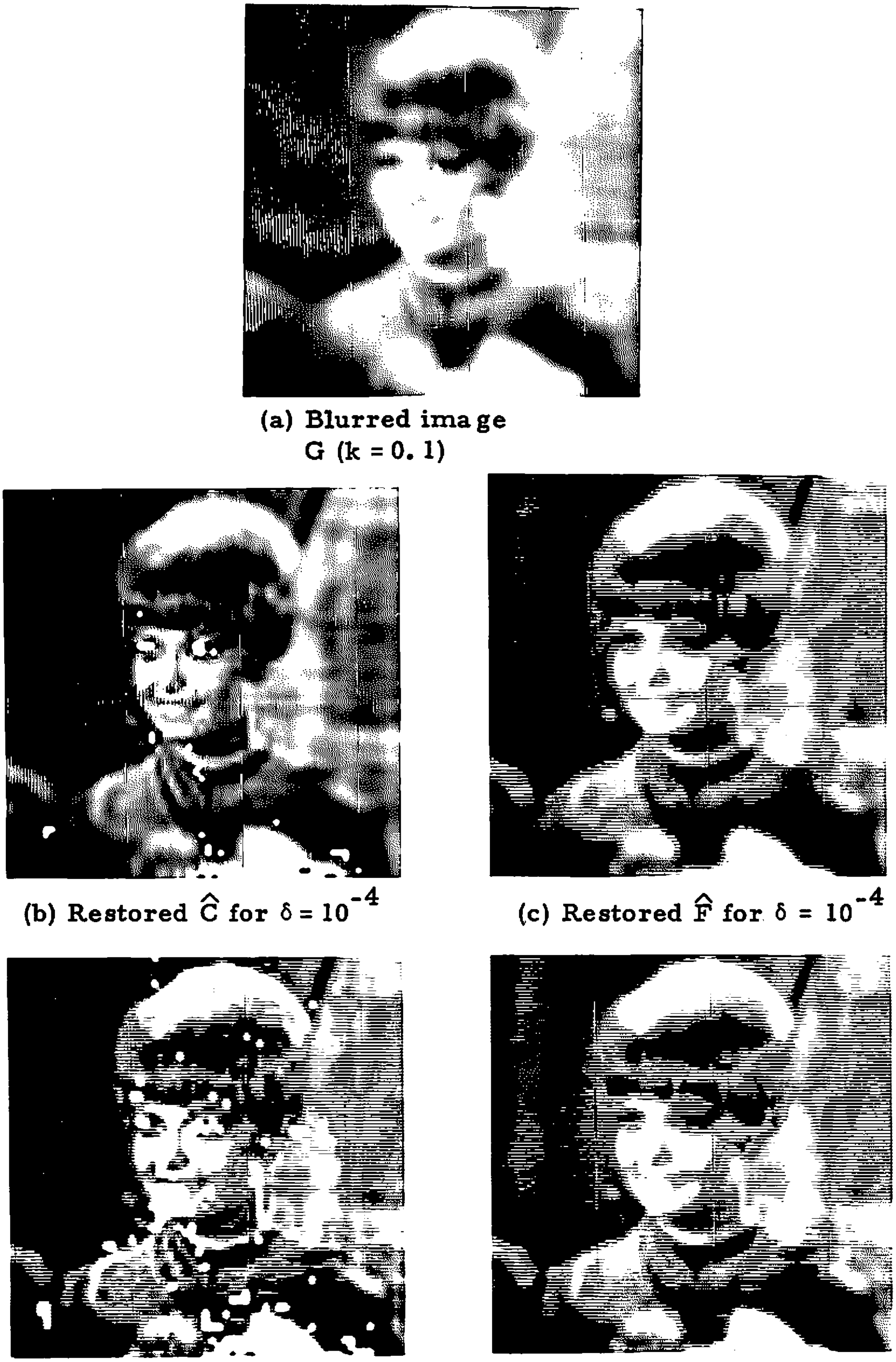
(b) Restored $\widehat{C}$ for $\delta = 10^{-4}$

(c) Restored $\widehat{F}$ for $\delta = 10^{-4}$

(d) Restored $\widehat{C}$ for $\delta = 0$

(e) Restored $\widehat{F}$ for $\delta = 0$

Figure 4.2-3 . Restoration from severe SIPSF blur (k = 4).



(a) Blurred image G (k = 0. 1)

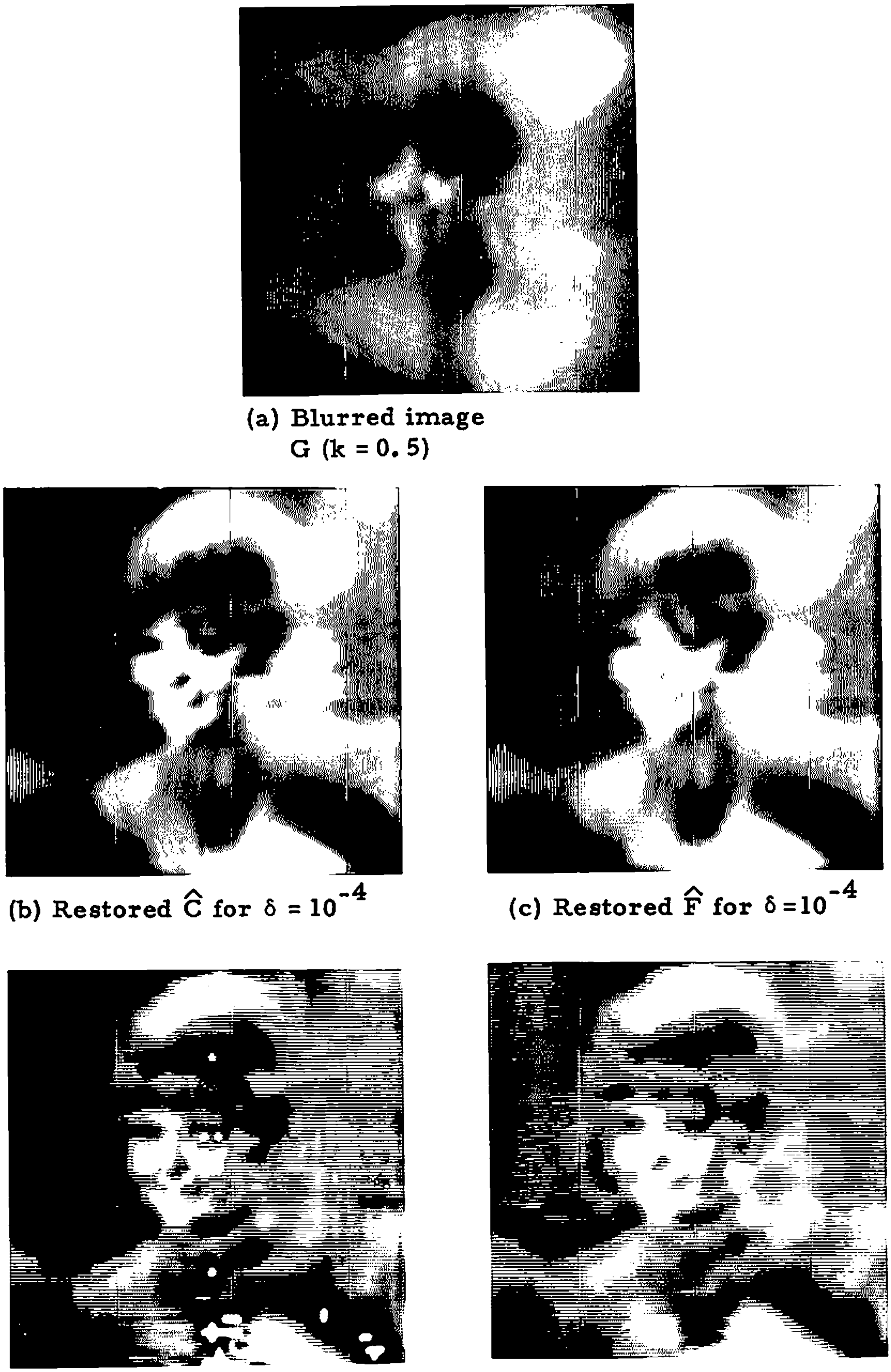
(b) Restored $\hat{C}$ for $\delta = 10^{-4}$

(c) Restored $\hat{F}$ for $\delta = 10^{-4}$

(d) Restored $\hat{C}$ for $\delta = 0$

(e) Restored $\hat{F}$ for $\delta = 0$

Figure 4.2-4. Restoration from moderate SVPSF blur (k = 0. 1).



(a) Blurred image
G (k = 0, 5)

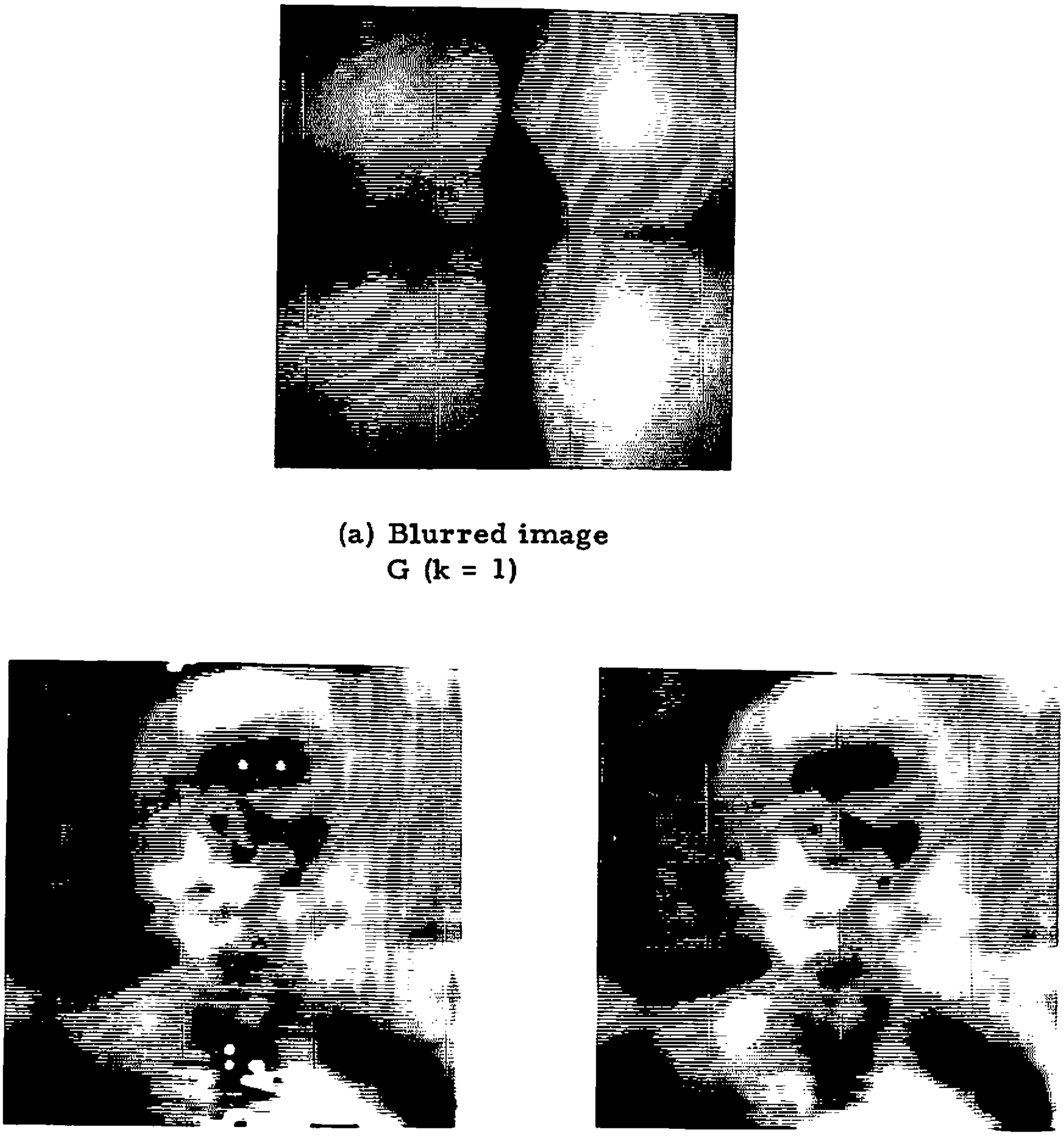
(b) Restored $\hat{C}$ for $\delta = 10^{-4}$

(c) Restored $\hat{F}$ for $\delta = 10^{-4}$

(d) Restored $\hat{C}$ for $\delta = 0$

(e) Restored $\hat{F}$ for $\delta = 0$

Figure 4.2-5. Restoration from severe SVPSF blur (k = 0, 5).



(a) Blurred image
G (k = 1)

(b) Restored $\hat{C}$ for $\delta = 0$

(c) Restored $\hat{F}$ for $\delta = 0$

Figure 4.2-6 . Restoration from very severe SVPSF blur (k = 1).

case; and the norm of the error matrix in the gradient algorithm oscillates. This is because the matrix $\underline{A}$ now is badly conditioned and approaches singularity. Theoretically, as $\gamma=0$, the conjugate gradient algorithm is the same as the pseudoinverse of $\underline{A}$. Under this condition, the ellipsoidal contour surface in the direction corresponding to zero eigenvalues shrinks, thus residual errors can no longer maintain orthogonality, and the computing time to convergence grows enormously.

As shown in figures 1 and 2, the tradeoff between the picture smoothness and sharpness which may be accompanied by oscillations becomes evident from the results for $\gamma=10^{-4}$ and $\gamma=0$ in both SIPSF and SVPSF cases. The price paid for sharp pictures is a long iteration time. Notice that in the SIPSF case, the restored object for $\gamma=10^{-8}$ is almost identical with that for $\gamma=0$. Hence, it is suspected that in the SVPSF case, the oscillation could be supressed by using $\gamma=10^{-6}$ or 10^{-7} without much impairment of the picture sharpness, but with the additional advantage of faster convergence.

The white spots appearing in all the $\hat{\underline{C}}$ pictures are the negative coefficients in the $\hat{\underline{C}}$ matrix. Because of the positive nature of the spline basis function the coefficients must have negative values in order to reconstruct $\hat{f}(\epsilon,\eta)$ properly. As expected, the white spots appear at the high contrast areas of the GIRL picture, such as along the edges of her scarf, or the flowers and in her eyes. As γ decreases, the number of white spots increases because the restored picture becomes sharper. For severely blurred images, the white spots are scarce and hence, the object is no longer sharply reconstructed.

4.3 A General Image Estimation Algorithm Applicable to Multiplicative and Non-Gaussian Noise

Nasser E. Nahi and Mohammed Naraghi

In statistical image enhancement, an image is described by a two-dimensional random process (field). These processes are often characterized by their mean and autocorrelation [3]. Denoting the image brightess function by $b(i,j)$, with i and j as the horizontal and vertical variables, the two moments are defined as

$$M(i,j) = E\{b(i,j)\} \qquad (1)$$

$$R(i,j,k,\ell) = E\{[b(i,j))-M(i,j)][b(k,\ell)-M(k,\ell)]\} \qquad (2)$$

where E is the mathematical expectation operator. The degraded image (commonly referred to as the observation) is denoted by $y(i,j)$ and specifies the functional relationship between signal, $b(i,j)$, and noise (i,j) given by

$$y(i,j) = f\{b(i,j),\gamma(i,j)\} \qquad (3)$$

where f may be nonlinear and $\gamma(i,j)$ may be vector valued.

Optimum filtering of images under the general condition of eq. (3) has received little attention. However, a variety of procedures have been developed for the special linear case, where

$$y(i,j) = b(i,j) + \gamma(i,j) \tag{4}$$

where $\gamma(i,j)$ is white and Gaussian [11 to 17]. Although, eq.(4) describes many natural forms of degradations [12 to 16], there are conceivably as many situations where this model does not apply. Examples are images with film grain noise and pictures cbserved through non-hcmogenecus clcud layers, where the noise is a random attenuation factor. In these examples, the observations take the form

$$y(i,j) = \gamma(i,j) b(i,j) \tag{5}$$

The majority of the existing linear estimation procedures require the correlaticn function $R(i,j,k,\ell)$ to be specified as an analytic function of a particular form [12 to 17]. This limits the generality of these methods since they cannot be applied to practical cases where, the function $R(i,j,k,\ell)$ is often specified numerically at only a small number of argument indices.

The purpose of this wcrk is to develop a general estimation method which requires numerical values of the autocorrelation function $R(i,j,k,\ell)$ only, and is applicable to nonlinear (as well as linear) observation systems. Furthermore, the estimation technique will be of recursive nature, and hence, computationally efficient.

Notation: An image is viewed as an $n \times n$ matrix with elements $b(i,j)$, where $b(i,j)$ is the intensity cf the image at pixel (i,j). To reduce notational ccmplexity the pixels are indexed by $1,2,\ldots,n$ consecutively frcm left to right and top to bottom. This ccnvention enables reference to the doubly indexed image b(i,j) as b(k), symbolically. Hence eqs. (1) to (3) can be written as

$$M(k) = E\{b(k)\} \tag{6}$$

$$R(k,\ell) = E\{[b(k) - M(k)][b(\ell) - M(\ell)]\} \tag{7}$$

$$y(k) = f\{b(k), \gamma(k)\} \tag{8}$$

Let the process x(k) be defined as

$$x(k) = b(k) - M(k) \tag{9}$$

for k=1,2,...,n . Thus, the problem of estimating b(k) reduces to estimating x(k).

<u>Estimation Method</u>: The minimum mean square (MMS) estimate $x^{\sigma}(k)$, of a process x(i) at time (pixel) k and for a given set of observation y(1),...,y(k) is given by [23]

$$x^{\sigma}(k) = E\{x(k) \mid y(1),\ldots,y(k)\} \tag{10}$$

Letting

$$Y(k) = \{y(1),\ldots,y(k-1)\} \tag{11}$$

then it can be shown [23,25] that $x^{\sigma}(k)$ and its error variance $\sigma_{\sigma}^{2}(k)$ are functionally related to Y(k) and y(k) by

$$x^{\sigma}(k) = \frac{\int x(k)p(y(k)\,|x(k))p(x(k)\,|\,Y(k))dx(k)}{\int p(y(k)\,|\,x(k))p(x(k)\,|\,Y(k))dx(k)} \tag{12}$$

$$\sigma_{\sigma}^{2}(k) = \frac{\int\left[x(k)-x^{\sigma}(k)\right]^{2} p(y(k)\,|\,x(k))p(x(k)\,|\,Y(k))dx(k)}{\int p(y(k)\,|\,x(k))p(x(k)\,|\,Y(k))dx(k)} \tag{13}$$

where p designates appropriate density functions. Equations (12) and (13), in turn suggest that the optimal estimation at k is achieved by first finding $p(x(k)|Y(k))$ and then using it along with $y(k)$ to arrive at $x^{\sigma}(k)$ and $\sigma_{\sigma}^{2}(k)$. The mean of $p(x(k)|Y(k))$ is the MMS one step prediction of the random variable $x(k)$ and its variance is the error variance of the predicted value. Thus, the optimal estimation at time k can be thought of as a two step procedure depicted in figure 1a, where blocks P and F may be identified as the prediction and filtering steps, respectively. In this system structure, $y(k)$ is isolated from other random variables and, assuming $p(x(k)|Y(k))$ is known, conceptually one can deal with its nonlinearities in block F, i.e. if $p(x(k)|Y(k))$ is given, then derivation of $x^{\sigma}(k)$ and $\sigma_{\sigma}^{2}(k)$ is accomplished by carrying out the integrations in eqs. (12) and (13). However, for the general observation of eq. (3), derivation of this probability density does not lend itself to analytic methods and available numerical approaches are computationally unfeasible [23, Chapter 7].

In this report an alternate procedure is considered, whereby an approximation to the probability density $p(x(k)|Y(k))$ is derived. The method is compatible with the logic of the estimator in figure 1a. This logic consists cf representing past information (i.e. information due to a pricri statistic and observations $y(1),\ldots,y(k-1)$ in the form of a protability density to be combined with $y(k)$ in block F. Based on this premise and the goal of algorithmic implementability, the estimator is constructed according to the following restrictions.

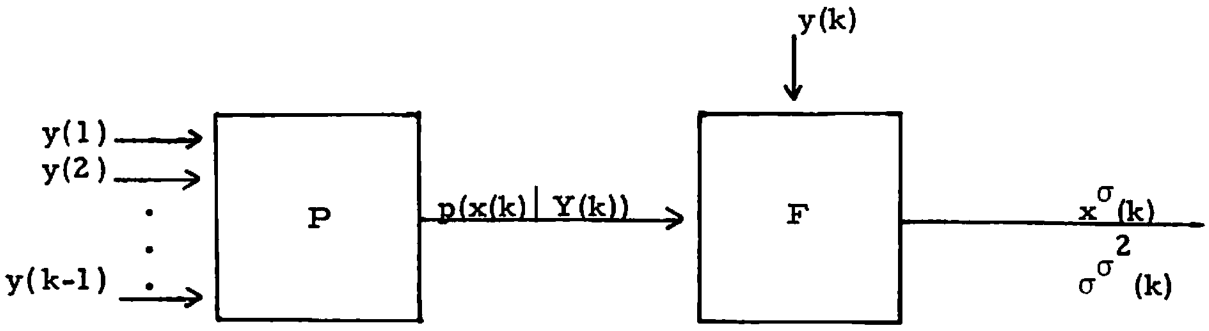
a). Only the first two moments of any random variable are computed.

b). The prediction process is chosen to be linear.

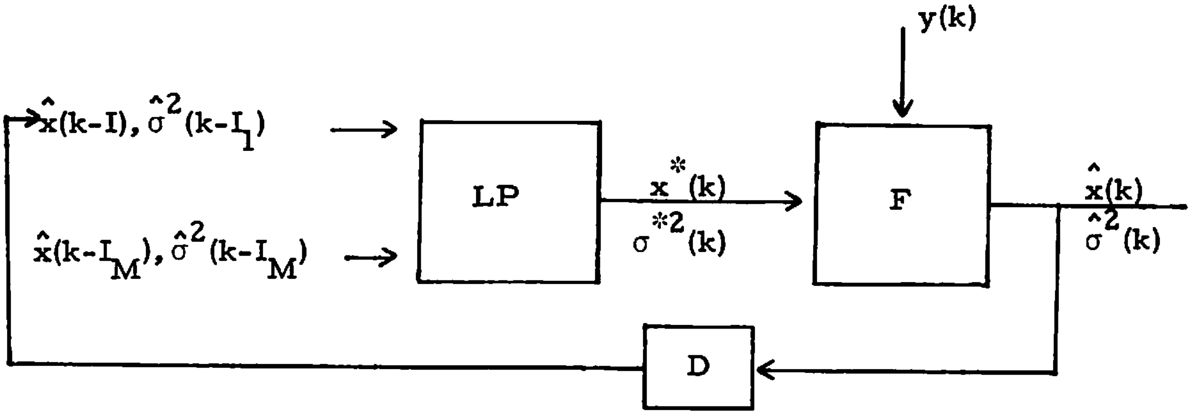
c). The prediction is to be based on a selected small number of past estimates. This will impose a desired limited memory requirement for the estimator.

Letting $\hat{x}(i)$ and $\hat{\sigma}^2(i)$ represent the estimate and its error variance, respectively, at time i, then the block diagram in figure 1b represents the structure of the proposed estimator. In this figure blocks LP, F and D signify linear prediction, filtering and one unit time delay, respectively. The subscript M is an indication of the size of memory and $x^*(k)$ and $\sigma^{*2}(k)$ are the one-step predicted value and its error variance. The set $\{k-I_1,\ldots,k-I_M\}$ is a set of two dimensional indices each distinct and prior to k.

Modeling Procedure: To derive the linear predictor (block LP of figure 1b), the a priori correlation information is first incorporated into a linear finite order model of the process $x(k)$ in the form of



(a) Optimal


y(k)
$\hat{x}(k-I), \hat{\sigma}^2(k-I_1)$
$\hat{x}(k-I_M), \hat{\sigma}^2(k-I_M)$
LP
$x^*(k)$
$\sigma^{*2}(k)$
F
$\hat{x}(k)$
$\hat{\sigma}^2(k)$
D

(b) Sub-Optimal

Figure 4.3-1. Estimator Configurations

$$x(k) = \sum_{i=1}^{M} \beta_i x(k-I_i) + B\,u(k) \qquad (14)$$

where $\beta_1, \ldots, \beta_M$ are constants and $\{u(k), u(k-1), \ldots\}$ is a set of independent identically distributed random variates with

$$E\{u(k)\} = 0 \qquad E\{u(m)\,u(n)\} = \begin{cases} 0 & \text{if } m \neq n \\ 1 & \text{if } m = n \end{cases} \qquad (15)$$

Consequently, eq.(14) is an autoregressive model [12], [18 to 20].

The problem of modeling consists of determining the order M, the coefficients $\beta_1, \ldots, \beta_M$, the set of two dimensional indices $k-I_1, \ldots, k-I_M$ and the variance of the white noise term B u(k) in eq. (14). In this work, first a procedure is developed to derive an autoregressive model for a given M followed by a discussion on the best choice of M. The modeling criterion is chosen to be minimization of E{B u(k)}. The procedure uses the numerical values of the correlation function and does not require analytic representation of R(m,n). The results are illustrated by the following example.

Consider the stationary two-dimensional correlation function

$$R(i,j,k,\ell) = R(|i-k|, |j-\ell|) = E\{x(i,j)\,x(k,\ell)\} = \exp\left[-\sqrt{(i-k)^2 + (j-\ell)^2}\right]$$

Application cf above procedure provides the following:

a). Best 2nd order model is

$$x(i,j) = 0.3\,x(i,j-1) + 0.3\,x(i-1,j) + 0.883\,u(I,j)$$

b). Best 3rd order model is

$$x(i,j) = 0.29\,x(i,j-1) + 0.25\,x(i-1,j) + 0.12\,x(i-1,j+1) + 0.877\,u(i,j)$$

c). Best 4th order model is

$$x(i,j) = 0.28\, x\,(i,j-1) + 0.24\, x(i-1,j) + 0.12\, x(i-1,j+1) + 0.03\, x(i-1,j-1) + 0.8769\, u(i,j)$$

d). Best 5th order model is

$$x(i,j) = 0.28\, x(i,j-1) + 0.24\, x(i-1,j) + 0.11\, x(i-1,j+1) + 0.03\, x(i-1,j-1) + 0.02\, x(i=1,j+2) + 0.8768\, u(i,j)$$

Hence, for example, to a third decimal place accuracy, the 3rd order model is a sufficient approximaticn. Note that, for example, the derivation of the 3rd order model requires the numerical values of R(C,C), R(0,1), B(1,0) and R(1,1).

<u>Linear Prediction</u>: Let the model of the random process x(k) (obtained in the previous section) be

$$x(k) = \sum_{i=1}^{M} \beta_i\, x(k-I_i) + B\,u(k) \qquad (16)$$

Given the estimate $\hat{x}(i)$, i=1,2,...,k-1 the linear prediction x (k), in general, is given by

$$x^*(k) = \sum_{j=1}^{k-1} \alpha_j\, \hat{x}(k-j) \qquad (17)$$

where $\alpha_1, \ldots, \alpha_{K-1}$ are to be chosen such that

$$E[x(k) - x^*(k)]^2 \qquad (18)$$

is minimized. This minimization is to be carried out subject to the system structure of figure 1b and is based on available information to the predictor. This information consists of the values of $x(i)$ and (i), $i<k-1$. Since each $\hat{x}(i)$ and $\hat{\sigma}^2(i)$ is the mean and variance, respectively, of a posterior density on $x(i)$ at time i (having used observations through $y(i)$), then the expectation in eq. (13) is well defined and operates on each random variabla $x(i)$ such that

$$E\{x(i)\} = \hat{x}(i)$$
$$E\{[x(i) - \hat{x}(i)]^2\} = \hat{\sigma}^2(i) \qquad (19)$$

Theorem 1: When the random process $x(k)$ satisfies eq. (16), then the (optimal) choice of $\alpha_1, \alpha_2, \ldots, \alpha_{K-1}$ in eq. (17) which minimizes eq. (18) is given by

$$\alpha_j = \begin{cases} \beta_i & \text{if } k-j = k-I_i \\ 0 & \text{otherwise} \end{cases}$$

The proof is given in [26].

This thecrem states that the best linear predictor is given as

$$x^*(k) = \sum_{i=1}^{M} \beta_i \hat{x}(k-I_i) \qquad (20)$$

The implementation of eq.(20) is very simple. This simplicity, along with the effectiveness of the result as illustrated in the next sections, are the justification behind the necessary approximations.

Filtering Step: Referring to figure 1b, the computational logic of block F is now developed. The predicted value $x^*(k)$ and its variance $\sigma_p^2(k)$, obtained from the linear predictor, represent the mean and variance of the a posteriori density on $x(k)$. This density represents the available knowledge on the random variable $x(k)$ prior to reception of $y(k)$. Since, for a given mean and variance the normal distribution represents the maximum uncertainty (entropy) [24., p. 132], this density function is assumed to be normal. Further uncertainty is associated with $x(k)$ if $\sigma^{*2}(k)$ is used in place of $\sigma_p^2(k)$. Consequently, an approximate and a rather conservative choice of the probability density for $x(k)$ is

$$p\left[x(k)\right] = \left[\sigma^*(k)\ \sqrt{2\pi}\right]^{-1} \exp\left\{-\frac{\left[x(k)-x^*(k)\right]^2}{2\sigma^{*2}(k)}\right\} \tag{21}$$

Observation $y(k)$ and $p(x(k))$ in eq.(21) are combined to derive the Bayes estimate, $\hat{x}(k)$

$$\begin{aligned}\hat{x}(k) = E\{x(k)\,|\,y(k)\} &= \int x(k)\ p(x(k)\,|\,y(k))\ dx(k) \\ &= \frac{1}{p(y(k))}\int x(k)\ p(y(k)\,|\,x(k))p(x(k))\ dx(k)\end{aligned} \tag{22}$$

But

$$p(y(k)) = \int p(x(k), y(k))\, dx(k) = \int p(y(k)\,|x(k))\, p(x(k))\, dx(k)$$

Hence, [25]

$$\hat{x}(k) = \frac{\int x(k)\, p(y(k)\,|\, x(k))\, p(x(k))\, dx(k)}{\int p(y(k)\,|x(k))\, p(x(k))\, dx(k)} \qquad (23)$$

Similarly,

$$\hat{\sigma}^2(k) = E\{[x(k) - \hat{x}(k)]^2\,|y(k)\} = \frac{\int (x(k)-\hat{x}(k))^2\, p(y(k)\,|x(k)) p(x(k)) dx(k)}{\int p(y(k)\,|x(k)) p(x(k))\, dx(k)} \qquad (24)$$

where p(x(k)) in eqs. (23) and (24) are given by eq.(21) and p(y(k)|x(k)) is obtained from the observation system structure.

In general, evaluation of x(.) and $\hat{\sigma}^2(.)$ in eqs. (23) and (24) will be performed numerically. This in turn, allows the procedure to be applicable to a broad class of observation systems including nonlinear forms of the observation y(k). The feasibility of this estimator is due to the structure of figure 1b which leads to eqs. (23) and (24).

Multiplicative Noise Term in Observation: Consider observations containing uniform multiplicative noise. In this case the observation is given by

$$y(k) = \gamma(k)\,[x(k) + M(k)] \qquad (25)$$

with

$$p(\gamma(k)) = \begin{cases} \dfrac{1}{\gamma_2^{(k)} - \gamma_1^{(k)}} & \text{if } 0 < \gamma_1(k) \le \gamma(k) < \gamma_2^{(k)} \\ 0 & \text{otherwise} \end{cases} \qquad (26)$$

With x(k) + M(k) as the image intensity at pixel k, eqs. (23) and (24) become [25]

$$\hat{x}(k) = \frac{1}{G}\int_a^b \frac{x(k)}{x(k)+M(k)} \exp -\left[\frac{(x(k) - x^*(k))^2}{2\sigma^{*2}(k)}\right] dx(k) \qquad (27)$$

$$\hat{\sigma}^2(k) = \frac{1}{G}\int_a^b \frac{[x(k) - x(k)]^2}{x(k) + M(k)} \exp\left[-\frac{(x(k) - x^*(k))^2}{2\sigma^{*2}(k)}\right] dx(k) \qquad (28)$$

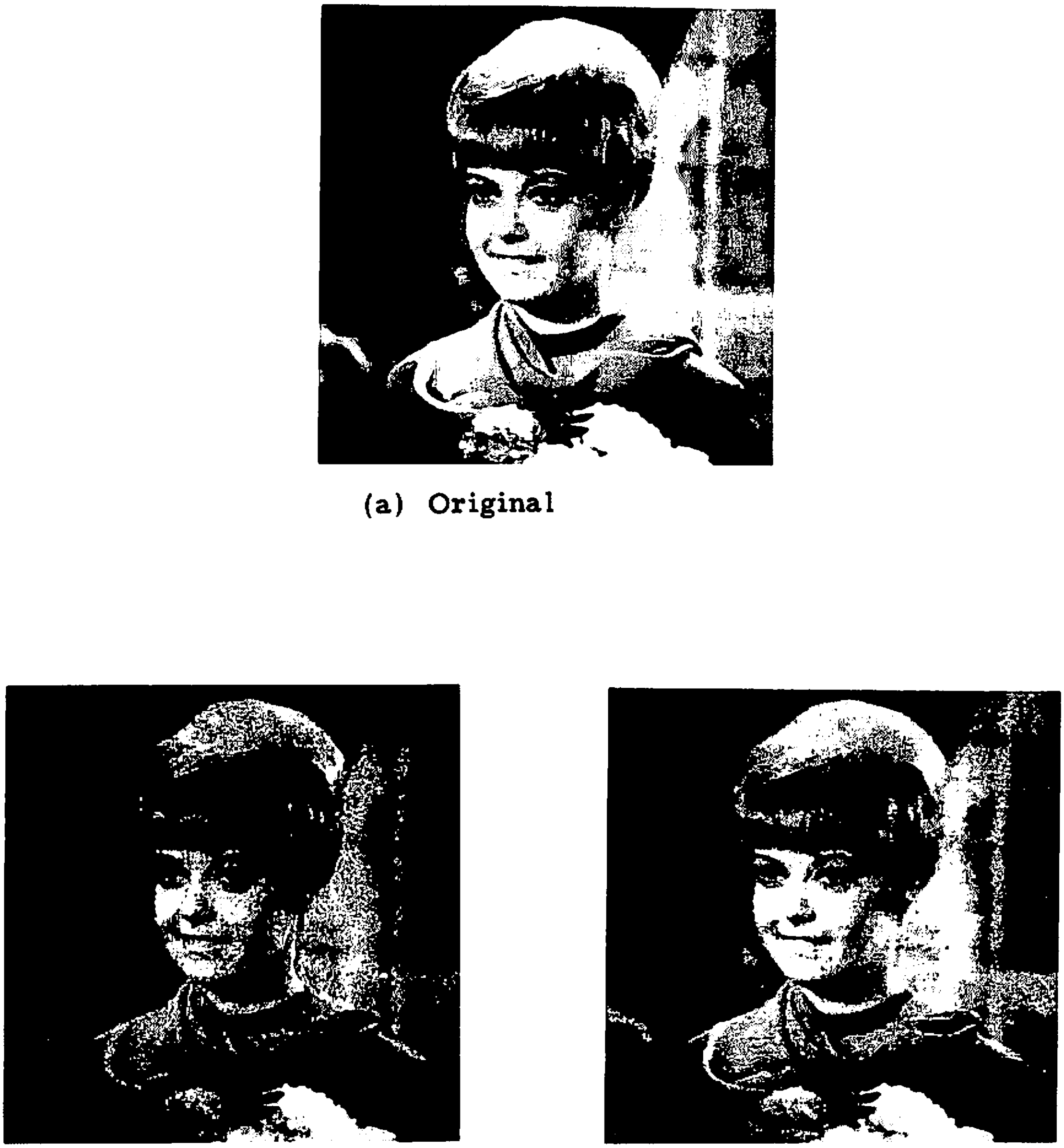
where

$$G = \int_a^b \frac{1}{x(k) + M(k)} \exp\left[-\frac{x(k) - x^*(k))^2}{2\sigma^{*2}(k)}\right] dx(k) \qquad (29)$$

and

$$\begin{aligned} a &= \frac{y(k)}{\gamma_2(k)} - M(k) \\ b &= \frac{y(k)}{\gamma_1(k)} - M(k) \end{aligned} \qquad (30)$$

Since eqs. (27) tc (29) are definite integrals, they can be evaluated numerically. All noisy images contain uniform multiplicative noise with noise bounds as indicated in these figures. The estimated images of figure 2 to 4 provide 5.48, 7.58 and 7.7 db. improvement, respectively. Aside frcm this quantitative improvement, the

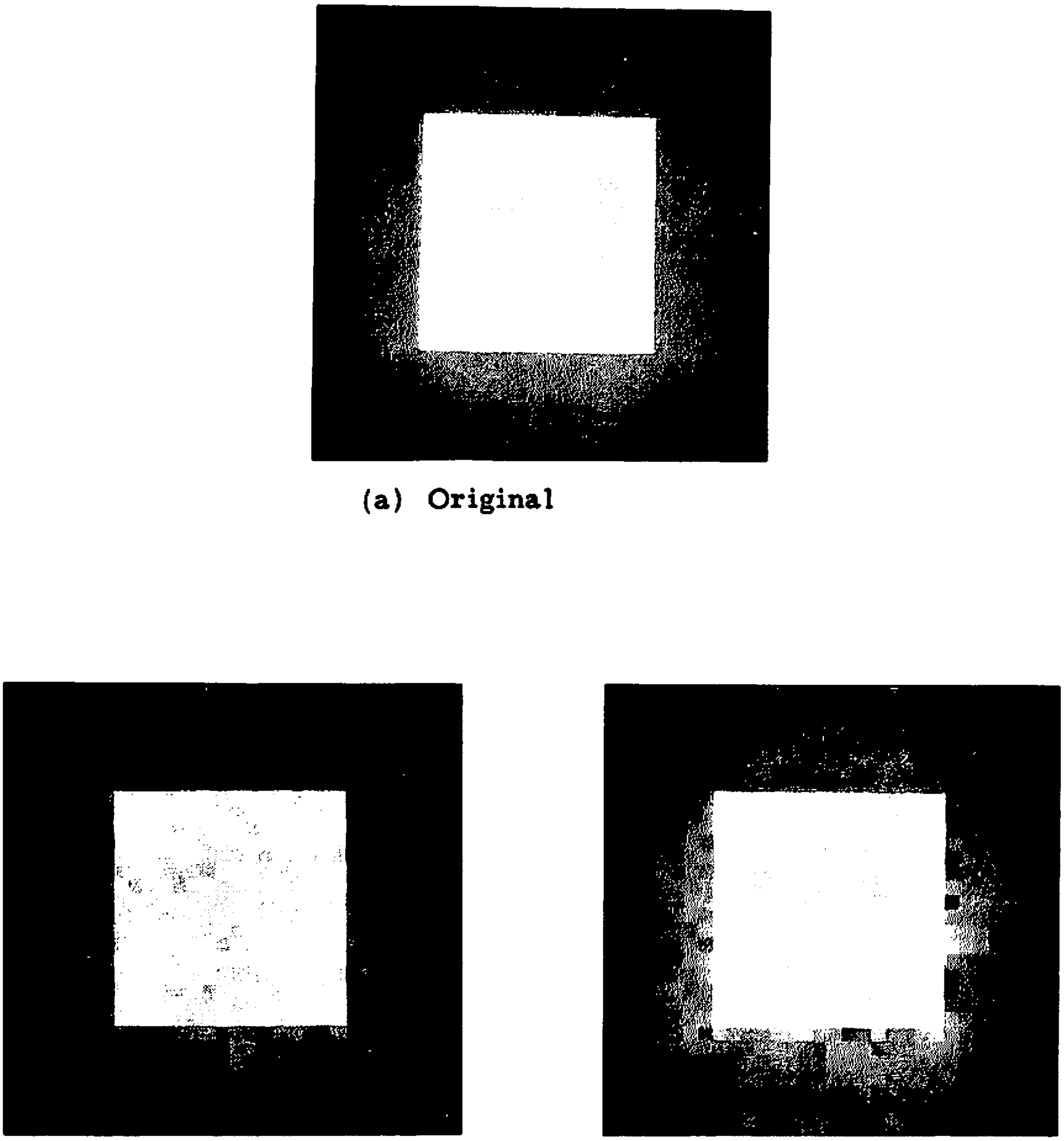


(a) Original

(b) Noisy, noise=0.7-1

(c) Estimate

Figure 4.3-2. Uniform multiplicative noise

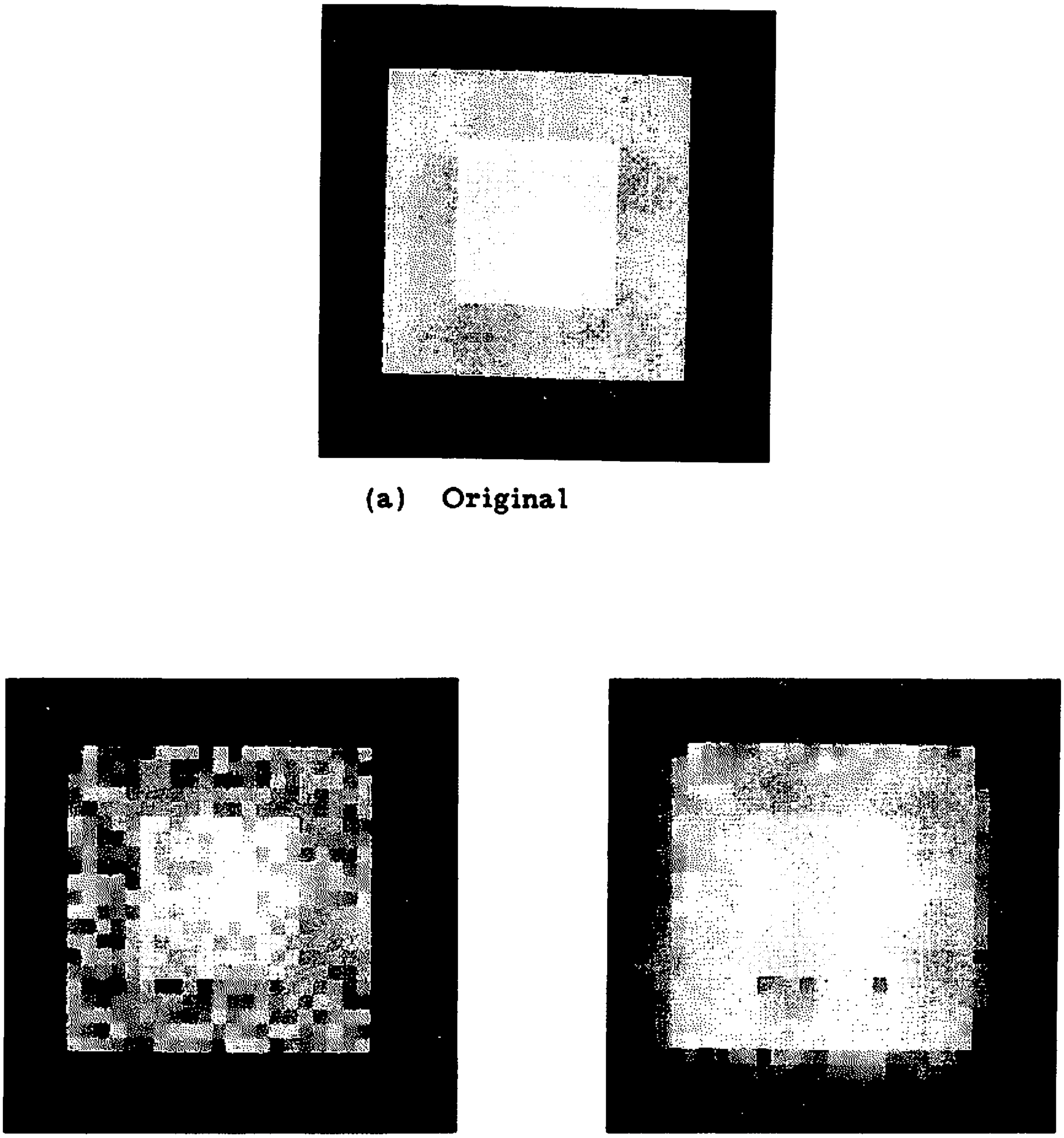


(a) Original

(b) Noisy, noise=0.7-1

(c) Estimate

Figure 4.3-3. Uniform multiplicative noise



(a) Original

(b) Noisy, noise=0.7-1

(c) Estimate

Figure 4.3-4. Uniform multiplicative noise

preservation of edges in the estimated images should be noted. The responsiveness of the estimator to abrupt pixel tc pixel intensity changes is due to the estimator structure of figure 1b.

The estimaticn procedure can also be applied to more general observation systems. As an example consider the case where

$$y(k) = \gamma(k)\left[x(k) + M(k)\right] + v(k) \tag{31}$$

where $\gamma(k)$ and $v(k)$ are both uniform. Letting the density of $\gamma(k)$ be given by eq. (25) and that cf $v(k)$ be

$$p(v(k)) = \begin{cases} \dfrac{1}{v_2(k)-v_1(k)} & \text{if } v_1(k) \le v(k) \le v_2(k) \\ 0 & \text{otherwise} \end{cases} \tag{32}$$

then $p(y(k)|x(k))$ can be obtained in terms of the convolution of $p(\gamma(k))$ and $p(v(k))$ [22]. This density, then, can be substituted in eqs. (23) and (24) to obtain pertinent filtering equations [25].

References

1. L.E. Franks, "A Model for the Random Video rocess," Bell System Technical Journal, April, 1966.

2. T.S. Huang, "Subjective Effect of Two Dimensional ictorial Noise," IEEE Transactions on Information Theory, Vol. IT-11, pp. 43-53, January, 1965.

3. V.M. Moraz, "Description of Images of Visual Objects through Correlation Functions," Automatika, Vol. 14, pp. 80-82, October, 1969.

4. C.W. Helstrom, "Image Restoration by the Method of Least Squares," Journal of the Optical Society of America, Vol. 57, pp. 297-303.

5. A.V. Oppenheim, R.W. Schafer and T.G. Stockham, Jr., "Nonlinear Filtering of Multiplied and Convolved Signals," ProceeDings of the IEEE, Vol. 56, No. 8, August, 1968, pp. 1264-1291.

6. B.R. Frieden, "Optimum Nonlinear Processing of Noisy Images," Journal of the Optical Society of America, Vol. 58, No. 9, September, 1968, pp. 1272-1275.

7. R.S. Bucy, M.J. Merritt and D.S. Miller, "Hybrid Computer Synthesis of Optimal Discrete Nonlinear Filter," University of Southern California, Technical Report No. 71-38, September, 1971.

8. J. Ting-Ho Lo, "Finite-Dimensional Sensor Orbits and Optimal Nonlinear Filtering," IEEE Transactions on Information Theory, Vol. IT-18, No. 5, September, 1972.

9. R.E. Kalman and R.C. Bucy, "New Results in Linear Filtering and Prediction Theory," ASME Transactions (J. Basic Eng.), Vol. 83D, March, 1961, pp. 95-108.

10. R.E. Kalman, "A New Approach to Linear Filtering and Prediction Problem," ASME Transaction (J. Basic Eng.), Vol. 82D, March, 1960,

pp. 35-45.

11. W.K. Pratt, "Generalized Wiener Filtering Computation Techniques," IEEE Transactions on Computers, Vol. C-21, No. 7, July, 1972, pp. 636-641.

12. A. Habibi, "Two Dimensional Bayesian Estimate of Images," Proceedings of the IEEE, Vol. 60, July, 1972, pp. 878-883.

13. A.K. Jain and E. Angel, "Image Restoration, Modeling, and Reduction of Dimensionality," IEEE Transactions on Computers, Vol. C-23, No. 5, May, 1974, pp. 470-477.

14. N.E. Nahi, "Role of Recursive Estimation in Statistical Image Enhancement," Proceedings of the IEEE, Vol. 60, July, 1972, pp. 872-877.

15. N.E. Nahi and T. Assefi, "Bayesian Recursive Image Estimation," IEEE Transactions on Computers, Vol. C-12, No. 7, July, 1972, pp. 734-738.

16. N.E. Nahi and L. Franco, "Recursive Image Enhancement - Vector Processing," IEEE Transactions on Communications, Vol. COM-21, No. 4, April, 1973, pp. 305-311.

17. S.R. Powell and L.M. Silverman, "Modeling of Two Dimensional Random Fields with Application to Image Restoration," IEEE Transactions Autc. Contr., Vol. AC-19, No. 1, February, 1974, pp. 8-13.

18. K.S. Miller, "A Noise on Stochastic Difference Equations," Ann. Math. Statist., Vcl. 39, No. 1, 1968, pp. 270-271.

19. H. Akaike, "Fitting Autoregressive Models for Prediction," Ann. Inst. Statist. Math., Vol. 21, 1969, pp. 243-247.

20. H. Akaike, "Power Spectrum Estimation through Autoregressive Mcdel Fitting," Ann. Inst. Statist. Math., Vol. 21, 1969, pp. 4C7-419.

21. P.B. Liebelt, An Introduction to Optimal Estimation, Addison-Wesley, Massachusetts, 1967.

22. A. Papculis, Probability, Random Variables, and Stochastic Processes, McGraw-Hill, New York, 1965.

23. N.E. Nahi, Estimation Theory and Applications, John Wiley and Sons, New York, 1969.

24. C.R. Rao, Linear Statistical Inference and its Applications, John Wiley and Scns, New Ycrk, 1965.

25. M. Naraghi, "A General Image Estimation Method," Dissertation, Department of Electrical Engineering, University of Southern California, June, 1975.

26. N.E. Nahi and M. Naraghi, "A General Image Estimation Algorithm Applicable to Multiplicative and Non-Gaussian Noise," Proceedings of Eighteenth Midwest Symposium on Circuits and Systems, August 11-12, 1975, Montreal, P.Q., Canada.

4.4 Image Restoration by Smoothing Spline Functions

Mohammad J. Peyrovian and Alexander A. Sawchuk

In a linear space-invariant imaging system with point-spread function h(x), the image g(x) is given by

$$g(x) = \int h(x-u)\,f(u)du + n(x) \tag{1}$$

where n(x) represents measurement Noise. In order to estimate the object function f(u) from image g(x) by a digital computer, the above continuous model must be discretized. A common method is to sample the functions h and g at a finite number of points. Spline functions, because of their highly desirably interpolating and approximating characteristics, are an interesting alternative to the above method. For uniformly spaced knots, a class of spline functions, called B-splines, has the following properties

(i) shift invariance

(ii) strictly positive

(iii) convolutional property

(iv) local basis property

Using B-splines for interpolation or approximation, the functions f and h can be represented by B-splines of degrees m and n, respectively

$$f(x) = \sum_{i=-\infty}^{\infty} f_i \, B_m \, (x-x_i) \qquad (2)$$

$$h(x) = \sum_{j=-\infty}^{\infty} h_j \, B_n(x-x_j) \qquad (3)$$

Substituting eqs. (2) and (3) in the convolution integral of eq. (1) gives

$$g(x) = \sum_{i=-\infty}^{\infty} \sum_{j=-\infty}^{\infty} f_i \, h_j \, B_m(x-x_i) * B_n(x-x_j) + n(x) \qquad (4)$$

From the convclutícnal property of B-splines

$$B_m(x-x_i) * B_n(x-x_j) = B_{m+n}(x-x_i-x_j) \qquad (5)$$

and representing g(x) by B-spline,s of degree m + n and assuming $\Delta x=x_{i+1}-x_i$ gives

$$\sum_{k=-\infty}^{\infty} g_k \, B_{m+n}(x-k\Delta x) = \sum_{i=-\infty}^{\infty} \sum_{j=-\infty}^{\infty} f_i h_j \, B_{m+n}(x-(i+j)\Delta x)+n(x) \qquad (6)$$

Equations (4), (5) and (6) show that the B-spline, which is interpolating the deterministic part of the degraded image, must be of higher degree than the B-splines interpolating object and point-spread function. In cther words, since the blurred image is always smoother than the object, a higher degree spline can follow the image function better than the one approximating the object function. This can be explained in the Fourier domain by observing that the Fourier transform of an m-th degree B-spline is a Sinc function to the power m. As m increases the amplitude of higher frequencies decreases. Since a blurred image has less higher frequency content than the object, a higher B-spline can represent the image better than the one representing the object.

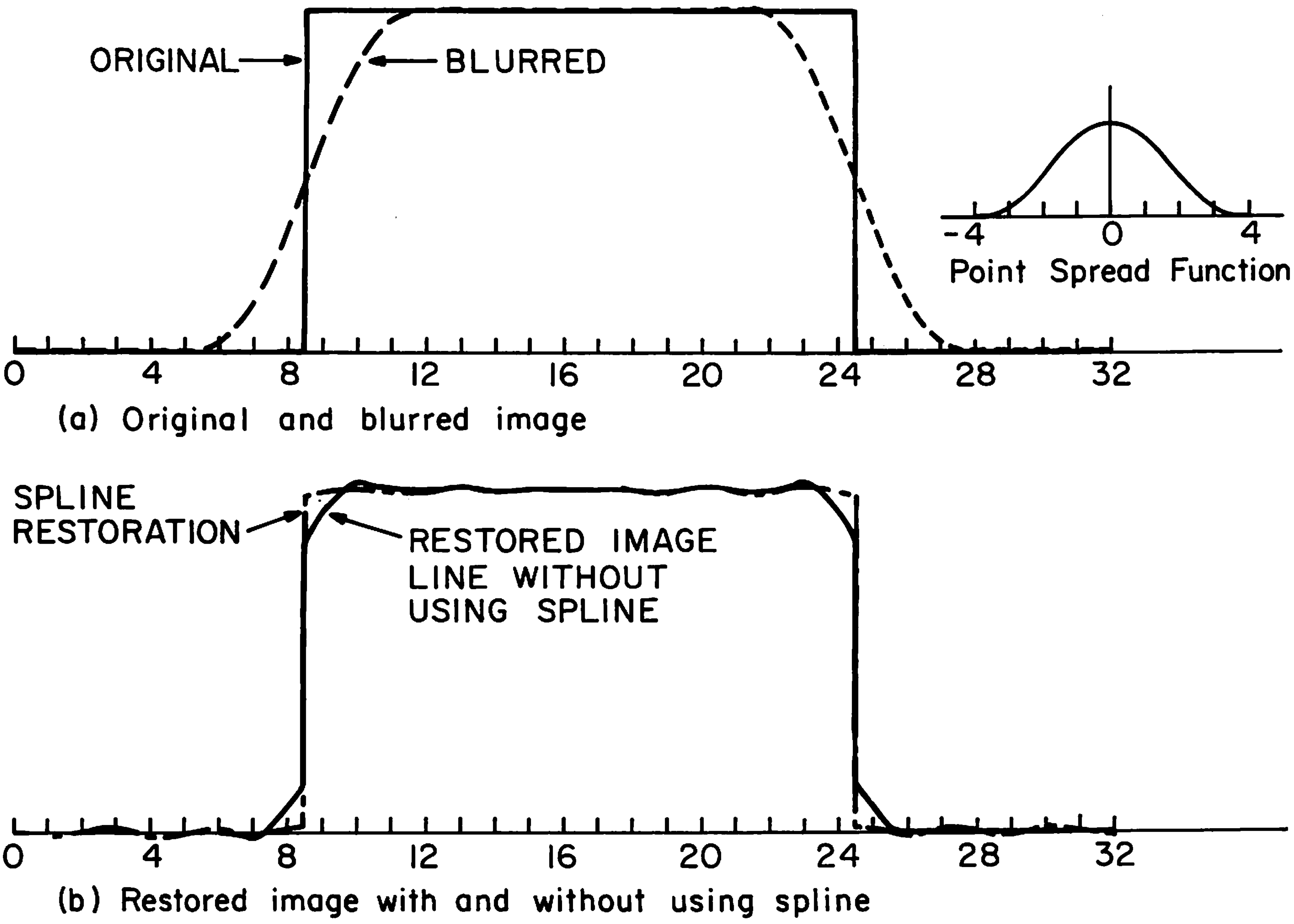
In a noiseless imaging system, eq. (6) may be written in the matrix form

$$\underline{g} = \underline{H}\,\underline{f} \tag{7}$$

If the point spread function is of finite width, the matrix $\underline{H}$ is banded. Figure 1a is a rectangular object which is blurred analytically by a 4th order polynomial

$$h(x) = \begin{cases} \frac{15}{56}\left(1 - \left(\frac{x}{3.5}\right)^2\right)^2 & -3.5 \le x \le 3.5 \\ 0 & \text{, elsewhere} \end{cases} \tag{8}$$

The object is a stop function, therefore it is interpolated by a zero order B-spline. The second derivative of h at points x=-3.5 and x=3.5 is a step function and it is interpolated by a second order B-spline. Since the convclution of a zero and second order B-spline is a cubic B-spline, the image is interpolated by a cubic B-spline. Figure 1b, the restored image with and without splines, shows that the spline



(a) Original and blurred image

(b) Restored image with and without using spline

Figure 4.4-1. Spline restoration.

restores the edges much sharper than the common pulse approximation method. Figure 2 is another example of spline restoration applied to a two dimensional blur with point spread function

$$H(x,y) = h(x)h(y) \tag{9}$$

where h is defined in eq.(8).

For a noisy image, the image data is first smoothed by minimizing

$$\int [g''(x)]^2 dx$$

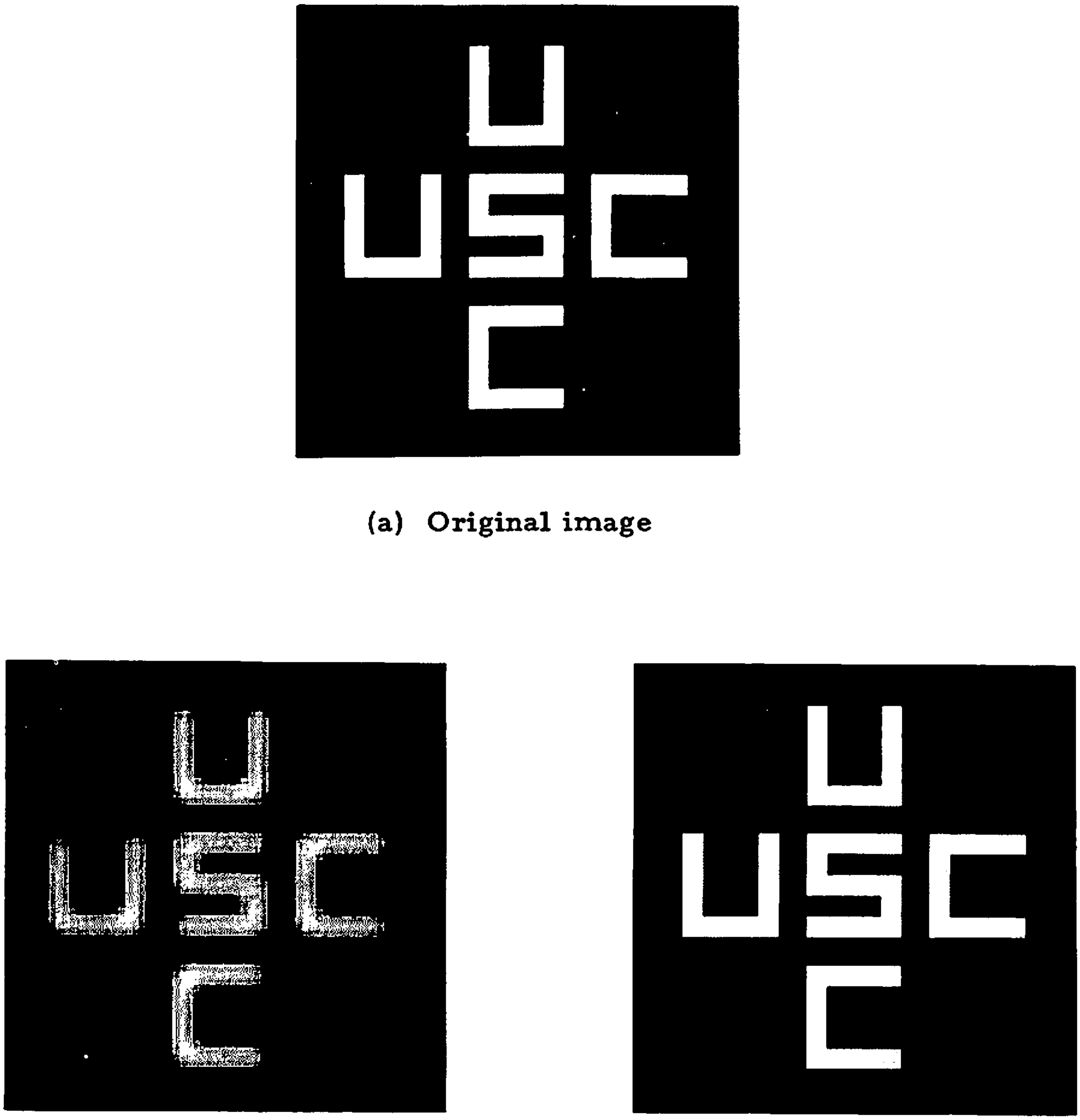
among all functions $g \epsilon c^2$ such that

$$\sum \left(\frac{g(x_i)-y_i}{\sigma_i} \right)^2 \leq S \tag{10}$$

Here y_i is the noisy image measured at point x_i...s>0 and $\sigma_i>0$ are given numbers. Setting S=0 leads to an interpolation problem. The factor σ_i control the smoothing window at point x_i and S controls the extent of smoothing. If the standard deviation of y_i is available, it may be used as σ_i . In this case, natural values of S lie within the confidence interval of the left hand side of eq.(10) as given by

$$N - (2N)^{\frac{1}{2}} \leq S \leq N + (2N)^{\frac{1}{2}}$$

where N is the number of data points. Reinsch [3] has shown that the solution to eqs.(9) and (10) is a cubic spline, and more generally, is



(a) Original image

(b) Blurred image

(c) Restored image using spline functions

Figure 4.4-2. Examples of spline restoration

a spline function of degree 2K-1 for least square minimization of the K-th derivative instead of the second derivative. In smoothing (S>0), the shape of the function is much more influenced by the minimum principle of eq.(9) than in interpolation (S=0).

The above smoothing criterion will be subject of further research on noisy blurred images, particularly the case K=2 because it leads to cubic splines which read simpler algorithms and less computation.

References

1. M.H. Schultz, Spline Analysis, Prentice-Hall, Incorporated, Englewood Cliffs, New Jersey, 1972.

2. T.N.E. Greville, Theory and Applications of Spline Functions, Academic Press, New York, 1969.

3. C.H. Reinsch, "Smoothing by Spline Functions,", Numer. Math. 10, 1967,pp. 177-183..

4.5 Detection and Estimation of Image Degradedby Film-Grain Noise

Firouz Naderi and Alexander A. Sawchuk

The goal of this research has been to analyze the problem of film-grain noise in the context of detection and estimation theory. The first step is the development of a mathematical model that reflects some of the complexities of image formation process, and yet is tractable in the subsequent restoration of the image.

Denoting by y(i,j) the observed optical density of photogsaphic film as measured by a micrc densitcmeter let

$$y(i,j) = S(i,j) + n(i,j) \qquad (1)$$

where S(i,j) denotes the density that would have been registered in the absence of grain noise and n(i,j) is the noise. Experiments by researchers in the field of Photography Science have indicated that n(i,j) is approximately Gaussian distributed with zero mean and a variance that is dependent on the type of the films used, the size of the scanner apertwre and the value of S(i,j). Clearly the observation model described in eq. (1) is additive with signal-dependent noise. Equivalently, the additivity of this modell may be sacrificed to obtain a signal-independent noise model. The result of doing so is the nonlinear observation model

$$y(i,j) = S(i,j) + g[S(i,j)]\, n(i,j) \qquad (2)$$

where the noise n(i,j) is zero mean and unit variance Gaussian. The form of the function g(.) has been subject cf some discussion. The experimental form

$$g[S(i,j)] = k[S(i,j)]^b \qquad (3)$$

has been found to be in agreement with many different theoretical and experimental results. Simplified photographic emulsion models such as Nottings, result in a value of 1/2 for the exponent b in the above equation. Data taken by Higgens and Stultz [1] suggest values of b in the range 0.3 to 0.4 if the scanning aperture is allowed to vary within a reasonable range.

With this model the restoration problem is considered in two different contexts: detection and estimation. In many image processing problems, it is necessary to use a high magnification to extract image information out of a photographic recording. A digital image of size 256 x 256 can be obtained by scanning a square region of side approximately 1.25 mm using a 5 micron aperture. Measuring optical density in such a small region of a photographic film results in such a high level of grain noise that distinguishing between adjacent areas of small contrast with the naked eye becomes impossible. Recently Zueng and Barrett considered image detection by a method called the "Noise cheating algorithm." References [2,3] show that this equivalent to method is sub optimal maximum liklilhood detection.

To set up the problem in the framework of detection theory suppose that the portion of the photographic film which is to be scanned can be segmented into M spatially uniform or near uniform density regions $R_1,\ldots,R_M$. Let a square aperture of size a x a be used to measure the optical density of the film. It is then possible to formulate an M + 1 hypothesis problem. The first M hypotheses, H_i

are the hypotheses that a given densitometer reading was obtained when the aperture was entirely in one of the M regions R_i. The last hypothesis H_{M+1} corresponds to a reading taken when the aperture overlapped on two or more regions simultaneously as shown in figure 1. Conventional maximum likelihood or Bayesian detectors can now be utilized for optimal detection of the M + 1 hypotheses.

A simple suboptimal method to accomplish this procedure is to perform the M + 1 hypothesis detection in two different steps. In step one the hypotheses $H_{M+1}$is ignored and the other M hypothesis are optimally detected. Therefore, in the first step the possibility that some readings might have been taken when the aperture overlapped more than one region is ignored. In the second step, in regions when hypothesis H_{M+1} appears to be highly probable (i.e. the edges), the image is re-examined with a finer aperture to recover details. Figures 2c to 2e contain simulation results of this restoration procedure for the three detection strategies described below.

<u>Maximum liklihood detection for signal-independent noise:</u> Referring to figure 1 assume that the mean density in region R , called the background, is m_b and the variance of the readings taken with an aperture of size a x a in this region is σ_b^2. The scanned image is of size 256 x 256. A two by two spatial averaging is first performed on the scanned image (Note that in effect the averaged image is what we would have obtained had we scanned the film with a 2a x 2a apreture to begin with.) In the averaged image, pixels in the region R_1 will now have mean m_b and variance $\tilde{\sigma}_b^2 = \tilde{\sigma}_b^2/4$. Each pixel in the

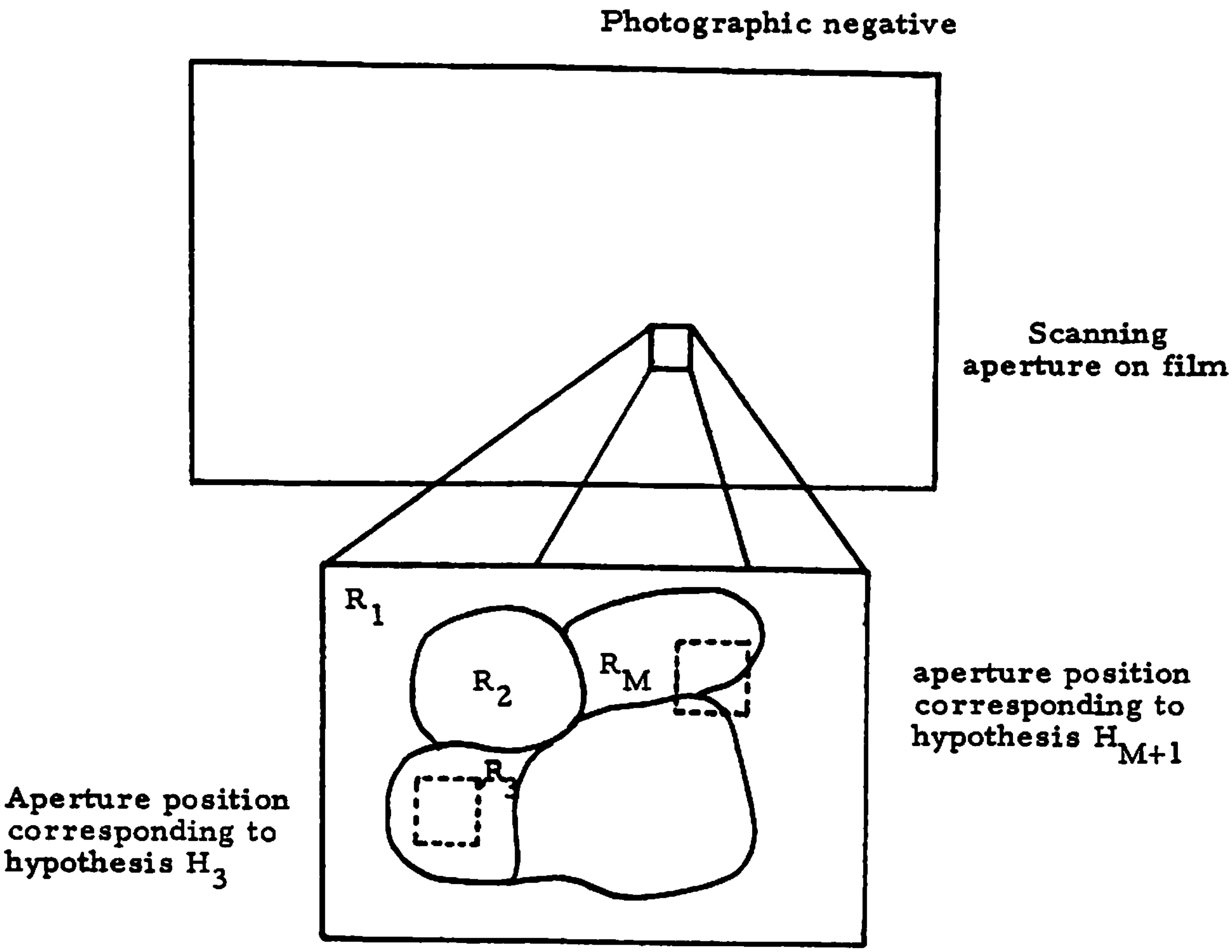
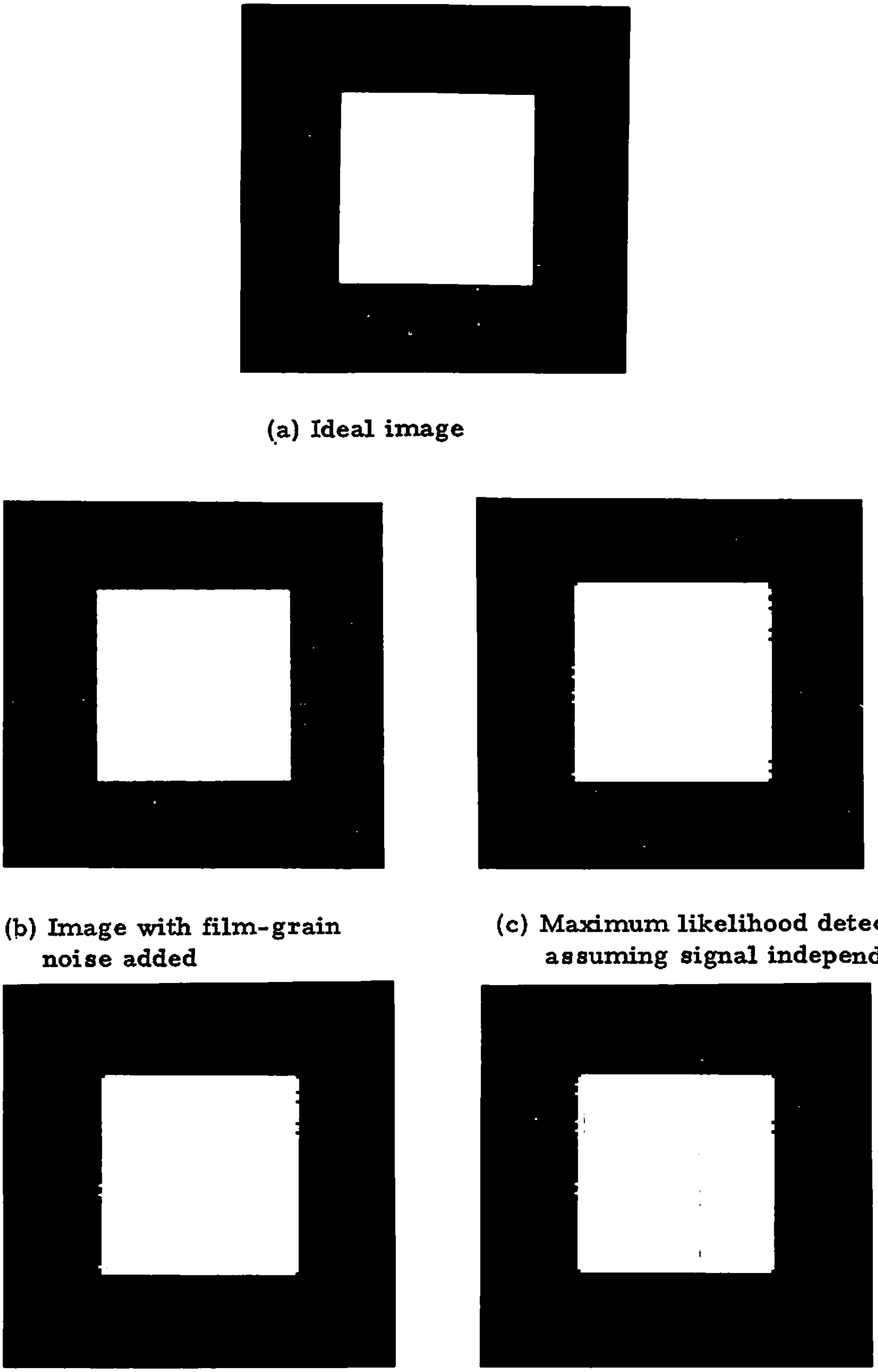

Photographic negative
Scanning aperture on film
R_1
R_2
R_M
R_3
aperture position corresponding to hypothesis H_{M+1}
Aperture position corresponding to hypothesis H_3

Figure 4.5-1. Image regions and aperture positions.



(a) Ideal image

(b) Image with film-grain noise added

(c) Maximum likelihood detected image assuming signal independent noise

(d) Maximum likelihood detected image assuming signal dependent noise

(e) Bayesian detected image

Figure 4.5-2. Image detection in the presence of film-grain noise.

averaged image is ncw quantized to one of M levels. These levels are chosen such that one of them will coincide with the mean of the background, m_b, and the others will be 4 G_b apart from each other. Since the distribution of the noise is Gaussian, if the decision levels for the quantization are set exactly at the mid-point between each quantization level then it is easy to demonstrate that the quantization is in fact maximum liklihood detection.

Since the levels are taken to be four standard deviations apart, all the image regicns which happen to have a mean density equal to one of the quantization levels will almost always be restored to their correct mean density following the quantizaton. Regions having mean densities that fall between two quantization levels will be "coded" into a percentage of these two levels.

The second step in the maximum liklihood detection process is to rework the edges in the quantized image by comparing the quantized image with the original scanned image which was scanned with the finer a x a aperture. Figure 2c is the detected image using this procedure.

Maximum liklihood detection for signal-dependent noise: The performance of the previous detector is dependent upon the distance between the quantization levels. If the levels are four standard deviations apart, it is certain that regions whose mean densities coincide with cne of the quantization level will be clear of the noise. As seen in eq. 3, the standard deviation of the noise is a function of the signal. Therefore for an image with high dynamic range it is necessary to increase the distance between the higher quantization level so as to keep the distance always four G . Furthermore, since the standard deviation varies, the decision level of the quantization which corresponds to the maximum likelihood detection, will no longer be at the mid-point between the quantization levels. Figure 2 shows the improvement over the previous detector when the signal dependence of the noise is taken into account with the proper quantization.

Bayesian Detection: As previously mentioned, quantization is, in effect, maximum likelihood detection. To take advantage of any a priori knowledge that might be available about the image, it is advantageous to perform Bayesian detection. Corresponding to the M hypothesis detection in the first step of the above two detectors, the mean densities of the M region may assume a distribution over a small range. Using the distribution as apriori statistics, the result of bayesian detection is shown in figure 2e.

Summary: Estimation algorithms are presently being applied to film-grain noise. Both Wiener filter and a nonlinear filtering reported in USC image processing institute report 580 [4], will be applied, and their performances will be compared and reported.

References

1. G. C. Higgins, and K. F. Stultz, "Experimental Study of rms Granularity as a Function of Scanning-spot size," Journal of Optical Society of America, Vol. 49, 1959, p.925.

2. H. J. Zweis,, and E. B. Barrett, "Noise Cheating Algorithms," Journal of Optical Society of America, Vol. 64, 1974.

3. F. Naderi and A.A. Sawchuk, "Nonlinear Filtering of Signal Dependent Noise" USC Image Processing Intitute Technical Report 560, 1 September 1974-28 February 1975, pp. 53-56.

4. M. Naraghi, "An Algorithmic Image Estimation Method Applied to Nonlinear Observation" USC Image Processing Institute Technical Report 580, 1975.

4.6 Vignetting and Density Correction for CRT Film Recording

Werner Frei

The acquisition of digitized image data and the restitution of processed pictures are generally costly, time-consuming, and yet essential steps of digital image processing. Errors and non-linearities introduced by the scanning and display equipment or the photographic process can add a surprising amount of unwanted and uncontrolled "image processing." These parasitic effects are by no means always readily visible in the finished product, but they may well invalidate the results of computer image manipulations. A careful control of the electro-optical machinery, the photographic process, as well as an understanding of human visual factors is therefore essential to insure the success and credibility of digital image processing.

Visual Factors: Optimum reflection prints, transparencies and television images practically never replicate the brightness distribution of original scenes, in the sense that color images do not reproduce the spectral energy distribution of colored lights. Although comprehensive fidelity criteria for images are yet to be discovered, a few simple rules have been found useful in the optimization of image acquisition and reproduction techniques.

Consider for example a black and white reflection print, which consists of a reflective backing coated with an emulsion of microscopic grains of silver. The image is formed by controlling the amount of silver in the emulsion and thus varying the relative light absorption of the print, within a typical dynamic range of 50 to 100:1. Such a photograph conveys its pictorial information to an observer irrespective of illumination variations over perhaps four to five orders of magnitude. This rather surprising phenomenon is caused by the ability of the visual system to "adapt" to ambient levels of lighting and thus to extract the reflection properties of objects [1,2]. Studies of the reproduction characteristics of optimal images [3] indicate indeed that although absolute brightness influences perceived quality, the quality criterion within the physical limitations of any given reproduction situation is greatly dependent upon its ability to reproduce relative brightness ratios. This fact is intuitively satisfying noting that pixel brightness ratios are a property of the scene reflectances that is invariant to the absolute intensity of a uniform illumination.

The implications of the above visual phenomenon are that the digital representation of light intensities sensed by a scanning device should ideally be a measure of image brightness ratios rather than arbitrary absolute intensity values. This is easily implemented in practice by recording the logarithm of the measured image intensities. Many commercially available scanners provide for such an option, usually called density (as opposed to transmittande or reflectance) scanning. On the reproduction side, care has then to be taken to preserve the recorded brightness ratios, a process that is facilitated by the inherent characteristics of the photographic process to be discussed in the next section.

The Photographic Process: Exposure of a black and white emulsion to light and subsequent development produces a light absorbing layer characterized by its optical density D which is defined as the logarithm of the ratio of transmitted to incident light. With all other parameters fixed, the optical density is ideally related to the intensity of the exposing light I by the function [4]

$$D = \gamma \log [I t] \qquad (1)$$

where t is the duration of the exposure. This function, well known in photography, is the Hurter-Driffield or D-log E curve. Actual photographic materials depart from this idealized law at both ends of their useful dynamic range. The factor describes the "contrast" of the emulsion and is positive for an ordinary negative material, and negative for a reversal process. Because the unexposed emulsion and its substrate are not perfectly transparent, an additional "fog" level D_0 is incorporated into the above equation yielding

$$D = D_0 + \gamma \log [It] \qquad (2)$$

The light reflected from a print or transmitted through a slide is related to the incident light I by [4]

$$I = I_0 10^{-D} \qquad (3)$$

The reproduced light intensity I' is given by

$$I^1 = I_0 10^{-D}[It] \qquad (4)$$

Note that if $\gamma = -1$, the conditions for an optimum reproduction as discussed in the previous section are met.

It is not easy to meet the relationship of eq.(4) with actual image processing equipment. Film is typically exposed by a CRT, LED or laser as a series of discrete dots which partly overlap; the exposure may not be uniform over the area of the image, etc. It is possible though to correct for such defects with a numerical pre-distortion of the digital image data. A simple model, appropriate for the correction of a CRT scanner, is discussed next.

<u>Calibration of I/O Devices:</u> Actual image acquisition and reproduction devices have a number of inherent imperfections which distort the final product. For example, the measurement of pixel intensity in scanners is usually not perfectly logarithmic (often linear); the pixel intensities displayed on television monitors are a

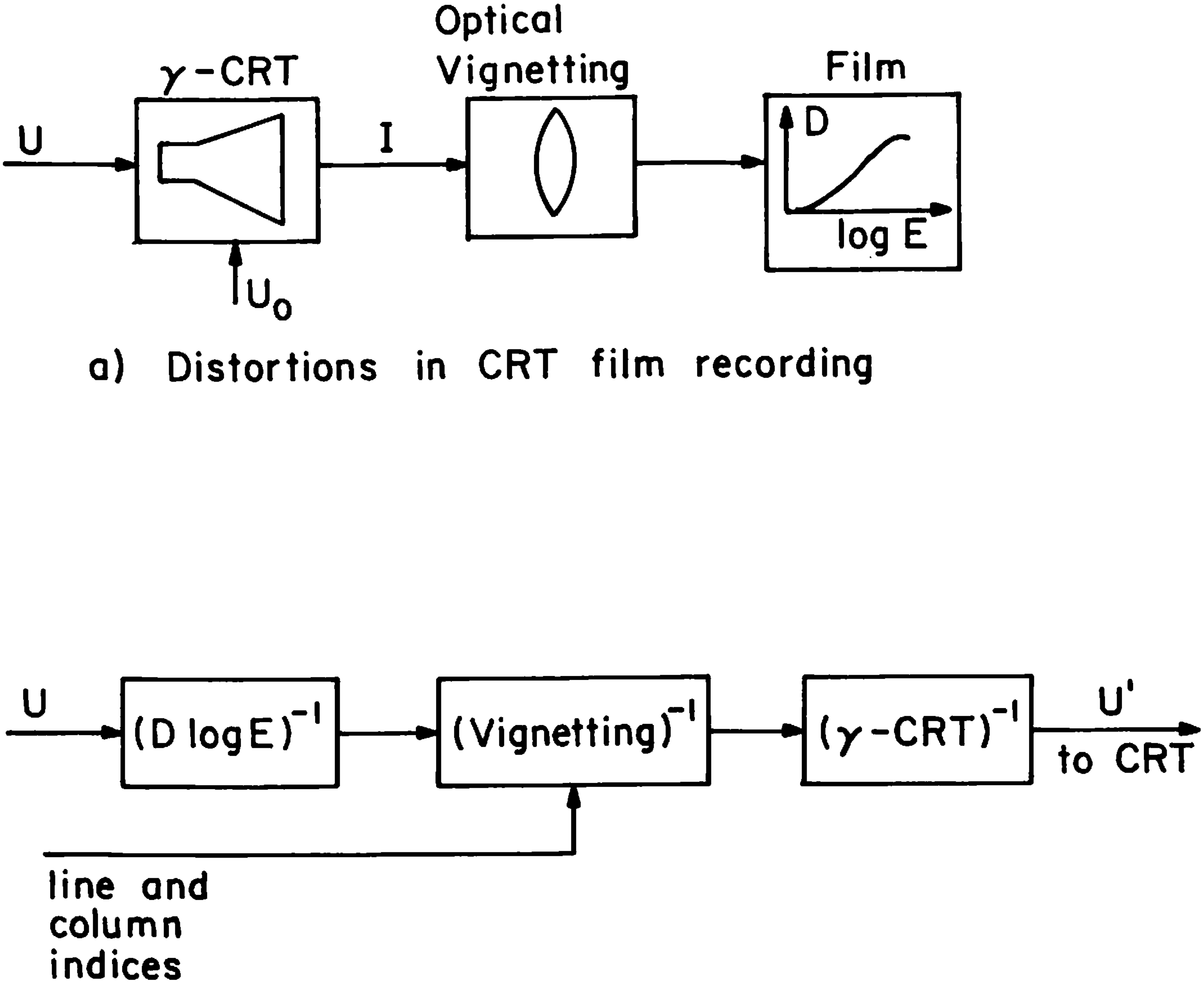
power function of the image signals; the light sensitive or light emitting surfaces of electron beam devices are not perfectly homogeneous; optical systems may introduce significant vignetting, etc. A number of procedures have been devised to cope with such imperfections [5,6]. For example, table look-up or polynomial approximations may be used to correct for the average deviations of the electro-optical transfer function from the desired behaviour. A more refined (and expensive) solution is to vary the coefficients of the correction as a function of the geometric image coordinates.

A true assessment of I/O device performane and the gathering of physical data for the design of correction schemes is best done by producing test patterns such as step tablets and measuring the optical density functions obtained on hardcopy or transparency.

To illustrate the above, a new software correction technique for CRT scanners is presented. It is of medium complexity, but computationally very fast and has given excellent results with a CRT scanner. The major sources of distortions in this case are schematized in figure 1. The CRT light emission I as a function of the drive and bias voltages U and U_0 respectively [7], as approximated by

$$I = [U + U_0 + U_1]^{\gamma}CRT \qquad (3)$$

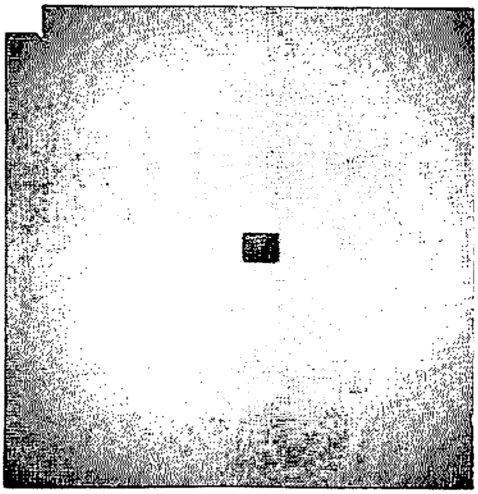
where U_1 represents the cut-off voltage of the CRT. Optical vignetting produces a darkening towards the image corners (figure 2),



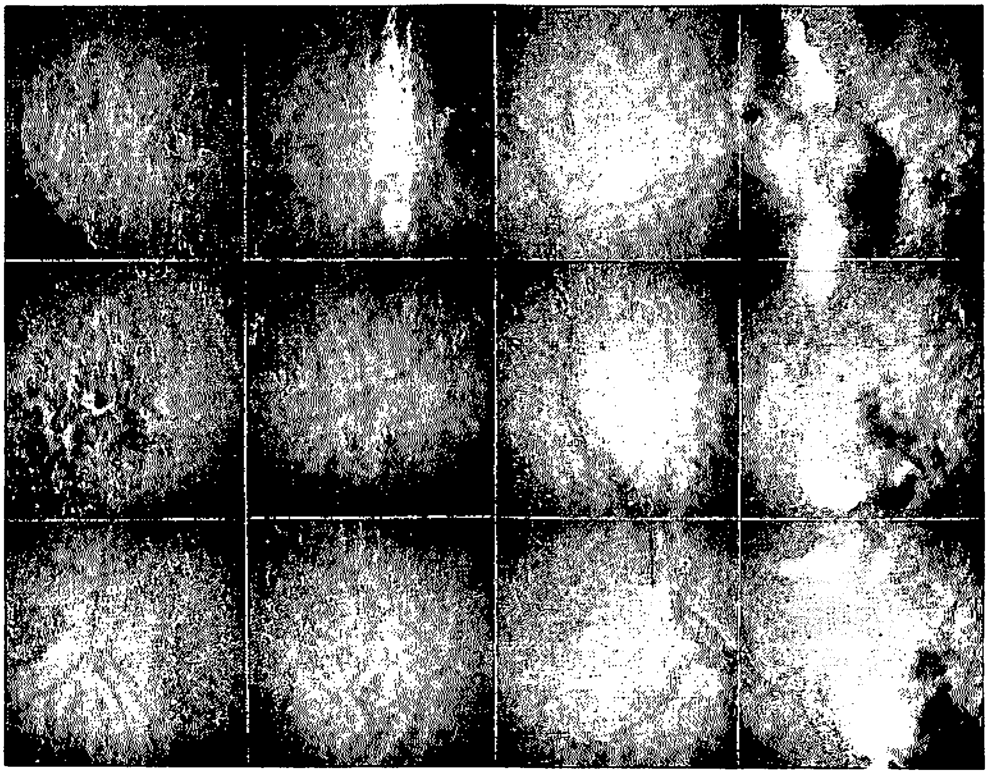
a) Distortions in CRT film recording

b) Numerical pre-distortion for recording correction

Figure 4.6-1. Distortions in CRT film recording and numerical pre-distortion for correction.



(a) Constant brightness values photographed with a polaroid camera. The darkening of the corners is evidenced by the small cut-off pasted in the middle of the photograph.



(b) The effect of vignetting on a mosaique

Figure 4.6-2. Demonstration of the vignetting effect.

(particularly annoying if one attempts to produce a mosaique, see figure 2b. Assuming that the vignetting is the only space-variant distortion, a fast table look-up algorithm has been implemented, such that each source of distortion mentioned above is corrected for in the appropriate order. The γ_{CRT}^{-1} and D-log E correction of figure 1b are straight foward look-up tables based upon measured data. Perhaps the most interesting pre-distortion step is the vignetting correction. Assuming circular symmetry, a second order polynomial of the form

$$I' = I\left[A + B(x^2 + y^2)\right] \quad (7)$$

has been used to boost the light intenities towards the image corners where x and y are the image coordinates referenced to the screen center. The values A/2+Bx are stored in a one dimensional array C and the correction is made by looking up this array twice given the pixel line and column indicies x_i and y_i. The results from this fast correction technique are shown in figure 5. The variations in density across a uniform surface are less than 0.1 density units, whereas the uncorrected image had corners darkened by as much as 0.35 density units.

References

1. T.H. Cornsweet, Visual Perception, Academic Press, New York, 1970.

2. T.G. Stockham, "Image Processing in the Context of a Visual Model," Proceedings of the IEEE, Vol. 60, July, 1972, pp. 828-842.

3. C. J. Battleson and E.J. Breneman, "Brightness Perception in Complex Fields," JOSA, pp. 953-957, July, 1967.

4. R.M. Evans, W.T. Hanson, and W.L. Brewer, *Principles of Color Photography*, John Wiley and Sons, New York, 1953.

5. R. Nathan, "Digital Video Data Handling," Technical Report 32-877, Jet Propulsion Laboratory, Pasadena, California, 1966.

6. F.C. Billingsley, "Applications of Digital Image Processing," Applied Optics, Vol. 9, 1970, pp. 289-299, 1970.

7. F. Kretz and W. Frei, "Optimal Logarithmic Quantization for Picture Processing," USCEE Report No. 530, 1974, pp. 11-19.

4.7 Spectral Sensitivity Estimation of a Color Image Scanner

Clanton E. Mancill and William K. Pratt

The spectral sensitivity of a color scanner must be determined in order to calibrate its response. Direct spectral measurements over the continuum of the spectral band are often difficult to obtain. However, responsivity measurements can be made through spectrally selective filters to estimate the continuous spectral sensitivity of the color scanner.

Spectral Radiance Estimation: Many tasks in color and multispectral image restoration involve the estimation of the spectral radiance function $c(\lambda)$ from a series of observations of the form

$$x_i = \int c(\lambda)\, s_i(\lambda)\, d\lambda + n_i \qquad (1)$$

where $s_i(\lambda)$ is the spectral sensitivity of the spectral measurement filter for i=1,2,...,P observations. The term n_i represents additive noise or uncertainty in the measurement. Discrete estimation techniques can be applied to this problem solution <1>. The first step is to discretize the continuous integral to form the vector equation

$$x_i = \underline{s}_i^T \underline{c} + n_i \qquad (2)$$

where $\underline{s}_i$ and $\underline{c}$ are Q x 1 vectors of quadrature samples of $s_i(\lambda)$ and $c(\lambda)$, respectively. Then, the set of P observations may be arranged into the P x 1 vector

$$\underline{x} = \underline{S}\,\underline{c} + \underline{n} \qquad (3)$$

where the vector $\underline{s}_i^T$ occupies the i th row of the matrix S. The system of equations represented by eq.(3) is normally highly underdetermined if sufficient quadrature mesh points are taken to reduce the quadrature error to reasonable bounds.

An estimate $\hat{\underline{c}}$ of the true spectral energy distribution c can be obtained by the generalized inverse estimate <2>

$$\hat{\underline{c}} = \underline{S}^{-}\,\underline{x} = \underline{S}^T(\underline{S}\,\underline{S}^T)^{-1}\,\underline{x} \qquad (4)$$

Although the generalized inverse provides a minimum mean square error, minimum norm estimate of $\underline{c}$, ill-conditioning of $\underline{S}$ coupled with observational errors can lead to oscillatory estimates. Since c is generally quite smooth, it is reasonable to impose some smoothing constraints on the solution. A common type of smoothing estimate is given by <3>

$$\underline{\hat{c}} = \underline{M}^{-1}\underline{S}^{T}(\underline{S}\,\underline{M}^{-1}\underline{S}^{T})^{-1}\,\underline{x} \qquad (5)$$

where $\underline{M}$ is a smoothing matrix of the typical form

$$\underline{M} = \begin{bmatrix} 1 & -2 & 1 & 0 & 0 & 0 & 0 & \cdots & \cdots & \cdots & 0 \\ -2 & 5 & -4 & 1 & 0 & 0 & 0 & & & & \cdot \\ 1 & -4 & 6 & -4 & 1 & 0 & 0 & & & & \cdot \\ 0 & 1 & -4 & 6 & -4 & 1 & 0 & & & & \cdot \\ 0 & 0 & 1 & -4 & 6 & -4 & 1 & & & & \cdot \\ & & & & \cdot & & & & & & \\ & & & & & \cdot & & & & & \\ & & & & & & \cdot & & & & \\ & & & & & 1 & -4 & 6 & -4 & 1 & 0 \\ & & & & & 0 & 1 & -4 & 6 & -4 & 1 \\ & & & & & 0 & 0 & 1 & -4 & 5 & -2 \\ 0 & \cdots & \cdots & \cdots & \cdots & 0 & 0 & 0 & 1 & -2 & 1 \end{bmatrix} \qquad (6)$$

A third alternative is to apply Wiener estimation methods <4>. With Wiener estimation, the vector $\underline{c}$ to be estimated is assumed to be a sample of a vector random process with known mean and covariance matrix $\underline{K}_c$. The Wiener estimate is given by

$$\underline{\hat{c}} = \underline{K}_c\underline{S}^{T}(\underline{S}\,\underline{K}_c\underline{S}^{T}+\underline{K}_n)^{-1}\underline{x} \qquad (7)$$

where $\underline{K}_n$ is the covariance matrix of the additive observational noise assumed independent of $\underline{c}$. As a convenient approximation the covariance matrix can be modelled as a first order Markov process covariance matrix of the fcrm

$$\underline{K}_c = \frac{\sigma_c^2}{Q} \begin{bmatrix} 1 & \rho & \rho^2 & \cdots & \rho^{Q-1} \\ \rho & 1 & \rho & \cdots & \rho^{Q-2} \\ \cdot & & & & \cdot \\ \cdot & & & & \cdot \\ \cdot & & & & \cdot \\ \rho^{Q-1} & \cdots & & & 1 \end{bmatrix} \qquad (8)$$

where $0 \leq \rho \leq 1$ is the adjacent element correlation factor and represents the energy of $\underline{c}$. Observaticn noise is commonly modelled as a white noise process with covariance equal to

$$\underline{K}_n = \frac{\sigma_n^2}{Q} \underline{I} \qquad (9)$$

where σ_n^2 is the noise energy and $\underline{I}$ is an identity matrix.

<u>Color Image Scanner Calibration</u>: A common problem in the evaluation and calibration of color image scanners is to determine the total spectral respcnse cf the scanner taking into account the spectral radiance of the illumination source, spectral absorption and scattering of the optics, and spectral sensitivity of the photodetector. Direct measurements are often not feasible. Referring to eq.(1), let $c(\lambda)$ be redefined to represent the spectral sensitivity response of the scanner and $s_i(\lambda)$ be one of P spectral test functions. The measurement procedure then proceeds as follows. An optical filter of known spectral characteristics, such as an absorption filter or narrowband interference filter is introduced into the scanner and an output reading is obtained. The process is repeated for a number of

filters whose peak transmissivities span the spectral region of interest. The measurements form the vector of observations, and an estimation operation is then invoked to obtain an estimate of the scanner spectral response.

In order to evaluate the estimation procedure, a computer simulation experiment was performed in which simulated measurements were taken of a Gaussian shaped spectral function through simulated absorption filters. Figure 1 contains a plot of the spectral shapes of the filters. The simulated measurements were then utilized as spectral observations for estimation of $c(\lambda)$. Figure 2 illustrates the performance of the three estimation methods for simulated measurements through the filters. In these experiments the mean square fit between the actual spectral function and its estimate was least for the simulated interference filter measurements using a Wiener estimate with $\rho = 0.9$ and a signal-to-noise ratio of 1000.

The spectral estimation procedures have also been applied to the estimation of the spectral response of an Optronics Model S 2000 flat bed scanning microdensitometer. Figure 3 shows the estimate obtained with absorption and interference filters for the three estimation methods. No direct measurements are available for the scanner so that no "ground truth" can be established. But, on the basis of the simulation experiments, it is concluded that the Wiener estimate obtained with the set of interference filters is a reasonable estimate of the actual spectral response.

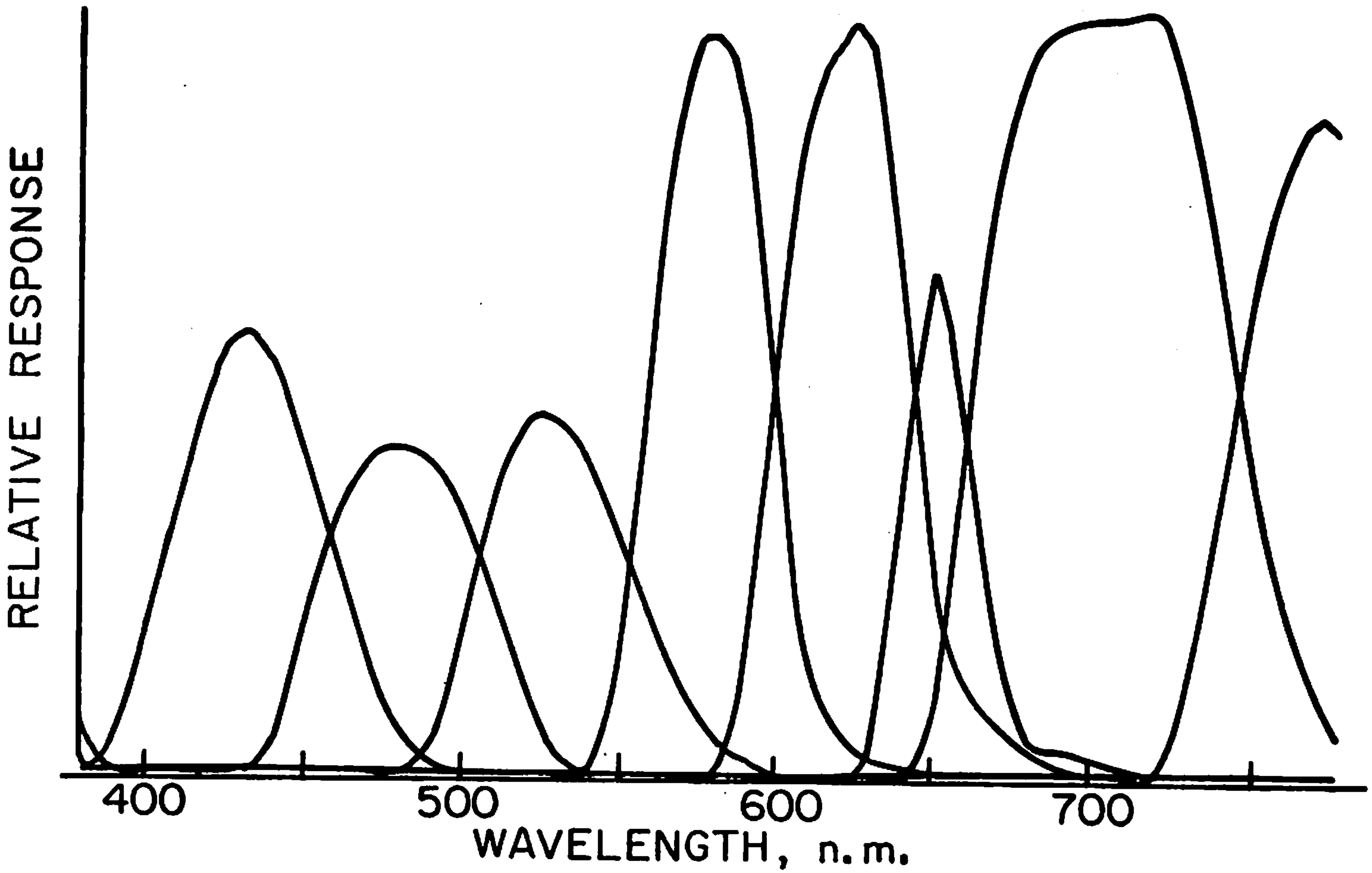
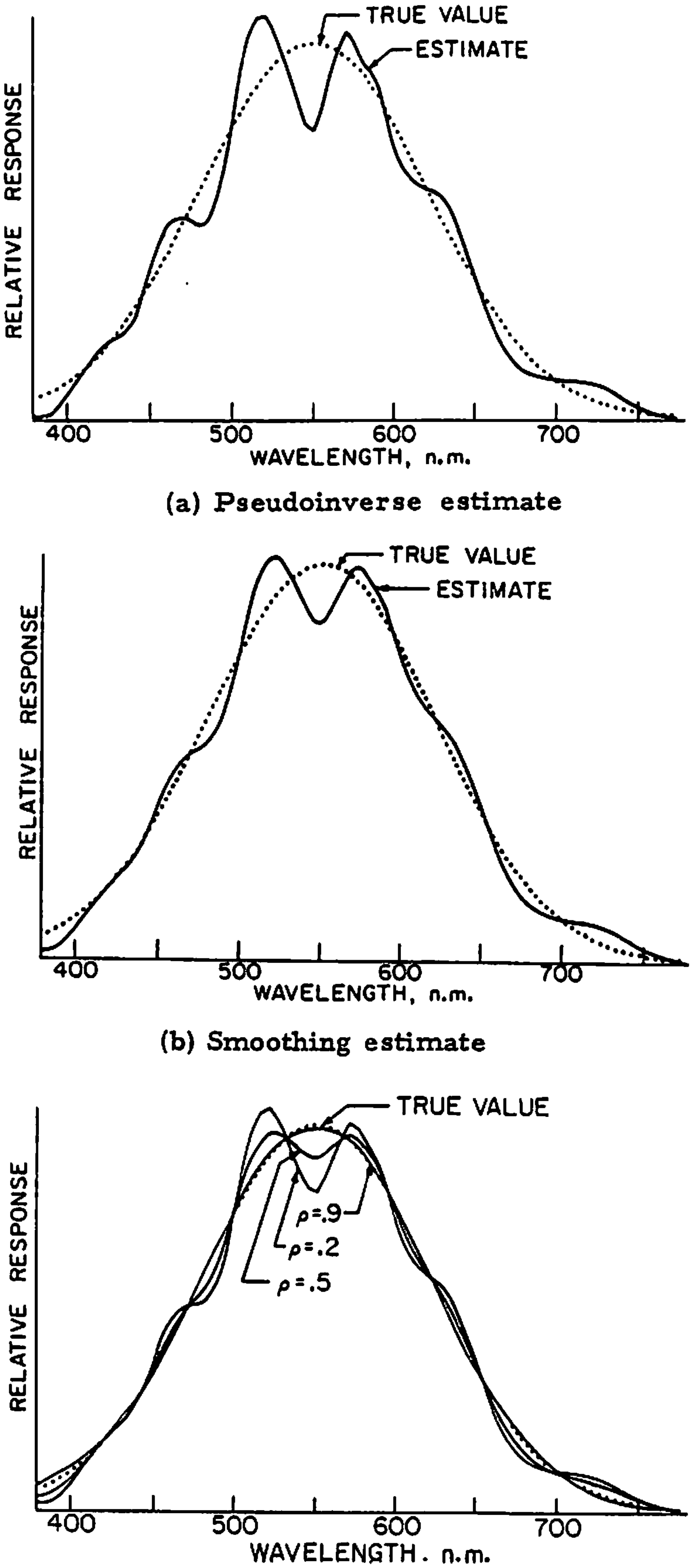


Figure 4.7-1. Spectral shapes of absorption filters

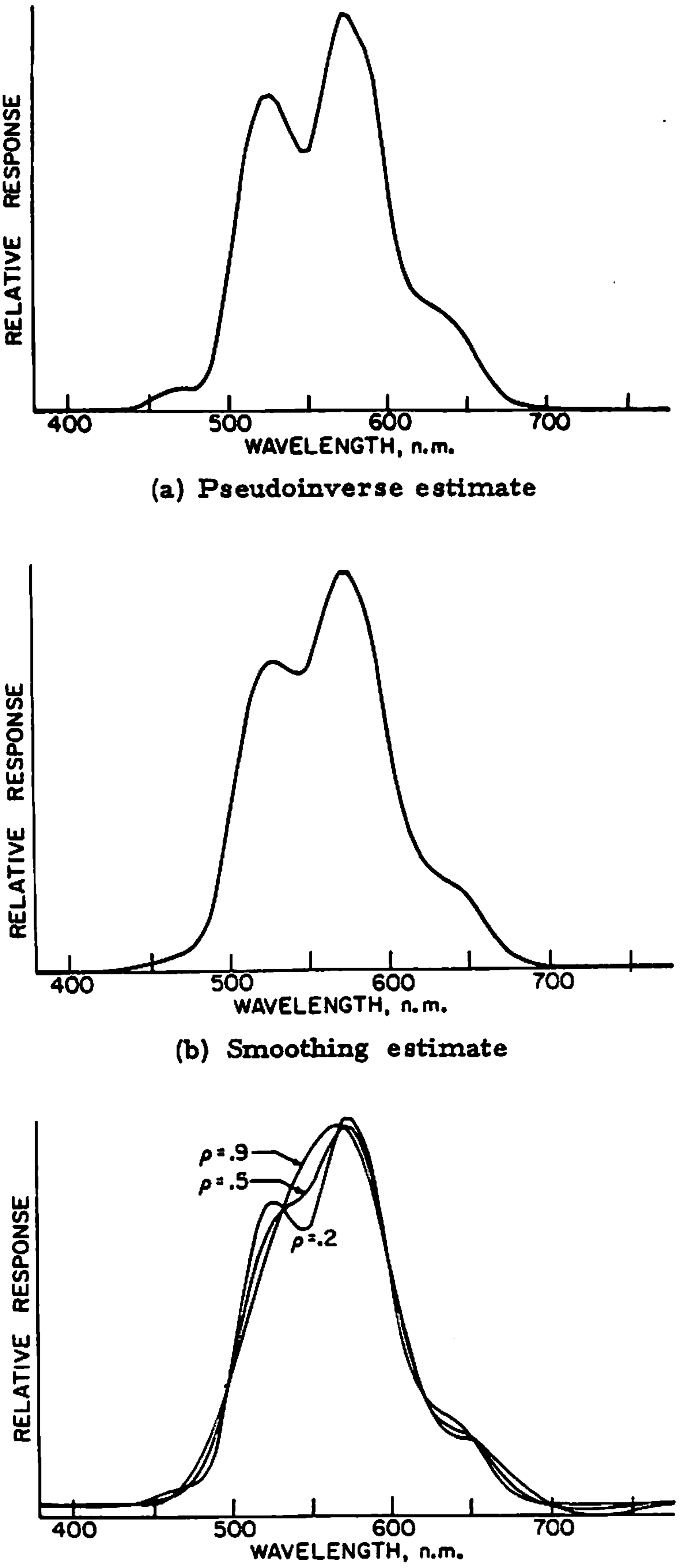


(a) Pseudoinverse estimate

(b) Smoothing estimate

(c) Wiener estimate, SNR= 1000

Figure 4.7-2. Comparison of actual and estimated spectral response for absorption filters obtained by computer simulation.



(a) Pseudoinverse estimate

(b) Smoothing estimate

(c) Wiener estimate, SNR=1000

Figure 4.7-3. Estimated spectral response for absorption filters for microdensitometer color scanner.

References

1. B. W. Rust and W. R. Burrus, Mathematical Programming and the Numerical Solution of Linear Equations, American Elsevier, New York, 1972.

2. F. A. Graybill, Introduction to Matrices with Applications in Statistics, Wadsworth, Belmont, Cal.,.1969.

3. C. R. Rao and S. K. Mitra, Generalized Inverse of Matrices and its Applications, John Wiley and Sons, New York, 1971.

4. P. B. Liebelt, An Introduction to Optimal Estimation, Addison-Wesley, Reading, Mass.,1967.

4.8 Median Filtering

William K. Pratt

The median filter is a nonlinear signal processing technique developed by Tukey <1> which is useful for noise suppression in images. In one dimensional form, the median filter consists of a sliding window encompassing an odd number of pixels. The center pixel in the window is replaced by the median of the window pixels. The median of a discrete sequence a ,a ,....,a , for N odd is that member of the sequence for which (N-1)/2 elements are smaller or equal in value, and (N-1)/2 elements are larger or equal in value. For example, if the values of the pixels within a window are 80,90,200,110,120, the center pixel would be replaced by the value 110 which is the median value of the sorted sequence 80,90,110,120,200.

In this example, if the value 200 was a noise spike in a monotonically increasing sequence, the median filter would result in considerable improvement. On the other hand, the value 200 might represent a valid signal pulse for a wide bandwidth sensor, and the resultant image would suffer some loss of resolution. Thus, in some cases the median filter will provide noise suppression, and in other cases it will cause signal suppression.

Figure 1 illustrates some examples of the operation of a median filter and a mean (smoothing) filter for a discrete step function, ramp function, pulse function, and triangle function with a window of five pixels. It is seen from these examples that the median filter has the usually desirable property of not affecting step functions or ramp functions. Pulse functions whose periods are less than one-half the window width are suppressed. Also, the peak of the triangle function is flattened.

Operation of the median filtered can be analyzed to a limited extent. It can be shown that the median of the product of a constant K and a sequence f(j) is

$$\mathrm{med}\{K f(j)\} = K\ \mathrm{med}\{f(j)\} \qquad (1)$$

Furthermore,

$$\mathrm{med}\{K + f(j)\} = K + \mathrm{med}\{f(j)\} \qquad (2)$$

However, for two arbitrary sequences f(j) and g(j) it does not follow that the median of the sum of the sequences is equal to the sum of

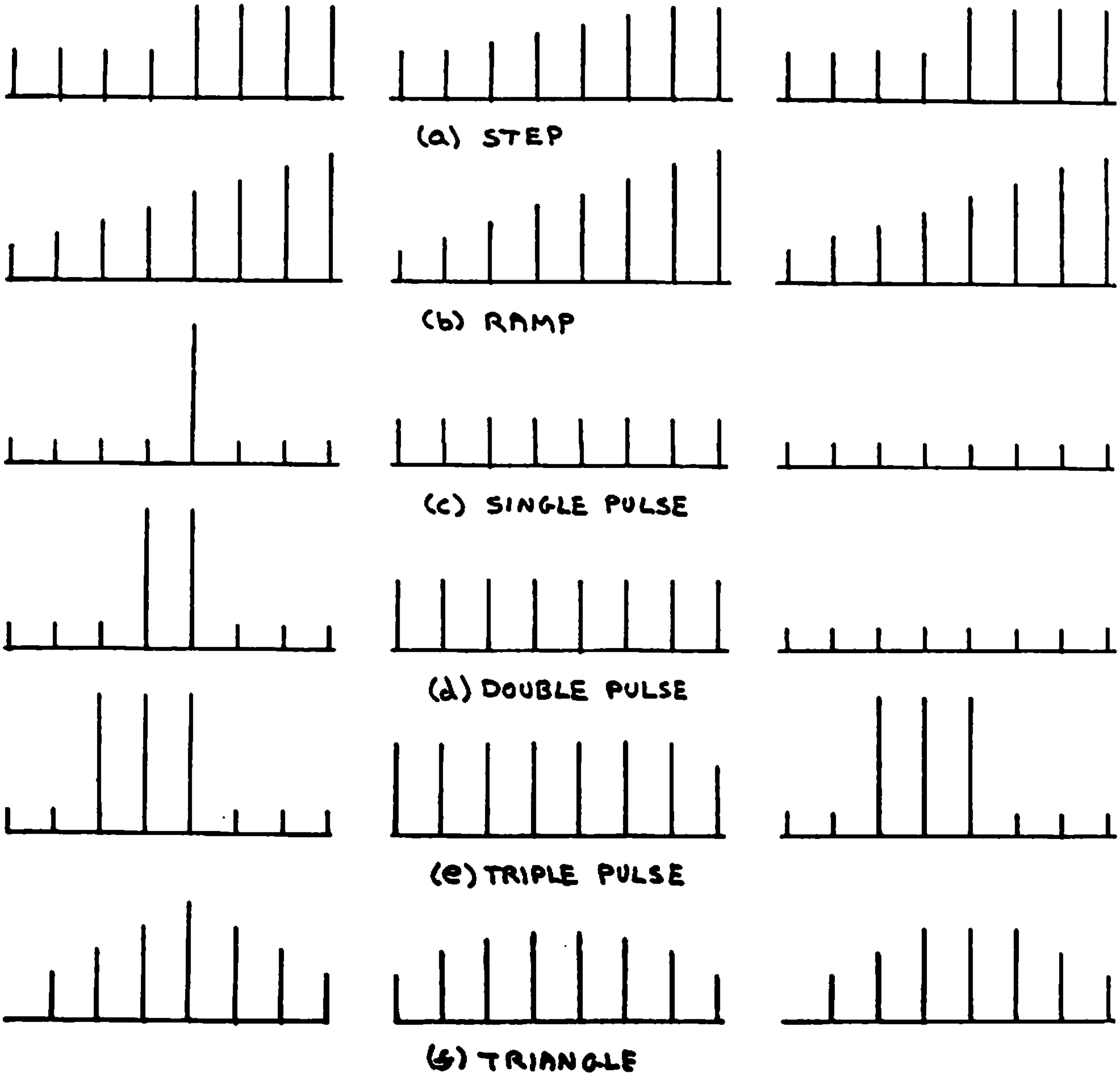


Figure 4.8-1. Examples of median filtering on primitive signals - L = S.

their medians. That is, in general

$$\text{med}\{f(j)+g(j)\} \neq \text{med}\{f(j)\} + \text{med}\{g(j)\} \qquad (3)$$

The sequences 80,90,100,110,120 and 80,90,100,90,80 are examples for which the additive linearity property does not hold.

There are various strategies for application of the median filter for noise suppression. One method would be to try a median filter with a window of length 3. If there is no significant signal loss, the window length could be increased to five for median filtering of the original. The process would be terminated when the median filter begins to do more harm than good. It is also possible to perform cascaded median filtering on a signal using fixed or variable length window.

The concept of the median filter can be easily extended to two dimensions by utilizing a two dimensional window of some desired shape such as a rectangle or a discrete approximation to a circle. It is obvious that a two dimensional L x L median filter will provide a greater degree of noise suppression than sequential horizontal and vertical processing with L x 1 median filters. But, two dimensional processing also results in greater signal suppression. Figure 2 illustrates the effect of two dimensional median filtering of a spatial pulse signal with a 3 x 3 square filter and a 5 x 5 plus sign shaped filter. In this example, the square median has deleted the corners, while the plus median filter has not affected the signal

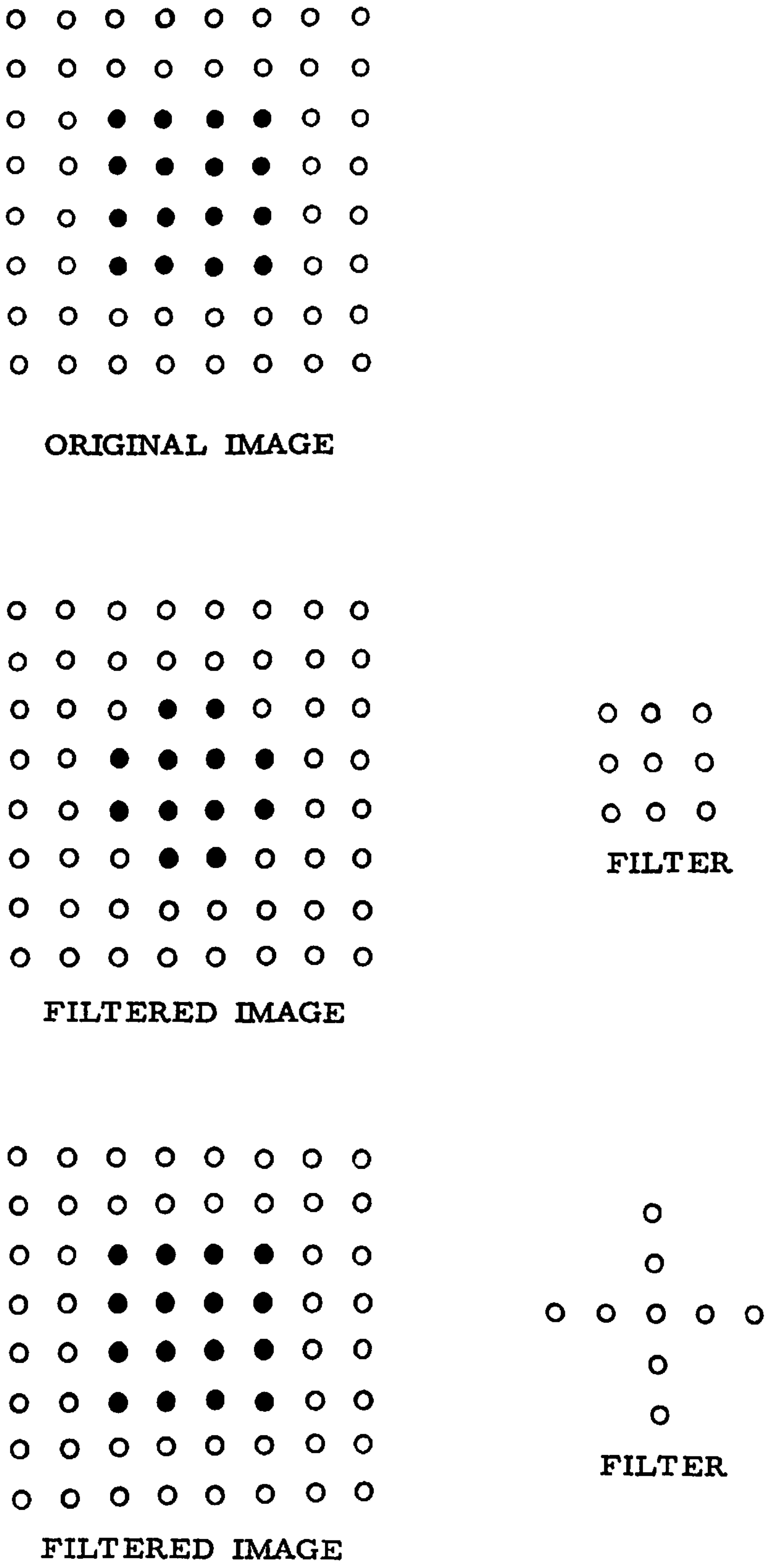


Figure 4.8-2. Example of two-dimensional median filtering

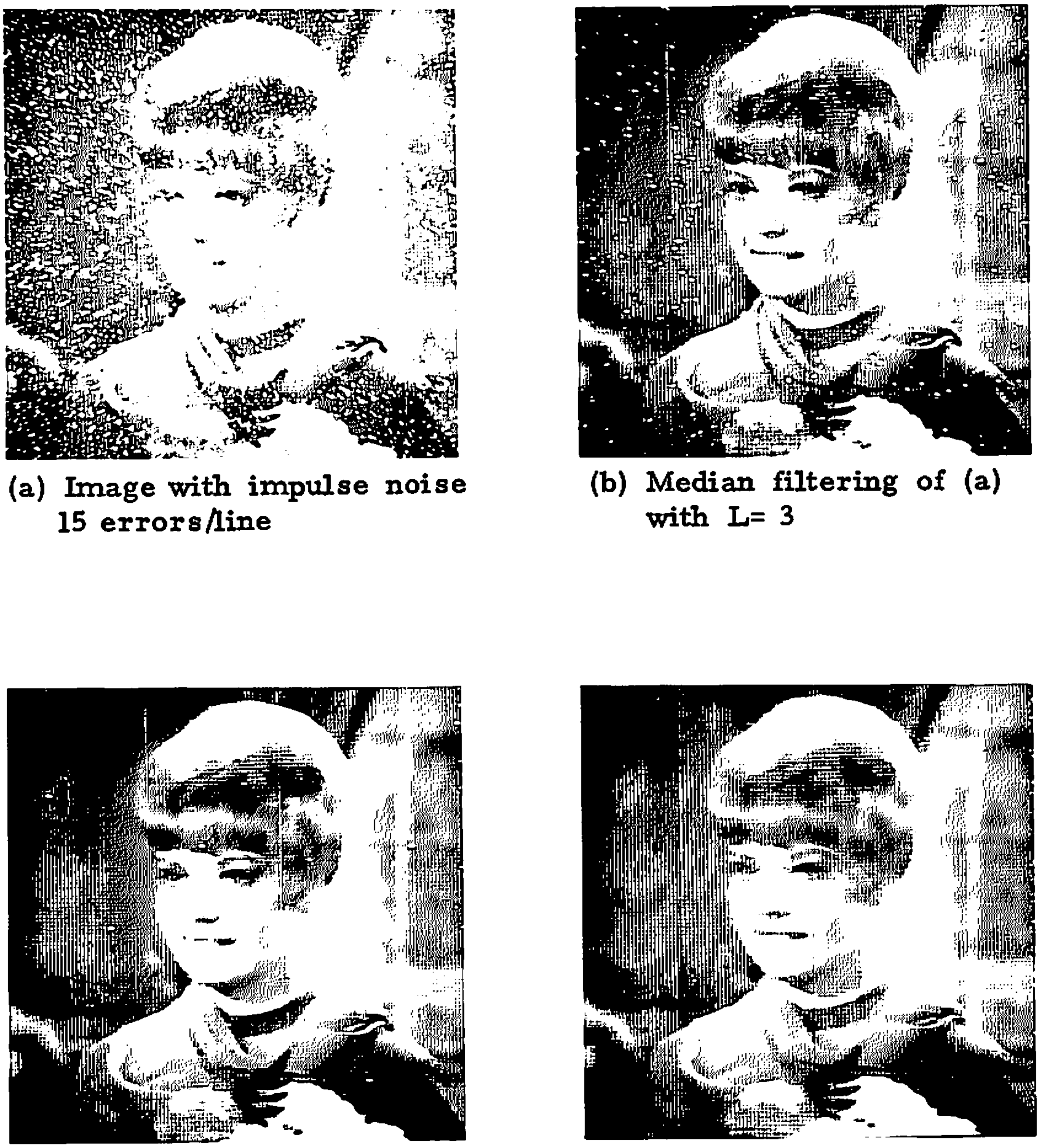
function.

Figures 3 and 4 contain examples of the application of median filtering for image noise suppression. In figure 3 impulse ncise was added to an image. One dimensional median filtering of length L=5 removed most of the noise impulses with only a small loss in resolution. Almost all errors were removed for a median filter with L=5, but edge distortion is noticeable. In figure 4 continuous Gaussian noise was added to an image. Median filtering resulting in a slight visual improvement.

For image enhancement applications, the median filter should simply be considered as an ad hoc tool for noise or interference suppression. It should not be used blindly, but rather its performance should be monitored to determine if its application is beneficial.

Reference

1. J. W. Tukey, Exploratory Data Analysis, Addison-Wesley, 1971.



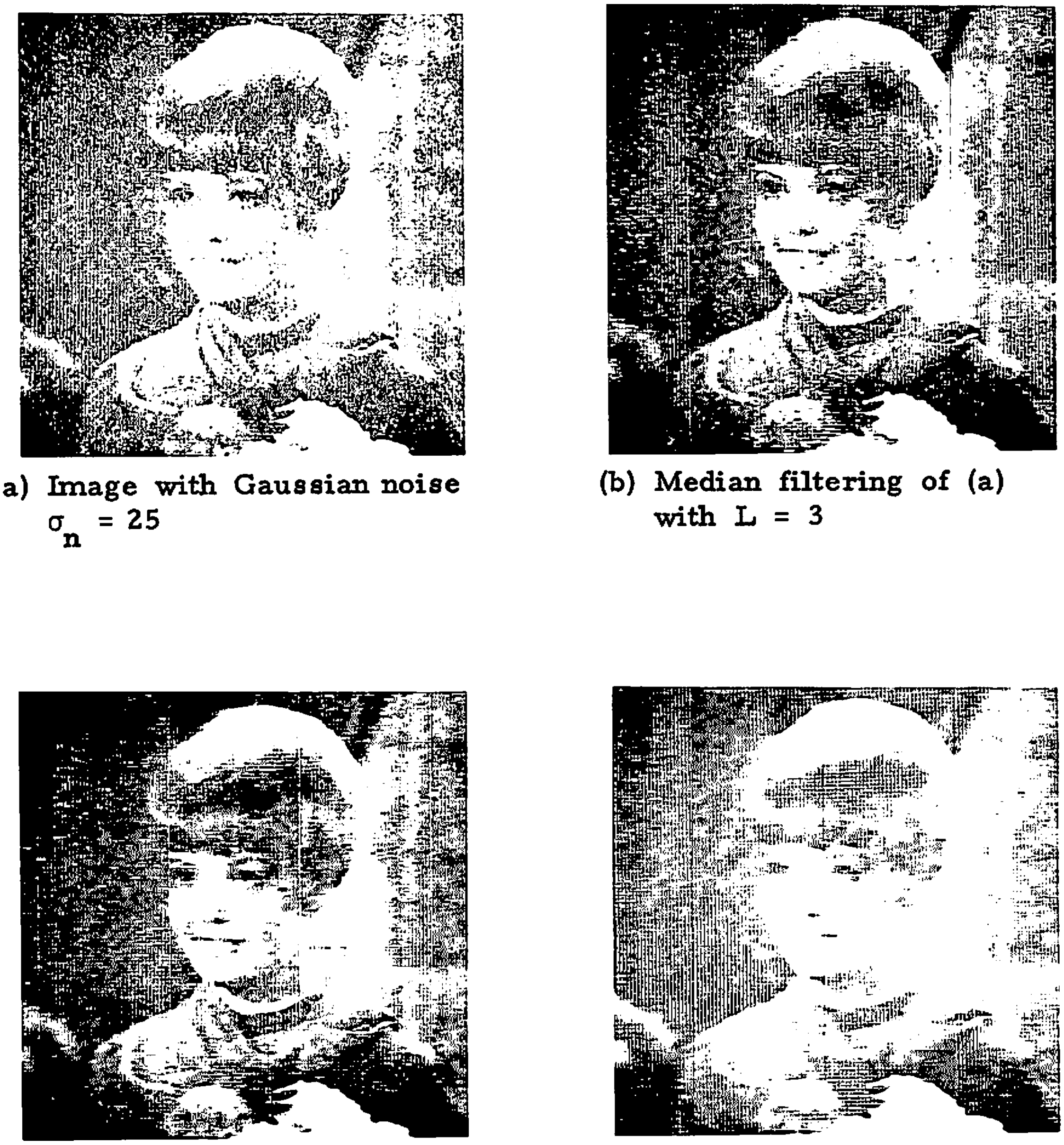
(a) Image with impulse noise 15 errors/line

(b) Median filtering of (a) with L= 3

(c) Median filtering of (a) with L = 5

(d) Median filtering of (a) with L = 7

Figure 4.8-3. Examples of one dimensional median filtering for images corrupted by impulse noise.



(a) Image with Gaussian noise $\sigma_n = 25$

(b) Median filtering of (a) with L = 3

(c) Median filtering of (a) with L = 5

(d) Median filtering of (a) with L = 7

Figure 4.8-4. Examples of one dimensional median filtering for images corrupted by Gaussian noise

5. Image Data Extraction Projects

Image data extraction activities include the extraction and measurement of image features, the detection of objects within pictures, the spatial registration of images, and the generation of images from one dimensional projections. Another facet of the effort covers image pre-processing operations which enable more efficient machine data extraction.

5.1 Textural Boundary Analysis

William B. Thompson

Previous reports have described the development of a textural distance function which accurately estimates the perceived dissimilarity between two textural regions. The textural distance function model allows the incorporation of textural cues into many of the existing approaches to scene segmentation. Texture may then be used, along with brightness, color, and any desired semantic processing in determining object boundaries. The utility of textural boundary detection will be demonstrated in an edge oriented system.

Many authors have developed edge finding systems which search for major discontinuities in the brightness function of the image [1]. This is normally done by computing an estimate of the derivative or gradient of the image and then finding the peaks in derivative function. Many functions have been suggested for this purpose. A common and often successful function is called the modified Roberts cross operator [2] and is defined as

$$R(i,j) = |p(i,j) - p(i+1,j+1)| + |p(i+1,j) - p(i,j+1)| \qquad (1)$$

The Roberts "gradient" is found by summing brightness differenes in two orthogonal directions. Many more sophisticated operators are possible. In particular, an operator which returns edge crientation may be quite useful.

A procedure has been developed to search for edges defined by textural properties in a manner similar to the Roberts operator. At specified intervals in the scene to be processed, four image regions arranged in a square were considered (see figure 1). The sum of the estimated perceived textural differences between regions a and d and between regions b and c was found. As with conventional gradient operations, it was postulated that larger values of this sum corresponded to textural edges running approximately through the intersection of the four regions. In addition, an edge direction was calculated. Let d(i,j) be the computed dissimilarity measure between two regions i and j (d(i,j)>0 for any two image regions). Then a textural boundary operator at the point in the scene shown in figure 1 may be defined as

$$T = d(a,d) + d(b,c) \qquad (2)$$

To determine the orientation of the edge, observe that

$$\text{ang} = \pm \arctan\left[\frac{d(a,d)}{d(b,c)}\right] \qquad (3)$$

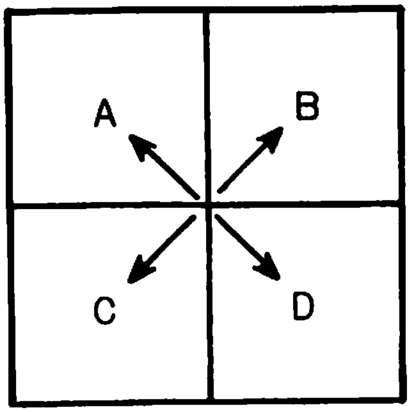


Figure 5.1-1. Template for textural edge operator

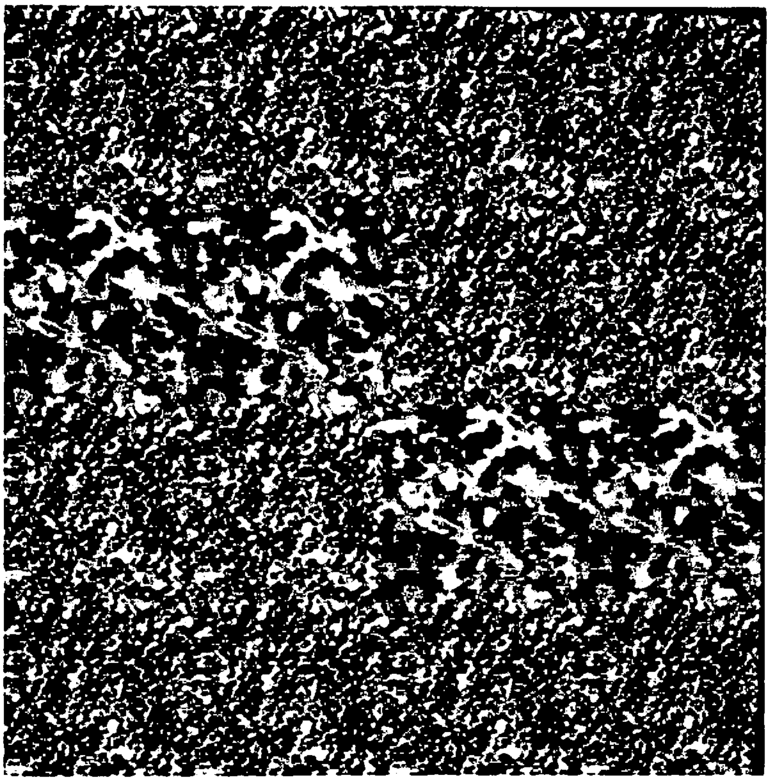
where ang = 0 implies an edge with negative slope at 45 degrees to the x-axis. Two angles are possible since d(a,d)=d(b,c) may correspond to either a vertical or horizontal edge. This ambiguity is straight forwardly resolved by considering d(a,c), d(b,d), d(a,b), and d(c,d).

In the current system, an edge map is first produced by applying the textural boundary operator at selected points in an image. A second edge map is produced by smearing each point in the first map along the direction of edge orientation. This is done to emphasize collinear edges. Finally, actual edge points are isolated by locating "ridge points" in the edge map. A ridge point is defined as an image point sufficiently greater than its neighbors along some direction. Much of the code to process the edge maps was adapted with little modification from a system originally designed to operate only on intensity information [4].

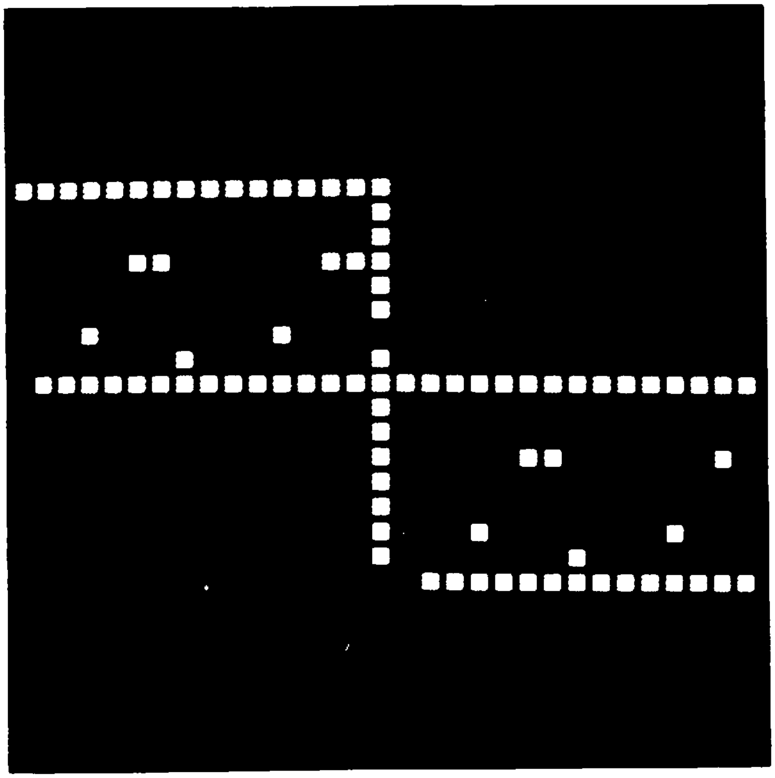
While most analysis systems designed to operate on natural imagery will use texture as only one of a set of multiple cues to determine image organization, some way is needed to evaluate the utility of the textural boundary operator on its own. As a result, this operator was applied to pictures in which the edges could be described as "purely textural." These test images were created as mosaics of textural patterns taken from pictures of natural scenes. Each component of the mosaic was normalized in the same manner as the patterns used in the resolution experiments. Thus, it was impossible to distinguish patterns based on average brightness or contrast criteria.

Figure 2 shows a representative mosaic pattern. Note that to a human observer, there are several quite prominent edges. Thus, it is clear that human perception can identify boundaries on criteria other than differences in average brightness. Figure 2a is another mosaic pattern. Figure 3b indicates the different textural regions present in figure 3a. In figure 3a, a very prominent boundary exists between patterns a and b. The boundary between b and d is relatively noticeable while the edge between a and d is hardly detectable. Region c may be viewed at one level as a uniform textural region. On another level, however, the region may be thought of as being composed of many smaller regions corresponding to the predominantly light and predominantly dark areas in the pattern.

The textural edge operator was applied to these and several other mosaic patterns using several different sizes for the basic blocks in the operator (i.e. the blocks in figure 1). The original mosaics were 256 by 256 picture elements in size. Figure 4 is an edge map for figure 3a using a basic block size of 16 by 16 picture elements. No post-processing other than the oriented smearing (e.g. edge linking, noise cleaning, etc.) was applied. An effective job has been done at identifying the visually prominent boundaries in the mosaic. The textural resolution experiments would indicate, however, that it should be possible to achieve higher resolution. Thus, it is possible to use block sizes as small as 6 or 8 pixels on a side. Figure 2b is an edge map for figure 2 using an 8 by 8 basic block size, all of the perceived boundaries have been well located. Figure 4a is an edge map for the mosaic in figure 3a using the same 8 by 8 basic block size.

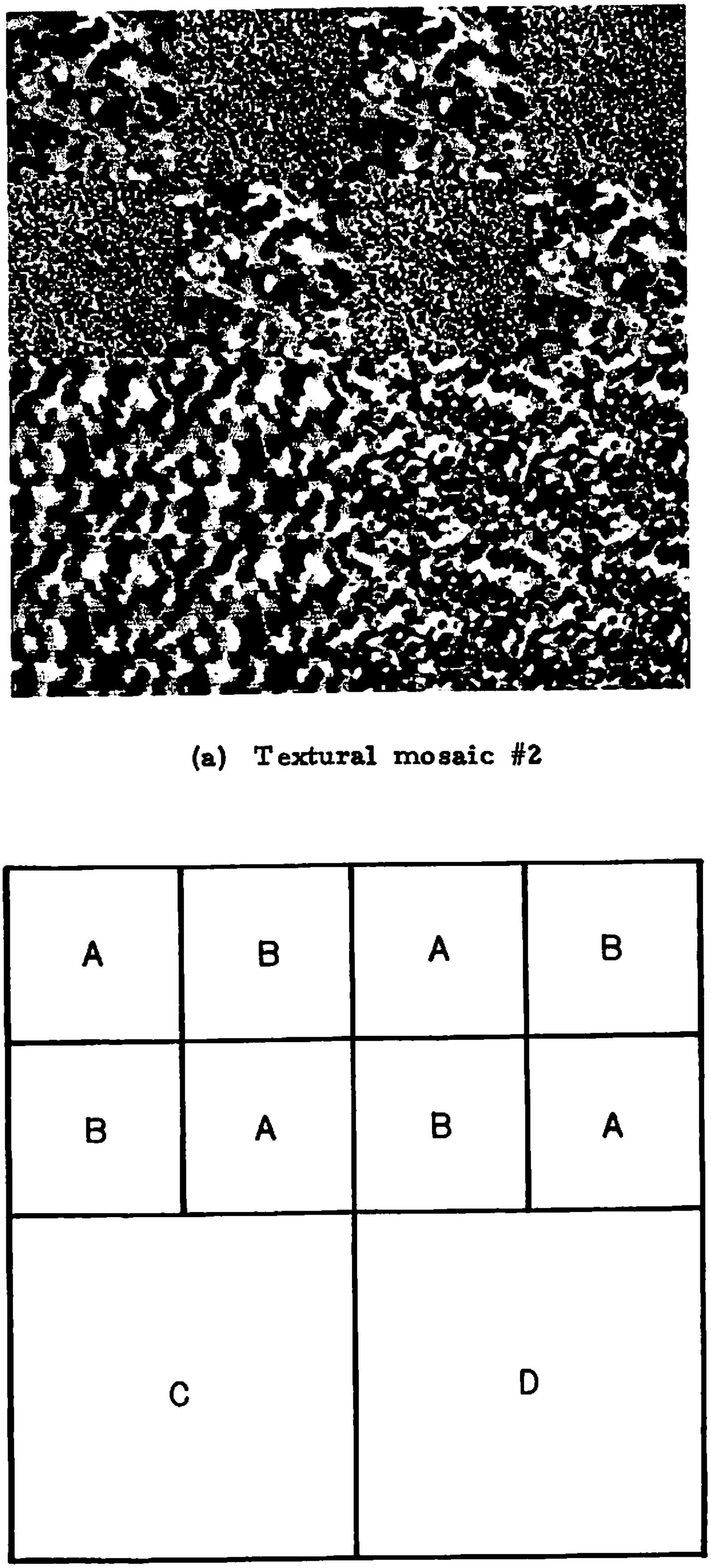


(a) Textural mosaic #1



(b) Edge map for (a) using 8 x 8 regions

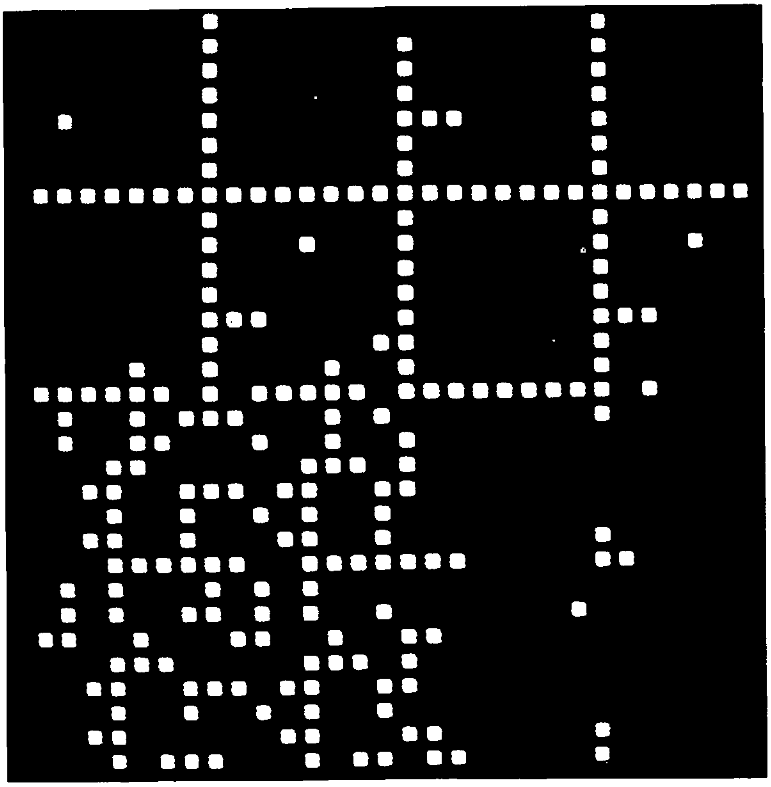
Figure 5.1-2. Examples of textural mosaics with edge map.



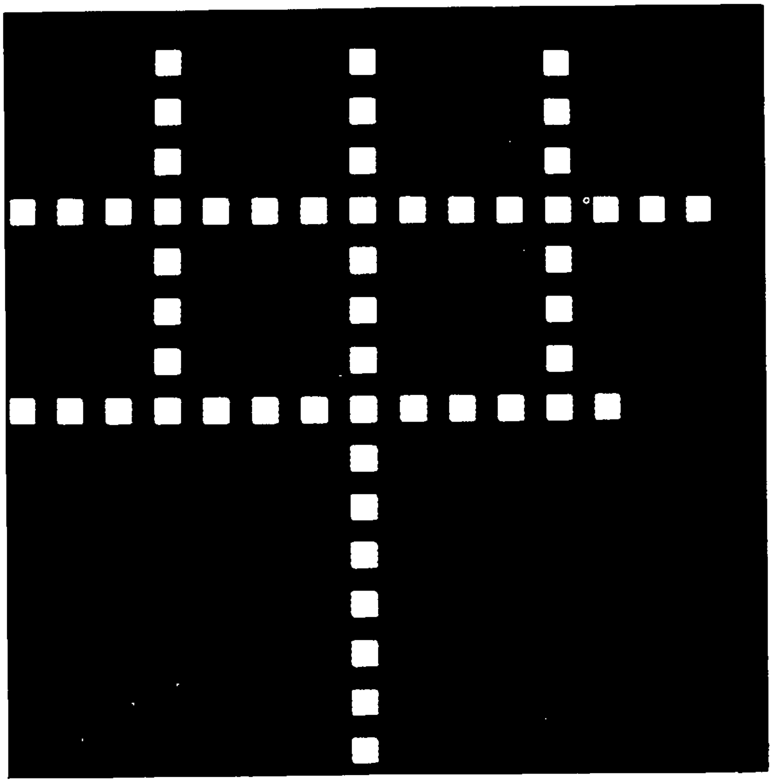
(a) Textural mosaic #2

(b) Identification of regions in (a)

Figure 5.1-3. Textural mosaic with region identification



(a) Edge map for figure -3a using 8 x 8 regions.



(b) Edge map for figure -3a using 16 x 16 regions.

Figure 5.1-4. Edge map differentiation using 8 and 16 block regions

Again, the boundaries are well identified. The operator completely degenerates in region c, however. A look at the original picture will show that many of the elementary light and dark areas are of comparable size to the 8 by 8 basic block. Thus, at this resolution, the micro-edges are a dominant effect. This is another example of the importance of realizing that perceived edges have a "size" associated with them that is a function of the size of the objects being searched for. Comparable results were obtained on the other mosaic test patterns.

A difficulty with many of the problems in automated image description is that it is often almost impossible to quantify the success of any given approach. For example, the utility of a particular object isolation procedure is really only meaningful in the context of the processing to follow. Unfortunately, the nature of the problems are so complex as to make development of completed systems most difficult. As much of automated scene analysis involves the simulation of perceptual effects, the development of lower level operators described in this report has used human visual perception as a performance goal.

The existence of readily perceived textural edges should be apparent. In many cases, existing automated systems which depend on identifying brightness discontinuities will fail to find these edges. This report has demonstrated a way in which measures of textural dissimilarity may be incorporated into scene segmentation systems. A textural edge operator is developed which is able to accurately locate boundaries of a purely textural nature.

The size of the region over which a textural pattern is measured has a significant effect on how well that texture can be characterized. Experimental results show that a dominant influence on human textural resolution is the nature of the patterns surrounding the region of interest. There is a well defined trade off between spatial resolution of a textural boundary and the ability to distinguish between visually similar textures. The structural interpretation of textural patterns suggests several additional methods for estimating minimal resolution regions. Unfortunately, at least one of these measures (an auto-correlation ratio) is not supported experimentally. The performance of the textural edge operator for varying region sizes corresponds closely to the predicted visual response from the resolution experiments.

References

1. R.O. Duda and P.E. Hart, Pattern Classification and Scene Analysis, John Wiley and Sons, New York,, 1973.

2. L.G. Roberts, "Machine Perception of Three-Dimensional Solids," Optical and Electro-Optical Information Processing, J.T. Tippett, et. al., eds., Cambridge, Massachusetts: M.I.T. Press, 1965, pp. 159-197.

3. M. Hueckel, "A Local Visual Operator which Recognizes Edges and Lines," JACM, Vol. 20, No. 4, October, 1973, pp. 634-647.

4. E.L. Hall, G. Varsi, W.B. Thompson, and R. Gauldin, "Computer Measurement of Particle Sizes in Electron Microscope Images," to appear in IEEE Transactions on Systems, Man and Cybernetics.

5.2 Image Segmentation by Boundary Determination

Ram Nevatia

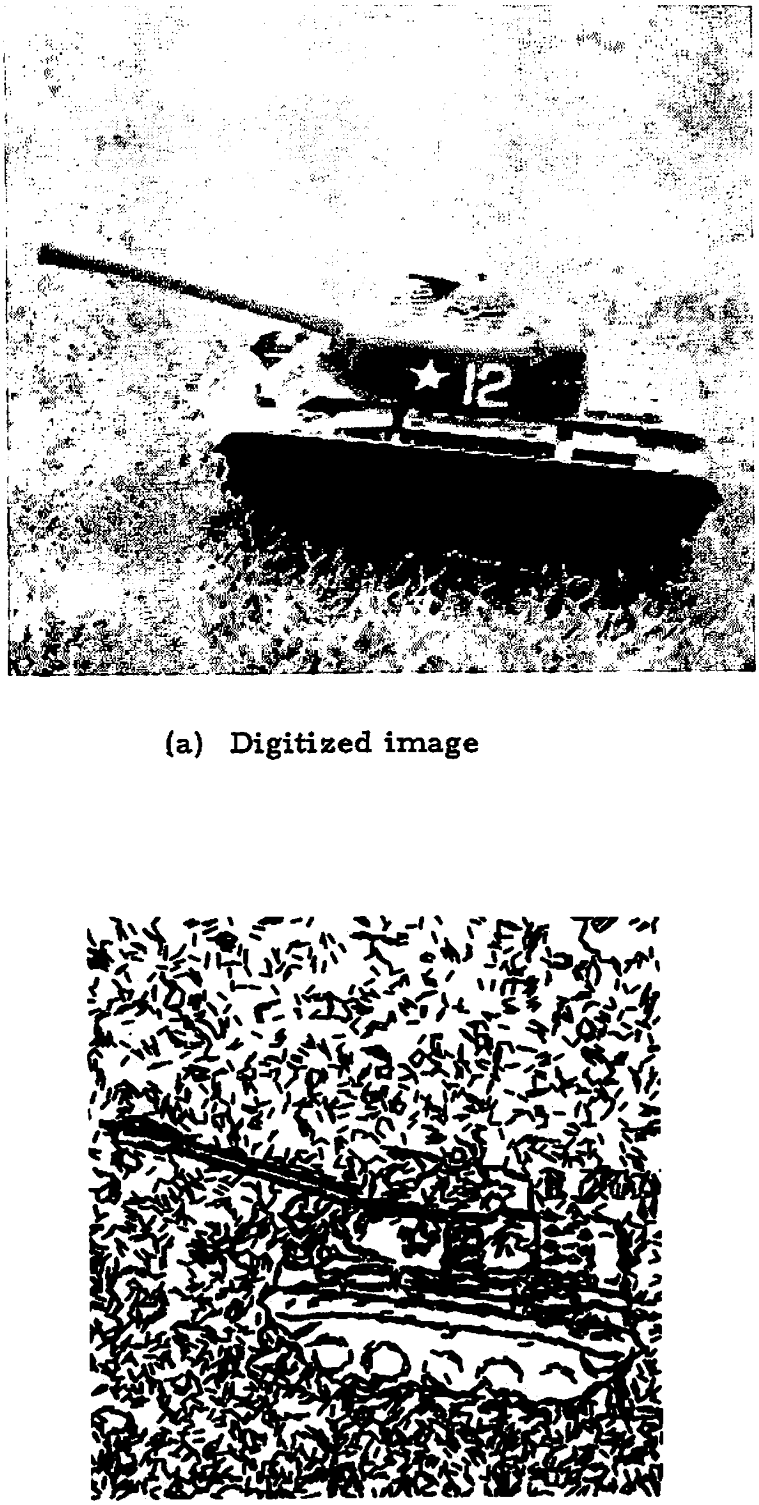
Finding boundaries of objects in an image is a major concern of scene analysis. The boundaries consititute a segmentation of the scene. Conversely, the boundaries may be derived from a given segmentation. A number of segmentation techniques have been suggested in the past, differing in their assumptions about the contents of the scene and in their control structure.

Using detailed specific knowledge of the objects likely to be present in an image simplifies the segmentation process [1-2], but these techniques suffer from loss of generality. Another distinction between various techniques is in their control structures, such as "top-down" vs. "bottom-up." The former treat an entire image as one object and successively sub-divide it into more parts as needed [3-4]; the latter start from small atomic regions (as small as a single pixel) or local edges and build larger parts from them.

The bottom-up techniques are usually referred to as being "edge" oriented or based on "region growing." The edge based techniques depend on detecting a discontinuity between some properties, such as brightness or color, of parts of an image and connecting these discontinuities to form boundaries. Region growing proceeds by clustering image points of "similar" properties in regions and further merging of regions of similar properties until a satisfactory segmentation has been obtained. Knowledge of image properties has been used to guide the merging of regions [5].

The edge based approaches were initially used for analysis of the scenes of polyhedral objects, the so-called "blocks world." The individual objects were of uniform, homogeneous surfaces and were seen against a uniformly light or dark background. Here, the edges detected by a local edge operator usually correspond to the desired object edges only. However, for more complex scenes, the local discontinuities do not necessarily correspond to the object boundaries only; shadows, surface imperfections and texture, and noise in the imaging devices being some of the causes.

Consider the picture in figure 1a showing a toy tank against a background of grass. Note the wheels of the tank are not visible in figure 1a because of display limitations. Figure 1b shows the intensity edges detected from figure 1a, by the application of a local edge detector, known as a Hueckel edge operator [6], at every second pixel in every other row of the image. This operator detects the presence of an edge in a circular neighborhood and returns the position as well as a direction for the edge. Figure 1b contains a large number of edges, most of which do not belong to the desired boundary of the tank. However, humans presented with this edge picture have no difficulty in perceiving the tank. The edges along



(a) Digitized image

(b) Edges detected in (a)

Figure 5.2-1. Edge detection for a picture of a toy tank.

the tank boundary connect in a coherent way, whereas the edges in the grass region are seen as being randomly distributed.

An algorithm to find groups of edges that connect in an approximate straight line, to be described later, is very successful in separating the tank boundary from the background for the above example. This method of segmentation has the advantage of being general, as no specific objects in the scene are assumed. Also, the schemes using texture properties defined over a region are sensitive to the choice of the region size, and it is difficult to locate the boundary accurately within a region.

The choice of linking edges into straight lines was based on the computational efficiency of this process. Many man-made and natural objects have boundaries with elongated segments. Further, any curve can be represented by piecewise linear segments; the linking algorithm only imposes a constraint on the maximum curvature of the segments linked.

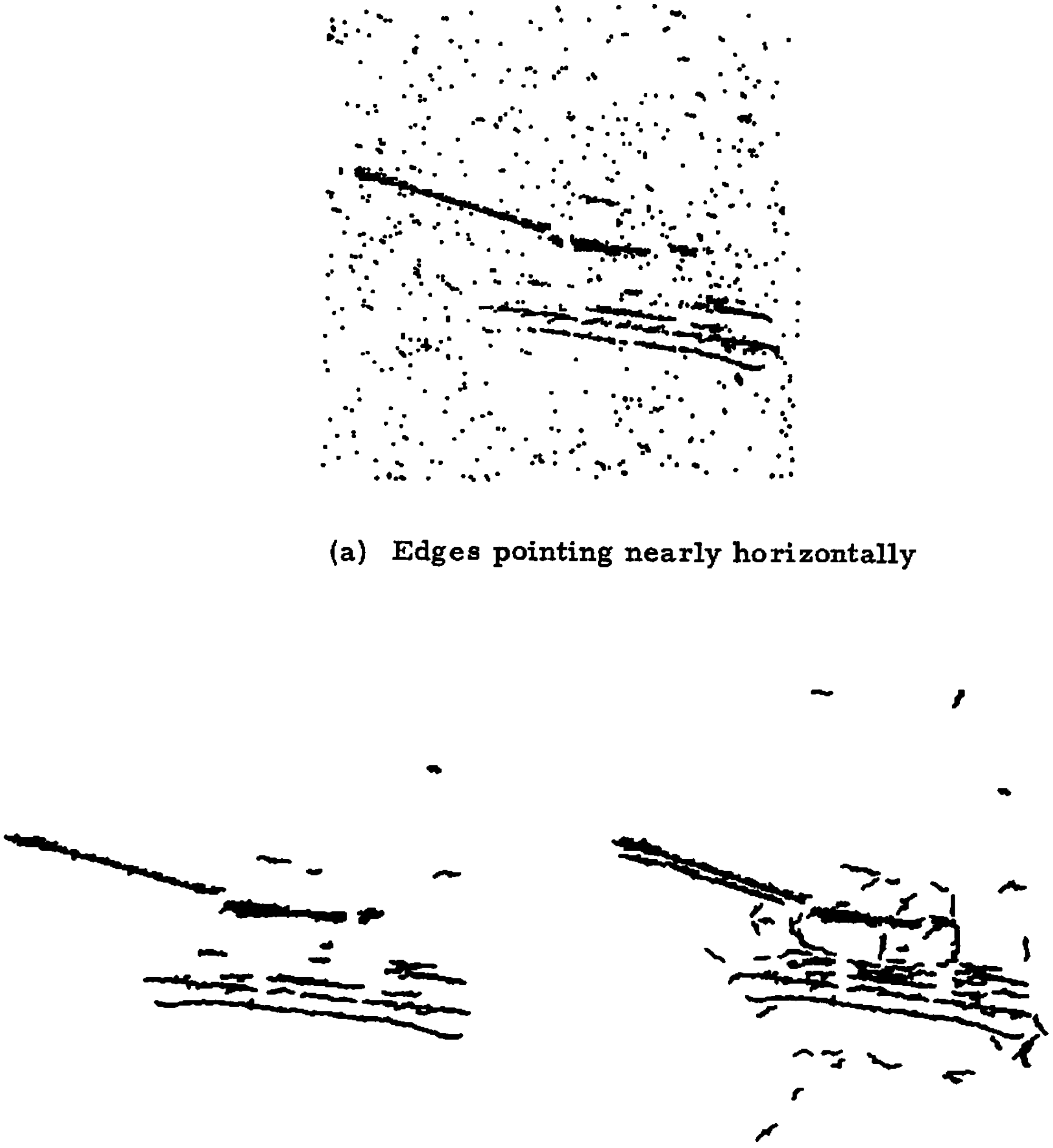
Linking Algorithm: Much work in the past has been concerned with linking local edge elements into straight line segments. Two broad classes of techniques are based on the use of the Hough transform [7-10], or the use of graph theoretic methods [11-12]. However, these techniques have been used in situations where the number of edge elements is small and most of these elements belong to the desired boundaries. Their effectiveness for the problems considered here is unclear, and in some cases the computational costs are likely to be unacceptable (e.g. the algorithms using minimal spanning trees, require computational times proportional to the cube of the number of edge elements to be linked). A detailed review may be found in [13]. A description of the algorithm developed follows.

For this discussion, each edge element, e_i, is considered to have a position p_i and an associated direction α_i. Two oppositely directed edge elements are considered to have different directions (differing by 180 degrees). Length of an edge element, determined by the size of the local edge operator, is unimportant.

The entire 360 degrees range of directions is divided in a number of equiangular intervals (say 12). Linking of edge elements along directions in each interval is examined. Linking in a chosen interval is constrained to edge elements having directions approximately within this interval. The following are the steps, in detail, for linking in an interval whose median angle is, say θ_j.

1. Examine each edge element and put in a set E_j if the direction of the edge, α_i is within a fixed, chosen range, $\Delta\theta$ of the direction θ_j. Note that $\Delta\theta$ need not be the same as the width of the angular interval. Figure 2a shows the edge elements for the tank from figure 1b, which are within a 60 degree range of horizontal direction (θ_j = 0 degrees).

2. Transform the co-ordinates so that the new x-axis, lies along θ_j. Let (x_i', y_i') be the transformed co-ordinates of the i-th edge element in set E_j.



(a) Edges pointing nearly horizontally

(b) Linked segments from (a)

(c) Linked segments from all directions

Figure 5.2-2. Linking of edges

3. Divide the image plane in parallel strips (buckets) of a fixed size (say 3 pixels wide), normal to X' (figure 3 shows schematically some buckets, with the rotated X-axis displayed horizontally). Each edge element e_i in E_j will fall into one of the buckets, determined by the co-ordinate x_i'. Store the edge elements in each bucket in a list ordered by the value of the y' co-ordinate.

4. Link edges in each bucket: If two consecutive edge elements in the edge list for a bucket differ in their y' co-ordinates by a distance smaller than a threshold TY, say 2 pixels, then the two elements belong to a common segment. e.g., bucket 2 in figure 3 is divided into segments S_1 , S_2, S_3.

5. Link segments in neighboring buckets: If the end points of two segments in adjacent buckets are within a distance of TY in their y' co-ordnates and also within a distance of TX in their x' co-ordinates, then the two segments are merged into one. Also, the merging must not result in a change of orientation of the segment, e.g. in figure 3, S_4 and S_7, or S_5 and S_8 are merged but not S_6 and S_9.

6. Retain only segments of a length exceeding a fixed number (say 7).

Figure 2b shows the linked segments resulting from the edge elements of figure 2a, using the thresholds indicated in the description of the algorithm above. Figure 2c shows linked segments

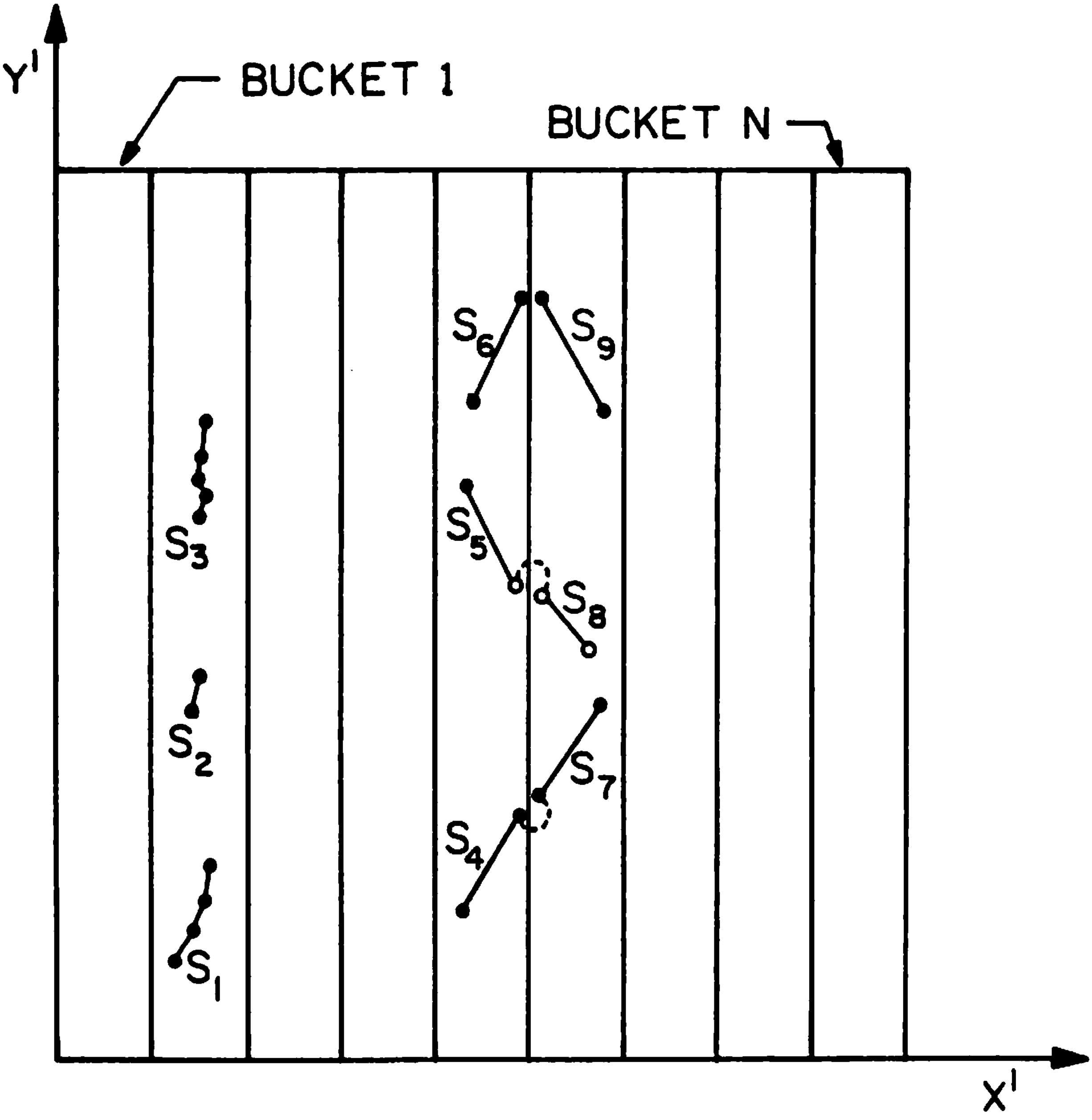


Figure 5.2-3. Schematic display of some buckets and segments.

from 12 intervals covering the entire 360 degrees range. Note that segments from different intervals are not linked though they appear so in the figure, and some edge elements are connected to more than one segment. Resclution of such overlaps and linking of intersecting inter-interval segments is straight-forward.

The above described algorithm uses many thresholds at various steps. However, the algcrithm is relatively robust to these choices and the programs work well on widely different scenes without changing these thresholds. The same program, without change of thresholds has been tried on different images, including the problem of rib detection in a chest X-ray, with encouraging results. The details of the basis of choice of threshclds are found in [13].

Computational Complexity: The various steps of this algorithm require the processing of an edge element either in isolaticn or in comparison with its immediate neighbors in an crdered list. Thus all computing costs are linearly proportional to the number of edges processed, except for the possible costs of sorting the edge lists in step 3 above.

The number of edges in any single bucket is normally a small propcrtion of the total number of edges. Taking advantage of the initial raster order of the edges, the sorting time can be limited to increase only linearly with the number of pixels in the image. The sorting details are not discussed here.

For the example of the tank, the total time to link in 12 directions was 20 seconds on a PDP-KI10 processor. The programs are written in the SAIL language. The total number of edges detected was about 5000. The maximum memory requirements were about 50K, 36 bit words.

The techniques described are limited to discovering elongated segments of edge boundaries. These segments have to be connected to form complete object boundaries. There is sufficient information to connect these segments as evidenced by our ability, as humans, to do so (in figure 1b for example) without recourse to the original grey level picture. The segments cannot be simply connected to their nearest neighbors; some notion of preferred configurations is required. Two long parallel segments are often boundaries of opposite sides of a part of an object; e.g., see the boundaries of the barrel of the tank in figure 1b. Information obtained by other forms of analysis of the image, such as texture or color analysis, will aid in the connection of these segments. Alternatively, these segments may be used to aid in such analysis.

References

1. R. Bajcsy and M. Tavakoli, "A Computer Recognition of Bridges, Islands, Rivers and Lakes from Satellite Pictures," Proceedings of the Symposium on Machine Processing of Remotely Sensed Data, Purdue University, October, 1973.

2. J.M. Tenenbaum, "On Locating Objects by their Distinguishing Features in Multisensory Images," Computer Graphics and Image

Processing, Vol. 2, No. 3/4, December, 1973, pp. 308-320.

3. R.B. Ohlander, "Analysis of Natural Scenes," Ph.D. Thesis, Computer Science Department, Carnegie Mellon University, Pittsburg, Pennsylvania.

4. S. Tsuji and F. Tomita, "A Structural Analyzer for a Class of Textures," Computer Graphics and Image Processing, Vol. 2, No. 3/4, December, 1973, pp. 216-231.

5. J.A. Feldman and Y. Yakimovsky, "Decision Theory and Artificial Intelligence: I. A Semantics-Based Region Analyzer," Artificial Intelligence, Vol. 5, No. 4, Winter, 1974, pp. 349-372.

6. M.H. Hueckel, "A Local Visual Operator Which Recognizes Edges and Lines," Journal of the ACM, Vol. 20, No. 4, October, 1973, pp. 634-647.

7. A.K. Griffith, "Edge Detection in Simple Scenes Using A Priori Information," IEEE Transactions on Computers, Vol. C-22, No. 4, April, 1973, pp. 371-381.

8. R.O. Duda and P.E. Hart, Pattern Recognition and Scene Analysis, John Wiley and Sons, New York, 1973.

9. W.A. Perkins and T.O. Binford, "A Corner Finder for Visual Feedback," Computer Graphics and Image Processing, Vol. 2, Nos. 3/4, December, 1973, pp. 355-376.

10. S.D. Shapiro, "Detection of Lines in Noisy Pictures Using

Clustering," Second International Joint Conference on Pattern Recognition, August 13-15, 1974, Copenhagen, Denmark, pp. 317-318.

11. E.V. Ramer, "Computer Edge Extraction from Photographs of Curved Objects," New York University Technical Report CRL-34, December, 1973.

12. C.T. Zahn, "Graph-Theoretical Methods for Detecting and Describing Gestalt Clusters," IEEE Transactions on Computers, Vol. C-20, No. 1, January, 1971, pp. 68-86.

13. R. Nevatia, "Object Boundary Determination in a Textured Environment," (to be presented) Annual ACM Conference, October, 1975, Minneapolis.

5.3 Color Edge Detection

Ram Nevatia and William D. Miller

A digital image may be represented as a matrix of values of a function I(x,y), defined at digitized points in the image. For a black and white image, I is a scalar valued function, corresponding to the brightness of the image at the digitized points. For a color image, I is a vector valued function having three components, say I_R, I_G and I_B, the intensity values in the red, green and blue color bands respectively.

In a black and white image, an edge is defined by a discontinuity in the scalar valued function I(x,y). An edge in a color image may be defined in several ways. If a metric were defined on the vector space spanned by I, edges could be detected in the new scalar space. Note, this is similar to reducing the color image to an equivalent grey level image. Alternatively, edges may be detected in the three components I_R, I_G and I_B of I independently and a single edge determined from their combination. A scheme for color edge detection is developed in the following.

First consider the details of edge detection in a single grey level image. It is useful to consider an edge as having a position and also a direction (a magnitude reflecting the discontinuity may also be included). A simple gradient operation followed by thresholding provides such edge output. An edge is often limited to belong to certain classes of discontinuities, e.g. a step-like or a line-like discontinuity. Consider step edges only. Edge detection may then be viewed as the best fit of a neighborhood of an image by a step function, and requires determination of the position, orientation and the magnitude of the step. Decision of the presence of an edge is based on the size of the step (and perhaps the quality of the fit).

It was suggested by Binford [1], that a color edge be determined by making best fits to the three functions I_R, I_G and I_B separately, but constraining the orientation of the step to be the same for all three components, and the decision of the presence of an edge based on the magnitudes of the three steps.

A popular edge detector for black and white image has been developed by Hueckel [2]. This operator determines the presence of an edge in a circular neighborhood and provides the position, orientation and th9 magnitude of the edge. Briefly, it proceeds by approximating the circular neighborhood by expansion in a finite number of terms of an orthogonal series of functions. Hueckel claims the chosen series to be optimal under certain assumptions. Let a_i be the coefficients of the expansion for a given neighborhood (i ranges from 0 to 7).

A best step function is fit to the approximated function next. A step function, parameterized by a tupl9, is expanded in the same series to yield coefficients s_i (tuple). Th9 parameters of the step are chosen to minimize the function

$$N^2 = \sum_{i=0}^{7} [a_i - s_i(\text{tuple})]^2 \qquad (1)$$

An attractive part of Hueckel's approach is that analytic solutions to this minimization problem can be found, avoiding expensive searches. In particular, the orientation of the optimal step can be determined independently of other parameters.

To extend this concept to a color edge, the function to be minimized may be formulated as

$$N^2 = N_R^2 + N_G^2 + N_B^2 \qquad (2)$$

The functions N_R, N_G and N_B are as defined in eq.(1) for the three components of the image I, i.e.

$$N_R^2 = \sum_{i=0}^{7} [a_{i_R} - s_i(\text{tuple})]^2 \qquad (3)$$

where the subscript R refers to the red component and similar expressions exist for N_G and N_B.

The minimization process now requires determination of three tuples of parameters, with the constraint that all three have the same orientation parameters. Again, it turns out that the orientation parameter can be determined independent of the other parameters. Further, once the orientation has been determined, the parameters in one tuple can be determined independently of parameters in the other tuples.

The algebraic details of the derivation are not presented here. A black and white, Hueckel edge operator program, coded in assembly language, has been in use at USC since last year. It is possible to use many parts of this program, as they are, in the development of a color edge operator. This new program is now being developed and debugged.

Other interesting considerations for color edge detection are in the weightings of the steps obtained for the three color components. It is expected that transformations of the R-G-B space to another three dimensional space, which is claimed to be Euclidean, based on models of human perception developed at USC [3], should aid in this task.

References

1. T.O. Binford, Private Communication.

2. M.H. Hueckel, "A Local Visual Operator which Recognizes Edges and Lines," Journal of the ACM, Vol. 20, No. 4, October, 1973, pp. 634-647.

3. W. Frei, "A Quantitative Model of Color Vision," USCIPI Report 540, September, 1974, pp. 69-83.

5.4 Image Boundary Estimation*

Nasser E. Nahi and Mohammad Jahanshahi

In visual perception, among the most effective stimulus configurations are the "edges" outlining objects within an image, [1]. This has motivated many researchers in the area of automated image processing, specifically scene analysis, to develop various techniques of edge detection and boundary estimation. An incentive for research in scene analysis is the study of robotics [2]. The available information about the shapes and sizes of physical objects constitute and total visual intelligence required by a robot. Such information can be provided through knowledge of object boundaries.

The oldest method known for boundary determination is that of thresholding [3]. This method, along with the later procedures of

*This research was partially supported by National Science Foundation ENG 75-03423.

locating the maximum gradients, are well known to be highly sensitive to the sources of degradation phenomenon [4]. Various refinements of the above methods, which to some extent account for the presence of noise, have been recently introduced [5].

In this report, a boundary estimator is introduced for a certain class of noisy images. The images considered contain an object of interest within a background. Defining the set of points which separate the object and the background as "object boundary," a recursive estimator is designed to yield an estimate of the object boundary. Extensions of the estimator to multi-object images are discussed. The performance of the estimator is illustrated through applications to a few images.

<u>Problem Statement</u>: Consider the class of images which can be partitioned into two regions: background and foreground. The foreground is assumed to form a "horizontally convex" object. Given a noisy version of such an image, the aim is to obtain an estimate of the object boundary.

<u>Modeling of Images by Replacement Processes</u>: An image whose grey level values, denoted by a two-dimensional function b(n,m), are unknown is commonly modeled by the given first and second order statistics of b(m,n). Literature in the area of digital image restoration includes use of this information, along with a set of observations, to derive a set of estimates (often a minimum mean square estimate) for b(m,n) [6,7]. However, consistent in the results has been the presence of blurry edges. Intuitively, it may be

ccncluded that an image model based solely on the first two moments of $t(m,n)$ might be suitable for reccnstruction of image grey level values, but it does not carry sufficient information to adeguately reconstruct the cbject boundary.

A model for the image signal $b(m,n)$ which explicitly represents the object boundary along with the background and object internal details is given by

$$b(m,n) = \gamma(m,n)b_o(m,n) + [1-\gamma(m,n)]b_b(m,n), \qquad (1)$$

where b_o and b_b represent the intensity values of the object and the background, respectively, and γ carries the boundary information of the object within the image. The two-dimensional functions $b_o(m,n)$ and $b_b(m,n)$ are assumed to be sample functions of two statistically independent, wide sense stationary random processes whose first two mcments are given. The mean values of b_o and b_b are indicative of the object and the background brightness similarities, whereas, their respective autocorrelation functions are measures of the object and the background textural information.

The binary valued function $\gamma(m,n)$, another random process, takes values of 1 or C corresponding to the points in the image belonging to the object or the background, respectively. In the literature, this function is usually known as the image "characteristic function" [8]. The statistical properties of γ will be described shortly.

The image model mathematically represented by eq.(1), is based on a concept called a "replacement process" where, by definition, a segment of a function or a random process is replaced by another function or random process according to a certain rule [9]. Considering that for typical images the object signal, in fact, "replaces" a portion of the background signal, the structure of this model is justified. In the model of eq.(1), replacement of the object process b_o with the background process b_b takes place according to the values of γ.

For future reference, note that the domains of the sample functions $b_o(m,n)$ and $b_b(m,n)$ are defined to be the entire image. This is, in fact, the main motivation behind introducing the concept of replacement processes in the image modeling.

A sequence of observations constructed as

$$y(m,n) = b(m,n) + \nu(m,n) \qquad (2)$$

are assumed available for measurement, where $b(m,n)$ is as defined by eq.(1), and $\nu(m,n)$ denotes an uncorrelated process representing the observation noise.

An image scanner will now be considered which transforms the two-dimensional data representing the noisy image, $y(m,n)$, into one-dimensional data. The scanner output, in the absence of observation noise, is denoted by $s(k)$, where

$$s(k) = \lambda(k)s_o(k) + [1-\lambda(k)]\, s_b(k) \qquad (3)$$

models the image in terms cf its grey level values and object boundary as viewed by the output of a line by line scanner.

The structure of the one-dimensional model of eq.(3) preserves the replacement processing concept. The functions $s_o(k)$ and $s_b(k)$ are associated with $b_o(m,n)$ and $b_b(m,n)$, respectively. That is, $s_o(k)$ and $s_b(k)$ denote the grey level values of the scanned object and background, and are assumed to be sample functions cf two statistically independent, cyclo-stationary randcm processes [10], whose first twc mcments are obtainable directly in terms of the first and second-order statistics of $b_o(m,n)$ and $b_b(m,n)$ [6]. As in the case of $b_o(m,n)$ and $b_b(m,n)$, the dcmains of the sample functions $s_o(k)$ and $s_b(k)$ are the entire scanned image.

The binary valued process $\lambda(k)$ is the one dimensional counterpart of $\gamma(m,n)$. Its statistics will be described below. Note that the statistics of λ ccmpletely define those of γ.

Let m_1 and m_2 indicate the first and the last lines of the object as viewed by the scanner, and α_ℓ, β_ℓ represent the beginning and end points of the object cn line ℓ, respectively. In general, m_1, m_2, α_ℓ, β_ℓ for $m_1<\ell<m_2$ are random.

The function $\lambda(k)$, appearing in eq.(3), is now defined in terms of α_ℓ and β_ℓ

$$\lambda(k) = \sum_{\ell=m_1}^{m_2} u[k-\alpha_\ell-(\ell-1)J] - u[k-\beta_\ell-(\ell-1)J] \qquad (4)$$

where u[ ] is the unit step function, J denotes the number of picture elements in one line of the image, and $\beta_\ell > \alpha_\ell$. The statistics of the process $\lambda(k)$ can now be given in terms of the statistics of m_1, m_2, and the sequence

$$W_\ell = (\alpha_\ell, \beta_\ell) \tag{5}$$

Assume that W forms a first-order Markov process. This assumption is made for the sake of computational simplicity, and it emphasizes the dependence of the object boundary points on line ℓ upon the points located on the previous line, $\ell - 1$. It is further assumed that the required density functions are given, and that

$$p(W_\ell \mid W_{\ell-1}, m_1, m_2) = p(W_{\ell-1}, m_1) \tag{6}$$

Notice also that

$$\begin{aligned} p(W_\ell \mid W_{\ell-1}, m_1) &= p(\alpha_\ell, \beta_\ell \mid \alpha_{\ell-1}, \beta_{\ell-1}, m_1) \\ &= p(\alpha_\ell \mid \alpha_{\ell-1}, \beta_{\ell-1}, m_1) \cdot p(\beta_\ell \mid \alpha_\ell, \alpha_{\ell-1}, \beta_{\ell-1}, m_1 \end{aligned} \tag{7}$$

The two dimensional observation sequence y(m,n) in eq.(2) will also be replaced by its scanned version given by

$$y(k) = s(k) + v(k) \tag{8}$$

where s(k) is as defined in eq.(3), and v(k) is a zero mean Gaussian white-noise process with variance σ^2.

To locate the object boundary, estimates of the first and last lines (m_1 and m_2) and estimates of the starting and ending points (α_ℓ and β_ℓ) of the object are sought. The estimation procedure developed here, as will be shown, requires the values of $s_o(k)$, and $s_b(k)$, 1<k<N, where N is the total number of pixels (picture elements) in the image. Since, in general, these values are not known (cases of images with known grey levels are exceptional), the estimates of $s_o(k)$ and $s_b(k)$ will be used in their place. Such estimates can, for example, be obtained by implementation of the results in [11] where only two-dimensional statistical information on s(k), or y(k) is used. Notice that the concept of replacement processes assures the existence of the estimates of $s_o(k)$ and $s_b(k)$ for all 1<k<N. Since the aim of this paper is estimation of the object boundary, it will be assumed that the values s_o and s_b (or their estimates) are given.

The boundary estimation problem, as evident from eqs.(3) and (8), is a nonlinear estimation problem. Furthermore, due to the type of nonlinearities involved (such as the binary nature of λ(k)), the available estimators based on linearization concepts (such as extended Kalman filters) do not yield satisfactory results.

In this work, a set of maximum a posteriori (MAP) estimates for the unknowns m_1, m_2, α_ℓ, and β_ℓ are obtained. It is shown that the MAP estimates will minimize the following expression

$$\min_{\underline{m},\ \underline{w}} \{-2\sigma^2 \ell n\ p(m_2 | m_1) - 2\sigma^2 \ell n\ p(m_1) + \sum_{\ell=m_1}^{m_a} [T(w_\ell) - 2\sigma^2 \ell n\ p(w_\ell | w_{\ell-1}, m_1)]\} \qquad (9)$$

Numerical Derivation cf Estimates: Acquisition of a numerical sclution for the minimization process of eq.(9) is an integral part of this presentation. Since a rigorous sclution of eq.(9), resulting in a set of optimal estimates for m_1, m_2, α_ℓ, and β_ℓ, is computationally unacceptable, approximate solutions are sought. Two approaches, shown later to yield satisfactory results, are described in the following.

One approach is to obtain the estimates of α_ℓ and β_ℓ over the range $m_1<\ell<m_2$, with the assumption that values of m_1 and m_2 are given. For example, values of m_1 and m_2 may be chosen as $m_1=1$ and $m_2=M$, implying that the object boundary points lie on every line of the image. Then, if necessary, one may utilize additional structural properties of the cbject to eliminate those boundary point estimates incompatible with the given structural information.

An alternative approach is to consider the problem in two steps; namely, solve for $\hat{\alpha}_\ell$ and $\hat{\beta}_\ell$, $m_1<\ell<m_2$, for a selected set of m_1 and m_2; then solve for the estimates of m_1 and m_2 by replacing the estimates $\hat{\alpha}_\ell$ and $\hat{\beta}_\ell$ for α_ℓ and β_ℓ. A recursive procedure will result if these two steps are performed at each scan line resulting in an algorithm which yields a set of estimates for m_1, m_2, α_ℓ, and β_ℓ, concurrently.

The former approach is computationally more attractive. However, it requires additicnal information, of a nonstatistical gecmetric nature, on the object, beyond the given statistical information, to ccmpletely specify the object boundary.

Computation: Assume that the first and the last lines of the object, i.e., m_1 and m_2, are given. Then, eq.(9) can be reduced to

$$\min_{\underline{w}} \left\{ \sum_{\ell=m_1}^{m_a} [T(w_\ell) - 2\sigma^2 \, \ell n \; p(w_\ell \,|\, w_{\ell-1} m_1)] \right\} \tag{10}$$

Now, from eq.(6)

$$\min_{w} \left\{ \sum_{\ell=m_1}^{m_2} [T(w_\ell) - 2\sigma^2 \ell n \, p(\alpha_\ell | \alpha_{\ell-1} \beta_{\ell-1}, m_1) \atop -2\sigma^2 \, \ell n \; p(\beta_\ell \,|\, \alpha_\ell, \alpha_{\ell-1}, \beta_{\ell-1}, m_1)] \right\} \tag{11}$$

Furthermore, since

$$T(w_\ell) = \sum_{k=(\ell-1)J+1}^{(\ell-1)J+\beta_\ell} K(k) - \sum_{k=(\ell-1)J+1}^{(\ell-1)J+\alpha_\ell-1} K(k) \tag{12}$$

where

$$K(k) = K_0(k) - K_b(k) \tag{13}$$

then eq.(11) can be written as

$$\begin{aligned} \min_{w} \Bigg\{ \sum_{\ell=m_1}^{m_a} \Big[& -2\sigma^2 \ell n \; p(\alpha_\ell \,|\, \alpha_{\ell-1}, \beta_{\ell-1}, m_1) \\ & -2\sigma^2 \, \ell n \; p(\beta_\ell \,|\, \alpha_\ell, \alpha_{\ell-1}, \beta_{\ell-1}, m_1) \\ & + \sum_{k=(\ell-1)J+1}^{(\ell-1)J+\beta_\ell} K(k) \quad - \sum_{k=(\ell-1)J+1}^{(\ell-1)J+\alpha_\ell-1} K(k) \Big] \Bigg\}. \end{aligned} \tag{14}$$

A recursive, easily implementable solution of eq.(14) is possible if the density functions of eq.(13) are approximated. Hence, the minimization in eq.(13) is replaced by

$$\min_{\underline{w}} \{ \sum_{\ell=m_1}^{m_a} [h(\beta_\ell) + g(\alpha_\ell)]\} \qquad (15)$$

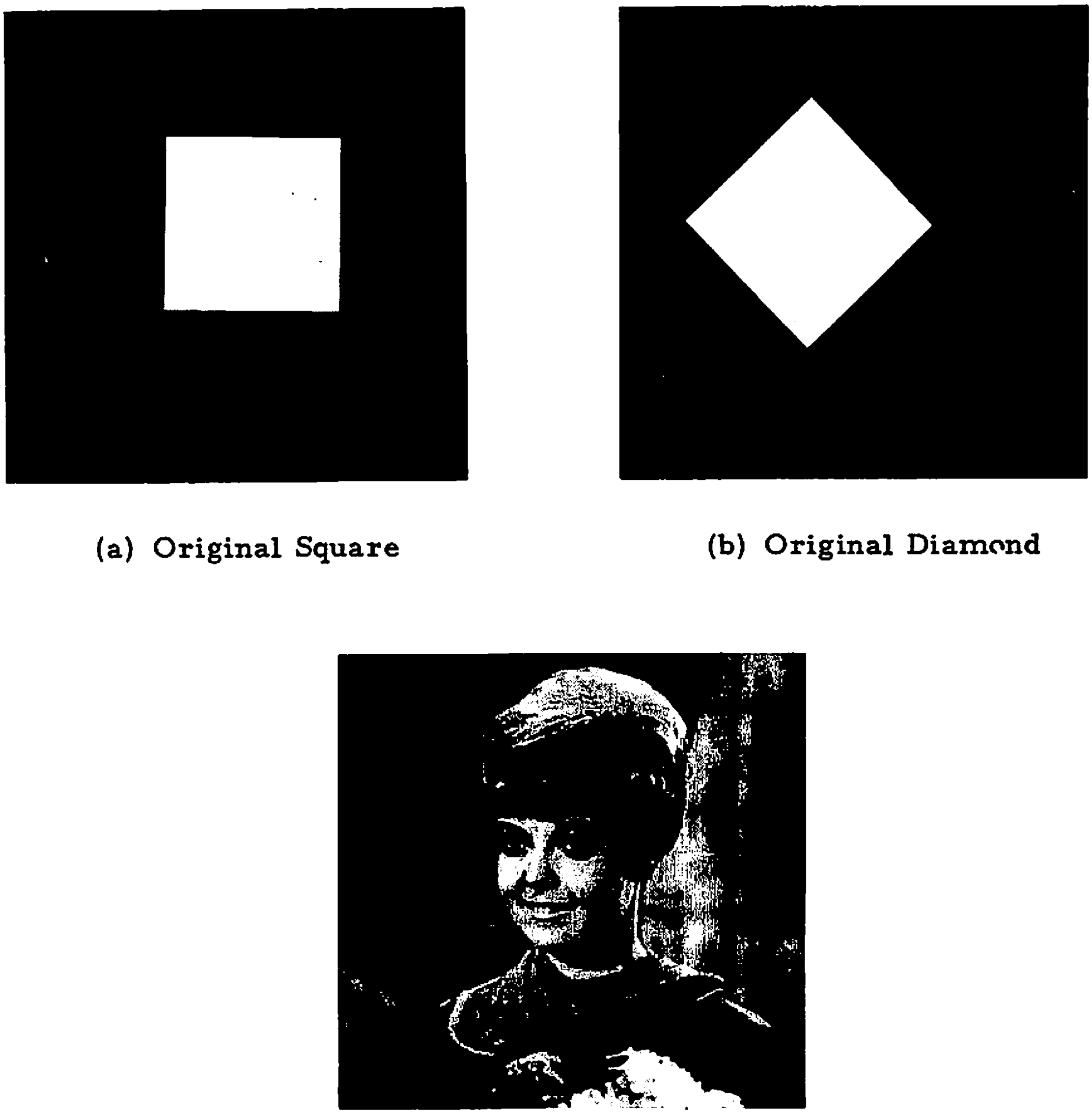
where

$$g(\alpha_\ell) = -2\sigma^2 \ell n\, p(\hat{\alpha}_\ell | \alpha_{\alpha-1}, \hat{\beta}_{\ell-1}, m_1) - \sum_{k=(\ell-1)J+1}^{(\ell-1)J+\alpha_\ell-1} K(k) \qquad (16)$$

$$h(\beta_\ell) = -2\sigma^2 \ell n\, p(\beta_\ell | \hat{\alpha}_\ell, \hat{\alpha}_{\ell-1}, \hat{\beta}_{\ell-1}, m_1) + \sum_{k=(\ell-1)J+1}^{(\ell-1)J+\beta_\ell} K(k) \qquad (17)$$

Examples: Several images have been considered to illustrate the results of this section. Figure 1 depicts three such examples. All the pictures have grid size of 256 by 256. In each case the mean and variance of the pictures are determined, and then a white Gaussian noise of specified variance is added to each picture (figure 2).

An arbitrary segmentation procedure was performed to produce the background, $s_b(k)$, and foreground, $s_o(k)$, $1<k<256*256$, sample functions for each picture. The segmentation procedure was based on replacing the object intensity values by the maximum background brightness value (forming the background sample) and the background

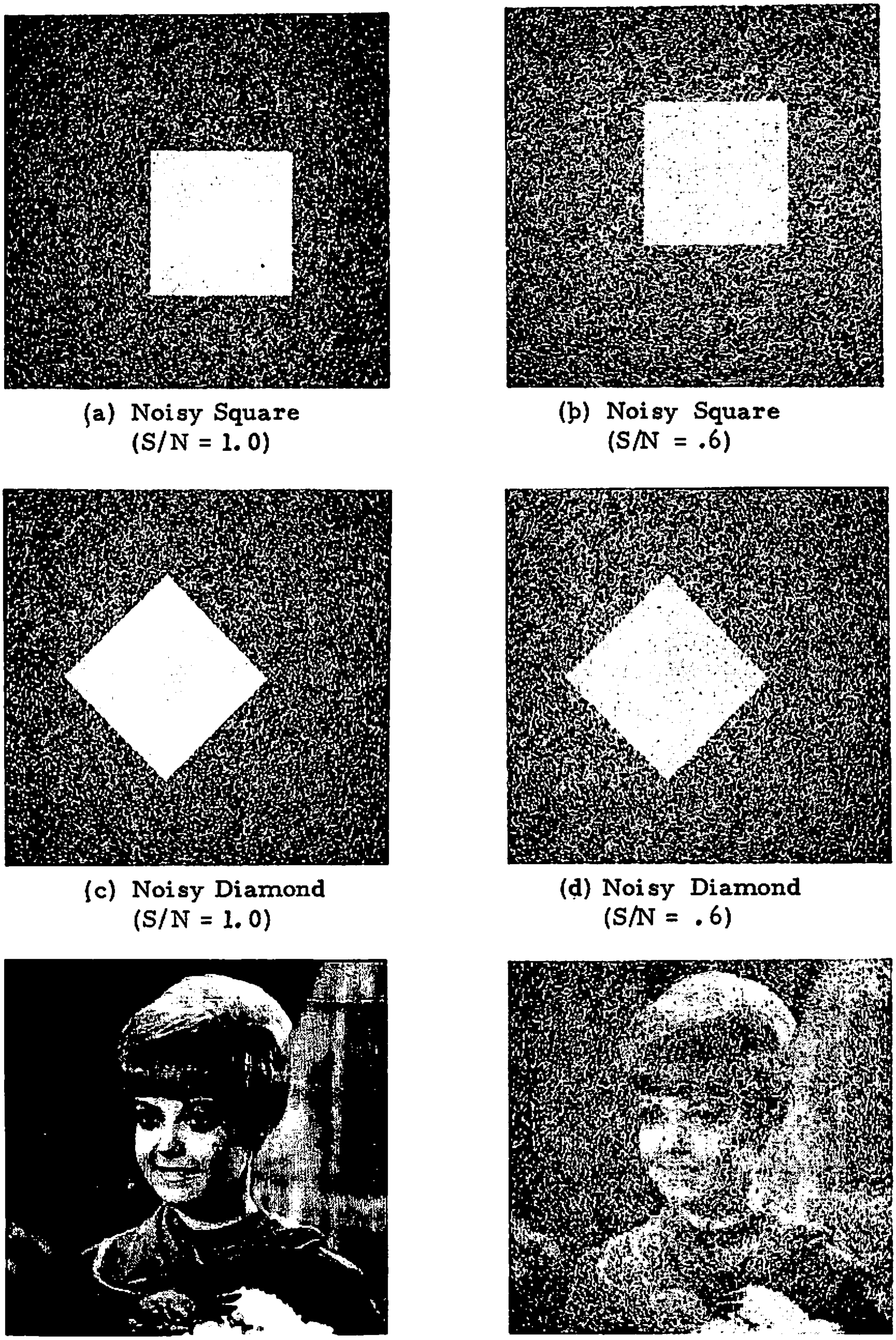


(a) Original Square

(b) Original Diamond

(c) Original "Girl"

Figure 5.4-1. Original images



(a) Noisy Square
(S/N = 1.0)

(b) Noisy Square
(S/N = .6)

(c) Noisy Diamond
(S/N = 1.0)

(d) Noisy Diamond
(S/N = .6)

(e) Noisy Girl
(S/N =10.0)

(f) Noisy Girl
(S/N = .9)

Figure 5.4-2. Images with additive noise

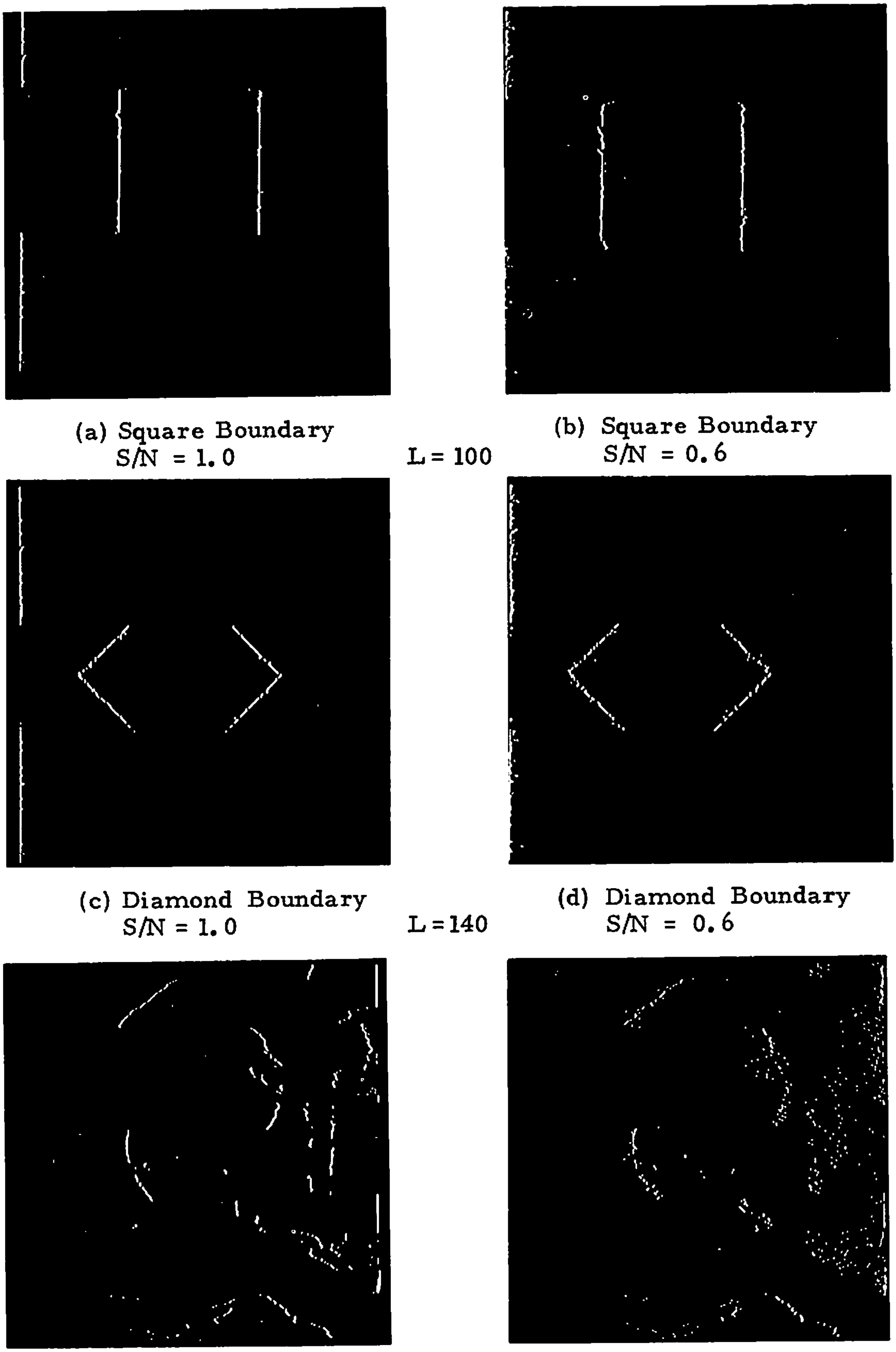
intensity values by the minimum object brightness value (forming the object sample). In general an estimator is needed to perform the segmentation; however, since the original images were available here (not usually the case), the above technique was a more convenient procedure. With values of m_1 and m_2 given as 1 and 256, respectively, the outputs of the boundary estimator are shown in figure 3 The signal to noise ratio (S/N=signal variance/noise variance) of the observed image and the conjectured values of the object maximum width, L, are indicated in each figure.

5.5 Principal Components and Ratioing for Multispectral Image Analysis

Guner S. Robinson and Werner Frei

Manual or machine classification of multi-spectral images is, in general, made difficult by the dimensionality of the problem and by the fact that the information of interest may reside in subtle differences between the spectral bands. However, the redundancy between multispectral images provides potentiality for a reduction in dimensionality without an appreciable information loss. Both linear and nonlinear transforms have been studied to achieve such a reduction and to enhance spectral dissimilarities for terrain classification of the four spectral bands of Earth Resources Satellite (ERTS) imagery.

The principal component transformation is a well-known linear method by which a linearly independent (uncorrelated) set of images is obtained. The energy compaction property of this transformation makes



(a) Square Boundary
S/N = 1.0

L = 100

(b) Square Boundary
S/N = 0.6

(c) Diamond Boundary
S/N = 1.0

L = 140

(d) Diamond Boundary
S/N = 0.6

(e) Girl Boundary
S/N = 10

L = 250

(f) Girl Boundary
S/N = 0.9

Figure 5.4-3. Boundary estimates

it particularly attractive for the reduction of dimensionality, but the computational load may be considered excessive in some cases.

Another popular technique is to generate ratio images in which each pixel value is equal to the rescaled ratio of the amplitudes of two spectral bands. The advantage of this ncnlinear transformation is that ratios are invariant to illumination variations and computationally fast. The disadvantage is that there are six possible ratio images (disregarding inverses) with rather similar energy contents.

Principal Component Analysis of Multispectral Images: Principal components analysis of ERTS bands is motivated by the desire to extract the most significant spectral components from the available four. This dimensionality reduction also results in preserving most of the ERTS information in a smaller number of components.

The principal component analysis of ERTS data involves finding a unitary transformation matrix which, when applied to the four bands, results in a new set of bands (principal components) having several desirable characteristics: the principal components are uncorrelated and each component has a variance less than the previous component.

The principal components are obtained from the original four spectral bands by the matrix multiplication

$$\underline{y} = \underline{A}\,\underline{x} \qquad (1)$$

where $\underline{x}$ is the vector of spectral intensities on four ERTS bands, $\underline{y}$ is the vector of principal components and $\underline{A}$ is the 4 x 4 Karhunen-Loeve transformation matrix. This matrix is derived by diagonalizing the spectral covariance matrix $\underline{C}_x$ of the spectral bands. The rows of A are the normalized eigenvectors of $\underline{C}_x$. The covariance matrix of the principal components is then

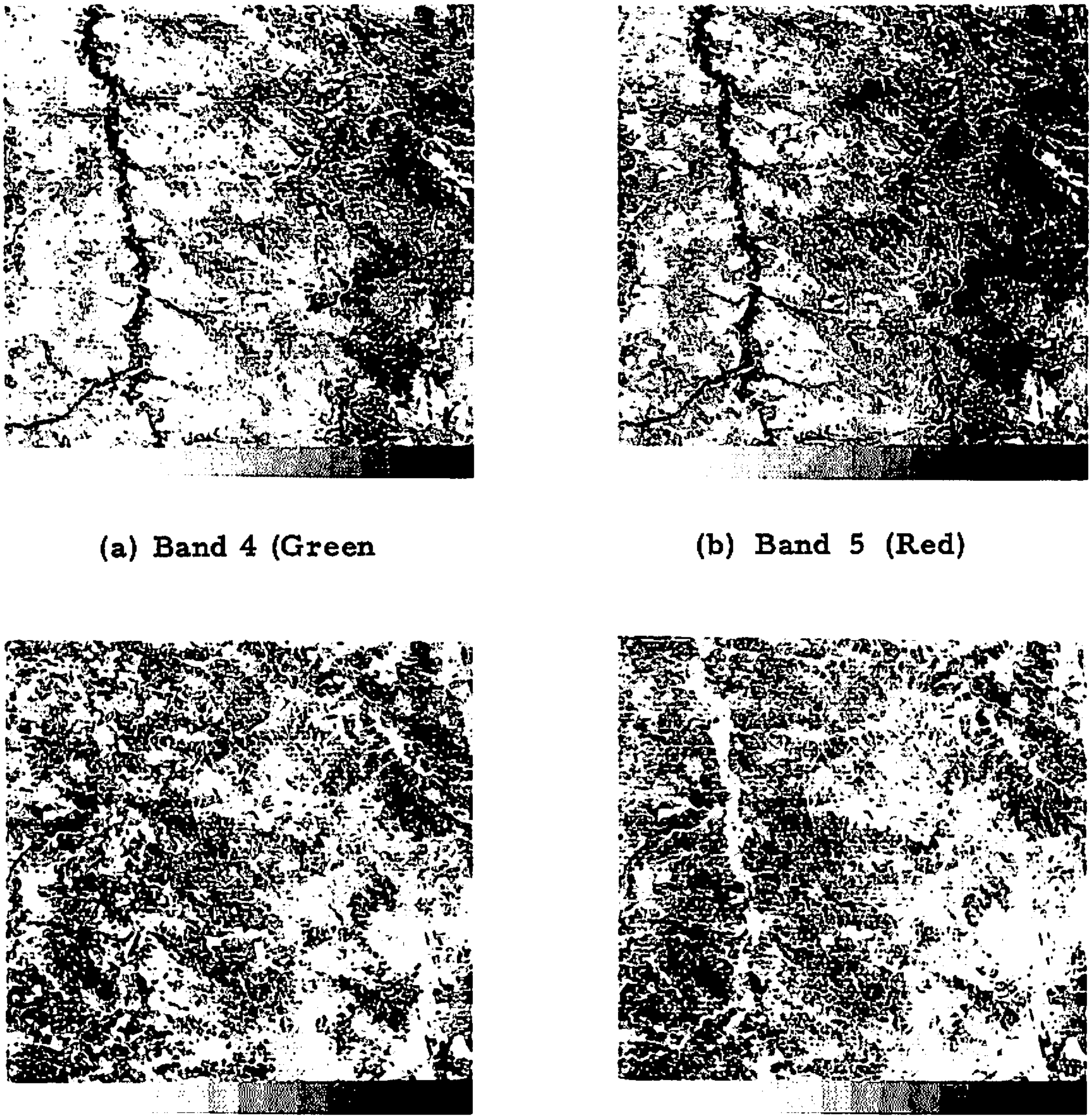
$$\underline{C}_y = \underline{A}\,\underline{C}_x\,\underline{A}^T = \begin{bmatrix} \lambda(1) & 0 & 0 & 0 \\ 0 & \lambda(2) & 0 & 0 \\ 0 & 0 & \lambda(3) & 0 \\ 0 & 0 & 0 & \lambda(3) \end{bmatrix} \qquad (2)$$

where $\lambda_1, \lambda_2, \lambda_3$, and (the variances of the principal components) are the eigenvalues of $\underline{C}_x$ ordered such that $\lambda_1 > \lambda_2 > \lambda_3 > \lambda_4$.

It should be noted that, since $\underline{A}$ is a unitary transformation, the total data energy is invariant. That is

$$\sum_{i=1}^{4} \sigma_i^2 = \sum_{i=1}^{4} \lambda_i \qquad (3)$$

where the σ_i, are the variances of the original ERTS bands. As an example, figure 1 shows four ERTS images, and figure 2 presents the principal components planes. All images have been enhanced by histogram manipulation before display. The spectral covariance matrix $\underline{C}_x$ of the four ERTS bands is obtained by computing the spectral covariance matrix on 64 x 64 blocks of ERTS pictures, (each 512 x 512 pixels) and then averaging over all the blocks. Exhibit 1 contains



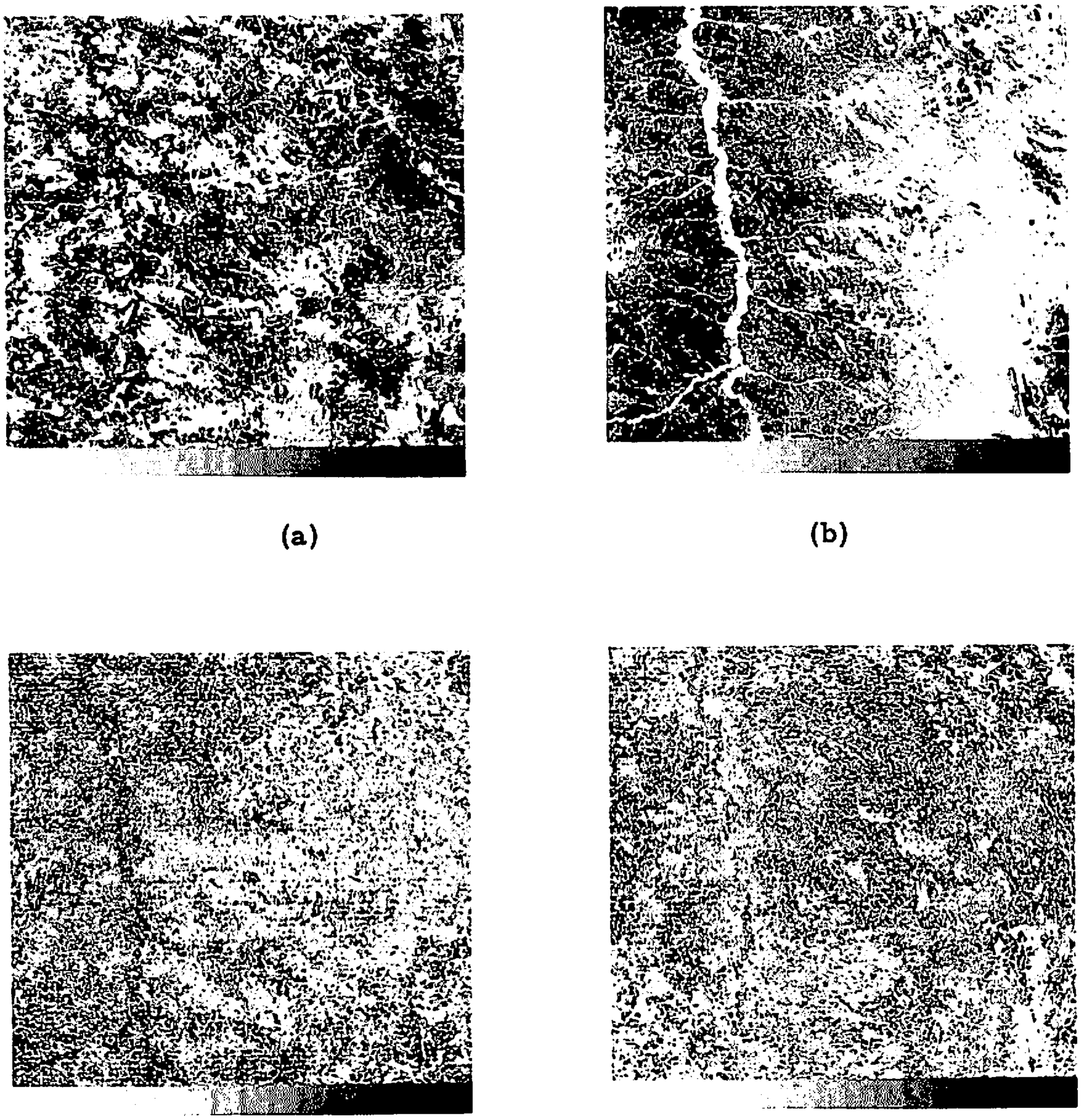
(a) Band 4 (Green

(b) Band 5 (Red)

(c) Band 6 (Infared 1)

(d) Band 7 (Infared 2)

Figure 5.5-1. Enhanced ERTS images



(a)

(b)

(c)

(d)

Figure 5.5-2. Principal Components of ERTS images

Exhibit 5.5-1

Statistical Data on Principal Components of ERTS Planes

spectral covariance matrix

57.16	75.80	39.23	18.46
75.80	113.69	53.76	24.50
39.23	53.76	68.97	64.78
18.40	24.50	64.78	85.53

normalized spectral covariance matrix

1.000	.117	.078	.033
.117	1.000	.075	.031
.078	.075	1.000	.105
.033	.031	.105	1.000

Karhunen-Loeve transform eigenmatrix

0.44465	0.63040	0.49520	0.39958
-0.32653	-0.49866	0.34168	0.72662
0.32957	-0.45586	0.67249	-0.48097
0.76619	-0.38227	-0.43103	0.28469

Karhunen-Loeve transform eigenvalues

i	λ_i	$\left(\frac{\lambda_i}{\sum_{k=1}^{4} \lambda_k}\right) \cdot 100\%$
1	224.92	69.14
2	90.78	27.91
3	5.42	1.66
4	4.13	1.27

the measured ERTS covariance matrix, the computed covariance matrix of the principal components planes, and the corresponding eigenvalues. It should be noted that the first two principal components represent 97% of the total energy.

Band Ratios: Ratioing of ERTS pictures is a useful pre-processing technique for multispectral recognition and classification. Signatures obtained from a training sample under one set of conditions may not have a good discrimination capability for a given classification scheme if the same area is observed under a different set of conditions. If the changes result from simple multiplicative factors such as the brightness level, then the ratio of the bands will be invariant.

Taking various ratios of the green, red and the two infrared bands (bands 4, 5, 6, and 7, respectively) of the ERTS data results in elimination of brightness variations due to topographic relief. Such ratio images have been shown to be more useful for determining boundaries between lithologic units and vegetation groups [1]. Ratios may be taken to emphasize variations due to color also. Such ratioing processes produce a color display whose color variations are more indicative of material variations than the simple pseudocolor displays.

Ordinarily, ratio images are obtained by forming a scaled ratio of two bands, (direct ratio). Logarithmic ratio images are produced by applying a logarithmic stretch to a ratio image. The advantage of the logarithmic ratio is a greater tolerance to quantization error.

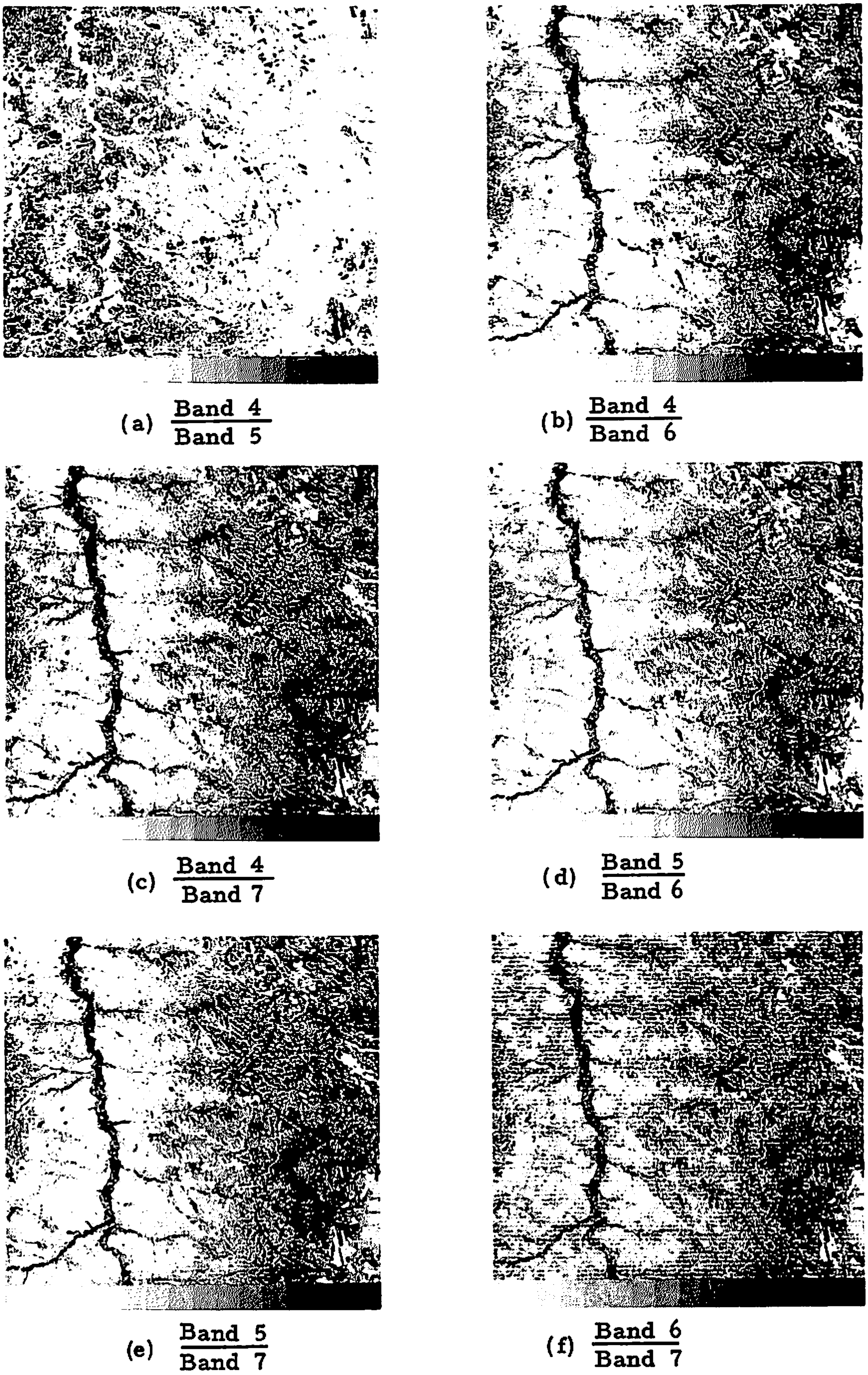
In the cases studied, it has appeared that logarithmic ratio images contain more visual information than direct ratio images. It is felt that experiments with more images are necessary to confirm the above conclusion.

As an example, figure 3 shows the logarithmic ratios of the ERTS pictures shown in figure 1. These ratio images have been enhanced using the same histogram manipulation algorithm as the original images. The choice of ratio images for a certain classification scheme depends on the data and the application.

The covariance matrix of various ratios could give some insight in choosing a set of ratios for a classification scheme: ratios that are uncorrelated are likely to produce better results than those that are highly correlated. This idea suggests the use of the principal components of ratios instead of ratios themselves. Exhibit 2 contains the normalized covariance matrix and eigenvalues of the logarithmic ratios. It is observed that the first two or three principal components contain most of the relevant information in ratio images. This can also be verified by studying the principal components shown in figure 4.

Reference

1. Goetz, A.F.H., et. al., "Application of ERTS Images and Image Processing to Regional Geologic Problems and Geologic Mapping in Northern Arizona," JPL Technical Report 32-1597, May 15, 1975.



(a) $\frac{\text{Band 4}}{\text{Band 5}}$

(b) $\frac{\text{Band 4}}{\text{Band 6}}$

(c) $\frac{\text{Band 4}}{\text{Band 7}}$

(d) $\frac{\text{Band 5}}{\text{Band 6}}$

(e) $\frac{\text{Band 5}}{\text{Band 7}}$

(f) $\frac{\text{Band 6}}{\text{Band 7}}$

Figure 5.5-3. Logarithmic Ratios of ERTS bands

Exhibit 5.5-2

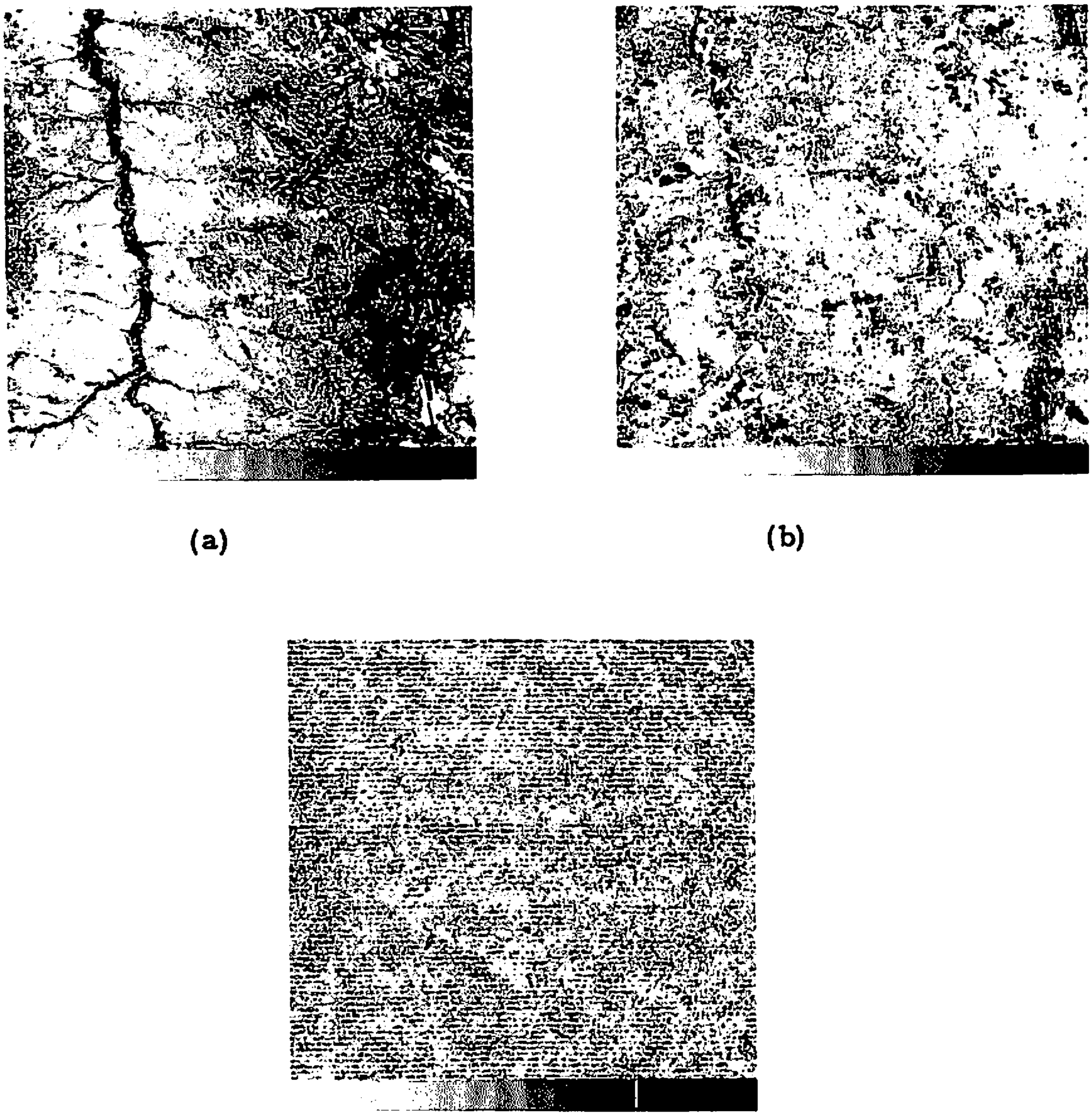
Statistical Data on Principal Components of ERTS Logarithmic Ratio Planes

normalized covariance matrix

Ratios	4:5	4:6	4:7	5:6	5:7	6:7
4:5	1.0	-0.297	-0.390	-0.746	-0.714	-0.399
4:6	-0.297	1.0	0.910	0.837	0.812	0.486
4:7	-0.390	0.910	1.0	0.840	0.912	0.771
5:6	-0.746	0.837	0.840	1.0	0.955	0.554
5:7	-0.714	0.812	0.912	0.955	1.0	0.751
6:7	-0.399	0.486	0.771	0.554	0.751	1.0

eigenvalues

i	λ_i	$\left(\frac{\lambda_i}{\sum_{k=1}^{6} \lambda_k}\right) .100\%$
1	35.495	86.0
2	3.270	8.0
3	1.592	3.9
4	0.084	0.2
5	0.082	0.2
6	0.080	0.2



(a)

(b)

(c)

Figure 5.5-4. The first three principal components of logarithmic ratio images.

6. Image Processing Systems Projects

The following describes the image processing systems projects which are concerned with the development of image processing hardware and software systems.

6.1 Hardware Projects

Toyone Mayeda

A real time color image magnetic tape recorder/playback system is under development. The recorder is to be used to record real time digitized television signals at a 600 ips rate and played back at a 1-7/8 ips rate to transfer the data to the PDP-10 computer. The inverse process is performed to produce real time television signals from coded computer records.

Delivery of the Emerson (Orion) digital magnetic tape recorder/playback unit has been delayed due to difficulty in meeting the bit error rate and maximum skew specifications. Emerson is presently redesigning the tape transport mechanism to reduce the problem. It is also planned to increase the deskew buffer capacity in the interface hardware which was developed at USC. Delivery is now planned for 1 January 1976.

A second digital image television display system, which is being developed, is presently in the check out and testing phase. This unit receives digital picture data from the ARPANET, acting as a virtual

TIP terminal, and produces a modulated television signal for connection to the antenna terminals of any commercial television receiver.

6.2 Software Projects

Dennis Smith

The software effort of the Image Processing Institute (IPI) programming group has been centered on two projects. The first has been the implementation of a network of mini-computers, and the second the augmentation of the library of image processing user programs.

The purposes of the network of mini-computers are to handle communication among the larger computers of the Engineering Computer Laboratory and the Image Processing Laboratory, and between these computers and machines at other sites, and to handle lowest level protocols with image processing devices.

The primary advantage of this network is the freeing of the larger computers from the task of minutely supervising complex devices, many of which cause frequent interrupts that are demanding upon a processor's time. All communications among the larger computers, and between them and the specialized devices are carried on in message packets which are blocks of data that can be passed about with a minimum of interrupts.

A second advantage of the network is one of reliability. Should the PDP-11 mini which is controlling a key device become ncn-operational, the software for that device can be easily moved to another mini, the device plugged into that machine, and service restored. Should the PDP-10, the principal ccmputer for user software, be unavailable, the HP-2100 or the IBM 360/44 can be used in this capacity, as the user software is written in portable FORTRAN.

To date, the two programs which will run on all the 11's, the supervisor prcgram, and the network contrcl program (NCP), which manages the routing of message packets from the source to the destination ccmputer, are both completed. Remaining to be finished are the service programs to handle each of the image processing devices on the 11's, and the NCPs for each of the larger computers.

The second area of concentration is user software. Several personal programs of the IPI faculty and staff were obtained from the individuals who wrote them and were added to the IPI library after modification to make them more useful to the general community. All of the fcllowing were standardized tc conform to parameter input conventions of the other library programs, and generalized to process images which are any power-of-two size smaller than or equal to 1024. All programs run in an interactive mode, asking the users questions as to what he wants done. These programs are described below.

CONVOL - a program for performing two dimensional convolution was generalized to provide a choice of impulse response arrays (or allow the user to enter his own) in sizes 3x3, 5x5, or 7x7.

HSTMOD - a program to perform modification of histograms, which does equalizaticn, exponentiation, cr a "gamma" function upon the histogram of a picture.

PICOUT - a program for contrast manipulation was expanded to perform the following: clipping, labeling, reformatting (packing, unpacking, integer-real, real-integer), and application of one of a variety of transfer functions: positive linear, negative linear, sawtooth, slicer, eye, half power, third power, log, or a user-defined step function (256 steps) with automatic scaling.

MEDIAN - a median filtering program which offers three choices of filtering: MTS, which ccmputes the median for each positicn of a rectangular window as it scans the picture file; MEDX, which computes the median for each position of a cross window; and MOVAVG, which computes the mean for each positon of a rectangular window. All of the above may be used with any window size 1 x 1 to 11 x 11.

CFIL - a program to dc image restoration and Wiener filtering. It allows specification of a blur, correlation coefficient, and signal-to-noise ratio, and an inplulse response matrix up to 31 x 31.

6.3 A Synthesis Procedure for Optical Nonlinearities

Stephen R. Dashiell and Alexander A. Sawchuk

A general technique for implementing nonlinear nonmcnotonic function incoherent optical parallel signal processing systems has been described in recent publications [1-4]. The technique operates by using special halftone screens and high contrast (binary) optical input devices to effectively pulse-width modulate the input. The selection of diffraction orders in a Fourier transform produces a desampled output which is a point nonlinear function of the input.

A very complete analysis of the entire process has been performed [4]. One generalization that has been found is that the halftone profiles (cells) themselves which determined the dot size b need not be monotonic. Thus, the effective periodicity of the preprocessing can be changed. The effect is to reduce the diffraction order necessary to achieve nonmonotonic operation. So many design variables are now available that the class of mathematical operations possible and ease of implementation has been greatly extended.

An exact synthesis procedure for nonlinearities using ordinary monotonic cells has been made and is summarized here for the case of linear one-dimensional scenes. Omitting wavelength and geometrical factors for clarity, the general expression for the amplitude in the transform plane resulting from an infinite grating of opaque bars of width, b, and period, a, with unit amplitude illumination is

$$\mathcal{F}\{\underline{t}(x)\} = \delta(f_X) - \sum_{n=-\infty}^{\infty} \delta(f_x - \frac{n}{a}) \cdot \frac{b}{a} \text{ sinc } (\frac{bn}{a}) \qquad (1)$$

where the y dimension is suppressed. By selecting these diffraction orders with simple spatial filters, the sinc terms in eq.(1) indicate that nonmonotonic behavior can be expected. In the special case of a zero order (n=0) selection, an intensity output

$$I_{out(0)} = (1 - b/a)^2 \qquad (2)$$

is expected frcm eq.(1) after inverse Fourier transforming and squaring. For a first order (n=1) selection, the output intensity

$$I_{out(1)} = \frac{1}{\pi^2} \sin^2 (\frac{\pi b}{a}) \qquad (3)$$

a function which is ncnmcnctonic in b.

Because of the halftone process, the value of b in these expressions is a function of the continuous input intensity I . A one-dimensional halftone screen can be described as periodic symmetrical cells centered on x=0 and extending from -a/2 to a/2, each with a density function f(x). The intensity I transmitted by the input-screen ccmbination in each cell is

$$I_t = I_{in} 10^{-f(x)} \qquad (4)$$

and this functicn is imaged or contact printed onto a binary clipping medium with effective cutcff I'. Since there is no exposure if $I_t \leq I'$ and full exposure if $I_t > I'$, opaque bars result where x is such that

$$I_{in} \geq I' 10^{f(x)} \qquad (5)$$

Taking logarithms yields

$$x \le f^{-1}\left[\log_{10}\left(\frac{I_{in}}{I'}\right)\right] \qquad (6)$$

where f^{-1} is the inverse of f. The cell size b is simply twice x for halftone cells symmetrical about the spacing a.

Combining eq.(6) with eqs.(2) or (3) for the appropriate order gives the overall mapping

$$I_{out} = g_0(I_{in}) = (1 - 2f^{-1}[\log_{10}(\frac{I_{in}}{I'})]/a)^2 \qquad (7)$$

for the zero (n=0) order, and

$$I_{out} = g_1(I_{in}) = \left(-\sin\left(\frac{2\pi}{a} \cdot f^{-1}\left[\log_{10}\frac{I_{in}}{I'}\right]\right)/\pi\right)^2 \qquad (8)$$

for the first order (n=1). Similar expressions can be obtained for two-dimensional cells and various selections of diffraction orders.

These expressions for transforms and dot sizes are valid only in local regions of constant input values. Input information produces low spatial frequency modulation, and the complete expression for the transform is much more complicated. The halftone process assumes input sampling at a rate sufficient to avoid aliasing, and these results describe the local nonlinear effects if desampling filters choose the low frequency input information.

The follcwing procedure can be used to obtain the cell profile f(x) and diffraction order for one-dimensional screens given a desired $I_{out}=h(I_{in})$:

I. Determine the minimum diffraction order n to be used by counting the number of sign changes, q, in the slope of the transfer function. If q is zero and the initial slope of h is negative, the n=0 order can be used directly. If q is zero with positive initial slope, the n=1 order must be used. For q greater than zero, add one to q if the initial slcpe is negative to obtain q'. If q is greater than zero and the initial slope is positive, then q'=q. The number of slope changes in the general $I_{out(n)}$ versus b curve is given by 2n-1 for b>0, thus n is selected so that q' is less than or equal to 2n-1. This procedure determines the minimum n, so that a larger order can be used if desired.

II. Normalize the desired function by scaling so the largest I_{out} equals the maximum I_{out} for the particular order used. For n=0, $I_{out} <1$; for n>C, $I_{out} <1/n$.

III. Equate $h(I_{in})$ with the appropriate general expression $g_n(I_{in})$ of the form eq.(10) or eq.(11) for the particular order n used. Solve this equation for $f^{-1}(\log_{10}(I_{in}/I'))$.

IV. Solve for f(x) by selecting a solution such that f(x) is mcnotonic and the initial slopes of $h(I_{in})$ and $g_n(I_{in})$ have the same sign. Whenever the slope of $h(I_{in})$ changes sign, the halftone cell size must atruptly increase so that the diffracticn output remains the same while jumping to a region of $g_n(I_{in})$ of opposite slope.

An example of this procedure is the synthesis of an optical level slicer, or intensity bandpass, with the characteristics shown in figure 1. This function is

$$I_{out} = h(I_{in}) = K, \qquad I_{c1} \le I_{in} \le I_{c2} \tag{9}$$
$$0, \qquad \text{otherwise}$$

and it has one sign change in slope, so q equals one. The initial slope is positive, so q' is one and the first (n=1) diffraction order can be used. Normalizing the function $h(I_{in})$, gives K equal to $1/\pi^2$, and equating $h(I_{in})$ with $g_1(I_{in})$ gives

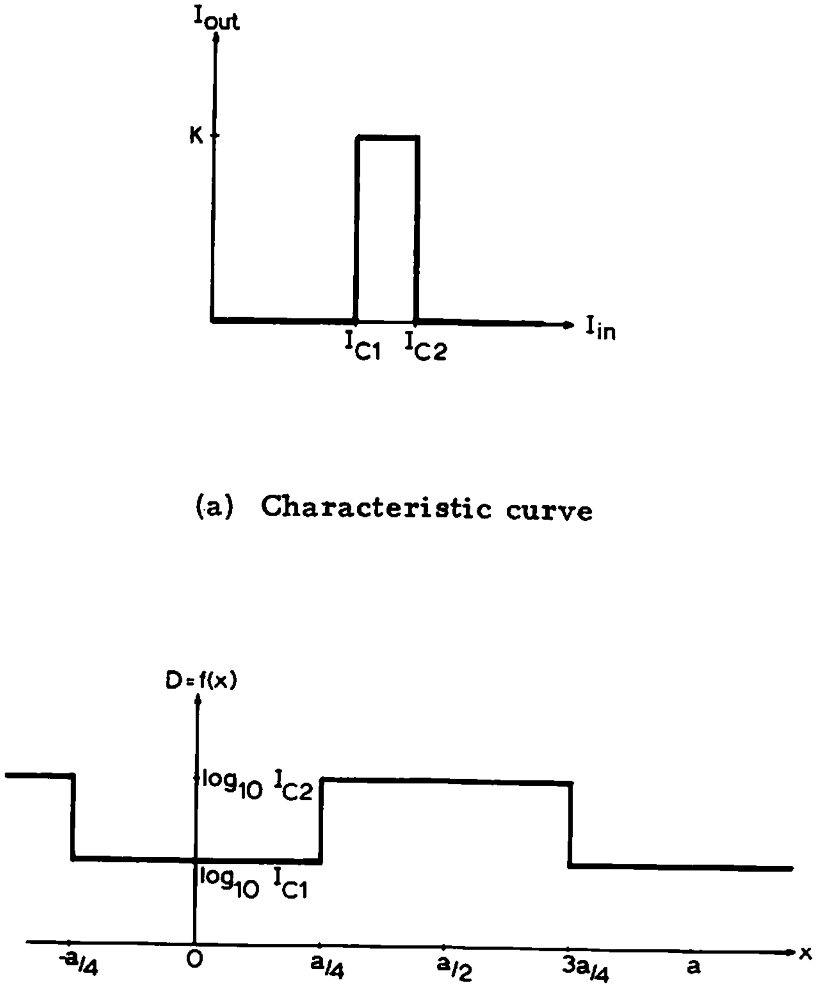
$$f^{-1}(\log_{10} I_{in}) = (a/2\pi)\sin^{-1}(\pi[h(I_{in})]^{\frac{1}{2}}) \tag{10}$$

where the clip level I' is assumed unity for simplicity. For I $<I$

$$f^{-1}(\log_{10} I_{in}) = (\frac{a}{2\pi}) \sin^{-1}(\pi[0]^{\frac{1}{2}}) \tag{11}$$
$$= 0 \text{ or } a/2$$

Selecting the zero solution to satisfy the monotonic cell condition, results in

$$f(0) = \log_{10} I_{c1} \tag{12}$$



(a) Characteristic curve

(b) Halftone cell profile

Figure 6.3-1. Level slicer function.

For $I_{c1} < I_{in} < I_{c2}$

$$f^{-1}(\log_{10} I_{in}) = \left(\frac{a}{2\pi}\right)\sin^{-1}\left(\pi\left(\frac{1}{\pi^2}\right)^{\frac{1}{2}}\right) = a/4 \qquad (13)$$

and solving gives

$$f\left(\frac{a}{4}\right) = \log_{10} I_{c2} \qquad (14)$$

as a point of discontinuity of f. For $I_{c2} < I_{in}$

$$f^{-1}(\log_{10} I_{in}) = \left(\frac{a}{2\pi}\right) \sin^{-1}\left(\pi[\,0\,]^{\frac{1}{2}}\right) = 0 \;\text{ or }\; a/2 \qquad (15)$$

Here the a/2 solution is selected to satisfy the monotonic cell condition. This is the end point of the profile having period a. This function f(x), 0<x<a/2 shown in figure 2, has been experimentally demonstrated [2-3]. The width of the level is controlled by the step size in f(x), and the level location is controlled by the preprocessing step. In general, the halftone cell profile may combine smooth and discontinuous functions, leading to transfer functions $h(I_{in})$ with both smooth and limiting nonlinear characteristics.

The analysis of system effects due to low contrast (finite gamma) input media is well underway. In the zero order the major effect is a change in the transfer function; in the first order, this effect is combined with a loss of diffraction efficiency. These effects are not serious in practice, and some techniques of pre-compensating halftone cells to correct for low gamma have been developed. These appear very promising for practical implemenation, particularly with real-time input devices. A series of computer routines have been written to iteratively synthesize cell profiles, produce input/output curves and study effects of parameter variation on the results.

A number of experimental halftone screens have been made and tested. A computer-controlled optronics flatbed microdensitometer has been used, and direct plots on highly resolution film have been adequate to make good quality screens. Most of the screens have been one-dimensional line gratings, and plotting aperture sizes down to 10 m. have been used, Kodak 50-427 sheet film is used for the screens because of its high resolution (>250 lines/mm.) and good line holding ability. Some of the functions which have plotted and tested with good results so far include: intensity level slicers, intensity notch filters, logarithms, and exponentials. Experimental verification of other functions is underway.

References

1. S.R. Dashiell and A.A. Sawchuk, "Nonlinear Optical Image Processing with Halftone Screens," USCIPI Semiannual Progress Report 540, 1 March 1974 - 31 August 1974, pp. 65-68.

2. S.R. Dashiell and A.A. Sawchuk, "Nonmonotonic Nonlinear Picture Operations," USCIPI Semiannual Progress Report 560, 1 September 1974 - 28 February 1975, pp. 99-103.

3. A.A. Sawchuk and S.R. Dashiell, "Nonmonotonic Nonlinearities in Optical Processing," Proceedings of the IEEE International Optical Computing Conference, Washington D.C., April 23-25, 1975, pp. 73-76.

4. S.R. Dashiell and A.A. Sawchuk, "Optical Synthesis of Nonlinear Nonmonotonic Functions," accepted for publication in Optics Communications.