INFORMATION TO USERS

This reproduction was made from a copy of a document sent to us for microfilming. While the most advanced technology has been used to photograph and reproduce this document, the quality of the reproduction is heavily dependent upon the quality of the material submitted.

The following explanation of techniques is provided to help clarify markings or notations which may appear on this reproduction.

1. The sign or "target" for pages apparently lacking from the document photographed is "Missing Page(s)". If it was possible to obtain the missing page(s) or section, they are spliced into the film along with adjacent pages. This may have necessitated cutting through an image and duplicating adjacent pages to assure complete continuity.

2. When an image on the film is obliterated with a round black mark, it is an indication of either blurred copy because of movement during exposure, duplicate copy, or copyrighted materials that should not have been filmed. For blurred pages, a good image of the page can be found in the adjacent frame. If copyrighted materials were deleted, a target note will appear listing the pages in the adjacent frame.

3. When a map, drawing or chart, etc., is part of the material being photographed, a definite method of "sectioning" the material has been followed. It is customary to begin filming at the upper left hand corner of a large sheet and to continue from left to right in equal sections with small overlaps. If necessary, sectioning is continued again—beginning below the first row and continuing on until complete.

4. For illustrations that cannot be satisfactorily reproduced by xerographic means, photographic prints can be purchased at additional cost and inserted into your xerographic copy. These prints are available upon request from the Dissertations Customer Services Department.

5. Some pages in any document may have indistinct print. In all cases the best available copy has been filmed.
Collaer, Marcia Lee

IMAGE DATA COMPRESSION: DIFFERENTIAL PULSE CODE MODULATION OF TOMOGRAPHIC PROJECTIONS

The University of Arizona  M.S.  1985

University Microfilms International

300 N. Zeeb Road, Ann Arbor, MI 48106
PLEASE NOTE:

In all cases this material has been filmed in the best possible way from the available copy. Problems encountered with this document have been identified here with a check mark ✓.

1. Glossy photographs or pages ✓
2. Colored illustrations, paper or print ______
3. Photographs with dark background ✓
4. Illustrations are poor copy ______
5. Pages with black marks, not original copy ______
6. Print shows through as there is text on both sides of page ______
7. Indistinct, broken or small print on several pages ______
8. Print exceeds margin requirements ______
9. Tightly bound copy with print lost in spine ______
10. Computer printout pages with indistinct print ______
11. Page(s) _________ lacking when material received, and not available from school or author.
12. Page(s) _________ seem to be missing in numbering only as text follows.
13. Two pages numbered ______. Text follows.
14. Curling and wrinkled pages ______
15. Dissertation contains pages with print at a slant, filmed as received _______
16. Other ______________________________________________________

University
Microfilms
International
IMAGE DATA COMPRESSION: DIFFERENTIAL PULSE CODE MODULATION OF TOMOGRAPHIC PROJECTIONS

by

Marcia Lee Collaer

A Thesis Submitted to the Faculty of the
DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING
In Partial Fulfillment of the Requirements
For the Degree of
MASTER OF SCIENCE
WITH A MAJOR IN ELECTRICAL ENGINEERING
In the Graduate College
THE UNIVERSITY OF ARIZONA

1985
STATEMENT BY AUTHOR

This thesis has been submitted in partial fulfillment of requirements for an advanced degree at The University of Arizona and is deposited in the University of Arizona to be made available to borrowers under rules of the Library.

Brief quotations from this thesis are allowable without special permission, provided that accurate acknowledgment of source is made. Requests for permission for extended quotation from or reproduction of this manuscript in whole or in part may be granted by the head of the major department or the Dean of the Graduate College when in his or her judgment the proposed use of the material is in the interests of scholarship. In all other instances, however, permission must be obtained from the author.

SIGNED: Marcia L. Collar

APPROVAL BY THESIS DIRECTOR

This thesis has been approved on the date shown below:

B. R. HUNT
Professor of Electrical and Computer Engineering

8-19-85
ACKNOWLEDGMENTS

I would like to thank Dr. B. R. Hunt, who provided the original impetus for this project and shaped the direction of the research. Another provider of major assistance, Dr. Robin Strickland, helped with day-to-day dilemmas and answered a great many of my foolish questions.

Finally, I want to thank my husband, David Peters, for all of his support, both tangible and intangible, and his remarkably good humor throughout the process.
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>LIST OF ILLUSTRATIONS</th>
<th>vi</th>
</tr>
</thead>
<tbody>
<tr>
<td>LIST OF TABLES</td>
<td>ix</td>
</tr>
<tr>
<td>ABSTRACT</td>
<td>x</td>
</tr>
<tr>
<td>CHAPTER</td>
<td></td>
</tr>
<tr>
<td>1. INTRODUCTION</td>
<td>1</td>
</tr>
<tr>
<td>Compression Techniques</td>
<td>3</td>
</tr>
<tr>
<td>Orthogonal Transform Techniques</td>
<td>3</td>
</tr>
<tr>
<td>Prediction Techniques</td>
<td>4</td>
</tr>
<tr>
<td>Interpolation Techniques</td>
<td>5</td>
</tr>
<tr>
<td>Nonorthogonal Techniques</td>
<td>5</td>
</tr>
<tr>
<td>Variable Word Length Encoding (Huffman Codes)</td>
<td>6</td>
</tr>
<tr>
<td>Current Method</td>
<td>6</td>
</tr>
<tr>
<td>2. TOMOGRAPHY</td>
<td>10</td>
</tr>
<tr>
<td>Fourier Slice Theorem</td>
<td>11</td>
</tr>
<tr>
<td>Backprojection</td>
<td>15</td>
</tr>
<tr>
<td>Digitization and Simulation</td>
<td>19</td>
</tr>
<tr>
<td>3. REDUNDANCY</td>
<td>28</td>
</tr>
<tr>
<td>Measures of Information</td>
<td>28</td>
</tr>
<tr>
<td>Redundancy of Digital Information</td>
<td>30</td>
</tr>
<tr>
<td>Redundancy Measurement Method</td>
<td>32</td>
</tr>
<tr>
<td>Redundancy Results</td>
<td>36</td>
</tr>
<tr>
<td>Redundancy Conclusions</td>
<td>42</td>
</tr>
<tr>
<td>4. DIFFERENTIAL PULSE CODE MODULATION</td>
<td>45</td>
</tr>
<tr>
<td>Prediction Methods</td>
<td>48</td>
</tr>
<tr>
<td>Minimum Mean Square Error Methods</td>
<td>48</td>
</tr>
<tr>
<td>One-dimensional—First Order</td>
<td>51</td>
</tr>
<tr>
<td>One-dimensional—Second Order</td>
<td>53</td>
</tr>
<tr>
<td>One-dimensional—Third Order</td>
<td>55</td>
</tr>
<tr>
<td>Two-dimensional—Second Order</td>
<td>56</td>
</tr>
</tbody>
</table>
# TABLE OF CONTENTS — Continued

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Slope Method</td>
<td>57</td>
</tr>
<tr>
<td>Error Quantization</td>
<td>59</td>
</tr>
<tr>
<td><strong>5. RESULTS</strong></td>
<td>64</td>
</tr>
<tr>
<td>DPCM Coding and Backprojection</td>
<td>68</td>
</tr>
<tr>
<td>MSE Measures</td>
<td>69</td>
</tr>
<tr>
<td>Prediction Methods</td>
<td>69</td>
</tr>
<tr>
<td>Within vs. Across Projections</td>
<td>72</td>
</tr>
<tr>
<td>Bit Rates</td>
<td>72</td>
</tr>
<tr>
<td>Reconstructed Images</td>
<td>73</td>
</tr>
<tr>
<td>Errors</td>
<td>75</td>
</tr>
<tr>
<td>Within vs. Across Projections</td>
<td>90</td>
</tr>
<tr>
<td>Analysis of the Difference Distribution</td>
<td>93</td>
</tr>
<tr>
<td><strong>6. CONCLUSIONS</strong></td>
<td>96</td>
</tr>
<tr>
<td>REFERENCES</td>
<td>98</td>
</tr>
<tr>
<td>Figure</td>
<td>Description</td>
</tr>
<tr>
<td>--------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>1.</td>
<td>Stages in compression and decompression of image data</td>
</tr>
<tr>
<td>2.</td>
<td>The projection of an image at an angle of zero</td>
</tr>
<tr>
<td>3.</td>
<td>Two projections generated using a rotated coordinate axis</td>
</tr>
<tr>
<td>4.</td>
<td>Fourier slice theorem</td>
</tr>
<tr>
<td>5.</td>
<td>Fourier slice theorem for multiple projections</td>
</tr>
<tr>
<td>6.</td>
<td>Optical tomographic encoder</td>
</tr>
<tr>
<td>7.</td>
<td>Model of the digital tomographic encoder</td>
</tr>
<tr>
<td>8.</td>
<td>Positioning and number of detectors required</td>
</tr>
<tr>
<td>9.</td>
<td>Projection matrix (sinogram) from a square image</td>
</tr>
<tr>
<td>10.</td>
<td>Data within vs across projections</td>
</tr>
<tr>
<td>11.</td>
<td>Hypothetical probability distribution of a projection matrix</td>
</tr>
<tr>
<td>12.</td>
<td>Images used in the redundancy calculations</td>
</tr>
<tr>
<td>13.</td>
<td>Trimmed projection matrix</td>
</tr>
<tr>
<td>14.</td>
<td>Redundancy within projections as a function of projection angle</td>
</tr>
<tr>
<td>15.</td>
<td>The effect of angle on redundancy</td>
</tr>
<tr>
<td>16.</td>
<td>Redundancy across projections as a function of detector position</td>
</tr>
<tr>
<td>17.</td>
<td>Block diagram of DPCM system</td>
</tr>
<tr>
<td>18.</td>
<td>Model of pixel arrangement for prediction schemes</td>
</tr>
<tr>
<td>19.</td>
<td>Partitioning of a trimmed projection matrix</td>
</tr>
<tr>
<td>Figure</td>
<td>Illustration Description</td>
</tr>
<tr>
<td>--------</td>
<td>-----------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>20.</td>
<td>Nonuniform quantization</td>
</tr>
<tr>
<td>21.</td>
<td>Clock image and reconstruction</td>
</tr>
<tr>
<td>22.</td>
<td>Test image and reconstruction</td>
</tr>
<tr>
<td>23.</td>
<td>River image and reconstruction</td>
</tr>
<tr>
<td>24.</td>
<td>Clock: prediction within projections, coding at approximately 4 bits/pixel</td>
</tr>
<tr>
<td>25.</td>
<td>Clock: prediction within projections, coding at approximately 3 bits/pixel</td>
</tr>
<tr>
<td>26.</td>
<td>Clock: prediction within projections, coding at approximately 2 bits/pixel</td>
</tr>
<tr>
<td>27.</td>
<td>Test: prediction within projections, coding at approximately 4 bits/pixel</td>
</tr>
<tr>
<td>28.</td>
<td>Test: prediction within projections, coding at approximately 3 bits/pixel</td>
</tr>
<tr>
<td>29.</td>
<td>Test: prediction within projections, coding at approximately 2 bits/pixel</td>
</tr>
<tr>
<td>30.</td>
<td>Clock: prediction across projections, coding at approximately 4 bits/pixel</td>
</tr>
<tr>
<td>31.</td>
<td>Clock: prediction across projections, coding at approximately 3 bits/pixel</td>
</tr>
<tr>
<td>32.</td>
<td>Clock: prediction across projections, coding at approximately 2 bits/pixel</td>
</tr>
<tr>
<td>33.</td>
<td>Test: prediction across projections, coding at approximately 4 bits/pixel</td>
</tr>
<tr>
<td>34.</td>
<td>Test: prediction across projections, coding at approximately 3 bits/pixel</td>
</tr>
<tr>
<td>35.</td>
<td>Test: prediction across projections, coding at approximately 2 bits/pixel</td>
</tr>
<tr>
<td>Figure</td>
<td>Description</td>
</tr>
<tr>
<td>--------</td>
<td>--------------------------------------------------</td>
</tr>
<tr>
<td>36</td>
<td>Clock: two-dimensional prediction</td>
</tr>
<tr>
<td>37</td>
<td>Test: two-dimensional prediction</td>
</tr>
<tr>
<td>38</td>
<td>Reconstruction error vs. order of prediction</td>
</tr>
<tr>
<td>39</td>
<td>Error distributions for four prediction methods</td>
</tr>
</tbody>
</table>
## LIST OF TABLES

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Redundancy within and across projections</td>
<td>43</td>
</tr>
<tr>
<td>2.</td>
<td>Mean square error of reconstructions from prediction within projections</td>
<td>70</td>
</tr>
<tr>
<td>3.</td>
<td>Mean square error of reconstructions from prediction across projections</td>
<td>71</td>
</tr>
<tr>
<td>4.</td>
<td>Actual bit rates</td>
<td>74</td>
</tr>
</tbody>
</table>
ABSTRACT

Image data compression through the simulation of optical tomography combined with differential pulse code modulation (DPCM) was investigated. Tomographic projections were formed from digital image data. Data redundancy, both within and across projections, was calculated and found to range from a maximum of 77% redundant to a minimum of 8% redundant. DPCM was used to encode the tomographic projections. Five different methods were used to form the predictions used in DPCM. Image encoding to approximately 2, 3 and 4 bits was performed. The images were reconstructed by decoding the projections and backprojecting the decoded data. Modest amounts of compression (2.5:1) were achieved with fairly good reconstructions. The main limitation to the amount of compression which can be achieved seems to lie in the fact that the tomographic process tends to blur images even without DPCM coding of the projection data.
CHAPTER 1

INTRODUCTION

Images are among the most potent tools available for communication. Increasingly, images are being represented in digital form as a result of innovations in image processing and storage equipment. This digitization of images has made it feasible to apply standard digital signal techniques to images; for example, it is possible to transmit pictures using any digital transmitting technique, and images can be stored on standard computer memory devices. As digital images become increasingly common, it will be essential to deal with them in an efficient manner.

Digital images often require large amounts of data to represent a single scene. There are disadvantages to dealing with these large batches of information; in storing images, significant amounts of data can tie up large chunks of computer memory and, in the transmission of images, a great deal of time is necessary to send or receive a single picture. It would be attractive, therefore, to be able to represent images by a smaller amount of data. A reduction in image data requirements is called compression and is the topic of this research. Compression consists of eliminating unnecessary or less essential data. The brief introduction to digital image representation given below will cover some basic concepts and vocabulary.
In a black and white digital image, a picture is composed of a two-dimensional array of discrete picture elements (pixels) or dots. Each pixel is quantized to a specific grey level (GL) which can range along the grey scale from white to black. It is the arrangement of the grey levels in the array which produces the visually perceived image. The grey level usually is represented by a fixed length binary word. For example, an 8 bit binary number will provide for \(2^8\) or 256 grey levels per pixel. Zero would correspond to black and 255 to white, and between these two extremes 254 different grey levels exist. To digitally represent an image, the total number of bits needed is equal to the size of the image array (total number of pixels) times the number of bits per pixel. Using some typical values, a square image array with 512 pixels on each side and 8 bits per pixel, requires over 2 million bits to specify the image.

Compression can help to reduce the data load of images. In general, compression works as follows. An original digital image, which is represented by \(M\) bits of data, is compressed using some technique which allows the image to be specified by only \(N\) bits of data, assuming \(N\) is less than \(M\). The state of the data after compression may have no visual resemblance to the original image. It is in the storage or transmittal of this compressed data that the memory sparing or time and energy saving benefits of this reduction are realized. Later, when the data is needed in image form again, the stored or transmitted data must be "decompressed" to some facsimile of the original scene.
Compression Techniques

Two basic types of image compression algorithms are possible. Error-free compression schemes retain all of the information of the original image but usually result in a modest amount of data compression (a factor of 2-3 times). The second type is compression relative to some fidelity criterion in which errors are introduced into the decompressed image but more substantial amounts of compression are achieved (Rosenfeld and Kak, 1982; Gonzalez and Wintz, 1977). In the latter case, a trade-off exists between the amount of distortion introduced into the decompressed image and the bit rate (the average number of bits per pixel) of the compressed data. Lower bit rates (greater degrees of compression) generally produce more errors in the reconstructed image.

One main objective of compression schemes is to eliminate redundant information, thereby retaining only uncorrelated or orthogonal data. A second objective is to code more efficiently; unlike the decorrelation scheme above, this does not reduce the amount of redundancy in the information but searches for more efficient ways to represent it. Compression schemes commonly employ one or both of these principles.

A variety of methods has been designed to achieve compression. A brief survey of some well-known methods follows.

Orthogonal Transform Techniques

Orthogonal transforms consist of a mapping procedure. The original data are mapped or projected onto a set of orthogonal
functions so that the original collection of dependent data samples can be represented by a group of uncorrelated coefficients. The mapping procedure is accomplished through a reversible linear transformation, such as the Fourier, Karhunen-Loeve, Hadamard, or cosine transforms (Rosenfeld and Kak, 1982; Netravali and Limb, 1980). Additionally, once these coefficients are known, they can be ranked in importance, and those of lesser consequence can be eliminated, reducing the amount of information to be coded.

Prediction Techniques

The grey levels of adjacent pixels within an image are likely to be well correlated. According to Gonzalez and Wintz (1977), for an image with 256 grey levels, the average difference between adjacent pixels is approximately 20 grey level values. This similarity can be exploited by means of predictive coding (Gonzalez and Wintz, 1977; Jayant, 1974; Jayant and Noll, 1984; Netravali and Limb, 1980; Rosenfeld and Kak, 1982). Using this method, we attempt to predict some pixel, \( x(n) \), based on the value of its previously observed neighbor(s), e.g., \( x(n-1) \), \( x(n-2) \). The discrepancy between the predicted and actual values can be obtained, resulting in an error or difference value for each pixel:

\[
d(n) = x(n) - f(x(n-i))
\]

where \( f(.) \) is some prediction method and \( x(n-i) \) is one or several previous data points. Assuming an accurate prediction scheme, the difference values, \( d(n) \), will have a much smaller range than the
original pixel values. Therefore, these pixel difference values can be retained and quantized using fewer bits than would be required by the original pixel data.

Interpolation Techniques

Like predictive coding mentioned above, interpolation techniques rely on correlations in adjacent pixel values. A subset of the image pixels are retained while all others are discarded (Hunt, 1978; Hunt, Strickland and Schowengerdt, 1983; Netravali and Limb, 1980). For example, alternate lines in an image could be discarded and within the lines which are not discarded, every other pixel could be retained. Subsequently, the values of the "lost" pixels are interpolated from those that remain using a variety of techniques.

Nonorthogonal Techniques

One example of nonorthogonal techniques is contour tracing (Gonzalez and Wintz, 1977). In contour tracing or encoding, the discrete grey levels of the image can be visualized as a series of plateaus, similar to a topographic map. The plateau boundaries trace out edges or contours in the picture. A large plateau would result if many pixels in an area were at the same grey level while if there were fewer adjoining pixels at a specific grey level a smaller plateau would be produced. The entire image can be determined from knowledge of the height (grey level), shape and position of each plateau or contour. Therefore, the image can be broken down into a number of contours, each specified by a grey level, its location or initial point, and the
direction of travel to trace the outer edge. Pixels located inside the contour are all assigned the same grey level. This method can be used to provide error-free encoding.

Variable Word Length Encoding (Huffman Codes)

Techniques using variable word length codes depend on the fact that different grey levels have different probabilities of occurrence (Gonzalez and Wintz, 1977). Compression can be accomplished by assigning different length binary code words based on the probability of grey level values. First, the image is usually decorrelated. Next the probability density of the resulting decorrelated values is calculated and codes, consisting of different length binary words, are assigned to the various values. The most probable levels have the shortest codes, while the least frequent levels have the longest.

Current Method

The method of image compression which will be investigated in this study can be roughly considered to be a composite of the first two techniques, namely a variation on orthogonal transform compression combined with predictive coding. In this approach, a computer simulation of tomography (related to the Fourier transform) is performed on the image. Following this, the projections, which are the result of the tomographic procedure, are encoded using differential pulse code modulation (DPCM) which is a type of predictive coding. The compressed image data can be stored or transmitted in the form of these encoded projections. When the actual image is needed again, the projections
are decoded and subsequently "backprojected" to regenerate an approximation of the original image (see Figure 1).

One attractive feature of the tomographic process is that it is not restricted to implementation on digital computers. It is possible to use a simple optical system to produce and backproject tomographic projection data. If computer simulation of compression via encoded projections is found to be promising, construction of such an optical system could be undertaken. This system would greatly increase the speed and simplicity of the tomographic process involved in this method of compression (Fraser, Hunt and Su, 1985).

The compression resulting from both tomography and the predictive coding method is achieved by exploiting correlations in the data. It is well known that image data are highly correlated, thereby justifying the use of an orthogonalizing technique such as tomography (typical correlations range from 0.85 to 0.95 (Gonzalez and Wintz, 1977)). However, it is less clear whether tomographic projection data are correlated. Therefore, before DPCM should be used to encode tomographic projections, the existence of correlations in the data needs to be established.

Both of these techniques, tomography and DPCM, used by themselves, can provide some measure of compression. It is the aim of this research to determine whether the combination of these two methods will increase the amount of compression which can be achieved.

To summarize the approach of this research: first, it is necessary to verify that redundancy exists in tomographic projections.
Assuming that redundancy is present, the objective of this study is to determine how well the combination of tomography and DPCM encoding work for compressing images. The main variable which will be investigated within the framework described in Figure 1 is the efficacy of various types of DPCM encoding algorithms. A number of prediction schemes are possible. Several are examined in this study, and their success is evaluated with respect to their computational complexity, the quality of the reconstructed image, and the amount of compression achieved.
Figure 1. Stages in compression and decompression of image data. -- (a) Compression stage; and (b) Decompression stage.
CHAPTER 2

TOMOGRAPHY

This section will present the mathematical and physical foundations of tomography and introduce the concept of redundancy as it applies to tomographic projections.

Tomography is a technique in which an N-dimensional object is decomposed into a series of (N-1)-dimensional projections. Each projection is made at a different angle through the original object. Radiographs or X-rays are one common application of tomography. In the case of a three-dimensional object, such as a human body which is being X-rayed, for example, the three-dimensional form is collapsed along one dimension and produces a two-dimensional projection or representation at one angle. In X-rays of bodies, the variable of interest is the tissue density of the form. When tomography is performed on images, however, the two-dimensional picture is broken down into a series of one-dimensional projections, and the variable of interest is the pixel intensity (grey level) in the image.

A projection is produced by integrating the data in one direction across the image. Along the horizontal or x-axis, for example, a projection is defined by:

\[ P_\theta(x) = \int_{-\infty}^{\infty} f(x,y)dy \]  

(1)
where $P_\theta$ is the projection at angle $\theta$ and $f(x,y)$ is the original image. Figure 2 shows an example of a projection made at an angle of $\theta = 0^\circ$. For the next projection, the angle must be incremented; either the coordinate system or the object is rotated and another projection is calculated. A rotated coordinate system, $x'$ and $y'$, can be calculated using:

$$
x' = x \cos \theta + y \sin \theta
$$
$$
y' = -x \sin \theta + y \cos \theta
$$

(2)

Generalizing Equation 1, projections can be calculated at any angle by

$$
P_\theta(x') = \int_{-\infty}^{\infty} f(x',y') dy'
$$

(3)

Figure 3 shows two projections generated from a simple geometric image using a rotated coordinate system. The projecting beams are parallel to the $y'$ axis and perpendicular to the $x'$ axis.

A fundamental relationship exists between the one-dimensional Fourier transform of the projections and the two-dimensional Fourier transform of the original image (Dudgeon and Mersereau, 1984; Rosenfeld and Kak, 1982). This relationship is explained in the following section.

**Fourier Slice Theorem**

Let $S_\theta(w)$ be the Fourier transform of the projection $P_\theta(x')$. Then,

$$
S_\theta(w) = \int_{-\infty}^{\infty} P_\theta(x') \exp(-j2\pi wx') dx'
$$

(4)
Figure 2. The projection of an image at an angle of zero. -- $P_0$ is the projection at angle zero and $f(x,y)$ is the image.
Figure 3. Two projections generated using a rotated coordinate axis.
Also, let \( F(u,v) \) be the two-dimensional Fourier transform of the original image \( f(x,y) \).

\[
F(u,v) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x,y) \exp(-j2\pi[ux + vy]) \, dx \, dy \tag{5}
\]

where \( u \) and \( v \) are the axes in the frequency or Fourier plane. Consider \( F(u,v) \) in Equation 5 for the case where \( v = 0 \). By substituting in Equation 1, \( F(u,v) \) can be reduced to

\[
F(u,0) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x,y) \exp(-j2\pi ux) \, dx \, dy
= \left[ \int_{-\infty}^{\infty} f(x,y) \, dy \right] \exp(-j2\pi ux) \, dx
= \int_{-\infty}^{\infty} p_\theta(x) \exp(-j2\pi ux) \, dx
= s_\theta(w) \quad \text{for the case } \theta = 0 \tag{6}
\]

The lines above indicate that the one-dimensional Fourier transform of the projection at \( \theta = 0^\circ \) equals the two-dimensional Fourier transform of the image along the line \( v = 0 \). If we let

\[
F(w,\theta) = F(u,v)
\]

by switching to polar coordinates, the previous result in Equation 6 can be generalized.

\[
P_\theta(x') = \int_{-\infty}^{\infty} f(x',y') \, dy'
\]
\[
S_\theta(w) = \int_{-\infty}^{\infty} P(x') \exp(-j2\pi wx') dx'
\]

\[
= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x',y') dy' \exp(-j2\pi wx') dx'
\]

(7)

and substituting for the coordinate system of Equation 2 into Equation 5, we get

\[
S_\theta(w) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x,y) \exp[-j2\pi w(x\cos\theta + y\sin\theta)] dx dy
\]

\[
= F(u,v) \quad \text{for } u = w \cos\theta, \ v = w \sin\theta
\]

\[
= F(w,\theta)
\]

\[
S_\theta(w) = F(w,\theta)
\]

(8)

This result shows that the one-dimensional Fourier transform of a tomographic projection made at angle \(\theta\) is equal to the two-dimensional Fourier transform of the image evaluated along a line passing through the origin of Fourier space at an angle \(\theta\). Figure 4 demonstrates that relationship.

In other words, if projections were made at an infinite number of angles and Fourier transformed, then \(F(u,v)\) would be determined at every point in the frequency domain. For a finite number of projections, \(F(u,v)\) is known along a number of radial lines, as shown in Figure 5.

**Backprojection**

Knowledge of this relationship between projections and the Fourier domain helps to clarify the algorithm used to reconstruct the
Figure 4. Fourier slice theorem. -- Relationship between one projection of a two-dimensional function and a slice through Fourier space.
Figure 5. Fourier slice theorem for multiple projections. -- Relationship between several projections and their Fourier transformed versions.
image from the projections. This technique of reconstruction is known as backprojection. Continuing with the use of polar coordinates, it is evident that the original image can be reconstructed by performing the inverse Fourier transform (IFT) on the frequency domain data.

\[
f(x,y) = \int_{0}^{2\pi} \int_{0}^{\infty} F(w,\theta) \exp\left[j2\pi w(x\cos\theta + y\sin\theta)\right] dw d\theta
\]

\[
= \int_{0}^{\pi} \int_{0}^{\infty} F(w,\theta) \exp\left[j2\pi w(x\cos\theta + y\sin\theta)\right] dw d\theta + \int_{\pi}^{2\pi} \int_{0}^{\infty} F(x, \theta+\pi) \exp\left[j2\pi w(x\cos(\theta+\pi) + y\sin(\theta+\pi))\right] dw d\theta
\]

(9)

and since \( F(w,\theta+\pi) = F(-w,\theta) \) and \( x' = x\cos\theta + y\sin\theta \)

\[
f(x,y) = \int_{0}^{\pi} \int_{-\infty}^{\infty} F(w,\theta) \left|w\right| \exp(j2\pi wx') dw d\theta
\]

\[
= \int_{\pi}^{2\pi} \int_{0}^{\infty} S_{\theta}(w) \left|w\right| \exp(jw2\pi x') dw d\theta
\]

(10)

The term in square brackets can be considered to be the projections filtered by a high pass (w) filter and subsequently inverse Fourier transformed. For notational simplicity, let
\[ Q_\theta(x') = \text{IFT}[S_\theta(w) \cdot |w|] \]
\[ = \int_{-\infty}^{\infty} S_\theta(w) \cdot |w| \exp(j2\pi wx') \, dw \]
\[ = P_\theta(x') * h(x') \quad (11) \]

where '*' denotes convolution and \( h(x') \) denotes a filter whose frequency response is given by \( H(w) = |w| \). Therefore
\[ f_p(x, y) = \int_0^\pi Q_\theta(x') \, d\theta \]
\[ = \int_0^\pi Q_\theta(x \cos \theta + y \sin \theta) \, d\theta \quad (12) \]

where \( f_p \) is the backprojected or reconstructed version of the image.

**Digitization and Simulation**

The previous section introduced the mathematical basis for tomography and backprojection in the domain of continuous variables. Simulation of the procedure on a computer, however, requires discrete data. Within the constraints imposed by digitization, only a finite number of projections can be made; each projection will consist of a finite number of samples, and the sample values will be quantized into discrete levels.

The angles at which projections are made can be designated as \( \theta_0, \theta_1, \ldots, \theta_{N-1} \), where:
\[ \Delta \theta_i = \theta_i - \theta_{i-1} \quad \text{where } i = 0, 1, \ldots, N-1 \quad (13) \]
The discrete version of Equation 3, which is used in projecting the images, becomes:

\[ P_{\theta_i}(x'_n) = \Delta y_k \sum_{k=-\infty}^{\infty} f(x'_n, y'_k) \]  

where \( P_{\theta_i} \) and \( f \) are discrete functions and \( x'_n \) and \( y'_k \) represent integer samples along the \( x' \) and \( y' \) axes, and \( n \) and \( k \) can range from negative to positive infinity.

Similarly, to backproject the data, a discrete version of Equation 12 is used:

\[ f(x_n, y_k) = \sum_{i=-\infty}^{\infty} \Delta \Theta_i \Theta_i(x_n \cos \Theta_i + y_k \sin \Theta_i) \]  

A digital image is a discrete array of grey levels. In determining projections from such an array, a specific pixel model is assumed. The simplest method is to model the pixel as a concentration of intensity at one point, the center of the pixel (Kashyap and Mittal, 1975). A second model is to assume the intensity of the pixel is uniformly distributed along a line parallel to the rotated \( x \) axis (Hunt, Strickland and Schowengerdt, 1983). The second model is used in this study because it produces better reconstruction results.

The tomography simulation program used in this study was written by Mr. J. C. Su (Hunt, Strickland and Schowengerdt, 1983). The major components of the optical system which is being simulated are shown in Figure 6 (Fraser, Hunt and Su, 1985). A physical analog to the digital simulation is shown in Figure 7. In the case of an \( N \times N \) image, consider the system generator to be a series of ray sources, all
Figure 6. Optical tomographic encoder.
Figure 7. Model of the digital tomographic encoder.
rays having a specific, fixed width. If each ray is one pixel wide and
the projection is made at an angle of zero, each ray passes through one
column of pixels and strikes a single detector (also one pixel wide) in
the receiver. The "force" with which the ray impinges upon its
detector reflects the sum of grey level intensities found within that
column of pixels. Therefore, each projection is made up of a series of
discrete samples, each of which arrives at each discrete detector. At
an angle of zero, N detectors (one pixel wide) would be sufficient to
cover an N x N image. However, as the projection angle increases
towards 45° or 135°, additional rays and detectors are needed, assuming
a noncircular image. For a square image, at 45° or 135° the string of
detectors is parallel to a line drawn through the diagonal of the
image. Consequently, the maximum number of detectors and ray sources
needed is equal to ND where

\[ ND = \text{smallest integer } \geq 2^{1/2}N \]  \hspace{1cm} (16)

In order to use a constant number of detectors and sources, the
number is fixed at ND regardless of the angle of the projection. At an
angle of 45°, the string of detectors spans the spatial extent of the
data; however, at 0° or 90° the endpoints of the detector array swing
outside the image and register artificial zero data (see Figure 8).

A "projection matrix" can be compiled by arranging the
horizontal lines of projection data in a descending fashion. Figure 9
shows an example of a typical projection matrix where the numerical
results have been scaled to the range of a visual display. Each
Figure 8. Positioning and number of detectors required. -- The number of detectors varies as a function of the projection angle. (a) At a projection angle of 45°, seven detectors are required; and (b) At an angle of 0°, only five detectors register true data; the extra detectors swing out of the image.
Figure 9. Projection matrix (sinogram) from a square image.
horizontal line corresponds to the projection data gathered by all \( N \) detectors at one specific angle; a horizontal line, therefore, is referred to as a data sequence from "within" a single projection (see Figure 10). Conversely, a vertical column through the projection matrix represents the data integrated at one specific detector as it travels through all \( M \) angles; this is referred to as data "across" projections.

The projection matrix of any image displays certain consistent characteristics. As seen in the projection matrix of Figure 9, there is a sinusoidal design woven into the matrix. This pattern, in varying degrees, is present in the projection matrix derived from any image.

It has been hypothesized previously that compression could be achieved through the use of tomographic principles by taking advantage of the fact that adjacent projections, for small angle differences, are reasonably similar (Fraser, Hunt and Su, 1985). Variables such as the number of projections made, the number of samples per projection and the number of bits per sample can all be manipulated to achieve specific data rates. Exploitation of these redundancies between projections should make it possible to provide further compression possibilities.
Figure 10. Data within vs across projections.

a. Within projections: the original image, at left, is integrated along y' and sensed by each detector along x'. For example, the results at detectors 3 and -2 are the sums of values in the shaded areas of the original image. The data from projection angle $\theta_1$ are placed into the projection matrix along a horizontal line.

b. Across projections: a vertical line in the projection matrix, e.g., along detector 3, represents the integrated result from a single detector in the original image as it swings through all possible angles.
Figure 10. Data within vs across projections.
CHAPTER 3

REDUNDANCY

The last section dealt with tomographic principles and suggested that redundancies or correlations, if present in the projections, could be exploited to provide data compression. First, however, the presence and amount of redundancy must be verified. It is well known that the data in images are highly redundant (Gonzalez and Wintz, 1977). Based on this, one might surmise that the projection matrix also contains substantial amounts of redundant information. The methods of assessing information and redundancy are presented below.

*Measures of Information*

In an intuitive sense, information is the property of events or messages through which it is possible to reduce, to some degree, the amount of uncertainty about a situation (Papoulis, 1965; Stremler, 1982). The amount of information conveyed by some event "A," occurring with the probability $p(A)$, is defined as:

$$\text{I}(A) = \log_2 \frac{1}{p(A)} \text{ bits}$$  \hspace{1cm} (17)

where "log2" means log to the base 2 and a "bit" is defined as the basic quantum unit of information. For the case where any one of equally probable events can occur, each event carries the same amount of information:
\[ I(i) = \log_2 \left[ \frac{1}{p(i)} \right] \text{bits} \quad i = 1, 2, \ldots, N \quad \text{and} \quad p(i) = \frac{1}{n} \]

In this situation, the amount of information in each event is constant across all events. To generalize to the case where \( N \) events are possible but not all equiprobable, the average information associated with the set of events can be calculated. This average is called the entropy of the set, or \( H \), and is defined by:

\[ I_{\text{avg}} = H = \sum_{i=1}^{n} p(i) \log_2 \left[ \frac{1}{p(i)} \right] \text{bits} \]  

where \( p(i) \) is the probability of the \( i^{\text{th}} \) event. The information contained in a set of events is equal to the minimum number of binary digits which are needed to provide each member with a unique binary label.

Assume that a sequence of events or messages is generated from a set of possibilities. If the set contains \( N \) equiprobable elements, and they are selected independently, then the case of maximum entropy occurs where \( H = \log_2 N \) bits. However, a sequence is often not a collection of independent equiprobable events; instead, a sequence such as the series of letters used to form words, may be formed under specific constraints or rules of generation. Constraints which provide structure or regularity to a sequence can be of two types.

According to Sheridan and Ferrell (1974), distributional constraints result when set elements are not equiprobable and sequential constraints occur when adjacent elements are not independent, i.e.,
information about an element can be determined from preceding elements. In the presence of such constraints, the average information or entropy of each element is less than \( \log_2 N \). The redundancy, \( R \), is defined as the difference between the maximum and actual entropy values.

\[
R = H_{\text{max}} - H_{\text{actual}}
\]  

(20)

and the percent of redundancy can be computed as

\[
\%R = 100 \frac{H_{\text{max}} - H_{\text{actual}}}{H_{\text{max}}}
\]  

(21)

These equations can be applied to projections to determine their amount of redundancy.

Redundancy of Digital Information

As demonstrated in the previous equations, to calculate redundancies, the probability of events must be known. In the case of a digital sequence, an "event" is defined as the occurrence of a particular sample value. A tomographic projection is a digital sequence whose discrete samples represent summed grey levels. A probability distribution of these summed values can be determined. This distribution reflects the proportion of samples which have a particular value or fall in a particular range of values. Figure 11 shows how a hypothetical probability histogram may appear for a projection at a specific angle. The probability that the value of a given sample will fall between 255 and 512, in this example, is 0.15.

Once the actual probability distribution is known, the entropy can be calculated within or across projections using the formulae:
Figure 11. Hypothetical probability distribution of a projection matrix. \( P_i \) represents the probability of the occurrence of a particular sample value in the matrix.
\[ H_\theta = \sum_{i=1}^{n} p_\theta(i) \log_2[1/p_\theta(i)] \text{bits (within proj.)} \quad (22) \]

\[ H_d = \sum_{i=1}^{n} p_d(i) \log_2[1/p_d(i)] \text{bits (across proj.)} \quad (23) \]

where \( n \) is the number of discrete grey level ranges in the probability histogram, \( \theta \) is a specific angle, \( d \) is a specific detector location in the receiver, and \( p_\theta(i) \) or \( p_d(i) \) is the probability of a particular data point being contained in the \( i \)th cell of the histogram.

The redundancy in a line of data then can be determined:

\[ R_\theta = H_{\text{max}} - H_\theta \quad \text{(within projections)} \quad (24) \]
\[ R_d = H_{\text{max}} - H_d \quad \text{(across projections)} \quad (25) \]

where \( H_{\text{max}} \) is the maximum amount of information possible based on the assumption of equiprobable data ranges in the histogram.

**Redundancy Measurement Method**

Redundancy within and across projections was measured for five images. Each original image was 128 x 128 pixels in size. Projections were derived from these images using the tomographic simulation program mentioned in Chapter 2. One hundred projections were obtained, each made 1.8° apart; each projection "receiver" had 182 detectors, each detector representing one sample and being one pixel wide. The original images are shown in Figure 12 and will be referred to as Girl, Clock, Test, River and Map, respectively. Two of these images, Girl and Clock, are subsampled versions of larger images (512 x 512);
Figure 12. Images used in the redundancy calculations.

a. Girl.
b. Clock.
c. Test.
d. River.
e. Map.
subsampling usually reduces the amount of redundancy in an image by removing neighboring pixels which tend to have similar grey levels. The other three images were not subsampled but were produced by extracting a section of a larger image.

For every projection angle, i.e., each horizontal line in the projection matrix, a redundancy value was calculated. A histogram or probability distribution was computed for the data sequence at each angle. The probability distribution was divided into 16 equal length data ranges (cells) and the number of data points falling in each region was determined. Based on the population of each data range, a probability value was calculated. Subsequently, $H$ values and redundancies were calculated for the data sequence at each angle from Equation 20. The maximum amount of information ($H_{\text{max}}$) occurs when all cells of the histogram have equal probabilities. In the case where 16 data ranges are possible,

$$H_{\text{max}} = \log_2 (16) = 4 \text{ bits}$$

The same procedure was followed to calculate redundancies for the data sequence of each detector (each vertical line in the projection matrix).

To simplify calculations, the scalloped ends of the projection matrices are not used in the calculation of redundancies, nor are they used later in the compression and backprojection sections. As mentioned earlier, these end sections (see Figure 8) contain spurious, artificial zeros. If these artificial data are treated the same as
real data, erroneous results could occur. Therefore, in this and subsequent sections, the projection matrix under consideration is the data contained in the largest central rectangular region that can be formed without including any artificial zeros (Figure 13). The final dimensions are 100 horizontal lines (i.e., 100 angles) by 128 vertical lines (128 detectors). Backprojection using a rectangular projection matrix produces a circular image.

**Redundancy Results**

Figure 14 shows the redundancy calculated for the data within each individual projection as a function of the projection angle. Each graph represents redundancy measures from one image. The value of redundancy for the data at various projection angles ranges from a maximum of 2.98 bits (75% redundancy) to a minimum of 0.30 bits (8% redundancy). The degree of redundancy for data within projections appears to be strongly dependent on the specific angle of the projection. Redundancy tends to be highest near the angles of 0°, 90° and 180° and is lowest near angles of 45° and 135°. Greater redundancy at certain angles occurs due to the way the tomographic process operates on a square image. As the string of detectors swings through each projection angle, the number of pixels contributing to the sum at the individual detectors changes. At a given projection angle, some detectors in the string may have more pixels impinging on them than do other detectors. At angles near 0°, 90° and 180°, all detectors register the summation of an equal number of pixels (approximately). At angles near 45° and 135°, however, the number of pixels which
Figure 13. Trimmed projection matrix. -- The data from the outermost detectors have been eliminated, resulting in a square matrix.
Figure 14. Redundancy within projections as a function of projection angle. -- (A) Girl, (B) Clock, (C) Test, (D) River, and (E) Map.
contribute to the sum at the central detectors in the string is much greater than the number of pixels contributing to detectors on the ends of the string. Therefore, in the first case (angles near 0°, 90° and 180°) it is more likely that similar results will be obtained at all data points (detectors) in the sequence; this produces a high redundancy situation. However, in the second case (angles near 45° and 135°) it is more likely that a large variation in results will be achieved in the data sequence; large variations produce lower redundancy. Figure 15 shows how these different redundancy results for various projection angles may be produced by the tomographic process on a completely redundant image.

Redundancy across projections as a function of detector position is shown for each image in Figure 16. This is equivalent to the redundancy in each vertical line (detector number being held constant) of the projection matrix. The redundancy values for the individual detectors range from a maximum of 3.06 bits (77% redundancy) to a minimum of 0.85 bits (21% redundancy). There is less regularity to the curves of this figure compared to those in Figure 14. Going back to the single grey level image example in Figure 15, it is possible to see that the detectors on the very end points of the string, as well as the most central detectors, experience radical variations in the number of pixels which contribute to their summations; this tends to lower the redundancy at these points on the curve. Detectors located about one-quarter and three-quarters of the way through the detector string, however, have a more constant number
Figure 15. The effect of angle on redundancy. -- The redundancy varies as a function of projection angle due to how tomography operates on a square image. Assuming a completely redundant image, the projection data at $\theta = 0^\circ$ is total redundant; however, the projection at $\theta = 45^\circ$ is only about 50% redundant. The calculations assume a maximum entropy of 4 bits.
Figure 16. Redundancy across projections as a function of detector position. -- (A) Girl, (B) Clock, (C) Test, (D) River, and (E) Map.
of contributing pixels. Therefore, due solely to the tomographic process (exclusive of individual image differences), the redundancy curve as a function of detector position should be approximately 'M' shaped, e.g., lower at the end points and the center. A couple of the graphs in Figure 16 seem to support this slightly, especially those derived from the Map and Test images, but the redundancy seems to rise abruptly again at the extreme ends of the detector string. Table 1 gives the minimum, maximum and mean redundancy values in bits and percentages for each image, both within and across projections.

The results in this table confirm that the subsampled images (Clock and Girl) have the lowest values of redundancy. This was expected because subsampling removes neighboring pixels which tend to contribute to redundancy.

Redundancy Conclusions

The results confirm that significant redundancy does exist in projection data. Within projections there is a fairly predictable shape for the curve of redundancy versus angle; most redundancy occurs around angles of 0°, 90° and 180°. Across projections, there is less regularity or predictability to the curve; in most cases, however, it seems that the redundancy curve is fairly symmetrical about the center detector. In general, redundancy calculated across projections tends to be somewhat higher than that calculated within projections.

The results from this section have verified that redundancy does exist in the projection matrix, albeit to varying degrees. With this knowledge, there is now a solid foundation to justify a
Table 1. Reduncancy within and across projections. -- In bits* and percentages.

<table>
<thead>
<tr>
<th>Image</th>
<th>Within Projections</th>
<th>Across Projections</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Max</td>
<td>Min</td>
</tr>
<tr>
<td>Clock</td>
<td>1.31-33%</td>
<td>0.30-8%</td>
</tr>
<tr>
<td>Test</td>
<td>1.81-45%</td>
<td>0.51-13%</td>
</tr>
<tr>
<td>River</td>
<td>2.86-72%</td>
<td>0.60-15%</td>
</tr>
<tr>
<td>Map</td>
<td>2.98-75%</td>
<td>0.45-11%</td>
</tr>
<tr>
<td>Girl</td>
<td>2.18-55%</td>
<td>0.37-9%</td>
</tr>
</tbody>
</table>

* The maximum possible redundancy is 4 bits and the minimum possible is 0 bits.
compression procedure whose main aim is to reduce the amount of redundancy in the projection matrix.
CHAPTER 4
DIFFERENTIAL PULSE CODE MODULATION

The coding procedure known as differential pulse code modulation (DPCM) operates on the principle of quantizing prediction error values rather than actual data values (Jayant and Noll, 1984). For some data sequence, \( x(n) \) where \( n = 1, 2, \ldots, N \), a predictor \( \hat{x}(n) \) is generated, based on the recent history of the signal. Ideally, \( \hat{x} \) should be as close as possible to the actual signal value. Then, rather than quantizing and storing the value of \( x(n) \), the difference between the predicted and actual values:

\[
d(n) = x(n) - \hat{x}(n)
\]

is quantized and stored. When this operation is performed for each data point, the signal is represented by a series of difference values. Subsequently, when it is necessary to regenerate the original signal, \( x(n) \), a decoding operation is performed by integrating (or in the case of discrete data, summing) these quantized difference values. Figure 17 shows a block diagram of the basic closed-loop DPCM system. In addition to Equation 26, other important equations are:

\[
Q_d(n) = d(n) - q(n)
\]

\[
y(n) = \hat{x}(n) + Q_d(n)
\]
Figure 17. Block diagram of DPCM system. -- (a) Encoding, and (b) Decoding.
where $Q_d(n)$ is the quantized prediction error, $q(n)$, is the quantization error, and $y(n)$ is the decoded approximation to the coder input.

For a data sequence, $x(n)$, and its difference signal, $d(n)$, the variance of $d(n)$, the input to the quantizer in Figure 17, is significantly smaller than the variance of $x(n)$, the original data (Jayant and Noll, 1984). Since the quantization error variance is directly proportional to the variance of the quantizer input, it is possible, using DPCM, to lower the quantization bit rate to a specific level and maintain an equivalent signal to noise ratio (SNR). These reductions in the bit rate constitute compression of the signal.

The closed loop or feedback-around quantizer design of Figure 17 ensures that quantization errors do not accumulate into subsequent samples, i.e., the errors should be independent. For each sample, the reconstruction error, $r(n)$, is equal to the quantization error for that sample.

$$
\begin{align*}
\quad r(n) &= x(n) - y(n) \\
&= x(n) - [\hat{x}(n) + Q_d(n)] \\
&= x(n) - [\hat{x}(n) + d(n) - q(n)] \\
&= q(n)
\end{align*}
$$

Consider one line of the projection matrix (either horizontal or vertical) to be the signal of interest, $x(n)$. If the principles of DPCM are applied to this signal, it should be possible to obtain a difference signal, $d(n)$, which requires substantially fewer bits to
encode than the original line of projection data. The original data line, or an approximation to it, can be recovered from \( d(n) \) by decoding. Subsequently, the image can be regenerated by backprojecting the decoded lines of the matrix. The accuracy of the prediction, \( \hat{x} \), is of the utmost importance because it controls the size and variance of \( d(n) \). There is a variety of prediction schemes which could be used. The five prediction methods used in the study are described below.

**Prediction Methods**

In all of the prediction methods, coding of the projection matrix is carried out one line (horizontal or vertical) at a time. The first value in each line (or sometimes the first two or three values) is carried through the DPCM system without formation of a prediction or difference value; the remaining values in the line are quantized at a reduced (compressed) rate.

The predictors in this study fall into two general classes. They are:

1. Minimum Mean Square Error Predictors
2. Slope Predictors

**Minimum Mean Square Error Methods**

This fixed (nonadaptive) method (Jayant and Noll, 1984) involves linear prediction of the form

\[
\hat{x}(n) = \sum_{j=1}^{n} h_j x(n-j)
\]  

(27)
The intent is to choose coefficients (h_j's) which minimize the square of the error or difference values. By varying the number of coefficients used, a number of different schemes can be devised.

The prediction in Equation 27 assumes that the data are statistically stationary; this assumption usually is violated with image data. Equation 27 also neglects the effects of quantization, i.e., from Figure 17 it can be seen that the prediction actually is derived from previously encoded predictions, not from the uncoded original samples, as implied by the equation above.

In this study, four variations of the basic equation in Equation 27 are used. The different methods can be characterized by the order of the predictor (the value of j) and the number of dimensions (one or two) used to generate the prediction. Figure 18 demonstrates these ideas. Assume that prediction within projections is being done. A segment of the image can be imagined as shown in Figure 18. If we are trying to predict the value of pixel A, examples of one-dimensional prediction would be to predict pixel A based on past values in the current line, e.g.,

\[ A = h_1 B \]  \hspace{1cm} (28)  
\[ A = h_2 B + h_3 C \]  \hspace{1cm} (29)  
\[ A = h_4 B + h_5 C + h_6 D \] \hspace{1cm} (30)

The three equations above are all one-dimensional, but Equation 28 is a first-order predictor while Equations 29 and 30 are second- and third-order predictors, respectively.
Figure 18. Model of pixel arrangement for prediction schemes.
A two-dimensional predictor is inherently second-order or more. An example of a second-order, two-dimensional predictor for Figure 18 would be:

\[ A = h_7B + h_8E \]  \hspace{1cm} (31)

Pixel B is drawn from the current scan line and pixel E is taken from a previous line; therefore, two-dimensional characteristics are being utilized. Switching to a more general notation, let \( A = x(n) \), the pixel to be predicted, \( B = x(n-1) \), \( C = x(n-2) \), etc. for prediction within a projection. In the case of prediction across projections, \( A = x(n) \) and \( E = x(n-1) \) and \( F = x(n-2) \).

The next issue to be tackled is to determine how the coefficients are chosen for each of these types of prediction.

**One-dimensional—First Order.** Returning to the notation where \( x(n) \) is the original signal and \( d(n) \) is the difference signal, we can define the one-dimensional, first-order predictor as

\[ \hat{x}(n) = h_1 x(n-1) \]  \hspace{1cm} (32)

The difference or error value becomes:

\[ d(n) = x(n) - \hat{x}(n) \]
\[ d(n) = x(n) - h_1 x(n-1) \]  \hspace{1cm} (33)

Taking the square of Equation 33 and using the expected value operator:
\[ E[d^2(n)] = E[x^2(n) - 2h_1x(n-1)x(n) + h_1^2x^2(n-1)] \]
\[ = E[x^2(n)] - 2h_1E[x(n-1)x(n)] + h_1^2E[x^2(n-1)] \]
\[ = R(0) - 2h_1R(1) + h_1^2R(0) \] (34)

where \( R(T) \) is the autocovariance function of a signal and can be defined as:

\[ R(T) = E[x(t)x(t+T)] - E[x(t)]^2 \quad T = 0, 1, \ldots, N \] (35)

'T' denotes the time or spatial lag between data points in the correlation. The maximum autocovariance occurs at a lag of 0. By definition,

\[ R(0) = E[x^2(t)] - E[x(t)]^2 \]
\[ = \sigma_x^2, \text{ the variance} \]

The autocovariance at \( T = 1 \) (e.g., \( R(1) \)) will therefore be less than or equal to \( R(0) \). We can write this relationship:

\[ R(1) = \rho_1R(0) \] (36)

where '\( \rho \)' can range between 0 and \( \pm 1 \) (inclusive). Solving for the general case:

\[ \rho_k = R(k)/R(0) \]

Equation 36 can be substituted into Equation 34, producing:

\[ E[d^2(n)] = R(0) - 2h_1\rho_1R(0) + h_1^2R(0) \]
\[ = R(0)(1 - 2h_1\rho_1 + h_1^2) \] (37)
To minimize the squared difference, take the partial derivative of Equation 37 with respect to $h_1$ and set it equal to zero.

$$\frac{\partial E[d^2(n)]}{\partial h_1} = R(0)[-2\rho_1 + 2h_1] = 0$$

Solving for $h_1$, we see the term in square brackets must equal 0.

$$h_1 = \rho_1$$  \hspace{1cm} (38)

Therefore, to minimize the squared error or difference, using a first-order predictor, it is necessary to let $h_1$ be set equal to $\rho_1$, the normalizing coefficient for the autocovariance. The first-order predictor becomes

$$\hat{x}(n) = \rho_1 x(n-1)$$  \hspace{1cm} (39)

The first-order example above illustrates what is meant by minimizing the mean squared difference or error (MSE). If the mean square error is minimized, then the mean absolute error is also minimized. This implies that the input to the quantizer is minimized and consequently, fewer bits are needed to quantize the data.

**One-dimensional—Second Order.** Let the predictor be given by:

$$\hat{x}(n) = h_1 x(n-1) + h_2 x(n-2)$$  \hspace{1cm} (40)

Following the reasoning for the first-order case,

$$d(n) = x(n) - \hat{x}(n)$$

$$= x(n) - h_1 x(n-1) - h_2 x(n-2)$$
Take the partial derivation with respect to $h_1$:

$$\frac{\partial^2}{\partial h_1^2}[d^2(n)] = \frac{\partial}{\partial h_1}[x(n) - h_1x(n-1) - h_2x(n-2)]^2$$

$$= 2[x(n) - h_1x(n-1) - h_2x(n-2)][-x(n-1)]$$

Setting $\frac{\partial}{\partial h_1}d^2(n) = 0$, it can be shown that:

$$0 = -x(n)x(n-1) + h_1^2x^2(n-1) + h_2^2x(n-1)x(n-2)$$

$$= -R(1) + h_1R(0) + h_2R(1)$$

$$= -\rho_1R(0) + h_1R(0) + h_2\rho_1R(0)$$

$$= R(0)[-\rho_1 + h_1 + h_2\rho_1]$$

Again, setting the bracketed term to zero and solving for $h_1$:

$$h_1 = \rho_1 - \rho_1h_2 \quad \text{(41)}$$

Taking the partial derivative with respect to $h_2$ and setting it to zero:

$$\frac{\partial}{\partial h_2}d^2(n) = 0 = 2[x(n) - h_1x(n-1) - h_2x(n-2)]x(n-2)$$

$$= [R(2) - h_1R(1) - h_2R(0)]$$

$$= [\rho_2R(0) - h_1\rho_1R(0) - h_2R(0)]$$

and

$$h_2 = \rho_2 - \rho_1h_1 \quad \text{(42)}$$

where $\rho_2 = R(2)/R(0)$.

Substituting Equation 41 into Equation 42:
\[ h_2 = \rho_2 - (\rho_1 - h_2 \rho_1) \rho_1 \]
\[ = \rho_2 - \rho_1^2 + \rho_1 h_2 \]  
(43)

and finally:
\[ h_2 = \frac{\rho_2 - \rho_1^2}{1 - \rho_1} \]
\[ h_1 = \frac{\rho_1(1 - \rho_2)}{1 - \rho_1^2} \]  
(44)

One-dimensional—Third Order. For one-dimensional, third-order prediction, the estimate is given by:
\[ \hat{x}(n) = h_1 x(n-1) + h_2 x(n-2) + h_3 x(n-3) \]  
(45)

The difference equation becomes:
\[ d(n) = x(n) - h_1 x(n-1) - h_2 x(n-2) - h_3 x(n-3) \]  
(46)

Squaring Equation 46 and solving for minimum partial derivatives as before, the resulting coefficients are:
\[ h_1 = \rho_1 - h_2 \rho_1 - h_3 \rho_2 \]  
(47)
\[ h_2 = \frac{\rho_2 - \rho_1^2 + h_3 (\rho_1 \rho_2 - \rho_1)}{1 - \rho_1^2} \]  
(48)
\[ h_3 = \frac{\rho_3 - 2 \rho_1 \rho_2 - \rho_1^2 \rho_3 + \rho_1 \rho_2^2 + \rho_1^3}{1 - 2 \rho_1^2 + 2 \rho_1^2 \rho_2 - \rho_2^2} \]  
(49)
It has been noted by Netravali and Limb (1980), for image data, that use of the prior three samples usually does not provide much improvement over prediction using the previous two samples.

**Two-dimensional—Second Order.** In the previous sections, the estimate of \( x(n) \) was always based one or more values drawn from a single line (horizontal or vertical) in the projection array. A two-dimensional estimate is based on one or more values from the current line and one or more values from a previous line. In this study, the pixel in position A (Figure 18) was predicted from pixel B and pixel E.

\[
\hat{x}(n) = \hat{x}_A = h_1 x_B + h_2 x_E
\]  
(50)

and the difference is given by:

\[
d(n) = x_A - h_1 x_B - h_2 x_E
\]  
(51)

The coefficients \( h_1 \) and \( h_2 \) for this predictor are found to be:

\[
h_1 = \frac{\rho_{AB} - \rho_{AE} \rho_{BE}}{1 - \rho_{BE}^2}
\]  
(52)

\[
h_2 = \frac{\rho_{AE} - \rho_{AB} \rho_{AE}}{1 - \rho_{BE}^2}
\]  
(53)

where

\[
\rho_{AB} = R_h(1)/R_h(0)
\]  
(54)

\( R_h(T) \) is the average autocovariance for the horizontal lines of the projection matrix. Likewise,
where $R_v$ is the average autocovariance for the vertical lines. And finally,

$$\rho_{BE} = R_d(1)/R_d(0)$$  \hspace{1cm} (56)

where $R_d(T)$ is the autocovariance for a signal which consists of data taken diagonally through the matrix.

**Slope Method**

In contrast to the above prediction methods, the slope method is not concerned with minimizing the mean square error. Instead this method is based on the average rate of change of the data. As mentioned earlier, all projection matrices share a basic sinogram pattern. Therefore, it should be possible to make an accurate estimate of $x(n)$ based not only on recent signal history but also on the position of $x$ in the matrix and the average rate of change between adjacent pixels in that area of the matrix. If the trimmed projection matrix is partitioned into a set of equal-size blocks, an estimate can be made of the average rate of change between $x(n)$ and $x(n-1)$ in each block. This average rate of change, or slope, can be used as a predictor in the coding and decoding stages. Figure 19 shows how the matrix can be subdivided. The equation below shows how the estimate is formed:

$$\hat{x}(n) = x(n-1) + \text{slope}(m,i)$$  \hspace{1cm} (57)
Figure 19. Partitioning of a trimmed projection matrix. -- The matrix is divided into MI blocks and slope values are calculated for each block.

<table>
<thead>
<tr>
<th>Angle $\theta$</th>
<th>1,1</th>
<th>1,2</th>
<th>1,3</th>
<th>1,4</th>
<th>...</th>
<th>1,M</th>
</tr>
</thead>
<tbody>
<tr>
<td>1,1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1,1</td>
</tr>
<tr>
<td>2,1</td>
<td>2,1</td>
<td>2,2</td>
<td></td>
<td></td>
<td></td>
<td>2,1</td>
</tr>
<tr>
<td>3,1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>...</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>M,1</td>
<td>M,2</td>
<td>...</td>
<td></td>
<td></td>
<td></td>
<td>M,1</td>
</tr>
</tbody>
</table>
where \( x(n-l) \) is contained in the block \((m,i)\) of the matrix and slope \((m,i)\) is the average rate of change within the block \((m,i)\). Both horizontal and vertical slopes are computed for each block. The horizontal slope is used to form predictions within projections and the vertical slope is used for estimates across projections. For a block consisting of 10 x 10 pixels, for example, the slope can be calculated with:

\[
\text{slope}(m,i) = \frac{1}{100} \sum_{l=0}^{9} \sum_{n=0}^{9} x(10m+l, 10i+n)
\]

for \( m = 0, \ldots, M \) and \( i = 0, \ldots, I \)

\[(57)\]

After their calculation, these slope values are retained and are used in both the coding and decoding phases.

**Error Quantization**

Much of the compression achieved through the use of DPCM schemes is a result of not quantizing the prediction errors as finely as the original data. Quantization using nonuniform step sizes is called companding. This can be a very useful technique when the data to be quantized have a large dynamic range but much of the signal energy is concentrated within a narrow band centered around zero. Figure 20 shows a sample of nonlinear quantization.

If the central steps (near zero) of the quantizer are too coarse (large dynamic range), then data which are slowly varying are quantized too coarsely and give the appearance of having random noise
Figure 20. Nonuniform quantization. -- The quantization levels are of gradually increasing size.
added to the data. Conversely, if the dynamic range of the quantizer is too small, the phenomenon of "slope overload" occurs in which the quantizer is not able to "catch up" to abrupt, large-magnitude changes in the input until several subsequent samples have passed.

A type of quantization which is nonadaptive and statistically based is derived from the work of Max (1960). If d(n) is the input to the quantizer with a probability density of p(d), it is possible to select decision and reconstruction levels which minimize the quantized error, QE.

\[
QE = \sum_{i=1}^{N} \int_{D_i}^{D_{i+1}} f(d-Qd_i)p(d)dd
\]

where \( D_1 < D_2 < \ldots < D_{N+1} \) and \( Qd_1 < Qd_2 < \ldots < Qd_N \) are the decision and reconstruction levels, respectively and \( f(\cdot) \) is a nonnegative function. Assume that for any input, \( d(n) \), where \( D_i < d \leq D_{i+1} \), the corresponding output is \( Qd_i \).

The prediction errors, \( d(n) \), the input to the quantizer tend to be distributed in a Laplacian (exponential) probability distribution (Jayant and Noll, 1984; Metravali and Limb, 1980). The decision and reconstruction values of the quantizer used in this study have been chosen to minimize the mean square error of quantizing a Laplacian. The levels, used in the quantizer of Figure 17, are given by Pratt (1978) and are companded as in Figure 20. The decision and reconstruction values of the quantizer vary as a function of the number
of bits available for quantization (number of quantization levels) and the variance of the input to the quantizer.

For a given bit rate, the value of the decision and reconstruction levels is set by specifying the standard deviation of the Laplacian distribution to be quantized. A complication arises due to the interaction between quantization levels and standard deviation. This interaction results from the feedback characteristic of the closed-loop DPCM system. As mentioned earlier, the prediction of $x(n)$ is derived from previously encoded predictions, not from the original data. These previously encoded predictions may contain quantization error ($q(n)$) and error due to slope overload. The magnitude of these errors subsequently influence the size of $d(n)$. Therefore, the value which is set for the standard deviation of the input determines the quantization levels but, in turn, the quantization levels then affect the size of the standard deviation of the input. Due to this interaction, the standard deviation of $d(n)$ obtained during prediction and quantization is different from that obtained during prediction without quantization. Consequently, setting the standard deviation of the Laplacian which is to be quantized equal to the standard deviation of the input which was determined without quantization is not a good match. To solve the problem of choosing the appropriate standard deviation, a compromise was reached. The prediction and encoding simulation was done iteratively, making successive approximations to the standard deviation, until the value used to set the quantization
levels was well matched to the standard deviation of the output (quantized data).
CHAPTER 5

RESULTS

The entire compression process, culminating in backprojection of the decoded projection matrix to reconstruct the image, was performed on three images: Clock, Test, and River. These three images were chosen because of their different visual characteristics. The Clock image is subsampled and contains a fairly wide range of grey levels as well as large expanse of white background; it includes areas of both fine detail and flat, uniform grey levels. The Test image is binary, being graphical in nature. The detail in this image consists mainly of straight lines in black superimposed on a white background. Finally, the River image shows an aerial view with a river flowing through the upper right corner, trees, fields and a road running parallel to the river in the lower left corner. This last image contains the most fine detail and most complicated pattern of grey levels.

Figures 21 through 23 show each of the original images and their backprojected versions without DPCM compression. Examination of these figures shows that considerable degradation occurs solely as a result of the tomographic process. The data in the original images all lie between the extremes of 0 and 255. It is possible for the backprojection to contain data outside this range due to imperfections.
Figure 21. Clock image and reconstruction.

a. Original image.

b. Reconstruction without DPCM coding.
Figure 22. Test image and reconstruction.

a. Original image.

b. Reconstruction without DPCM coding.
Figure 23. River image and reconstruction.

a. Original image.

b. Reconstruction without DPCM coding.
in the process. When displaying images on the monitor for photographing, the reconstructions were truncated to the 0-255 range. This means that negative values are represented as black (0), and values greater than 255 are displayed as white (255).

The backprojected images provide a reference for the DPCM compression results. Assuming that the DPCM compression was perfect and did not introduce any error, the best results which could be expected would be equivalent to these backprojections. Noticeable blurring, similar to low pass filtering, is evident in each reconstruction. The blurring is most detrimental in the case of the River image, which contains the most fine detail. The Test image, on the other hand, due to its graphical appearance, is least noticeably affected by the tomographic process.

**DPCM Coding and Backprojection**

The accuracy of the various prediction methods used during DPCM can be assessed in several ways. Objective measures include (1) the mean square error between the compressed projection matrix and the original projection matrix (MSE$_{proj}$), and (2) the mean square error between the backprojected (reconstructed) image following DPCM coding of its projection matrix vs. the backprojected image which did not experience coding (MSE$_{recon}$). A more subjective measure to consider is the overall quality of the image as judged by human observers.
MSE Measures

\( \text{MSE}_{\text{proj}} \) and \( \text{MSE}_{\text{recon}} \) values were obtained for the different reconstructions of the three images, each coded with all prediction methods, at each bit rate, both across and within projections. It was found that the two MSE values (proj. and recon.) calculated for each reconstruction (a total of 81 altogether) were highly correlated, as might be expected. The correlation between these two values was \( r = 0.95 \) for all reconstructions using prediction done within projections. For prediction across projections, the same correlation was \( r = 0.995 \). Because of this strong relationship and to avoid redundancy, only the \( \text{MSE}_{\text{recon}} \) values are presented in Tables 2 and 3. Table 2 is organized to present the \( \text{MSE}_{\text{recon}} \) values for all prediction methods and bit rates which used prediction within projections. Table 3 gives the same data for predictions across projections. The only exception is for the case of two-dimensional prediction which uses data for both within and across projections. \( \text{MSE}_{\text{recon}} \) results for two-dimensional prediction have been included in Table 2.

**Prediction Methods.** Examination of the \( \text{MSE}_{\text{recon}} \) values in these tables shows that, within projections, the best prediction methods usually were one-dimensional, second- or third-order and the worst methods were the one-dimensional, first-order and two-dimensional. Across projections, different results were obtained; the best prediction method was always via the slope method and the worst was always through one-dimensional, third-order predictors.
Table 2. Mean square error of reconstructions from prediction within projections.

<table>
<thead>
<tr>
<th>Image</th>
<th>Approximate Bit Rate*</th>
<th>1 DIM 1st Order</th>
<th>1 DIM 2nd Order</th>
<th>1 DIM 3rd Order</th>
<th>2-Dim Slope</th>
</tr>
</thead>
<tbody>
<tr>
<td>Clock</td>
<td>4</td>
<td>124</td>
<td>48</td>
<td>41</td>
<td>141</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>242</td>
<td>108</td>
<td>94</td>
<td>272</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>572</td>
<td>314</td>
<td>261</td>
<td>-</td>
</tr>
<tr>
<td>Test</td>
<td>4</td>
<td>135</td>
<td>72</td>
<td>73</td>
<td>113</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>348</td>
<td>218</td>
<td>229</td>
<td>340</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>1099</td>
<td>820</td>
<td>846</td>
<td>1060</td>
</tr>
<tr>
<td>River</td>
<td>4</td>
<td>35</td>
<td>21</td>
<td>22</td>
<td>30</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>88</td>
<td>55</td>
<td>52</td>
<td>77</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>258</td>
<td>169</td>
<td>160</td>
<td>223</td>
</tr>
</tbody>
</table>

* Actual bit rates can be found in Table 4.
Table 3. Mean square error of reconstructions from prediction across projections.

<table>
<thead>
<tr>
<th>Image</th>
<th>Approximate Bit Rate*</th>
<th>1 DIM 1st Order</th>
<th>1 DIM 2nd Order</th>
<th>1 DIM 3rd Order</th>
<th>Slope</th>
</tr>
</thead>
<tbody>
<tr>
<td>Clock</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
<td>56</td>
<td>60</td>
<td>70</td>
<td>53</td>
</tr>
<tr>
<td>3</td>
<td></td>
<td>219</td>
<td>242</td>
<td>280</td>
<td>159</td>
</tr>
<tr>
<td>2</td>
<td></td>
<td>788</td>
<td>882</td>
<td>964</td>
<td>516</td>
</tr>
<tr>
<td>Test</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
<td>126</td>
<td>149</td>
<td>202</td>
<td>74</td>
</tr>
<tr>
<td>3</td>
<td></td>
<td>489</td>
<td>588</td>
<td>784</td>
<td>269</td>
</tr>
<tr>
<td>2</td>
<td></td>
<td>1650</td>
<td>1922</td>
<td>2451</td>
<td>977</td>
</tr>
<tr>
<td>River</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
<td>40</td>
<td>46</td>
<td>55</td>
<td>24</td>
</tr>
<tr>
<td>3</td>
<td></td>
<td>157</td>
<td>175</td>
<td>207</td>
<td>82</td>
</tr>
<tr>
<td>2</td>
<td></td>
<td>506</td>
<td>552</td>
<td>660</td>
<td>329</td>
</tr>
</tbody>
</table>

* Exact bit rates can be found in Table 4.
Within vs. Across Projections. At an approximate bit rate of four, the MSE values of prediction within vs across projections are fairly equivalent. However, at lower bit rates, the MSE from predictions within projections are usually smaller. The exception to this is that the slope prediction method produces better MSE results across projections than within projections.

Bit Rates

The term "approximate bit rate" used in Tables 2 and 3 indicates the number of quantization levels which were available (the number of quantization levels equals $2^{\text{number of bits}}$) to quantize the difference data during a compression trial. The actual bit rate varies somewhat due to the different amounts of "overhead," or ancillary information, required by each type of prediction. For example, in the slope prediction method, it is necessary to store or transmit the slope value for each block of pixels; this adds a fractional amount to the bit rate of each pixel.

Each line of projection data, whether within or across projections, starts a new cycle through the DPCM coder. At a minimum, the first sample in each line is retained at its full bit rate and carried through the coder without any error. In second- and third-order prediction schemes, the first two and three samples, respectively, are retained at their full bit rates to make prediction of subsequent values more accurate. Storage or transmission of these 16-bit numbers adds to the final bit rate of the image. Other parameters which must be saved and add to the bit rate are the
prediction coefficients \((h_1, h_2, \text{ etc.})\), the standard deviation of the quantization Laplacian and the mean value of the original projection matrix. The actual bit rates which correspond to the approximate ones are listed in Table 4.

The projection matrix consists of 16-bit data samples. Compression from 16 bits to approximately 3.2 bits, for example, represents a compression ratio of 5:1. However, since the original data is in the form of an image using 8-bit data before being transformed into projection data, a more realistic compression ratio is obtained by comparison of the compressed bit rates to the 8-bit original information. In this case, compression from 8 bits to 3.2 bits is a compression ratio of 2.5:1.

Reconstructed Images

The real value of any image compression or processing scheme lies in the degree to which the quality of the final image allows viewers to obtain information from it, and it is judged to be aesthetically pleasing. Figures 24 through 37 are compressed and reconstructed images of the Clock and Test results. For Figures 24 through 35, each page contains four reconstructions. For all four images on a page, quantization is done at the same bit rate, and prediction is performed in the same direction (within or across projections). The critical difference in the generation of the four images is the method by which prediction was performed. Position A (upper left) is always one-dimensional, first-order prediction; Position B (upper right) is one-dimensional, second-order prediction;
Table 4. Actual bit rates. — Exact bit rates are given for each method of prediction and each direction of prediction.

<table>
<thead>
<tr>
<th>Prediction Method</th>
<th>One-Dim</th>
<th>One-Dim</th>
<th>One-Dim</th>
<th>Slope</th>
<th>Two Dim</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1st Order</td>
<td>2nd Order</td>
<td>3rd Order</td>
<td>Dim</td>
<td></td>
</tr>
<tr>
<td>Approximate Bit Rate</td>
<td>W/in Acr</td>
<td>W/in Acr</td>
<td>W/in Acr</td>
<td>W/in Acr</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>4.08 4.10</td>
<td>4.17 4.22</td>
<td>4.26 4.34</td>
<td>4.18 4.21</td>
<td>4.20</td>
</tr>
<tr>
<td>3</td>
<td>3.09 3.12</td>
<td>3.19 3.25</td>
<td>3.29 3.38</td>
<td>3.20 3.23</td>
<td>3.22</td>
</tr>
<tr>
<td>2</td>
<td>2.10 2.13</td>
<td>2.21 2.27</td>
<td>2.32 2.41</td>
<td>2.21 2.24</td>
<td>2.24</td>
</tr>
</tbody>
</table>
Position C (lower left) is one-dimensional, third-order prediction; and Position D (lower right) is prediction using the slope method. Half of these figures were compressed by coding within projections (24 through 29) and the other half were by coding across projections (30 through 35).

The next two figures (36 through 37) show reconstructions whose projections were coded using two-dimensional prediction. In these figures, the various images on one page represent quantization at different bit rates.

Examination of the reconstructions suggests that the image quality after DPCM compression of the projection matrix to four bits is generally as good as the backprojections without DPCM. In some cases, the four-bit reconstructions appear to be slightly sharper than reconstructions without coding; this effect may result from the fact that the quantization step sizes are coarse enough to enhance the edges, without being so coarse that they interject distracting noise. The results of coding with three bits seem to lie at the threshold of noticeable degradation, compared to backprojections without coding. Comparison of the different prediction methods at this bit rate gives, perhaps, the most information about which prediction techniques work best. Reconstructions which follow from coding of the projection matrix at two bits generally are of poor quality, especially for coding across projections.

Errors. In flat areas, with two-bit coding, the noise appears to be randomly distributed (see the background in Figure 26). In some
Figure 24. Clock: prediction within projections, coding at approximately 4 bits/pixel.

a. 1-dimensional, first order, 4.08 bits/pixel.
b. 1-dimensional, second order, 4.17 bits/pixel.
c. 1-dimensional, third order, 4.26 bits/pixel.
d. Slope prediction, 4.18 bits/pixel.
Figure 25. Clock: prediction within projections, coding at approximately 3 bits/pixel.

a. 1-dimensional, first order, 3.09 bits/pixel.
b. 1-dimensional, second order, 3.19 bits/pixel.
c. 1-dimensional, third order, 3.29 bits/pixel.
d. Slope prediction, 3.20 bits/pixel.
Figure 26. Clock: prediction within projections, coding at approximately 2 bits/pixel.

a. 1-dimensional, first order, 2.10 bits/pixel.
b. 1-dimensional, second order, 2.21 bits/pixel.
c. 1-dimensional, third order, 2.32 bits/pixel.
d. Slope prediction, 2.21 bits/pixel.
Figure 27. Test: prediction within projections, coding at approximately 4 bits/pixel.

a. 1-dimensional, first order, 4.08 bits/pixel.

b. 1-dimensional, second order, 4.17 bits/pixel.

c. 1-dimensional, third order, 4.26 bits/pixel.

d. Slope prediction, 4.18 bits/pixel.
Figure 28. Test: prediction within projections, coding at approximately 3 bits/pixel.

a. 1-dimensional, first order, 3.09 bits/pixel.

b. 1-dimensional, second order, 3.19 bits/pixel.

c. 1-dimensional, third order, 3.29 bits/pixel.

d. Slope prediction, 3.20 bits/pixel.
Figure 29. Test: prediction within projections, coding at approximately 2 bits/pixel.

a. 1-dimensional, first order, 2.10 bits/pixel.
b. 1-dimensional, second order, 2.21 bits/pixel.
c. 1-dimensional, third order, 2.32 bits/pixel.
d. Slope prediction, 2.21 bits/pixel.
Figure 30. Clock: prediction across projections, coding at approximately 4 bits/pixel.

a. 1-dimensional, first order, 4.10 bits/pixel.

b. 1-dimensional, second order, 4.22 bits/pixel.

c. 1-dimensional, third order, 4.34 bits/pixel.

d. Slope prediction, 4.21 bits/pixel.
Figure 31. Clock: prediction across projections, coding at approximately 3 bits/pixel.

a. 1-dimensional, first order, 3.12 bits/pixel.

b. 1-dimensional, second order, 3.25 bits/pixel.

c. 1-dimensional, third order, 3.38 bits/pixel.

d. Slope prediction, 3.23 bits/pixel.
Figure 32. Clock: prediction on across projections, coding at approximately 2 bits/pixel.

a. 1-dimensional, first order, 2.13 bits/pixel.

b. 1-dimensional, second order, 2.27 bits/pixel.

c. 1-dimensional, third order, 2.41 bits/pixel.

d. Slope prediction, 2.24 bits/pixel.
Figure 33. Test: prediction across projections, coding at approximately 4 bits/pixel.

a. 1-dimensional, first order, 4.10 bits/pixel.

b. 1-dimensional, second order, 4.22 bits/pixel.

c. 1-dimensional, third order, 4.34 bits/pixel.

d. Slope prediction, 4.21 bits/pixel.
Figure 34. Test: prediction across projections, coding at approximately 3 bits/pixel.

a. 1-dimensional, first order, 3.12 bits/pixel.
b. 1-dimensional, second order, 3.25 bits/pixel.
c. 1-dimensional, third order, 3.38 bits/pixel.
d. Slope prediction, 3.23 bits/pixel.
Figure 35. Test: prediction across projections, coding at approximately
2 bits/pixel.

a. 1-dimensional, first order, 2.13 bits/pixel.
b. 1-dimensional, second order, 2.27 bits/pixel.
c. 1-dimensional, third order, 2.41 bits/pixel.
d. Slope prediction, 2.24 bits/pixel.
Figure 36. Clock: two-dimensional prediction.

a. Coding at 4.20 bits/pixel.

b. Coding at 3.22 bits/pixel.
Figure 37. Test: two-dimensional prediction.

a. Coding at 4.20 bits/pixel.
b. Coding at 3.22 bits/pixel.
c. Coding at 2.24 bits/pixel.
reconstructions, the placement of errors is affected by the direction of prediction. This becomes apparent by comparing Figures 29 and 35. For prediction within projections, errors tend to be grouped into horizontal lines (especially Figure 29b), while for prediction across projections the errors group in arc-like patterns (for example, note the errors emanating from the black dots in the upper right section of the images in Figure 35). These error arrangements are reasonable based on the tomographic process. Each sequence of data which is encoded across projections corresponds to data from a single detector; any errors which occur in this string should become evident as a curved line as the detector swings circularly through the image in backprojection (also see Figures 32a, b and c). Errors in data being predicted and coded within projections are more likely to be evident as a straight line; this follows from the fact that each line of data within projections corresponds to a projection made at a specific angle (straight line) through the image.

**Within vs. Across Projections.** Subjective evaluation of the images reconstructed from across projections agrees fairly well with the results of MSE analysis. At four bits, the reconstructions all look fairly equal. At lower bit rates, the results from prediction within projections is superior. Two-dimensional prediction produces results whose quality lies somewhere between the other two types.

An interesting development is that for prediction within projections, the quality seems to improve as the order of prediction rises (although third-order is not dramatically different from second-order),
while for prediction across projections the opposite occurs. The quality seems to become worse as the order of prediction increases for prediction across predictions. Figure 38 shows this trend graphically.

The two-dimensional predictors do not seem to have any distinct advantage over the one-dimensional variety. To calculate the coefficients for this two-dimensional method, it was necessary to know the autocovariance for data sequences running along a diagonal through the matrix \( \rho_{BE} \) (in Chapter 4) in addition to the autocovariance for data contained in horizontal and vertical strings. Problems with inconsistent record length, coupled with nonstationarity, made calculation of \( \rho_{BE} \) less accurate than the other normalized covariance measures \( \rho_{AB} \) or \( \rho_{AE} \). Considering the inconsistent record length, for example, the autocovariance of the horizontal and vertical lines of the projection matrix was based on calculations involving fixed length-data strings of 128 and 100 samples, respectively. However, data compiled by collecting pixels arranged diagonally through the matrix begins with strings whose length is one data point. Each successive data string increases in length by one pixel until a maximum of 141 pixels is reached. Later, data sequences decrease to one pixel again. Since the projection matrix contains a definite, somewhat fixed, sinusoidal pattern, the diagonal data sequences are not stationary, and the unequal number of samples in the data strings makes it unlikely that a realistic value for \( \rho_{BE} \) is achieved. This inability to obtain an accurate coefficient may have reduced the quality of results obtained using this method.
Figure 38. Reconstruction error vs. order of prediction.
Analysis of the Difference Distribution

A post-hoc analysis of the difference or error distribution was done to determine whether the assumption of a Laplacian distribution was justified. Figure 39 shows the probability distribution of $d(n)$ for different prediction methods at a four-bit coding rate on the Clock projection matrix. The minimum MSE predictors produce distributions which contain obvious deviations from an exponential, while the slope predictors appear to produce a more exponential, symmetrical distribution. A Laplacian distribution is of the form:

$$p(x) = a/2\exp (-a|x|)$$

Agreement between the obtained difference distributions and a true Laplacian was calculated using Pearson's chi square goodness of fit statistic (Hays, 1963). The chi square value obtained is proportional to the amount of discrepancy between the actual and expected frequencies in each cell of the probability distribution.

The chi square value for the difference distribution produced by the slope predictors, compared to its best fit Laplacian ($a = 0.0036$), was:

$$X^2 = 49.4$$

with 28 degrees of freedom. Referring to tabled chi square values, it can be said that the obtained distribution is roughly Laplacian in nature but not a close fit ($p < 0.005$). Analysis of the distributions from the other prediction methods was performed, and they were found to
Figure 39. Error distribution for four prediction methods. -- Using the clock projection matrix, the probability distribution of the prediction difference values, d, was calculated for prediction within projections using (A) 1-dimensional, first order, (B) 1-dimensional, second order, (C) 2-dimensional, second order, and (D) Slope.
be very non-Laplacian. For example, using the one-dimensional, second-order predictor, the best fit Laplacian, with an exponential decay of $a = 0.0025$, compared to a theoretical Laplacian, produced

$$X^2 = 1932$$

with 38 degrees of freedom. The tabled values of chi square (for 38 degrees of freedom) go to a maximum of 73.4; consequently, the obtained chi square is more than an order of magnitude greater than this. The error distribution obtained by one-dimensional second-order predictors, therefore, is not a very good match to a Laplacian. The distribution in this case is wider and less symmetrical than that obtained using slope prediction.
CHAPTER 6

CONCLUSIONS

The combination of tomography with DPCM has produced modest amounts of data compression. Compression ratios of approximately 2.5:1 were achieved with fairly good reconstruction results. The reconstructions whose projections had been compressed to four bits per pixel were generally as good as images whose projections had not been compressed. Compression to a bit rate of approximately three bits per pixel produced fairly good-quality images, but often some degree of degradation was evident. Rates of approximately 2 bits per pixel produced reconstructions which were very noisy and unacceptable for most uses.

The projection matrix was found to contain substantial redundancy. Several prediction methods were devised in an attempt to tap this redundancy. The coding schemes were designed to try to model the data and make predictions which would decorrelate the projection matrix. Various prediction schemes achieved some success in different data environments. For prediction within projections, one-dimensional, second- and third-order predictors were best. The slope prediction method worked best for predictions across projections.

In general, the redundancy across projections was greater than redundancy within projections. This greater amount of redundancy, however, did not lead to more accurate predictions and reconstructions.
Perhaps the prediction methods under investigation were less well matched to the character of the data obtained from across projections. A more accurate way to model and predict the data would lead to smaller values of $d(n)$ and better compression rates. Overall, reconstructions stemming from prediction within projections were superior to reconstructions from prediction across projections.

This technique of image data compression would not seem to be the method of choice for high-quality originals from which equally high-quality reconstructions were required. If sharpness of the reconstructed image is not essential, then using this technique, compression ratios of approximately 2:1 or 2.5:1 may be achieved.

The degradation of the backprojected image, even without DPCM coding, limits the degree of compression which can be achieved and the general usefulness of this method. A better simulation of backprojection may produce better reconstructions, and, consequently, a lower bit rate (more error) could be tolerated. A fundamental obstacle involved in the current simulation is that errors in the decoded projection matrix are amplified by the filtering done prior to backprojection. This high pass filter contributes to the noisy appearance of the reconstructions. Since this filtering process is an integral part of tomographic reconstruction, it appears that while further research may improve the results somewhat, the amount of improvement possible may not be sufficient to warrant additional investigation.
REFERENCES


Max, J. Quantizing for minimum distortion. IRE Transactions on Information Theory, 1960, IT-6, 7-12.


