LBL-31561

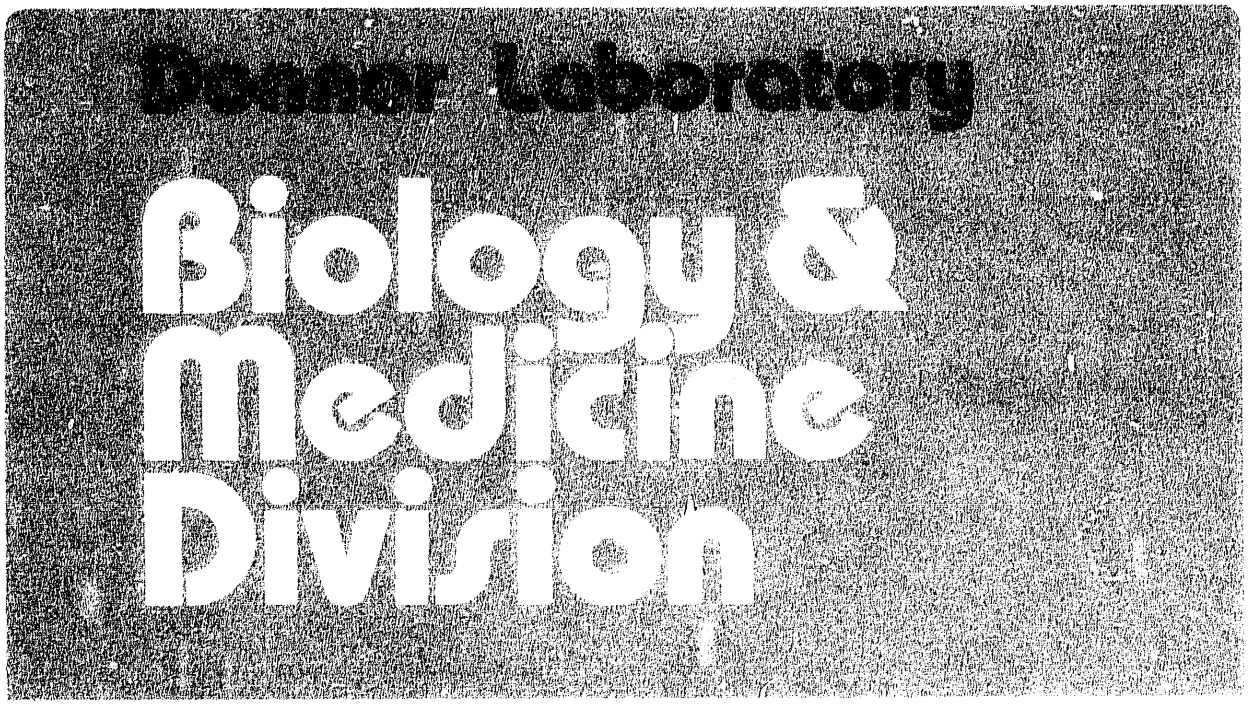
Lawrence Berkeley Laboratory
UNIVERSITY OF CALIFORNIA

Spatially Variant Tomographic Imaging: Estimation, Identification, and Optimization

J.R. Baker
(Ph.D. Thesis)

MAR 2 4 1992

November 1991



Prepared for the U.S. Department of Energy under Contract Number DE-AC03-76SF00098

DISTRIBUTION OF THIS DOCUMENT IS UNLIMITED

DISCLAIMER

This document was prepared as an account of work sponsored by the United States Government. Neither the United States Government nor any agency thereof, nor The Regents of the University of California, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial products process, or service by its trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof, or The Regents of the University of California. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof or The Regents of the University of California and shall not be used for advertising or product endorsement purposes.

Lawrence Berkeley Laboratory is an equal opportunity employer.

LBL--31561

DE92 010262

Spatially Variant Tomographic Imaging: Estimation, Identification, and Optimization

John Robert Baker

jrbaker@lbl.gov

(510) 486-5435

Department of Electrical Engineering and Computer Sciences

University of California, Berkeley

and

Research Medicine and Radiation Biophysics Division

Lawrence Berkeley Laboratory[1]

University of California

Berkeley, CA 94720

[1]This work was supported in part by the Director, Office of Energy Research, Office of Health and Environmental Research, Medical Applications and Biophysical Research Division of the U.S. Department of Energy under contract No DE-AC03-76SF00098 and in part by NIH grant HL07367.

MASTER

DISTRIBUTION OF THIS DOCUMENT IS UNLIMITED

Spatially Variant Tomographic Imaging: Estimation, Identification, and Optimization

Copyright © 1991

by

John Robert Baker

The U.S. Department of Energy has the right to use this thesis
for any purpose whatsoever including the right to reproduce
all or any part thereof

Spatially Variant Tomographic Imaging: Estimation, Identification, and Optimization

By

John Robert Baker

Abstract

This thesis is an investigation of methods for processing multidimensional signals acquired using modern tomography systems that have an anisotropic or spatially variant response function. The main result of this research is the discovery of a new method to obtain better estimators of an unknown spatial intensity distribution by incorporating detailed knowledge about the tomograph system response function and statistical properties of the acquired signal into a mathematical model. In this model, data acquisition is represented by a linear transformation, $F_{\theta k}$, of functions, b, on a continuous domain object space to functions, $p_{\theta k}$, on a discrete domain observation space as described by

$$p_{\theta k} \;=\; F_{\theta k} \cdot b \;=\; \int_{\mathbf{R}} dy \int_{\mathbf{R}} dx \, f_{\theta k}(x,y)\; b(x,y).$$

This data acquisition model and the singular value decomposition,

$$F_{\theta k} \;=\; \boldsymbol{U}_{\theta k} S \cdot V^T,$$

of this model into the left singular vectors, $\boldsymbol{U}_{\theta k}$, right singular functions, V, and the singular value operator, S, form a common, unifying structure for the three areas of

this work: identification, estimation, and optimization. For system identification, the model is easily adapted to represent tomography systems that are spatially invariant, spatially variant with symmetry, or spatially variant without symmetry. Least squares, normal maximum likelihood, and Poisson maximum likelihood estimators and the corresponding covariance have been formulated to compensate for all three types of spatial responses. While bias is reduced by these estimators, undesirable statistical and systematic fluctuations can result due to pixelization effects. To reduce these fluctuations new estimators such as the filtered singular value least squares estimator

$$\hat{b} \;=\; V \cdot S^T \boldsymbol{D} \left(S \cdot S^T \right)^+ \boldsymbol{U}^T \underline{\boldsymbol{p}}$$

with diagonal filter matrix, $\boldsymbol{D}$, were developed. These new estimators use an orthonormal pixel basis decomposition of the unknown spatial distribution to eliminate systematic error and minimize statistical errors that occur using square or polar pixel bases. Characterizing the effects which sampling density has on reconstructed image resolution and noise using singular value analysis has led to a method to optimize sampling strategies to obtain better estimates of the unknown spatial distribution given bounds for noise and resolution. Symmetries in the data acquisition process lead to very efficient implementations of the new inverse problem algorithms.

Acknowledgments

I wish to give my warmest thanks to Prof. Thomas F. Budinger for his advice, encouragement and funding for this work. Special thanks to Dr. Ronald H. Huesman for his continuous guidance and supports. Thanks to Dr. Pamela C. Coxson, Dr. Stephen E. Derenzo, Dr. Marc L. Kessler, Dr. Mark S. Roos, and Dr. Sam T. S. Wong for their helpful discussions. I am grateful for the help I have received from the graduate students and staff of the Research Medicine group.

Most of all, I am forever indebted to Dr. Janet M. Griffiths. Through her love and kindness this precious lady has helped me to have the strength and commitment to complete this work. While the path may not have always been clear and easy, we have been able to find our way together.

Contents

List of Figures

List of Tables

Symbol Table

Symbol[1]	Definition and page number
$p_{\theta k}$	Projection 7
θ	Angle index 7
k	Lateral index 7
b	Object distribution 7
F	Projection formation operator 7
x	Cartesian spatial coordinate 7
y	Cartesian spatial coordinate 7

[1] Lower case bold symbols denote vectors, lower case script symbols denote functions or scalars, upper case bold symbols denote matrices, and upper case script symbols denote operators.

Symbol	Definition and page number
Θ	Number of angles 7
K	Number of lateral bins 7
$\boldsymbol{p}$	Projection vector 8
L_2	Lebesgue square norm 8
$\boldsymbol{U}$	Left singular vectors 8
S	Singular values 8
V	Right singular functions 8
$\boldsymbol{A}$	Projection normal matrix 10
F_0	Ideal parallel beam projection 21
$\boldsymbol{G}$	Projection blurring matrix 21
H	Object blurring operator 21

Symbol	Definition and page number
$\Delta\theta$	Angle index difference 22
B	Generalized pixel basis 26
$\boldsymbol{c}$	Generalized pixel amplitude 26
$\tilde{\boldsymbol{p}}$	Reprojection vector 35
$\hat{\boldsymbol{c}}$	Pixel amplitude estimate 40
E	Expectation 40
$\underline{\boldsymbol{p}}$	Observed projection vector 40
$\boldsymbol{\Sigma}_{\hat{\boldsymbol{c}}}$	Pixel covariance matrix 42
$\boldsymbol{\Sigma}_{\boldsymbol{p}}$	Projection covariance matrix 42
$\hat{b}$	Object estimate 43
$\boldsymbol{D}$	Diagonal weighting matrix 44
J	Singular value truncation index 45

Symbol	Definition and page number
L	Likelihood 52
l	Log-likelihood 53
$\boldsymbol{Q}$	Expected reprojection covariance matrix 57
δ	Kronecker delta 57
$\bar{p}$	Homoscedastic projection variance 75
$\mathcal{F}$	Fourier matrix 87
$\boldsymbol{I}$	Identity matrix 87

Chapter 1

Introduction

1.1 Statement of thesis

My thesis is that by incorporating detailed knowledge about the tomograph system response function and statistical properties of the acquired signal into a mathematical model, better estimators of an unknown spatial intensity distribution will result. These estimators, which are based on a generalized orthonormal pixel basis decomposition that is fundamental to the tomograph, will have reduced local bias, will minimize systematic error due to pixelization, and will simplify error propagation when used as input to quantitative models of physiologically important processes such as the *in vivo* density of neuroreceptors.

1.2 Motivation

Many emission tomography systems have an anisotropic or spatially variant response function; i.e., resolution varies throughout the field of view. Traditionally, estimation (reconstruction) algorithms have not included knowledge about the processes that cause spatially variant resolution. When images reconstructed with a spatially invariant algorithm are used as input to a physiological model, significant bias can be introduced into the parameters of the physiological model. Consider the images shown in figure 1.2 which were reconstructed from data taken with a positron emission tomograph (PET). The upper image was reconstructed using a backprojection of filtered projections [1] algorithm that does not account for spatial variance and the lower image was reconstructed using a new separable algorithm [2] that includes spatial variance in the system response function. Qualitatively the lower image is a better representation of the 37-point phantom, shown in figure 1.1, which it depicts and will introduce less bias in the physiological model.

1.3 Previous work

For continuous space noiseless measurements of line integrals from parallel projections, an inverse Radon transform [3] [4] [5] can be used to perform reconstruction. An inverse of the attenuated or exponential Radon transform can be used for single photon emission computed tomography (SPECT) with known attenuation and continuous space parallel line integral projections [6] [7]. The reconstruction approach proposed by O [8] includes a continuous space model for detector response in addition

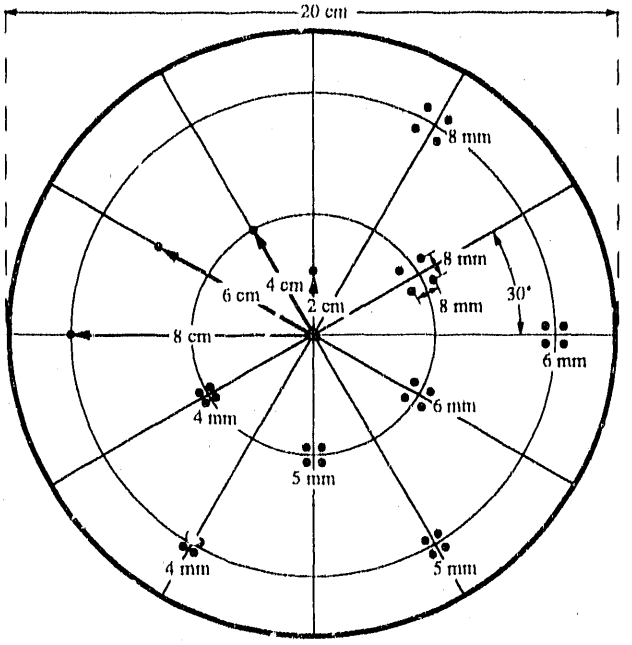


Figure 1.1: Schematic diagram of a phantom with 37 line sources in a 20 cm diameter cylinder of lucite.

to known attenuation.

Most physically realizable tomography systems however acquire discrete projection data. Approximations using a sampled Radon transform can be applied [9] [10] [11]. The filtered backprojection and backprojection of filtered projections algorithms are the most commonly used methods for reconstructing discrete parallel line integral projection data [12] [13]. Efficient implementations of these algorithms [14] using fast Fourier transforms [15] has led to their widespread use. Marr [16] developed a reconstruction method using orthogonal polynomials that results in what he calls pixels that are *natural* to the system. In his case, the system is modeled by line integrals on a circular domain.

For systems that do not acquire line integral projections, more complex techniques

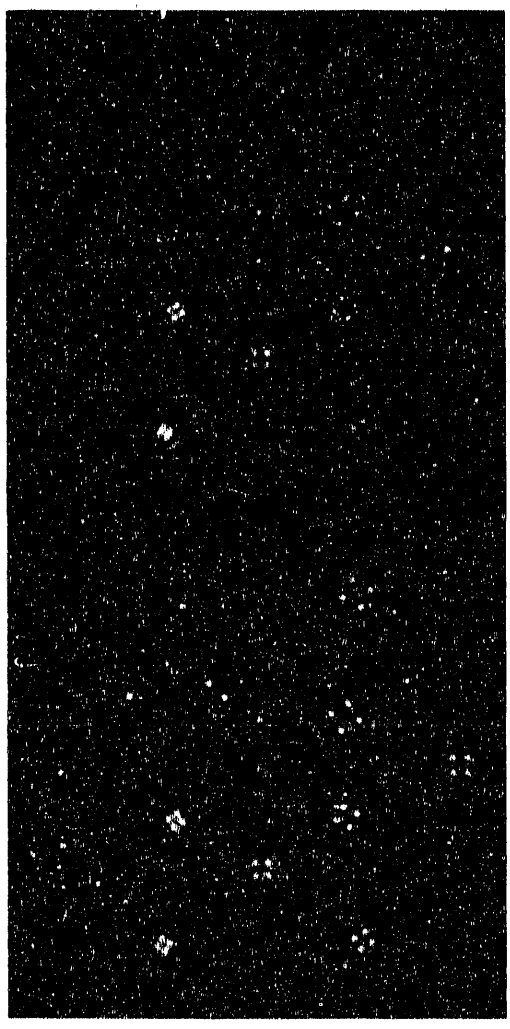


Figure 1.2: Images reconstructed from data taken with a positron emission tomograph with spatially variant system response function. Upper) backprojection of filtered projections reconstruction. Lower) new separable spatially variant reconstruction.

have been developed which include direct and optimization techniques [17]. The direct reconstruction methods are usually implemented as linear matrix algorithms; e.g., the pseudo-inverse [18], and the statistical properties of the resulting images are straightforward. Conversely, the optimization or series expansion methods are generally implemented as iterative algorithms; e.g., iterative least squares [19] algebraic reconstruction techniques [20], and maximum likelihood estimation [21] [22]. The statistical properties of images reconstructed using optimization techniques are often difficult to characterize [2]. Both techniques *ab initio* discretize the object space into square or sometimes polar pixels [23] [24]. Substantial systematic errors or artifacts in the reconstructed images can result from pixelization [25] [26].

Buonocore, *et al*, [27] also worked on a natural pixel decomposition that uses each projection ray as a pixel. Buonocore's definition for a pixel eliminates systematic errors due to pixelization; however, it usually leads to pixels that are not spatially orthogonal. This can be an undesirable property especially for interpreting covariance between estimates of the intensity at two different pixels that may spatially overlap.

Several researchers have used regularization techniques to de-emphasize reconstruction artifacts due to pixelization and ill-posedness of the tomographic inverse problem. The method of sieves by Snyder and Miller [28] [29] is one example. The sieve or blurring function modifies the expectation maximization (EM) [30] maximum likelihood approach of Shepp and Vardi [21] by constraining the estimate to a set of images where the likelihood is bounded. The choice of blurring function is object dependent and is found by trial and error. Evaluating the statistical properties of

their estimators is extremely difficult even in an asymptotic sense. The computation acceleration scheme of Lewitt and Muehllehner [31] can also be classified as a regularization technique.

Incorporation of prior information about the spatial properties of the distribution to be reconstructed has received attention by several researchers [32]. In their maximum *a posteriori* (MAP) algorithm, Levitan and Herman used a normal (Gaussian) prior [33] to reduce "irregular high amplitude patterns." Their MAP EM algorithm is a special form of penalized EM algorithm. Leahy *et al* have used Markov random fields as a prior in their MAP algorithm which includes depth dependent resolution in SPECT [34]. The Markov property ensures that values at nearby pixels do not differ too much from each other. This tends to smooth the reconstructed image and may add significant bias.

An extensive bibliography on tomographic reconstruction was presented by Rangayyan *et al* [35].

Chapter 2

Theoretical Framework

2.1 Data acquisition

Projection formation can be described by the discrete-continuous model [36], [37], [32]

$$p_{\theta k} = F_{\theta k} \cdot b \tag{2.1}$$

$$= \int_{\mathbf{R}} dy \int_{\mathbf{R}} dx \, f_{\theta k}(x,y) \, b(x,y) \tag{2.2}$$

where $p_{\theta k}$ is the measured projection at angle index θ and bin position k. $F_{\theta k}$ is a second order tensor functional operating on the two dimensional object distribution b. The operation of equation 2.1 represents the integration of the product of the impulse response $f_{\theta k}(x,y)$ and the object distribution $b(x,y)$ over the imaging field as depicted in figure 2.1. The symbol $\cdot$ indicates integration over the imaging field. The impulse response function is the spatial response of a projection at angle θ and

bin k to a point source moved to every position within the sampling domain. There are Θ different angles and K projection bins at each angle.

To simplify notation[1], the projection formation equation is written in vector form

$$\boldsymbol{p} = F \cdot b \tag{2.3}$$

by combining the θ and k indices into one index. Specifically[38],

$$b \in L_2\left[\mathbf{R}^2\right], \tag{2.4}$$

$$\boldsymbol{p} \in \mathbf{R}^{\Theta K}, \text{ and} \tag{2.5}$$

$$F : L_2\left[\mathbf{R}^2\right] \mapsto \mathbf{R}^{\Theta K}. \tag{2.6}$$

Because the model is based on the fact that the detection process is defined on a discrete domain and the original distribution is defined on a continuous domain, the model is easily adapted to include a variety of physical effects found in many imaging modalities. For positron emission tomography (PET), $F_{\theta k}$ can include radioactive decay, positron range, non-collinearity of photons, sampling geometry, attenuation, inter-crystal scatter, crystal penetration, and detection efficiency [2].

The singular value decomposition of the tomograph system response function is

$$F = \boldsymbol{U}S \cdot V^T. \tag{2.7}$$

$\boldsymbol{U}$ is an orthogonal matrix containing the left singular vectors of F and is defined by the eigenvalue decomposition of the projection normal matrix,

[1] Lower case bold symbols denote vectors, lower case script symbols denote functions or scalars, upper case bold symbols denote matrices, and upper case script symbols denote operators.

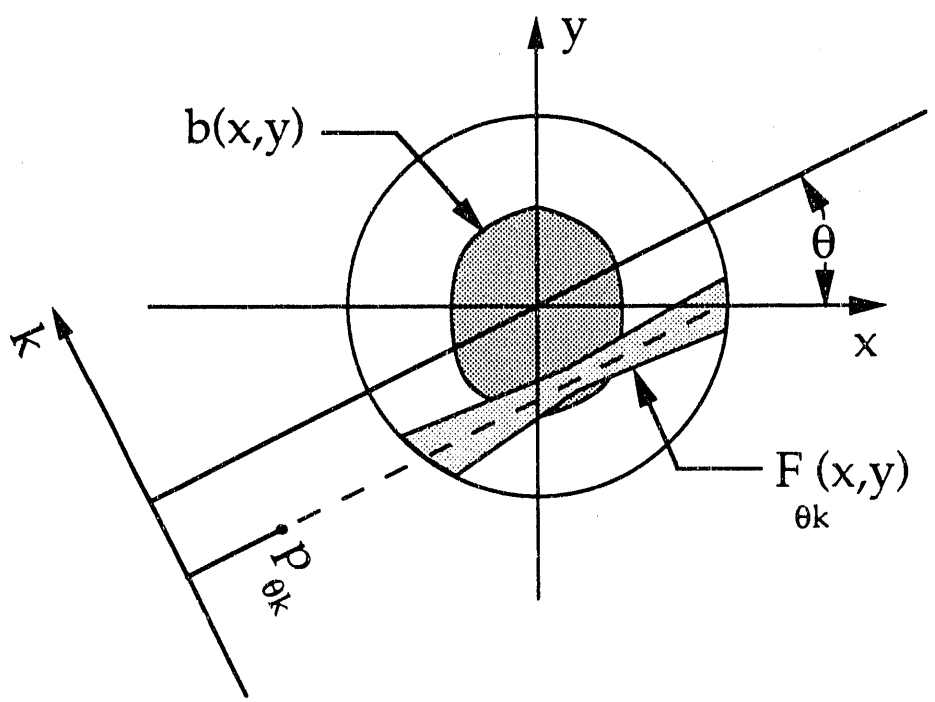


Figure 2.1: Schematic of projection formation.

$$\boldsymbol{A} \equiv F \cdot F^T \tag{2.8}$$

$$= \boldsymbol{U}S \cdot S^T\boldsymbol{U}^T. \tag{2.9}$$

An element $A_{\theta' k' \theta k}$ is the projection at angle θ' and bin k' of the backprojection at angle θ and bin k of a unit projection value, $p_{\theta k} = 1$. A schematic of this operation is shown in figure 2.2. As shown in section A.1, the projection normal matrix is symmetric and positive semidefinite. V are the right singular functions of F and are defined by the relationship

$$F^T F = V \cdot S^T S \cdot V^T. \tag{2.10}$$

A proof of equation 2.7 is given in appendix A.1.

For the real Hilbert space operator F, the corresponding adjoint operator denoted by the symbol F^T is defined by equating the inner product in the range space of F with the inner product in the domain space of F as follows

$$(\boldsymbol{p},\, F \cdot b) = \left(F^T \boldsymbol{p}, b\right) \tag{2.11}$$

which is the sum in the projection space and the integral in the object space

$$\sum_{\theta=0}^{\Theta-1} \sum_{k=0}^{K-1} p_{\theta k} \int_{\mathbf{R}} dy \int_{\mathbf{R}} dx\, f_{\theta k}(x,y)\, b(x,y) = \int_{\mathbf{R}} dy \int_{\mathbf{R}} dx \sum_{\theta=0}^{\Theta-1} \sum_{k=0}^{K-1} F_{\theta k}^T(x,y)\, p_{\theta k}\, b(x,y). \tag{2.12}$$

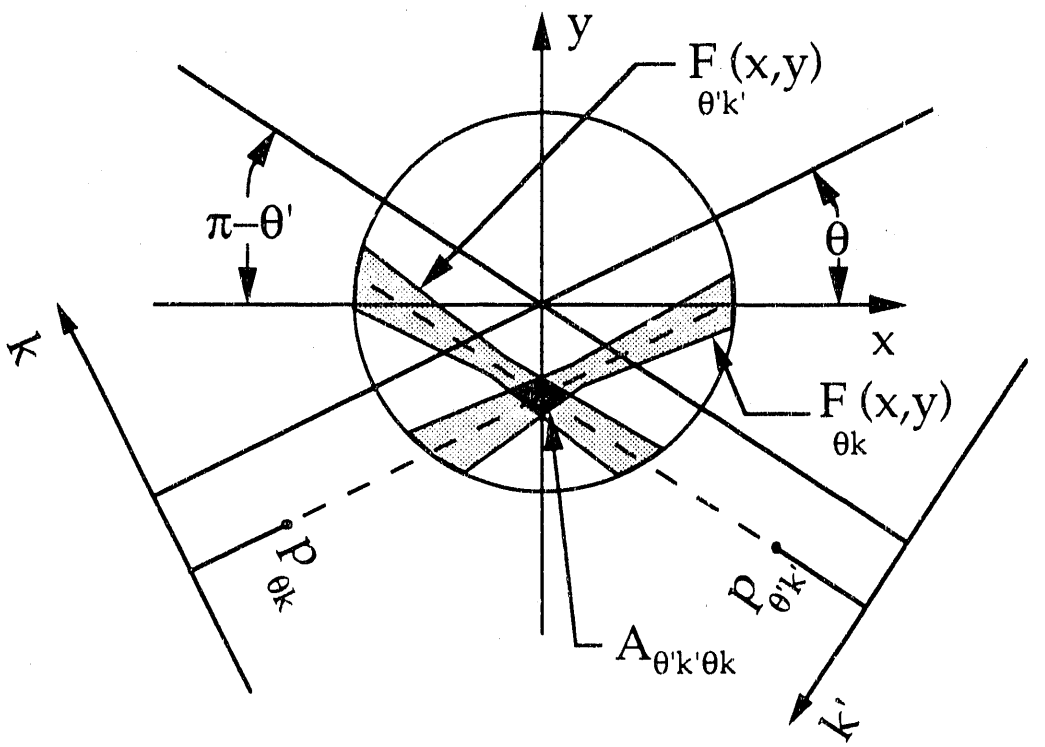


Figure 2.2: Schematic of projection normal matrix.

After exchanging the order of summation and integration,

$$\sum_{\theta=0}^{\Theta-1}\sum_{k=0}^{K-1} p_{\theta k}\int_{\mathbf{R}} dy \int_{\mathbf{R}} dx\, f_{\theta k}(x,y)\, b(x,y) \;=\; \sum_{\theta=0}^{\Theta-1}\sum_{k=0}^{K-1} p_{\theta k}\int_{\mathbf{R}} dy \int_{\mathbf{R}} dx\, F_{\theta k}^{T}(x,y)\, b(x,y). \tag{2.13}$$

Equation 2.13 must hold for an arbitrary object, b, and an arbitrary projection, $\boldsymbol{p}$, except for a space of measure zero. Therefore, the adjoint of F is almost everywhere

$$F_{\theta k}^{T}(x,y) \;=\; f_{\theta k}(x,y) \tag{2.14}$$

which is just the impulse response function. F can be interpreted as a column vector of integral functions and F^T as a rows vector of functions; hence, the use of the symbol ^{T}. The adjoint operation for a real matrix is the transpose of the matrix.

The functional S operates similarly to F in that it maps continuous domain functions to discrete domain samples. Thus, V performs an infinite dimensional rotation on the continuous domain object space, S selects and scales a finite number of the rotated functions, and $\boldsymbol{U}$ performs a finite dimensional rotation into the discrete domain projection measurement space as shown in equations 2.15-2.17.

$$\boldsymbol{U} \;:\; \mathbf{R}^{\Theta K} \mapsto \mathbf{R}^{\Theta K}, \tag{2.15}$$

$$S \;:\; L_2\left[\mathbf{R}^2\right] \mapsto \mathbf{R}^{\Theta K}, \text{ and} \tag{2.16}$$

$$V \;:\; L_2\left[\mathbf{R}^2\right] \mapsto L_2\left[\mathbf{R}^2\right]. \tag{2.17}$$

A geometric interpretation is given in figure 2.3. In the next chapter, this singular value decomposition will be used to define pixels and formulate reconstruction algorithms. To my knowledge, this work is the first time that a singular value decomposition has been formulated for a system that maps continuous domain functions to discrete domain samples.

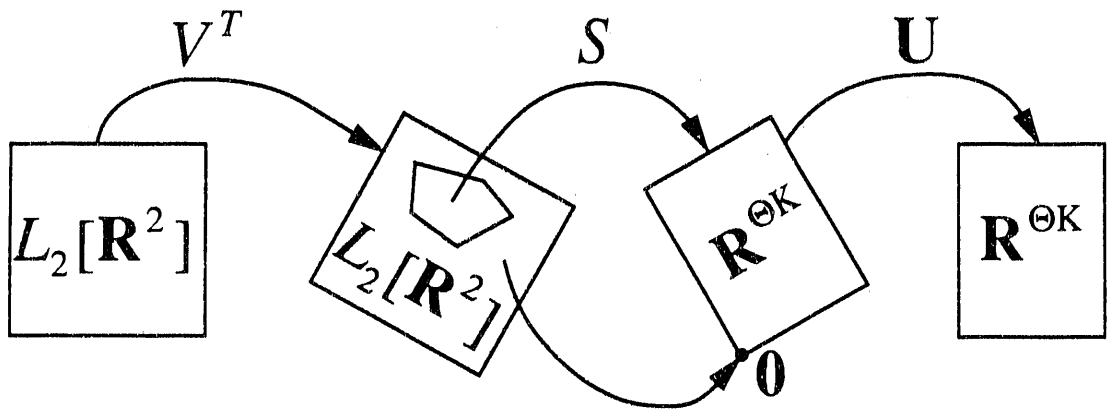


Figure 2.3: Singular value decomposition of the projection operator. The operator V^T performs an infinite dimensional rotation on the continuous domain object space, S selects selects and scales a finite number of the rotated functions, and $\boldsymbol{U}$ performs a finite dimensional rotation into the discrete domain projection measurement space.

As a simple example, consider the spatial sampling system of figure 2.4 that has three projection angles and two projection bins at each angle. The angles are equally spaced between zero and π radians. The impulse response functions are defined by

$$f_{\theta k}(x,y) = \begin{cases} 1 & \text{if } k-1 \leq -x\sin\left(\theta\frac{\pi}{3}\right) + y\cos\left(\theta\frac{\pi}{3}\right) < k \text{ and } x^2+y^2 \leq 1, \\ 0 & \text{otherwise} \end{cases} \tag{2.18}$$

and are shown in figure 2.4. The projection normal matrix and the singular value decomposition of the projection normal matrix are

$$
\begin{aligned}
\boldsymbol{A} &= F \cdot F^T \\
&= \frac{\pi}{6}\begin{bmatrix} 3 & 0 & 2 & 1 & 1 & 2 \\ 0 & 3 & 1 & 2 & 2 & 1 \\ 2 & 1 & 3 & 0 & 2 & 1 \\ 1 & 2 & 0 & 3 & 1 & 2 \\ 1 & 2 & 2 & 1 & 3 & 0 \\ 2 & 1 & 1 & 2 & 0 & 3 \end{bmatrix} \\
&= \boldsymbol{U}S \cdot S^T\boldsymbol{U}^T \\
&= \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & -1 & -1 & -1 & 1 & 1 \\ 1 & 1 & -1 & -1 & 0 & -2 \\ 1 & -1 & 1 & 1 & 0 & -2 \\ 1 & 0 & -2 & 1 & -1 & 1 \\ 1 & 0 & 2 & -1 & -1 & 1 \end{bmatrix}
\begin{bmatrix} \frac{1}{\sqrt{6}} & & & & & \\ & \frac{1}{\sqrt{4}} & & & & \\ & & \frac{1}{\sqrt{12}} & & & \\ & & & \frac{1}{\sqrt{6}} & & \\ & & & & \frac{1}{\sqrt{4}} & \\ & & & & & \frac{1}{\sqrt{12}} \end{bmatrix} \\
&\quad \begin{bmatrix} \frac{9}{6\pi} & & & & & \\ & \frac{4}{6\pi} & & & & \\ & & \frac{4}{6\pi} & & & \\ & & & \frac{1}{6\pi} & & \\ & & & & 0 & \\ & & & & & 0 \end{bmatrix} \\
&\quad \begin{bmatrix} \frac{1}{\sqrt{6}} & & & & & \\ & \frac{1}{\sqrt{4}} & & & & \\ & & \frac{1}{\sqrt{12}} & & & \\ & & & \frac{1}{\sqrt{6}} & & \\ & & & & \frac{1}{\sqrt{4}} & \\ & & & & & \frac{1}{\sqrt{12}} \end{bmatrix}
\begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 & 0 & 0 \\ 1 & -1 & -1 & 1 & -2 & 2 \\ 1 & -1 & -1 & 1 & 1 & -1 \\ 1 & 1 & 0 & 0 & -1 & -1 \\ 1 & 1 & -2 & -2 & 1 & 1 \end{bmatrix}
\end{aligned}
\tag{2.19}
$$

(2.20)

where the left singular vectors, $\boldsymbol{U}$, are given as the product of a matrix and the inverse

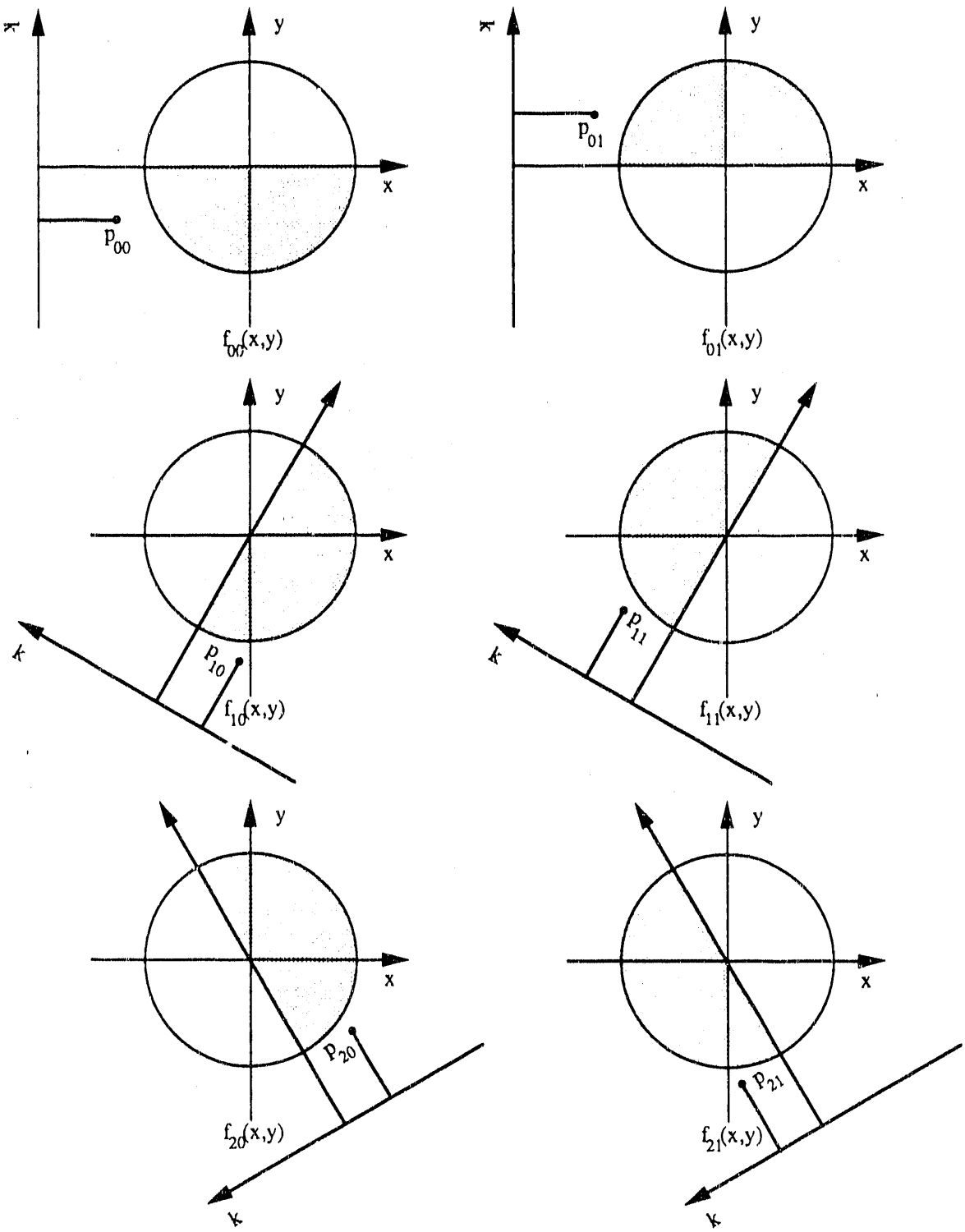


Figure 2.4: Schematic of spatial sampling functions for a simple parallel beam tomographic system with three equally spaced projection angles and two projection bins at each angle.

of the norm of that matrix. There are four non-zero singular values for this example parallel beam sampling system meaning that of the six measurements only four are linearly independent. For parallel beam sampling on a circular domain using uniform impulse response functions with Θ angles and K bins at each angle, there will be at most $\Theta(K-1)+1$ singular values that are non-zero. This new result is proved by considering all the impulse response functions at two different angles. Recall, that the impulse response function, $f_{\theta k}(x,y)$, is the spatial response of a particular detector at angle θ and bin k to a point source moved to every position, (x,y), within the sampling domain. By subtracting all but one of the impulse response functions at one angle from all of the impulse response functions at the other angle, the resulting difference is equal to the impulse response function that was not included in the difference; e.g., for the system of figure 2.4,

$$f_{00}(\cdot,\cdot) + f_{01}(\cdot,\cdot) - f_{10}(\cdot,\cdot) \;=\; f_{11}(\cdot,\cdot). \tag{2.21}$$

For one angle paired with the other $\Theta-1$ angles paired this will be true; thus, at least $\Theta-1$ of the impulse response functions are linearly dependent. Since there are ΘK measurements, there will be at most $\Theta K-(\Theta-1) = \Theta(K-1)+1$ linearly independent impulse response functions each corresponding to a non-zero singular value.

Figure 2.5 shows images of the impulse response functions, $f_{\theta k}(x,y)$, for a parallel beam sampling of a circular domain using sixteen projection angles and sixteen projection bins at each angle. Images across a row correspond to different projection

bins at a fixed projection angle and each row of images corresponds to a different projection angle for a total of 256 images. The impulse response functions have been discretized for display purposes only; in fact, these functions have a continuous domain on the unit circle. A plot of the singular values of the system is shown in figure 2.6. The singular values were obtained by performing the singular value decomposition of the projection normal matrix as in equation 2.8. It is interesting to note that the singular value changes by greater than two orders of magnitude between projection index 240 and projection index 241 which agrees with the theoretical result of 241 non-zero singular values for sixteen angles and sixteen bins (the projection index starts at zero). The details of this plot with be discussed further in chapter 3 and chapter 4. Figure 2.7 shows images of the left singular vectors, $\boldsymbol{U}$. Each singular vector is represented by an image in sinogram format; pixels across a row correspond to different projection bins at a fixed projection angle and each row corresponds to a different projection angle for an image size of sixteen by sixteen. The singular vector images have been ordered from left to right and top to bottom to correspond with the standard non-decreasing order of singular values; the image at the top left corresponds to the largest singular value and the image at the bottom right corresponds to the smallest singular value. Each left singular vector image has been scaled to give the largest contrast within the image; therefore, information about relative intensity between images has been lost.

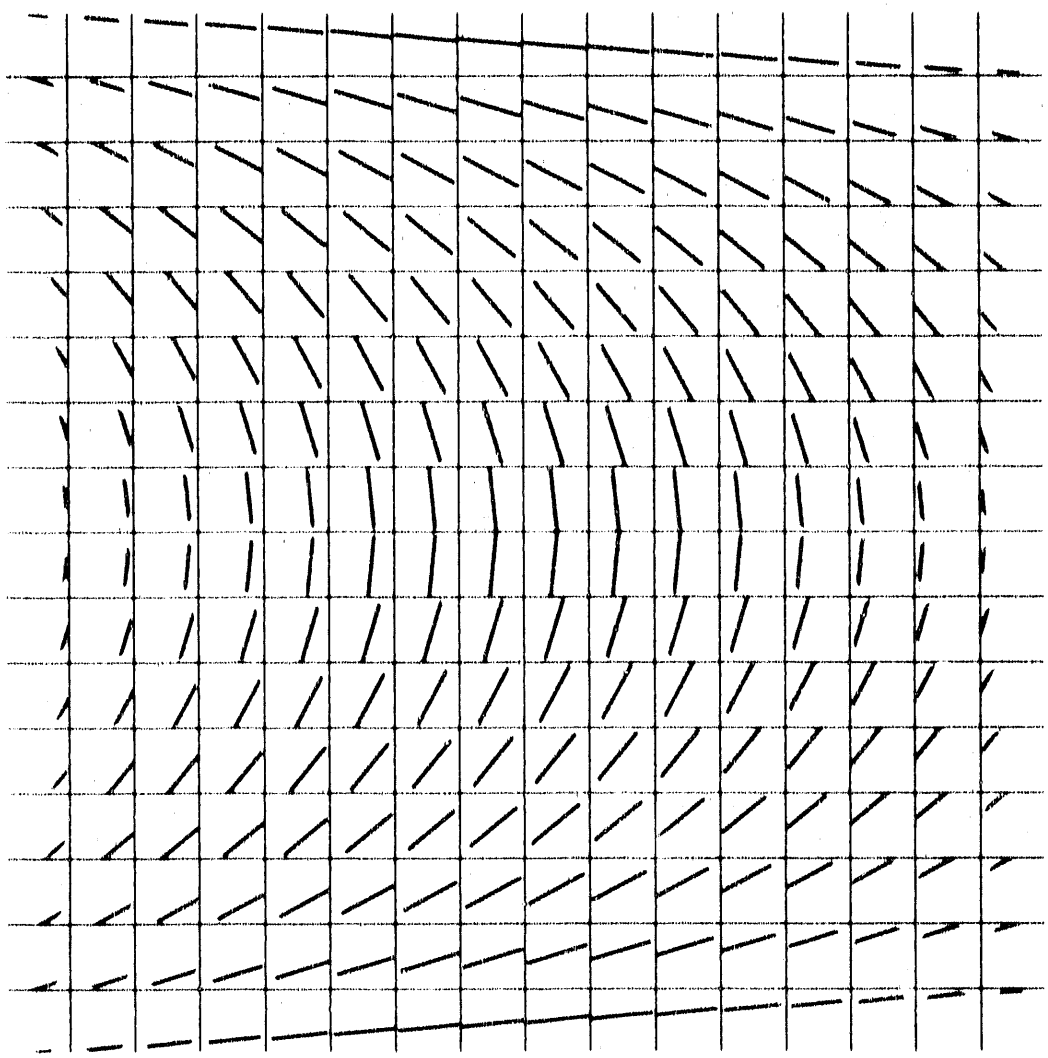


Figure 2.5: Images of spatial sampling functions for a parallel beam tomographic system with sixteen equally spaced projection angles and sixteen projection bins at each angle.

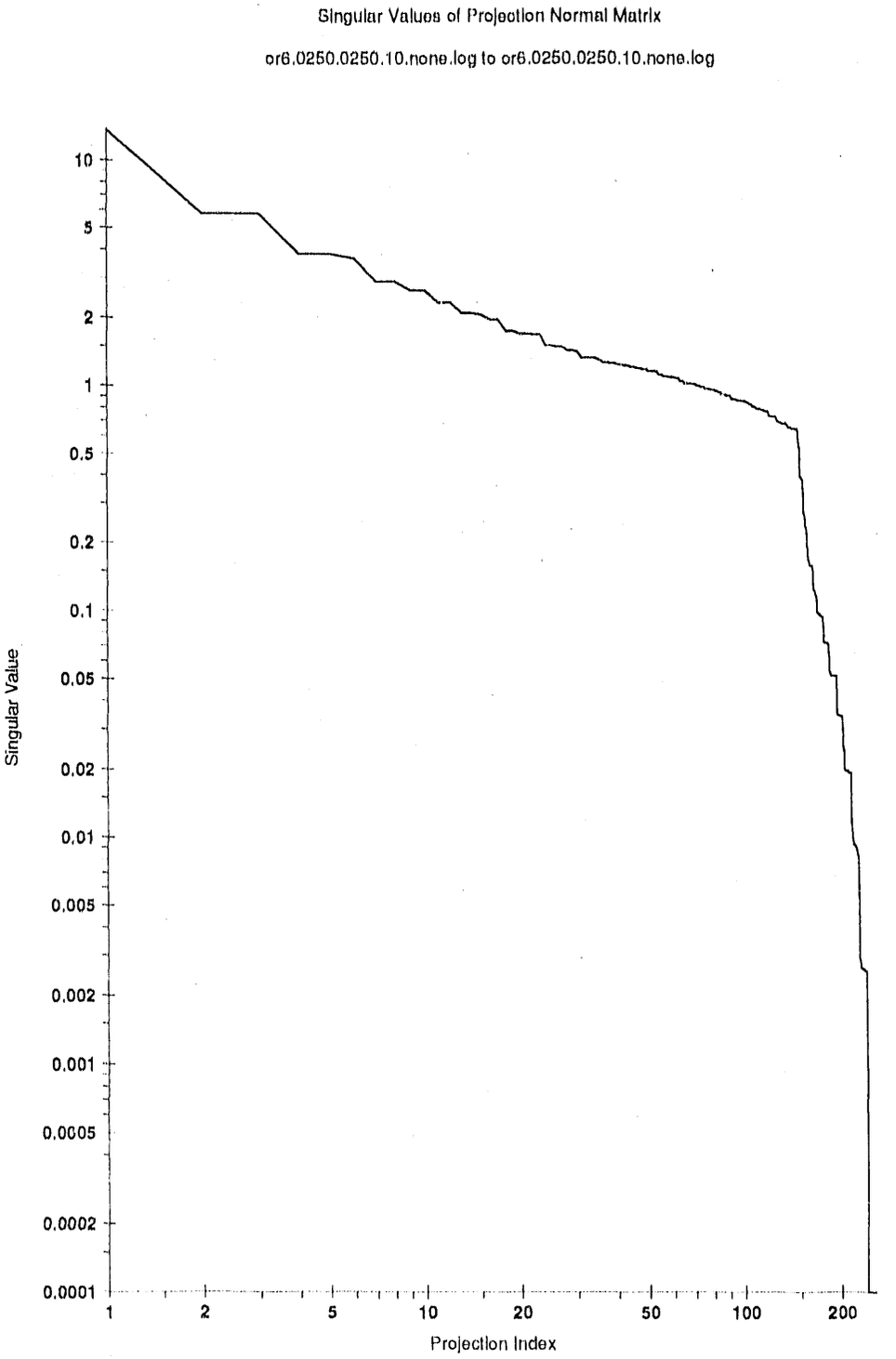


Figure 2.6: Singular value plot for a parallel beam tomographic system with sixteen equally spaced projection angles and sixteen projection bins at each angle.

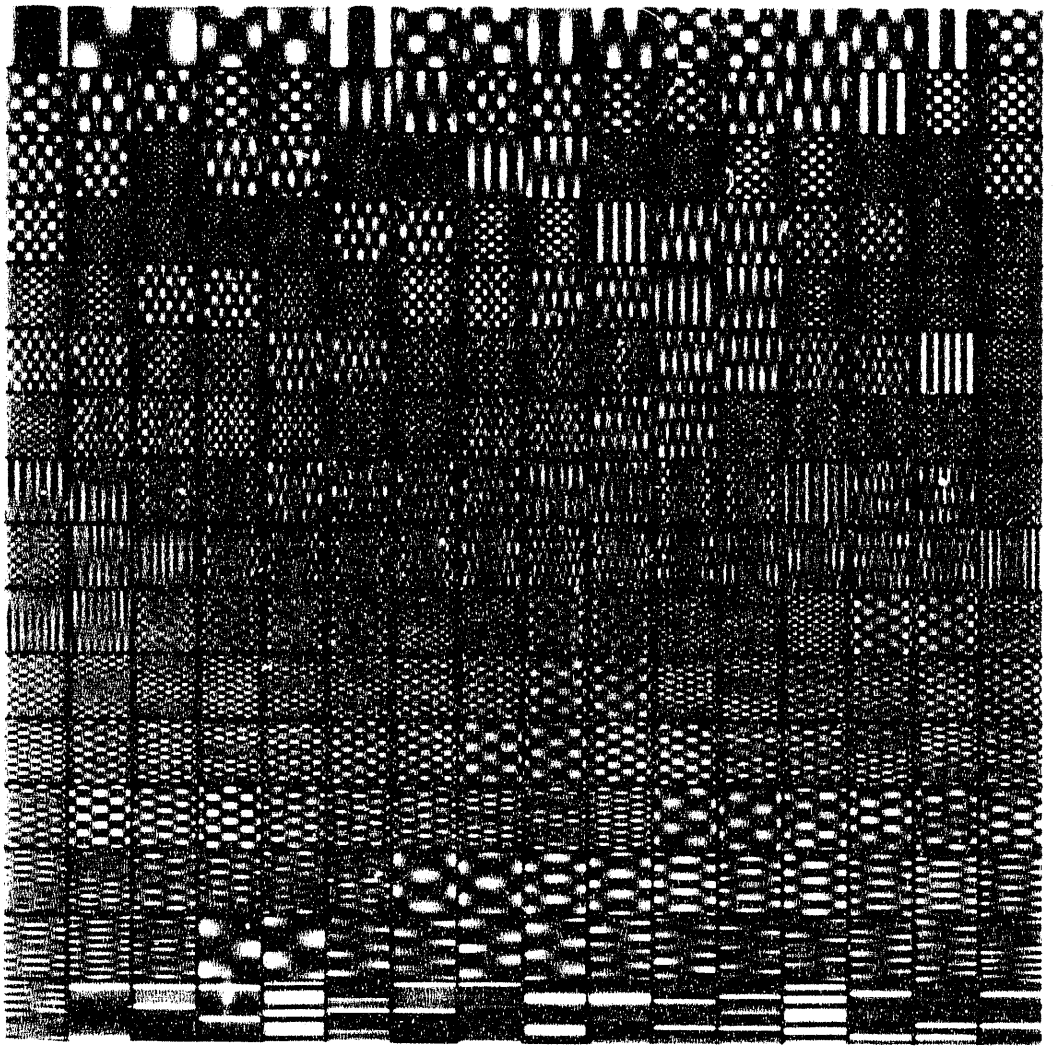


Figure 2.7: Images of left singular vectors for a parallel beam tomographic system with sixteen equally spaced projection angles and sixteen projection bins at each angle

2.1.1 Separable product operators

Separation of the tomograph system response into product operators has significant computational advantages when solving the tomographic inverse problem. There are two classes of physical effects that can be represented by a product decomposition of the tomograph response function [39]. The first of these, denoted by the operator H, can be interpreted as a modification of the object distribution before an ideal experiment; e.g., radioactive decay and the positron range effect. The second class of effects, denoted by the matrix $\boldsymbol{G}$, can be interpreted as modifying the dataset from an ideal experiment; e.g., attenuation in PET, intercrystal scatter, crystal penetration, and instrumental effects such as detection efficiency and dead-time. For these two classes of physical effects the non-ideal data acquisition process is represented by

$$\boldsymbol{p} = \boldsymbol{G}\vec{F}_0 \cdot H \cdot b \tag{2.22}$$

where F_0 represents line integral parallel projection.

2.1.2 Rotational invariance

When the elements of the projection normal matrix are a function of only the difference $\Delta\theta = \theta' - \theta$ modulo Θ as shown in equation 2.23, the system is *rotationally invariant*. If it is not a function of the difference between k and k' modulo K, then the system is *radially variant*. The system is *spatially variant* if it is rotationally or radially variant.

$$A_{\theta' k' \theta k} = A_{[(\theta-\theta')\bmod\Theta]k'0k} \tag{2.23}$$

$$= A_{[\Delta\theta\bmod\Theta]k'0k} \tag{2.24}$$

$$= \int_{\mathbf{R}} dy \int_{\mathbf{R}} dx\, f_{[\Delta\theta\bmod\Theta]k}(x,y)\, f_{0k}(x,y) \tag{2.25}$$

When the system is rotationally invariant, the projection normal matrix can be written in block circulant form. The block circulant structure is

$$\boldsymbol{A} = \begin{bmatrix} A_0 & A_1 & A_2 & .. & A_{\Theta-2} & A_{\Theta-1} \\ A_{\Theta-1} & A_0 & A_1 & .. & A_{\Theta-3} & A_{\Theta-2} \\ A_{\Theta-2} & A_{\Theta-1} & A_0 & .. & A_{\Theta-4} & A_{\Theta-3} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ A_2 & A_3 & A_4 & .. & A_0 & A_1 \\ A_1 & A_2 & A_3 & .. & A_{\Theta-1} & A_0 \end{bmatrix}. \tag{2.26}$$

There are $\Theta \times \Theta$ blocks each of size $K \times K$.

Let $f_k(x,y)$ denote the impulse response of a rotationally invariant, radially variant system, then equation 2.1 can be rewritten as

$$p_{\theta k} = \int_{\mathbf{R}} dy \int_{\mathbf{R}} dx\, f_k \Big[\; x\cos\left(\theta\frac{2\pi}{\Theta}\right) + y\sin\left(\theta\frac{2\pi}{\Theta}\right), \\ -x\sin\left(\theta\frac{2\pi}{\Theta}\right) + y\cos\left(\theta\frac{2\pi}{\Theta}\right)\Big]\; b(x,y). \tag{2.27}$$

The block circulant projection normal matrix has elements given by

$$
\begin{aligned}
A_{\theta' k' \theta k} &= \int_{\mathbf{R}} dy \int_{\mathbf{R}} dx \, f_{k'} \Big[\; x \cos\left(\theta' \frac{2\pi}{\Theta}\right) + y \sin\left(\theta' \frac{2\pi}{\Theta}\right), \\
&\qquad -x \sin\left(\theta' \frac{2\pi}{\Theta}\right) + y \cos\left(\theta' \frac{2\pi}{\Theta}\right) \Big] \\
&\qquad f_k \Big[\; x \cos\left(\theta \frac{2\pi}{\Theta}\right) + y \sin\left(\theta \frac{2\pi}{\Theta}\right), \\
&\qquad -x \sin\left(\theta \frac{2\pi}{\Theta}\right) + y \cos\left(\theta \frac{2\pi}{\Theta}\right) \Big] \qquad (2.28) \\
&= \int_{\mathbf{R}} dy \int_{\mathbf{R}} dx \, f_{k'} \Big[\; x \cos\left(\Delta\theta \frac{2\pi}{\Theta}\right) + y \sin\left(\Delta\theta \frac{2\pi}{\Theta}\right), \\
&\qquad -x \sin\left(\Delta\theta \frac{2\pi}{\Theta}\right) + y \cos\left(\Delta\theta \frac{2\pi}{\Theta}\right) \Big] \\
&\qquad f_k(x, y) \qquad (2.29)
\end{aligned}
$$

By using the rotational invariance properties of this matrix, computationally fast and efficient algorithms can be implemented for the procedures described in chapters 3 and 4. One such fast algorithm is described in chapter 5.

For the simple system of figure 2.4 the projection normal matrix can be written in block circulant form by reversing the direction of the projection axis for projection angle $\theta = 1$. With this change,

$$
\boldsymbol{A} = \frac{\pi}{6} \begin{bmatrix}
3 & 0 & 1 & 2 & 1 & 2 \\
0 & 3 & 2 & 1 & 2 & 1 \\
1 & 2 & 3 & 0 & 1 & 2 \\
2 & 1 & 0 & 3 & 2 & 1 \\
1 & 2 & 1 & 2 & 3 & 0 \\
2 & 1 & 2 & 1 & 0 & 3
\end{bmatrix}. \qquad (2.30)
$$

There are three blocks each of size two by two.

2.2 Organization of models

The data acquisition model presented in section 2.1 and the novel singular value decomposition of that model form a common, unifying structure for the rest of the material in this work. In chapter 3, it will be used for identification of the tomograph system response function and estimation of the unknown spatial distribution. Chapter 4 will discuss optimization of the system response function to obtain better estimates of the unknown spatial distribution using this formalism.

2.2.1 Identification

The first task is to identify the important phenomena in the data acquisition process for a particular imaging modality. The response of the tomograph is found using a combination of deterministic and stochastic simulations and measured point response data. For positron emission tomography, the effects modeled include radioactive decay, positron range, sampling geometry, attenuation, inter-crystal scatter, crystal penetration, and detection efficiency.

2.2.2 Estimation

An estimator for the unknown spatial distribution is formulated using the known response function and the statistical characteristics of the acquired data. Several of the above phenomena lead to position dependent resolution and reconstruction algorithms based on the spatially variant model have reduced bias when compared to spatially

invariant methods. Least squares, normal maximum likelihood, and Poisson maximum likelihood estimators and the corresponding covariance have been formulated to compensate for a spatially variant response. While bias is reduced by these estimators, undesirable statistical and systematic fluctuations can result due to pixelization effects. To study the effects of basis selection, each of the three estimators has been implemented using three different pixel bases; square pixels, projection ray natural pixels, and new orthonormal basis determined from the singular value decomposition of the tomograph response function.

2.2.3 Optimization

The data acquisition system may be optimized so better estimates of the unknown spatial distribution will result. The criterion used is to minimize the norm of the covariance matrix while keeping the estimator unbiased.

Chapter 3

Estimation and Identification

3.1 Pixel Basis Representation

The goal of tomography is to reconstruct the unknown distribution, b, from one realization, $\underline{\boldsymbol{p}}$, of the projection measurement process $\boldsymbol{p}$. Reconstructing the true continuous space distribution, b, from sampled projections is probably impossible without prior information about the distribution. Instead, a discretized representation, c_{mn}, is estimated from the measurements where $B_{mn}(x,y)$ defines a generalized pixel.

$$b(x,y) \approx \sum_{mn} B_{mn}^{T}(x,y)c_{mn} \tag{3.1}$$

In vector form the pixelization is

$$b \approx B^{T}\boldsymbol{c}. \tag{3.2}$$

A set of generalized pixel coefficients can be found by minimizing the square of the L_2 norm [38] of the difference between the continuous space object, b, and the generalized pixel representation of the object, $B^T\boldsymbol{c}$, over all possible generalized pixel coefficients, $\boldsymbol{c}$; i.e.,

$$\boldsymbol{c} \equiv \arg\min_{\boldsymbol{c}} \left\{ \left\| b - B^T\boldsymbol{c} \right\|_2^2 \right\}. \tag{3.3}$$

Following the derivation of section 3.2, the generalized pixel coefficients are given by

$$\boldsymbol{c} = (B \cdot B^T)^+ B \cdot b. \tag{3.4}$$

The operator $^+$ is the Moore-Penrose pseudo-inverse and satisfies the following relations [40] [41] [18].

$$MM^+M = M \tag{3.5}$$

$$M^+MM^+ = M^+ \tag{3.6}$$

$$MM^+ = (MM^+)^T \tag{3.7}$$

$$M^+M = (M^+M)^T \tag{3.8}$$

The explicit representation of the object by the generalized pixel basis of equation 3.1 has not appeared previously in the tomography literature.

The basis set used to describe the pixels influences the types of artifacts that appear in the reconstructed image. In this work, three bases are evaluated. The

first, $B1$, is the traditional square pixel or Heavyside basis. The second basis set, $B2$, consists of the set of functions that comprise the tomograph system response functional and was proposed by Buonocore [27]. The third, $B3$, is original to this work and is composed of the right singular functions V, defined by equation 2.7, that have been selected by S and normalized by the L_2 norm.

$$B1 \equiv \text{Heavyside} \tag{3.9}$$

$$B2 \equiv F \tag{3.10}$$

$$B3 \equiv \left[\left(S \cdot S^T\right)^+\right]^{\frac{1}{2}} S \cdot V^T \tag{3.11}$$

Note that the basis set $B3$ can be calculated from linear combinations of the basis set $B2$ as follows

$$B3 = \left[\left(S \cdot S^T\right)^+\right]^{\frac{1}{2}} \boldsymbol{U}^T B2. \tag{3.12}$$

This result can be proved by replacing the definition definition of the natural pixel basis from equation 3.10 in equation 3.12; whereby,

$$B3 = \left[\left(S \cdot S^T\right)^+\right]^{\frac{1}{2}} \boldsymbol{U}^T F \tag{3.13}$$

and after substituting the singular value decomposition of the tomograph system response function from equation 2.7,

$$B3 = \left[\left(S \cdot S^T\right)^+\right]^{\frac{1}{2}} \boldsymbol{U}^T \boldsymbol{U} S \cdot V^T. \tag{3.14}$$

Removing the identity $\boldsymbol{U}^T\boldsymbol{U}$ yields

$$B3 \;=\; \left[\left(S\cdot S^T\right)^+\right]^{\frac{1}{2}} S\cdot V^T \tag{3.15}$$

which is equivalent to equation 3.11.

3.1.1 Example

Figures 3.1-3.3 show, respectively, a possible set of basis functions for square pixels, Buonocore's natural pixels, and orthonormal natural pixels using the sampling defined by the impulse response functions of figure 2 4,

Consider the wedge shaped object of figure 3.4 which is defined as

$$b(x,y) \;=\; \begin{cases} 1 & \text{if } \left|\frac{y}{x}\right| < \frac{1}{2} \text{ and } x \geq 0 \text{ and } x^2+y^2 \leq 1, \\ 0 & \text{otherwise.} \end{cases} \tag{3.16}$$

The pixel coefficients using the square pixel basis of figure 3.1 are

$$\boldsymbol{c} \;=\; \begin{bmatrix} \frac{1}{3} & 0 & 0 & \frac{1}{3} \end{bmatrix}^T \tag{3.17}$$

and were found by solving equation 3.4. In Buonocore's natural pixel basis of figure 3.2, the pixel coefficients are

$$\boldsymbol{c} \;=\; \frac{1}{72}\begin{bmatrix} 4 & 4 & 13 & -5 & 13 & -5 \end{bmatrix}^T \tag{3.18}$$

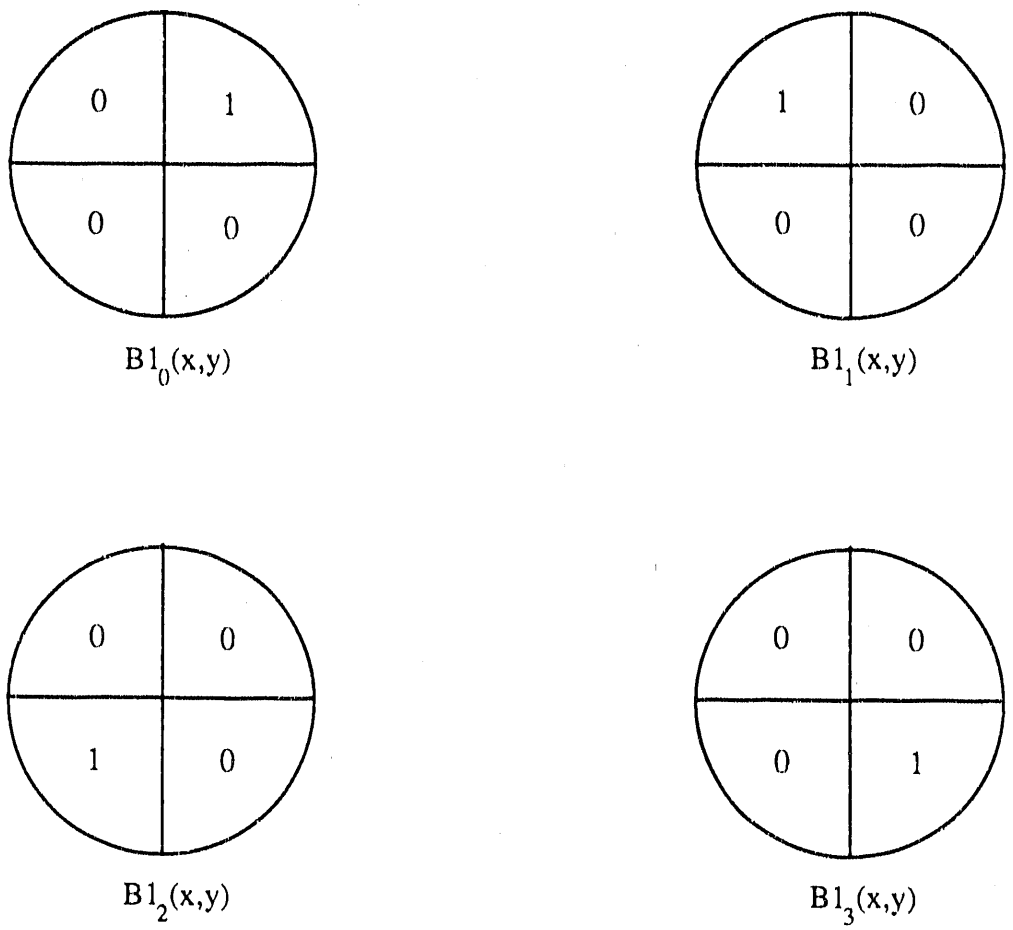


Figure 3.1: Square pixel or Heavyside basis for a simple parallel beam tomographic system with three equally spaced projection angles and two projection bins at each angle.

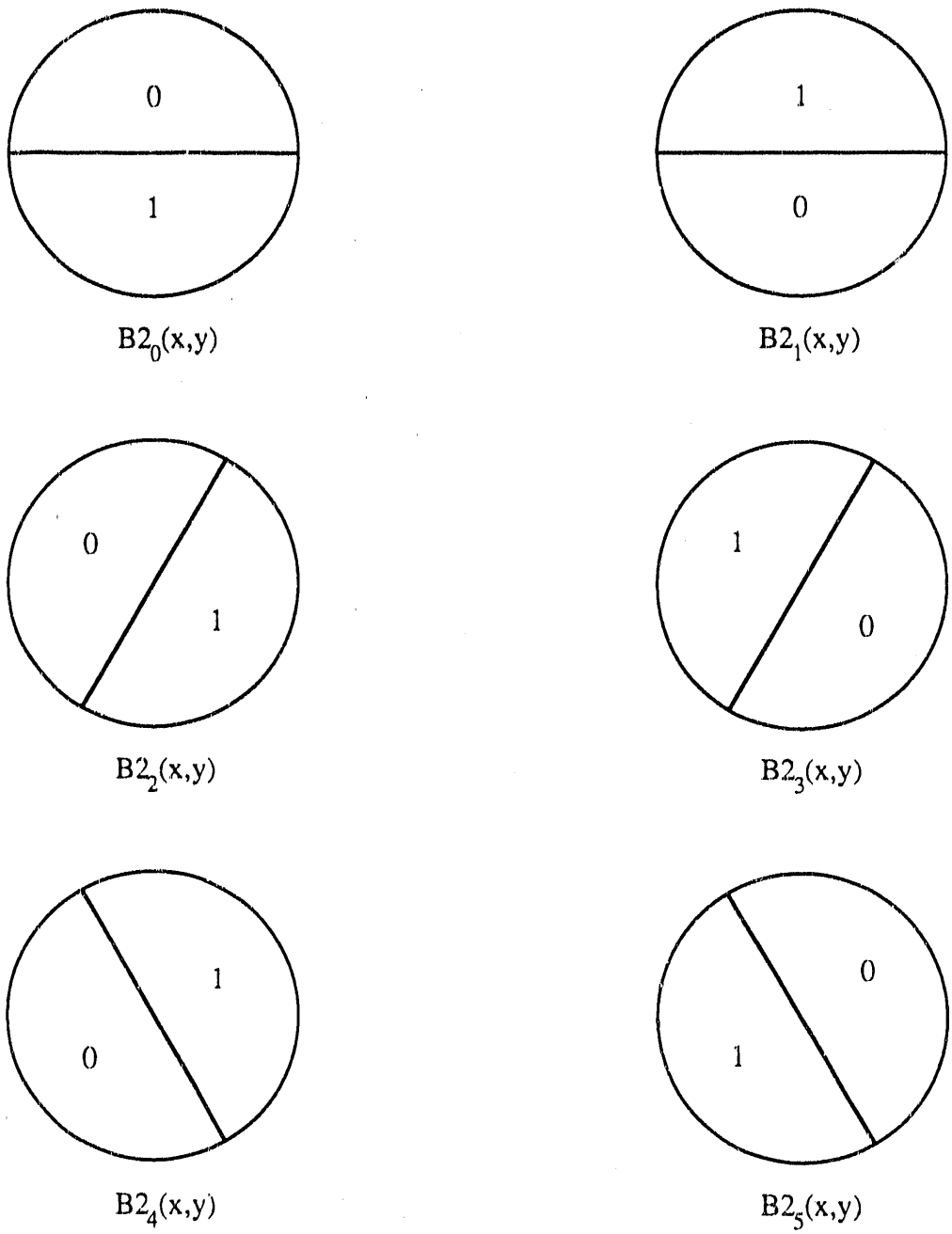


Figure 3.2: Buonocore's natural pixel basis for a simple parallel beam tomographic system with three equally spaced projection angles and two projection bins at each angle.

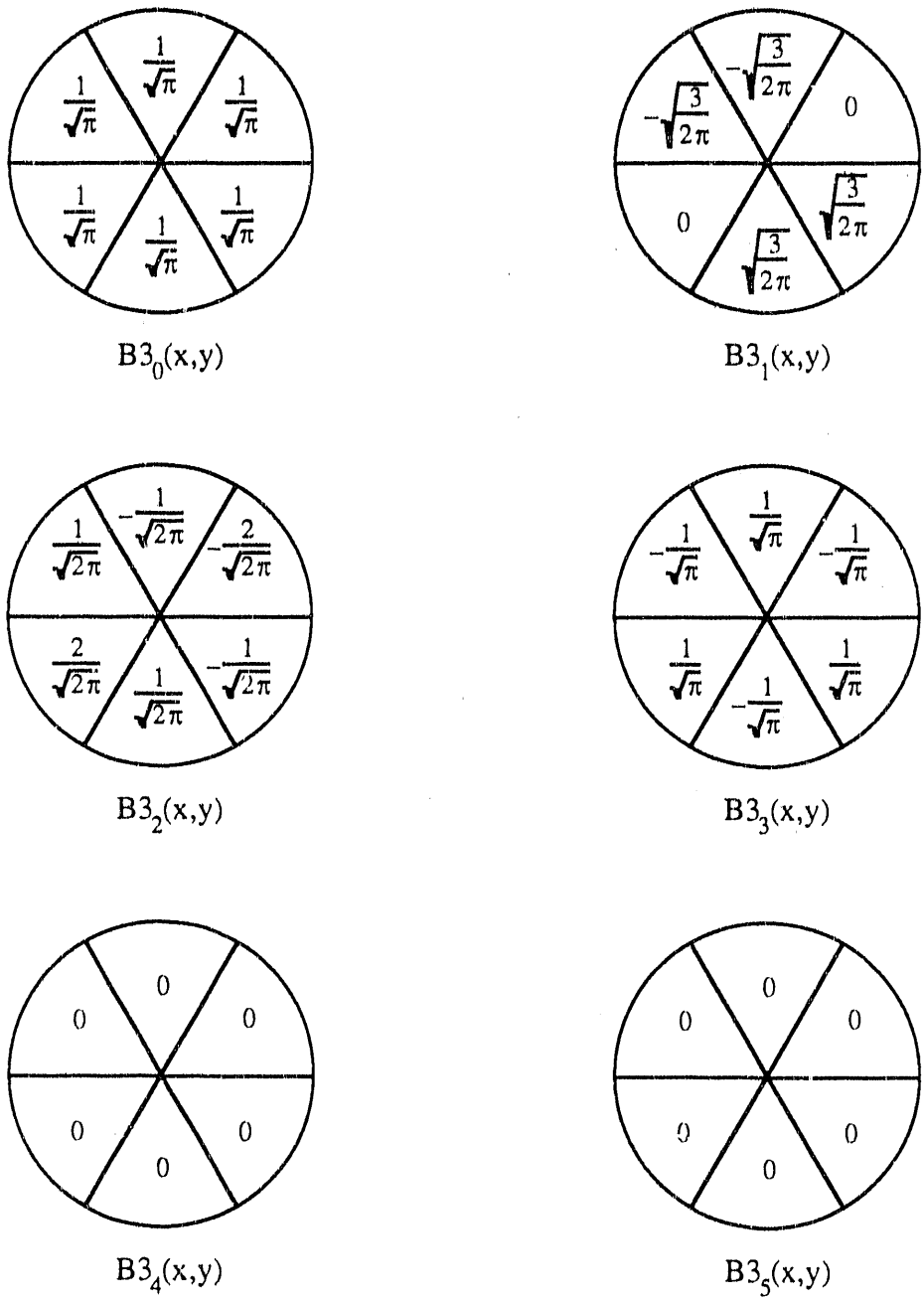


Figure 3.3: New orthonormal pixel basis for a simple parallel beam tomographic system with three equally spaced projection angles and two projection bins at each angle.

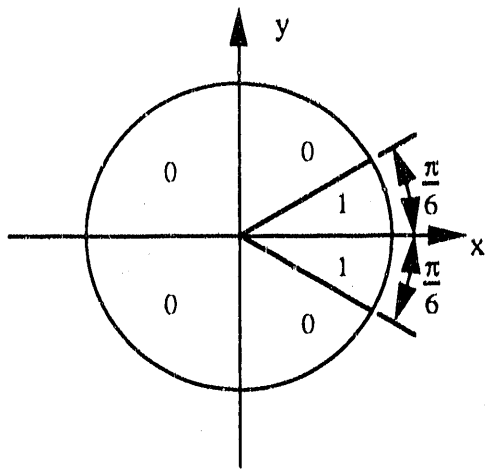


Figure 3.4: Wedge shaped phantom object.

and for the orthonormal natural pixel basis shown in figure 3.3

$$\boldsymbol{c} = \frac{\sqrt{\pi}}{4}\left[\begin{array}{ccccccc} 1 & \frac{1}{\sqrt{6}} & -\frac{1}{\sqrt{6}} & 0 & 0 & 0 \end{array}\right]^T. \tag{3.19}$$

Figure 3.5 shows, respectively, the continuous space representation of the wedge shaped object using square pixels, Buonocore's natural pixels, and orthonormal natural pixels with the sampling defined by the impulse response functions of figure 2.4.

3.1.2 Reprojection

When an object is pixelized by bases such as $B1$ or polar pixels [24] the projections of the pixelized object; i.e., $F \cdot B^T \boldsymbol{c}$, will not, in general, yield the same set of projections as the original object. This results in a systematic error in the estimates of unknown spatial distribution when algorithms that require backprojection and/or reprojection

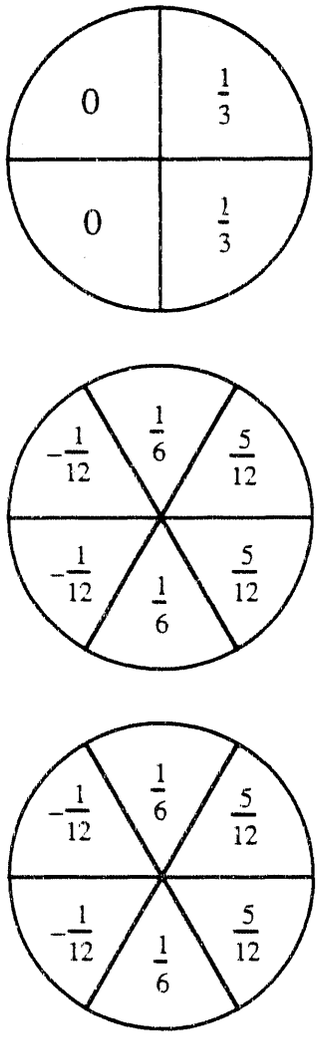


Figure 3.5: Representation of a wedge shaped object using top) square pixels, middle) Buonocore's natural pixels, and bottom) orthonormal natural pixels.

are used. Eliminating this systematic error is especially important in iterative algorithms [1] [42] [43]. When the wedge shaped object shown in figure 3.4 is sampled by the system of figure 2.4 gives projections

$$\boldsymbol{p} = \frac{\pi}{6}\begin{bmatrix} \frac{1}{2} & \frac{1}{2} & 1 & 0 & 1 & 0 \end{bmatrix}^T. \tag{3.20}$$

Projecting the square pixelized version of the wedge yields

$$\tilde{\boldsymbol{p}} = \frac{\pi}{6}\begin{bmatrix} \frac{1}{2} & \frac{1}{2} & \frac{5}{6} & \frac{1}{6} & \frac{5}{6} & \frac{1}{6} \end{bmatrix}^T. \tag{3.21}$$

Projection of a pixelized object using Buonocore's natural pixel basis, $B2$, is

$$\tilde{\boldsymbol{p}} = F \cdot B2^T \boldsymbol{c}. \tag{3.22}$$

Substituting the definition of Buonocore's natural pixel basis from equation 3.11 gives

$$\tilde{\boldsymbol{p}} = F \cdot F^T \boldsymbol{c} \tag{3.23}$$

and after substituting equation 3.11 into equation 3.4,

$$\tilde{\boldsymbol{p}} = F \cdot F^T \left(F \cdot F^T\right)^+ F \cdot b. \tag{3.24}$$

Using the properties of the pseudo-inverse in equations 3.6-3.7,

$$\tilde{\boldsymbol{p}} = F \cdot b \tag{3.25}$$

which from equation 2.3 yields

$$\tilde{\boldsymbol{p}} = \boldsymbol{p}. \tag{3.26}$$

Thus, the projections, $\tilde{\boldsymbol{p}}$, of the Buonocore pixelized object are identical to the projections, $\boldsymbol{p}$, of the original object. For the wedge example,

$$\tilde{\boldsymbol{p}} = \frac{\pi}{6}\begin{bmatrix} \frac{1}{2} & \frac{1}{2} & 1 & 0 & 1 & 0 \end{bmatrix}^T. \tag{3.27}$$

Likewise, for the orthonormal natural pixel basis, $B3$, projection of the discretized object is

$$\tilde{\boldsymbol{p}} = F \cdot B3^T \boldsymbol{c} \tag{3.28}$$

which after replacement with equation 3.11 becomes

$$\tilde{\boldsymbol{p}} = F \cdot V \cdot S^T \left[\left(S \cdot S^T\right)^+\right]^{\frac{1}{2}} \boldsymbol{c}. \tag{3.29}$$

Substituting for $\boldsymbol{c}$ the result of combining equation 3.11 and equation 3.4,

$$\tilde{\boldsymbol{p}} = F \cdot V \cdot S^T \left[\left(S \cdot S^T\right)^+\right]^{\frac{1}{2}} \left[\left(S \cdot S^T\right)^+\right]^{\frac{1}{2}} S \cdot V^T \cdot b. \tag{3.30}$$

Replacing F with the right hand side of equation 2.3 gives

$$\tilde{\boldsymbol{p}} \;=\; \boldsymbol{U}S \cdot V^T \cdot V \cdot S^T \left(S \cdot S^T\right)^+ S \cdot V^T \cdot b \tag{3.31}$$

and since the operation of the adjoint of the right singular functions on the right singular functions is the identity operator,

$$\tilde{\boldsymbol{p}} \;=\; \boldsymbol{U}S \cdot S^T \left(S \cdot S^T\right)^+ S \cdot V^T \cdot b. \tag{3.32}$$

The properties of the pseudo-inverse in equations 3.6-3.8 can be used to show

$$\tilde{\boldsymbol{p}} \;=\; \boldsymbol{U}S \cdot V^T \cdot b. \tag{3.33}$$

Upon substitution of equation 2.7,

$$\tilde{\boldsymbol{p}} \;=\; F \cdot b \tag{3.34}$$

which from equation 2.3 yields

$$\tilde{\boldsymbol{p}} \;=\; \boldsymbol{p}. \tag{3.35}$$

The projection of the wedge example using the orthonormal natural pixel basis, $B3$, is

$$\tilde{\boldsymbol{p}} = \frac{\pi}{6}\left[\begin{array}{cccccc} \frac{1}{2} & \frac{1}{2} & 1 & 0 & 1 & 0 \end{array}\right]^T. \tag{3.36}$$

All systematic reprojection errors due to pixelization are eliminated using a basis like $B2$ or $B3$ as shown in table 3.1 for the wedge phantom. While this is true in particular for $B2$ and $B3$, any basis that spans the subspace of functions defined by $S \cdot V^T$ will also have this property. A geometric interpretation of the error associated with computing projections of pixelized object is shown in figure 3.6. Many of these basis may represent the original object distribution, b, better than $B2$ or $B3$. However, no information about the coefficients for the functions that are outside the space $S \cdot V^T$ is available from the projection measurements. By using *apriori* information about the continuous space distribution of b, the formulation of Bayesian estimators that use basis functions not in $S \cdot V^T$ is an exciting area for future research. Thus, the basis subset contained in $S \cdot V^T$ is from a channel model for F and the subset contained in $V^T - S \cdot V^T$ is from a process model for b.

3.2 Least Squares

In this section, a least squares estimator (LSE) is formulated to estimate the mean intensity of the generalized pixels, $\hat{\boldsymbol{c}}$, used to described the unknown spatial distribution from one measured projection dataset, $\underline{\boldsymbol{p}}$. While the resulting LSE formula is quite general, specific application to image reconstruction using square pixels, Buonocore's natural pixels, and the new orthonormal natural pixels is shown. The least squares estimator for the mean intensity of the generalized pixel image is found by minimizing

Basis	Object Estimate Squared Error $\left\|b - B^T\boldsymbol{c}\right\|_2^2$	Projection Estimate Squared Error $\left\|\boldsymbol{p} - F \cdot B^T\boldsymbol{c}\right\|_2^2$
$B1$	$\frac{4}{3}$	$\frac{\pi^2}{324}$
$B2$	$\frac{7}{6}$	0
$B3$	$\frac{7}{6}$	0

Table 3.1: Errors due to pixelization for a wedge shaped phantom sampled with a parallel beam tomographic system with three equally spaced projection angles and two projection bins at each angle.

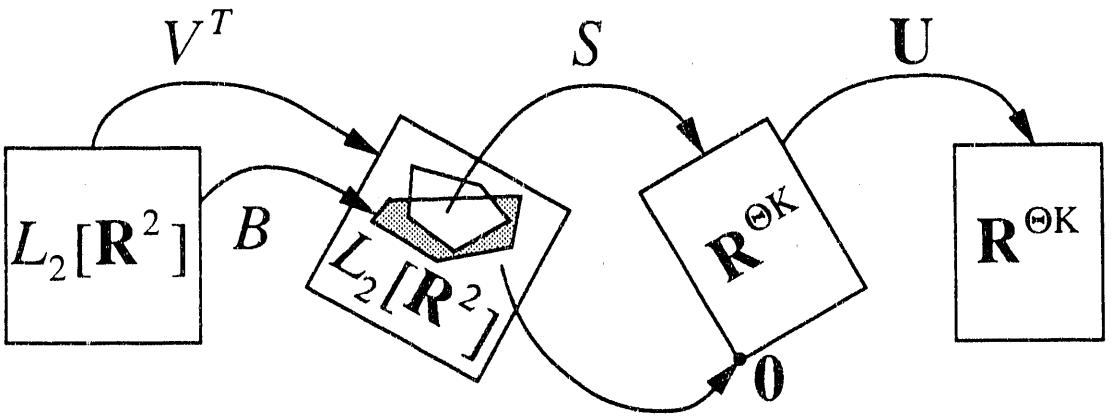


Figure 3.6: Simplified geometric representation of the systematic error that results from computing model projections from pixelized versions of an object. Pixel bases that include the subspace $S \cdot V^T$ eliminate systematic pixelization error.

the square of the L_2 norm of the difference between the projection vector, $\boldsymbol{p}$, and the estimated projection, $F \cdot B^T \boldsymbol{c}$, over all possible image vectors, $\boldsymbol{c}$; i.e.,

$$\hat{\boldsymbol{c}} \equiv \arg\min_{\boldsymbol{c}} \left\{ \left\| \boldsymbol{p} - F \cdot B^T \boldsymbol{c} \right\|_2^2 \right\}. \tag{3.37}$$

After substituting the definition of the L_2 norm,

$$\hat{\boldsymbol{c}} = \arg\min_{\boldsymbol{c}} \left\{ \left(\boldsymbol{p} - F \cdot B^T \boldsymbol{c} \right)^T \left(\boldsymbol{p} - F \cdot B^T \boldsymbol{c} \right) \right\}. \tag{3.38}$$

Finding the gradient vector with respect to unknown parameters, $\boldsymbol{c}$, and equating it with the zero vector, yields a $\Theta K \times \Theta K$ system of simultaneous linear equations.

$$\hat{\boldsymbol{c}} = \arg \left\{ \left[\left(F \cdot B^T \right)^T \left(\boldsymbol{p} - F \cdot B^T \boldsymbol{c} \right) \right] = 0 \right\} \tag{3.39}$$

Upon rearrangement, the equations in normal form are

$$B \cdot F^T F \cdot B^T \hat{\boldsymbol{c}} = B \cdot F^T \boldsymbol{p}. \tag{3.40}$$

The solution for this system using the second property, equation 3.7, and the third property, equation 3.8, of the pseudo-inverse is

$$\hat{\boldsymbol{c}} = \left(B \cdot F^T F \cdot B^T \right)^+ B \cdot F^T \boldsymbol{p}. \tag{3.41}$$

Substitution of the singular value decomposition of the projection formation operator, equation 2.7, yields

$$\hat{\boldsymbol{c}} = \left(B \cdot V \cdot S^T S \cdot V^T \cdot B^T\right)^+ B \cdot V \cdot S^T \boldsymbol{U}^T \boldsymbol{p}. \tag{3.42}$$

By equation 3.7, the minimum L_2 norm least squares estimator (LSE) for the mean value of the intensity of the generalized pixel image is

$$\hat{\boldsymbol{c}} = \left(S \cdot V^T \cdot B^T\right)^+ \boldsymbol{U}^T \boldsymbol{p}. \tag{3.43}$$

Using the measured projection vector, $\underline{\boldsymbol{p}}$, as a single sample estimate of the mean projection vector gives

$$\hat{\boldsymbol{c}} = \left(S \cdot V^T \cdot B^T\right)^+ \boldsymbol{U}^T \underline{\boldsymbol{p}}. \tag{3.44}$$

The fluctuations of the generalized pixel least squares estimator due to random variations in the measurements can typically be characterized in terms of the covariance between pixel estimates. The covariance matrix for the generalized pixel estimator is defined by

$$\boldsymbol{\Sigma}_{\hat{\boldsymbol{c}}} \equiv E\left[\left(\hat{\boldsymbol{c}} - E\hat{\boldsymbol{c}}\right)\left(\hat{\boldsymbol{c}} - E\hat{\boldsymbol{c}}\right)^T\right]. \tag{3.45}$$

Substitution of equation 3.43 yields

$$\begin{aligned}\boldsymbol{\Sigma}_{\hat{\boldsymbol{c}}} = E\Big[&\left\{\left(S \cdot V^T \cdot B^T\right)^+ \boldsymbol{U}^T \boldsymbol{p} - E\left(S \cdot V^T \cdot B^T\right)^+ \boldsymbol{U}^T \boldsymbol{p}\right\} \\ &\left\{\left(S \cdot V^T \cdot B^T\right)^+ \boldsymbol{U}^T \boldsymbol{p} - E\left(S \cdot V^T \cdot B^T\right)^+ \boldsymbol{U}^T \boldsymbol{p}\right\}^T\Big].\end{aligned} \tag{3.46}$$

Applying the relationship for the transpose of a product yields

$$\begin{aligned}\boldsymbol{\Sigma}_{\hat{c}} &= E\left[\left\{\left(S\cdot V^T\cdot B^T\right)^+\boldsymbol{U}^T\boldsymbol{p}-E\left(S\cdot V^T\cdot B^T\right)^+\boldsymbol{U}^T\boldsymbol{p}\right\}\right. \\ &\qquad \left.\left\{\boldsymbol{p}^T\boldsymbol{U}\left(B\cdot V\cdot S^T\right)^+-E\boldsymbol{p}^T\boldsymbol{U}\left(B\cdot V\cdot S^T\right)^+\right\}\right].\end{aligned} \tag{3.47}$$

The reconstruction filter can be brought outside the expectation so

$$\boldsymbol{\Sigma}_{\hat{\boldsymbol{c}}} = \left(S\cdot V^T\cdot B^T\right)^+\boldsymbol{U}^T E\left[\left(\boldsymbol{p}-E\boldsymbol{p}\right)\left(\boldsymbol{p}^T-E\boldsymbol{p}^T\right)\right]\boldsymbol{U}\left(B\cdot V\cdot S^T\right)^+. \tag{3.48}$$

Again using the relationship for transpose,

$$\boldsymbol{\Sigma}_{\hat{\boldsymbol{c}}} = \left(S\cdot V^T\cdot B^T\right)^+\boldsymbol{U}^T E\left[\left(\boldsymbol{p}-E\boldsymbol{p}\right)\left(\boldsymbol{p}-E\boldsymbol{p}\right)^T\right]\boldsymbol{U}\left(B\cdot V\cdot S^T\right)^+. \tag{3.49}$$

The term inside the expectation is the covariance of the projections; i.e.,

$$\boldsymbol{\Sigma}_{\boldsymbol{p}} = E\left[\left(\boldsymbol{p}-E\boldsymbol{p}\right)\left(\boldsymbol{p}-E\boldsymbol{p}\right)^T\right], \tag{3.50}$$

therefore, the covariance matrix for the generalized pixel least square estimator is

$$\boldsymbol{\Sigma}_{\hat{\boldsymbol{c}}} = \left(S\cdot V^T\cdot B^T\right)^+\boldsymbol{U}^T\boldsymbol{\Sigma}_{\boldsymbol{p}}\boldsymbol{U}\left(B\cdot V\cdot S^T\right)^+. \tag{3.51}$$

The estimator for the mean of the intensity of the object in continuous space is found by applying the adjoint of the basis operator, B, to the generalized pixel

estimator of the intensity mean. For the continuous space object, the least squares estimator for the mean of the intensity is

$$\hat{b} = B^T \hat{\mathbf{c}} \tag{3.52}$$

$$= B^T \left(S \cdot V^T \cdot B^T \right)^+ \boldsymbol{U}^T \boldsymbol{p}. \tag{3.53}$$

When using the Heavyside basis, the generalized pixel least squares estimator for the mean of the intensity, the covariance for that estimator, and the continuous space least squares estimator for the mean of the intensity become

$$\hat{\boldsymbol{c}} = \left(S \cdot V^T \cdot B1^T \right)^+ \boldsymbol{U}^T \underline{\boldsymbol{p}} \tag{3.54}$$

$$\boldsymbol{\Sigma}_{\hat{\boldsymbol{c}}} = \left(S \cdot V^T \cdot B1^T \right)^+ \boldsymbol{U}^T \boldsymbol{\Sigma}_{\boldsymbol{p}} \boldsymbol{U} \left(B1 \cdot V \cdot S^T \right)^+ \tag{3.55}$$

$$\hat{b} = B1^T \left(S \cdot V^T \cdot B1^T \right)^+ \boldsymbol{U}^T \underline{\boldsymbol{p}} \tag{3.56}$$

and for Buonocore's natural pixel basis

$$\hat{\boldsymbol{c}} = \boldsymbol{U} \left(S \cdot S^T \right)^+ \boldsymbol{U}^T \underline{\boldsymbol{p}} \tag{3.57}$$

$$\boldsymbol{\Sigma}_{\hat{\boldsymbol{c}}} = \boldsymbol{U} \left(S \cdot S^T \right)^+ \boldsymbol{U}^T \boldsymbol{\Sigma}_{\boldsymbol{p}} \boldsymbol{U} \left(S \cdot S^T \right)^+ \boldsymbol{U}^T \tag{3.58}$$

$$\hat{b} = V \cdot S^T \left(S \cdot S^T \right)^+ \boldsymbol{U}^T \underline{\boldsymbol{p}}. \tag{3.59}$$

Finally, the estimators and covariance for the orthonormal natural pixel basis are

$$\hat{\boldsymbol{c}} = \left[\left(S \cdot S^T \right)^+ \right]^{\frac{1}{2}} \boldsymbol{U}^T \underline{\boldsymbol{p}} \tag{3.60}$$

$$\boldsymbol{\Sigma}_{\hat{c}} = \left[\left(S \cdot S^T\right)^+\right]^{\frac{1}{2}} \boldsymbol{U}^T \boldsymbol{\Sigma}_{\boldsymbol{p}} \boldsymbol{U} \left[\left(S \cdot S^T\right)^+\right]^{\frac{1}{2}} \tag{3.61}$$

$$\hat{b} = V \cdot S^T \left(S \cdot S^T\right)^+ \boldsymbol{U}^T \underline{\boldsymbol{p}}. \tag{3.62}$$

Note that the least squares estimator for the mean of the continuous space distribution, $\hat{b}$, using both Buonocore's natural pixel basis and the orthonormal natural pixel basis are the same.

3.2.1 Singular Value Filtering

Small singular values in the singular value filter, $\left[\left(S \cdot S^T\right)^+\right]^{\frac{1}{2}}$, of equation 3.60 can lead to large statistical errors in the reconstructed image, $\hat{b}$. By applying a diagonal weighting matrix, $\boldsymbol{D}$, to the filter, the mean square error of the object estimates may be decreased [32]. The resulting estimates and covariance are

$$\hat{\boldsymbol{c}} = \boldsymbol{D} \left[\left(S \cdot S^T\right)^+\right]^{\frac{1}{2}} \boldsymbol{U}^T \underline{\boldsymbol{p}} \tag{3.63}$$

$$\boldsymbol{\Sigma}_{\hat{c}} = \boldsymbol{D} \left[\left(S \cdot S^T\right)^+\right]^{\frac{1}{2}} \boldsymbol{U}^T \boldsymbol{\Sigma}_{\boldsymbol{p}} \boldsymbol{U} \left[\left(S \cdot S^T\right)^+\right]^{\frac{1}{2}} \boldsymbol{D} \tag{3.64}$$

$$\hat{b} = V \cdot S^T \boldsymbol{D} \left(S \cdot S^T\right)^+ \boldsymbol{U}^T \underline{\boldsymbol{p}}. \tag{3.65}$$

Determining the weighting values is the subject of ongoing research. Since basis vectors are assumed to be arranged so the singular values are in non-increasing order, one possibility is to truncate the number of singular values used in the singular value filter so only the J largest singular values will be included. A weighting matrix with elements

$$D_{j'j} = \begin{cases} 1 & \text{if } j' = j \text{ and } j < J, \\ 0 & \text{otherwise} \end{cases} \tag{3.66}$$

will select only the J largest singular values. Since the basis is orthonormal, the resulting object estimate is the sum of the estimates of each pixel that was multiplied by one; i.e.,

$$\hat{b} = \sum_{j=0}^{J-1} B3_j^T \hat{c}_j \tag{3.67}$$

$$= \sum_{j=0}^{J-1} V \cdot S_j^T \left(S \cdot S^T \right)^+ \boldsymbol{U}^T \underline{\boldsymbol{p}}. \tag{3.68}$$

3.2.2 Noiseless Example

Returning the example of figure 3.4, the estimated pixel coefficients using the square pixel basis of figure 3.1 are found from equation 3.54 to be

$$\hat{\boldsymbol{c}} = \begin{bmatrix} \frac{5}{12} & -\frac{1}{12} & -\frac{1}{12} & \frac{5}{12} \end{bmatrix}^T. \tag{3.69}$$

In Buonocore's natural pixel basis of figure 3.2, the pixel coefficients are found from equation 3.57 to be

$$\hat{\boldsymbol{c}} = \frac{1}{72} \begin{bmatrix} 4 & 4 & 13 & -5 & 13 & -5 \end{bmatrix}^T \tag{3.70}$$

and for the orthonormal natural pixel basis shown in figure 3.3 the solution of equation 3.60 is

$$\hat{c} = \frac{\sqrt{\pi}}{4}\left[\begin{array}{cccccc} 1 & \frac{1}{\sqrt{6}} & -\frac{1}{\sqrt{2}} & 0 & 0 & 0 \end{array}\right]^T. \tag{3.71}$$

Figure 3.7 shows, respectively, the continuous space reconstruction of the wedge shaped object using square pixels, Buonocore's natural pixels, and orthonormal natural pixels with the sampling defined by the impulse response functions of figure 2.4. The object and projection estimate squared errors when using the least squares estimator are shown in table 3.2. In all three basis there is zero projection estimate error for the wedge object. However, the natural pixel and orthonormal natural pixel estimates have smaller object error than the square pixel estimates.

Basis	Object Estimate Squared Error $\left\|b - B^T\hat{c}\right\|_2^2$	Projection Estimate Squared Error $\left\|p - F \cdot B^T\hat{c}\right\|_2^2$
$B1$	$\frac{17}{12}$	0
$B2$	$\frac{7}{6}$	0
$B3$	$\frac{7}{6}$	0

Table 3.2: Least squares reconstruction errors for a wedge shaped phantom sampled without noise by a parallel beam tomographic system with three equally spaced projection angles and two projection bins at each angle.

3.2.3 Noisy Example

The previous example computations were done using a known value for the projections, i.e., no noise. To test the LSE in the presence of noise, a noisy projection

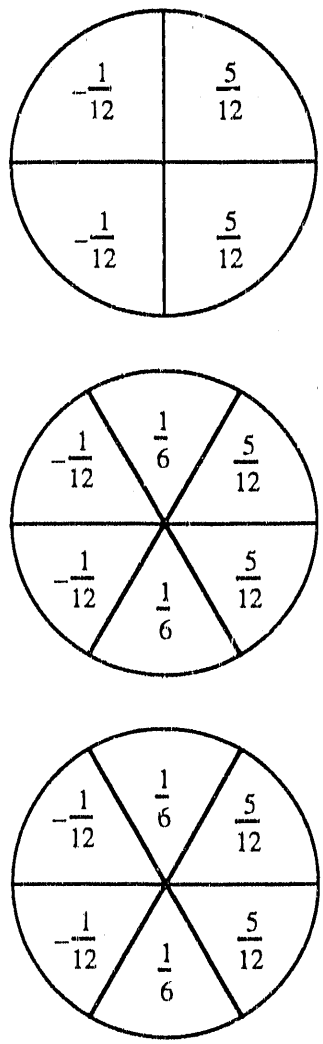


Figure 3.7: Least squares reconstruction of a wedge shaped object using top) square pixels, middle) Buonocore's natural pixels, and bottom) orthonormal natural pixels.

dataset,

$$\boldsymbol{p} = \begin{bmatrix} 0.8579 & 0.2920 & 1.4252 & 0.0000 & 0.4829 & 0.0000 \end{bmatrix}^T \tag{3.72}$$

will be used. This projection dataset was created by sampling an independent multivariate normal (Gaussian) distribution with mean and variance equal to the noiseless projections of equation 3.20. The estimated pixel coefficients and the covariance of those estimates using the square pixel basis of figure 3.1 are

$$\hat{\boldsymbol{c}} = \begin{bmatrix} 0.5508 & -0.3602 & 0.0981 & 1.0092 \end{bmatrix}^T \tag{3.73}$$

$$\boldsymbol{\Sigma}_{\hat{\boldsymbol{c}}} = \begin{bmatrix} 0.1738 & 0.0014 & -0.0854 & 0.0870 \\ 0.0014 & 0.0677 & -0.0191 & -0.0854 \\ -0.0854 & -0.0191 & 0.0677 & 0.0014 \\ 0.0870 & -0.0854 & 0.0014 & 0.1738 \end{bmatrix}. \tag{3.74}$$

In Buonocore's natural pixel basis of figure 3.2, the pixel coefficient estimates and covariance are

$$\hat{\boldsymbol{c}} = \begin{bmatrix} 0.1534 & 0.0629 & 0.5383 & -0.3220 & 0.1336 & 0.0827 \end{bmatrix}^T \tag{3.75}$$

$$\Sigma_{\hat{c}} = \begin{bmatrix} 0.1810 & -0.1771 & -0.1428 & 0.1468 & 0.1556 & -0.1517 \\ -0.1771 & 0.1810 & 0.1556 & -0.1517 & -0.1428 & 0.1468 \\ -0.1428 & 0.1556 & 0.1899 & -0.1771 & -0.1384 & 0.1512 \\ 0.1468 & -0.1517 & -0.1771 & 0.1722 & 0.1512 & -0.1561 \\ 0.1556 & -0.1428 & -0.1384 & 0.1512 & 0.1899 & -0.1771 \\ -0.1517 & 0.1468 & 0.1512 & -0.1561 & -0.1771 & 0.1722 \end{bmatrix} \quad (3.76)$$

and for the orthonormal natural pixel basis shown in figure 3.3

$$\hat{c} = \begin{bmatrix} 0.5751 & 0.6879 & -0.3641 & -0.2124 & 0.0000 & 0.0000 \end{bmatrix}^T \quad (3.77)$$

$$\Sigma_{\hat{c}} = \begin{bmatrix} 0.0556 & 0.0340 & -0.0589 & 0.0000 & 0.0000 & 0.0000 \\ 0.0340 & 0.1250 & 0.0000 & 0.0000 & 0.0000 & 0.0000 \\ -0.0589 & 0.0000 & 0.1250 & 0.0000 & 0.0000 & 0.0000 \\ 0.0000 & 0.0000 & 0.0000 & 0.5000 & 0.0000 & 0.0000 \\ 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 \\ 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 \end{bmatrix}. \quad (3.78)$$

Figure 3.8 shows, respectively, the continuous space reconstruction of the wedge shaped object using square pixels, Buonocore's natural pixels, and orthonormal natural pixels with the sampling defined by the impulse response functions of figure 2.4. Table 3.3 shows the mean and observed object squared error and the the observed projection squared error for these reconstructions. The natural pixel and orthonormal natural pixel estimates have better observed squared error than the square pixel estimates. The square pixel estimator has better mean squared error characteristics than the unfiltered estimators based on natural pixels or orthonormal natural pixels

for the wedge shaped object.

Basis	Mean Object Squared Error $E\left\|b - B^T\hat{\boldsymbol{c}}\right\|_2^2$	Observed Object Squared Error $\left\|Eb - B^T\hat{\boldsymbol{c}}\right\|_2^2$	Observed Projection Squared Error $\left\|E\boldsymbol{p} - F \cdot B^T\hat{\boldsymbol{c}}\right\|_2^2$
$B1$	1.1775	3.2636	0.3001
$B2$	1.3571	2.6294	0.2348
$B3$	1.3571	2.6294	0.2348

Table 3.3: Least squares reconstruction errors for a wedge shaped phantom sampled with noise by a parallel beam tomographic system with three equally spaced projection angles and two projection bins at each angle.

Table 3.4 shows the effects of using the diagonal weighting matrix defined in equation 3.66 with the value J varied from one to four for the example of figure 3.4. The projection estimate error decreases with the inclusion of each orthonormal natural pixel basis function; however, adding the image corresponding to the third basis vector, $J = 4$, increases the mean object squared error. The increase in mean object squared error is due to noise being added to the reconstructed image while no new information about the object is being added since the wedge phantom only has non-zero projections onto the first three orthonormal natural basis vectors as shown in equation 3.19.

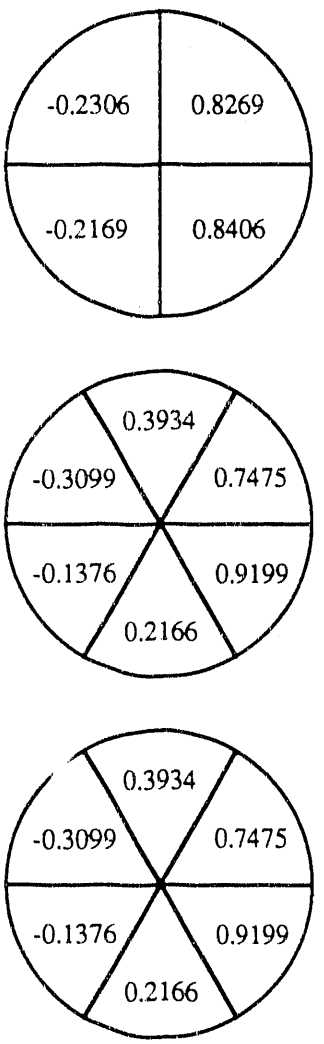


Figure 3.8: Least squares reconstruction of a wedge shaped object using top) square pixels, middle) Buonocore's natural pixels, and bottom) orthonormal natural pixels.

Truncation Index J	Mean Object Squared Error $E\left\|b - B^T\hat{\mathbf{c}}\right\|_2^2$	Observed Object Squared Error $\left\|Eb - B^T\hat{\mathbf{c}}\right\|_2^2$	Observed Projection Squared Error $\left\|E\boldsymbol{p} - F \cdot B^T\hat{\mathbf{c}}\right\|_2^2$
1	1.4762	1.9655	1.5270
2	1.3839	2.8223	0.5360
3	0.8571	2.4571	0.2584
4	1.3571	2.6294	0.2348

Table 3.4: Effects of truncating the number of singular values included in the singular value filter on orthonormal natural pixel least squares reconstruction errors for a wedge shaped phantom sampled with noise by a parallel beam tomographic system with three equally spaced projection angles and two projection bins at each angle.

3.3 Normal Maximum Likelihood

The least squares estimator in the previous section did not use any information about the variance of the measurements. In this section, it is assumed that the projection measurements are samples from multi-normal random variables with known variance. The likelihood for the measurements is given by

$$L(\boldsymbol{c}) \equiv \Pr\{\boldsymbol{p}|\boldsymbol{c}\} \tag{3.79}$$

$$= (2\pi)^{-\frac{\Theta K}{2}} \left|\boldsymbol{\Sigma}_{\boldsymbol{p}}\right|^{-\frac{1}{2}} \exp\left[-\frac{1}{2}\left(\boldsymbol{p} - F \cdot B^T\boldsymbol{c}\right)^T \boldsymbol{\Sigma}_{\boldsymbol{p}}^{-1}\left(\boldsymbol{p} - F \cdot B^T\boldsymbol{c}\right)\right]. \tag{3.80}$$

The normal maximum likelihood estimator (NMLE) can be found by maximizing this expression or alternatively the log-likehood,

$$
\begin{aligned}
l(\boldsymbol{c}) &\equiv \log\{L(\boldsymbol{c})\} && (3.81)\\
&= -\frac{\Theta K}{2}\log(2\pi) - \frac{1}{2}\log\left|\boldsymbol{\Sigma_p}\right| \\
&\quad -\frac{1}{2}\left(\boldsymbol{p} - F\cdot B^T\boldsymbol{c}\right)^T \boldsymbol{\Sigma_p}^{-1}\left(\boldsymbol{p} - F\cdot B^T\boldsymbol{c}\right), && (3.82)
\end{aligned}
$$

since the natural logarithm is a monotonically increasing function. Derivation of the NMLE follows.

$$
\begin{aligned}
\hat{\boldsymbol{c}} &\equiv \arg\max_{\boldsymbol{c}} l(\boldsymbol{c}) && (3.83)\\
&= \arg\max_{\boldsymbol{c}}\left\{-\frac{\Theta K}{2}\log(2\pi) - \frac{1}{2}\log\left|\boldsymbol{\Sigma_p}\right|\right. \\
&\quad \left. -\frac{1}{2}\left(\boldsymbol{p} - F\cdot B^T\boldsymbol{c}\right)^T \boldsymbol{\Sigma_p}^{-1}\left(\boldsymbol{p} - F\cdot B^T\boldsymbol{c}\right)\right\} && (3.84)
\end{aligned}
$$

As was done for the least squares estimator, the gradient vector with respect to unknown parameters, $\boldsymbol{c}$, is computed and equated with the zero vector, yielding a $\Theta K \times \Theta K$ system of simultaneous linear equations

$$
\hat{\boldsymbol{c}} = \arg\left\{\left(F\cdot B^T\right)^T \boldsymbol{\Sigma_p}^{-1}\left(\boldsymbol{p} - F\cdot B^T\boldsymbol{c}\right) = 0\right\}. \qquad (3.85)
$$

In normal form, the system is

$$
B\cdot F^T\boldsymbol{\Sigma_p}^{-1}F\cdot B^T = B\cdot F^T\boldsymbol{\Sigma_p}^{-1}\boldsymbol{p}. \qquad (3.86)
$$

Applying the pseudoinverse of the the normal matrix yields

$$\hat{\boldsymbol{c}} = \left(B \cdot F^T \boldsymbol{\Sigma}_{\boldsymbol{p}}^{-1} F \cdot B^T\right)^+ B \cdot F^T \boldsymbol{\Sigma}_{\boldsymbol{p}}^{-1} \boldsymbol{p} \tag{3.87}$$

which is equivalent to

$$\hat{\boldsymbol{c}} = \left(\boldsymbol{\Sigma}_{\boldsymbol{p}}^{-\frac{1}{2}} F \cdot B^T\right)^+ \boldsymbol{\Sigma}_{\boldsymbol{p}}^{-\frac{1}{2}} \boldsymbol{p}. \tag{3.88}$$

The estimator shown in equation 3.88 is commonly called a *weighted least squares estimator* because it is equivalent to the least squares method with the model and measurements preweighted by the inverse of the concomitant standard deviation.

The covariance for the normal maximum likelihood estimator is

$$\boldsymbol{\Sigma}_{\hat{\boldsymbol{c}}} = \left(B \cdot F^T \boldsymbol{\Sigma}_{\boldsymbol{p}}^{-1} F \cdot B^T\right)^+ \tag{3.89}$$

For the Heavyside basis, the estimators and the covariance become

$$\hat{\boldsymbol{c}} = \left(\boldsymbol{\Sigma}_{\boldsymbol{p}}^{-\frac{1}{2}} F \cdot B1^T\right)^+ \boldsymbol{\Sigma}_{\boldsymbol{p}}^{-\frac{1}{2}} \underline{\boldsymbol{p}} \tag{3.90}$$

$$\boldsymbol{\Sigma}_{\hat{\boldsymbol{c}}} = \left(B1 \cdot F^T \boldsymbol{\Sigma}_{\boldsymbol{p}}^{-1} F \cdot B1^T\right)^+ \tag{3.91}$$

$$\hat{b} = B1^T \left(S \cdot V^T \cdot B1^T\right)^+ \boldsymbol{U}^T \underline{\boldsymbol{p}} \tag{3.92}$$

and for the natural pixel basis

$$\hat{\boldsymbol{c}} = \boldsymbol{U} \left[S \cdot S^T \boldsymbol{U}^T \boldsymbol{\Sigma}_{\boldsymbol{p}}^{-1} \boldsymbol{U} S \cdot S^T\right]^+ S \cdot S^T \boldsymbol{U}^T \boldsymbol{\Sigma}_{\boldsymbol{p}}^{-1} \underline{\boldsymbol{p}} \tag{3.93}$$

$$\boldsymbol{\Sigma}_{\hat{\boldsymbol{c}}} = \boldsymbol{U}\left[S \cdot S^T \boldsymbol{U}^T \boldsymbol{\Sigma}_{\boldsymbol{p}}^{-1} \boldsymbol{U} S \cdot S^T\right]^+ \boldsymbol{U}^T \tag{3.94}$$

$$\hat{b} = V \cdot S^T \left[S \cdot S^T \boldsymbol{U}^T \boldsymbol{\Sigma}_{\boldsymbol{p}}^{-1} \boldsymbol{U} S \cdot S^T\right]^+ S \cdot S^T \boldsymbol{U}^T \boldsymbol{\Sigma}_{\boldsymbol{p}}^{-1} \underline{\boldsymbol{p}}. \tag{3.95}$$

The normal maximum likelihood estimators and covariance for the orthonormal natural pixel basis are

$$\hat{\boldsymbol{c}} = \left[\left(S \cdot S^T\right)^{\frac{1}{2}} \boldsymbol{U}^T \boldsymbol{\Sigma}_{\boldsymbol{p}}^{-1} \boldsymbol{U} \left(S \cdot S^T\right)^{\frac{1}{2}}\right]^+ \left(S \cdot S^T\right)^{\frac{1}{2}} \boldsymbol{U}^T \boldsymbol{\Sigma}_{\boldsymbol{p}}^{-1} \underline{\boldsymbol{p}} \tag{3.96}$$

$$\boldsymbol{\Sigma}_{\hat{\boldsymbol{c}}} = \left[\left(S \cdot S^T\right)^{\frac{1}{2}} \boldsymbol{U}^T \boldsymbol{\Sigma}_{\boldsymbol{p}}^{-1} \boldsymbol{U} \left(S \cdot S^T\right)^{\frac{1}{2}}\right]^+ \tag{3.97}$$

$$\hat{b} = V \cdot S^T \left[\left(S \cdot S^T\right)^+\right]^{\frac{1}{2}} \left[\left(S \cdot S^T\right)^{\frac{1}{2}} \boldsymbol{U}^T \boldsymbol{\Sigma}_{\boldsymbol{p}}^{-1} \boldsymbol{U} \left(S \cdot S^T\right)^{\frac{1}{2}}\right]^+ \left(S \cdot S^T\right)^{\frac{1}{2}} \boldsymbol{U}^T \boldsymbol{\Sigma}_{\boldsymbol{p}}^{-1} \underline{\boldsymbol{p}}. \tag{3.98}$$

When the system is non-singular, the matrix $S \cdot S^T$ is invertible and the natural pixel estimator and orthonormal natural pixel estimator will not depend on the covariance of the projections. The same is true when the projection data acquisition process is homoscedastic; i.e., the covariance matrix is a constant times the identity matrix. In these cases, the least squares solution and the normal maximum likelihood solutions are equivalent. But, most importantly, the estimators can be precomputed and applied to multiple projection datasets for time-series analysis of dynamic processes. Singularities in the system response will be considered further in chapter 4.

3.4 Poisson Maximum Likelihood

In emission tomography, the expected number of photons counted in a sampling time interval may not be large enough to justify approximating the measurement process

as multi-normal. When the count rate is low, a more realistic model is to assume the measurements are samples from a multi-variate Poisson random distribution. The measurements are assumed to be statistically independent and the resulting likelihood function is

$$L(\boldsymbol{c}) = \prod_{\theta k} \frac{\exp\left(-F_{\theta k} \cdot B^T \boldsymbol{c}\right)\left(F_{\theta k} \cdot B^T \boldsymbol{c}\right)^{p_{\theta k}}}{p_{\theta k}!} \tag{3.99}$$

As in section 3.3, the Poisson maximum likelihood estimator (PMLE) can be found by maximizing this expression or alternatively the log-likehood,

$$l(\boldsymbol{c}) = \sum_{\theta k}\left[F_{\theta k} \cdot B^T \boldsymbol{c} - \log p_{\theta k} + p_{\theta k} \log\left(F_{\theta k} \cdot B^T \boldsymbol{c}\right)\right] \tag{3.100}$$

over all possible image vectors, $\boldsymbol{c}$; i.e.,

$$\hat{\boldsymbol{c}} \equiv \arg\max_{\boldsymbol{c}} l(\boldsymbol{c}) \tag{3.101}$$

$$= \arg\max_{\boldsymbol{c}}\left\{\sum_{\theta k}\left[F_{\theta k} \cdot B^T \boldsymbol{c} - \log p_{\theta k} + p_{\theta k} \log\left(F_{\theta k} \cdot B^T \boldsymbol{c}\right)\right]\right\}. \tag{3.102}$$

Computing the gradient vector with respect to the unknown parameters, $\boldsymbol{c}$, and equating it with the zero vector results in a set of ΘK non-linear equations

$$\hat{\boldsymbol{c}} = \arg\left\{\sum_{\theta k}\left[F_{\theta k} \cdot B^T\left(1 - \frac{p_{\theta k}}{F_{\theta k} \cdot B^T \boldsymbol{c}}\right)\right] = 0\right\}. \tag{3.103}$$

There are no known closed form solutions for the Poisson maximum likelihood estimator (PMLE) of equation 3.103. Instead, an iterative solution is used. A technique

by Dempster [30] known as expectation maximization (EM) was first applied to tomography by Shepp and Vardi [21]. Their algorithm has been extended to generalized pixels as

$$\hat{\mathbf{c}}^{(t+1)} = \hat{\mathbf{c}}^{(t)} \frac{1}{\sum_{\theta k} B \cdot F_{\theta k}^T} \sum_{\theta k} B \cdot F_{\theta k}^T \frac{p_{\theta k}}{F_{\theta k} \cdot B^T \hat{\mathbf{c}}^{(t)}}. \tag{3.104}$$

The iteration step is denoted by t. Under appropriate conditions [44], the covariance matrix for the generalized pixel estimates is approximated by

$$\boldsymbol{\Sigma}_{\hat{\mathbf{c}}} \approx \left[E \frac{-\partial^2 l(\mathbf{c})}{\partial c_{m'n'} \partial c_{mn}} \right]^+ \tag{3.105}$$

$$= \left(B \cdot F^T \boldsymbol{Q}^{-1} F \cdot B^T \right)^+ \tag{3.106}$$

where $\boldsymbol{Q}$ is a diagonal matrix given in tensor form by

$$Q_{\theta' k' \theta k} = F_{\theta k} \cdot B^T \hat{\mathbf{c}} \; \delta_{\theta' \theta} \delta_{k' k}. \tag{3.107}$$

For the Heavyside basis, the Poisson maximum likelihood estimators and the covariance become

$$\hat{\mathbf{c}}^{(t+1)} = \hat{\mathbf{c}}^{(t)} \frac{1}{\sum_{\theta k} B1 \cdot F_{\theta k}^T} \sum_{\theta k} B1 \cdot F_{\theta k}^T \frac{p_{\theta k}}{F_{\theta k} \cdot B1^T \hat{\mathbf{c}}^{(t)}} \tag{3.108}$$

$$\boldsymbol{\Sigma}_{\hat{\mathbf{c}}} = \left(B1 \cdot F^T \boldsymbol{Q}^{-1} F \cdot B1^T \right)^+ \tag{3.109}$$

$$Q_{\theta' k' \theta k} = F_{\theta k} \cdot B1^T \hat{\mathbf{c}} \; \delta_{\theta' \theta} \delta_{k' k} \tag{3.110}$$

and for the natural pixel basis

$$\hat{\boldsymbol{c}}^{(t+1)} = \hat{\boldsymbol{c}}^{(t)} \frac{1}{\sum_{\theta k} \boldsymbol{U} S \cdot S^T \boldsymbol{U}_{\theta k}^T} \sum_{\theta k} \boldsymbol{U} S \cdot S^T \boldsymbol{U}_{\theta k}^T \frac{p_{\theta k}}{\boldsymbol{U}_{\theta k} S \cdot S^T \boldsymbol{U}^T \hat{\boldsymbol{c}}^{(t)}} \quad (3.111)$$

$$\boldsymbol{\Sigma}_{\hat{\boldsymbol{c}}} = \boldsymbol{U} \left(S \cdot S^T \boldsymbol{U}^T \boldsymbol{Q}^{-1} \boldsymbol{U} S \cdot S^T \right)^+ \boldsymbol{U}^T \quad (3.112)$$

$$Q_{\theta' k' \theta k} = \boldsymbol{U}_{\theta k} S \cdot S^T \boldsymbol{U}^T \hat{\boldsymbol{c}} \; \delta_{\theta' \theta} \delta_{k' k}. \quad (3.113)$$

The Poisson maximum likelihood estimators and covariance for the orthonormal natural pixel basis are

$$\hat{\boldsymbol{c}}^{(t+1)} = \hat{\boldsymbol{c}}^{(t)} \frac{1}{\sum_{\theta k} \boldsymbol{U} \left(S \cdot S^T \right)_{\theta k}^{\frac{1}{2}}} \sum_{\theta k} \boldsymbol{U} \left(S \cdot S^T \right)_{\theta k}^{\frac{1}{2}} \frac{p_{\theta k}}{\boldsymbol{U}_{\theta k} \left(S \cdot S^T \right)^{\frac{1}{2}} \hat{\boldsymbol{c}}^{(t)}} \quad (3.114)$$

$$\boldsymbol{\Sigma}_{\hat{\boldsymbol{c}}} = \left[\left(S \cdot S^T \right)^{\frac{1}{2}} \boldsymbol{U}^T \boldsymbol{Q}^{-1} \boldsymbol{U} \left(S \cdot S^T \right)^{\frac{1}{2}} \right]^+ \quad (3.115)$$

$$Q_{\theta' k' \theta k} = \boldsymbol{U}_{\theta k} \left(S \cdot S^T \right)^{\frac{1}{2}} \hat{\boldsymbol{c}} \; \delta_{\theta' \theta} \delta_{k' k}. \quad (3.116)$$

3.5 Separable Inversion

In section 2.1.1, a separable model for projection formation was proposed. Estimators for the unknown spatial distribution for systems that are separable have special properties that lead to computationally fast and efficient implementations. Substituting the model of equation 2.22 into the generalized pixel least squares estimator in equation 3.43 yields

$$\hat{\boldsymbol{c}} = \left(B \cdot H^T \cdot F_0^T \boldsymbol{G}^T \boldsymbol{G} F_0 \cdot H \cdot B^T \right)^+ B \cdot H^T \cdot F_0^T \boldsymbol{G}^T \underline{\boldsymbol{p}}. \quad (3.117)$$

When the basis set chosen to represent the object has the property that $B^T B$ is the product of delta functions, e.g., the basis is composed of unique point samples, the least squares estimator becomes

$$\hat{\boldsymbol{c}} = \left(B \cdot H^T \cdot B^T B \cdot F_0^T \boldsymbol{G}^T \boldsymbol{G} F_0 \cdot B^T B \cdot H \cdot B^T\right)^+ B \cdot H^T \cdot B^T B \cdot F_0^T \boldsymbol{G}^T \underline{\boldsymbol{p}}. \tag{3.118}$$

If the matrix $B \cdot H \cdot B^T$ which is a pixelized version of the object blurring operator, H, is invertible, then

$$\hat{\boldsymbol{c}} = \left(B \cdot H \cdot B^T\right)^{-1} \left(B \cdot F_0^T \boldsymbol{G}^T \boldsymbol{G} F_0 \cdot B^T\right)^+ B \cdot F_0^T \boldsymbol{G}^T \underline{\boldsymbol{p}}. \tag{3.119}$$

Haber's [45] spatially invariant method to correct for positron range blurring in PET with no projection blurring, i.e., $\boldsymbol{G}$ is an identity matrix, is an example of this least squares technique even though it was not expressed formally as equation 3.119.

Additionally, if the projection blurring matrix, $\boldsymbol{G}$, is invertible, an estimate for the mean of the generalized pixel representation of the object distribution is

$$\hat{\boldsymbol{c}} = \left(B \cdot H \cdot B^T\right)^{-1} \left(B \cdot F_0^T F_0 \cdot B^T\right)^+ B \cdot F_0^T \boldsymbol{G}^{-1} \underline{\boldsymbol{p}} \tag{3.120}$$

and the covariance of those estimates is

$$\begin{aligned} \boldsymbol{\Sigma}_{\hat{\boldsymbol{c}}} = & \left(B \cdot H \cdot B^T\right)^{-1} \left(B \cdot F_0^T F_0 \cdot B^T\right)^+ B \cdot F_0^T \boldsymbol{G}^{-1} \boldsymbol{\Sigma}_{\boldsymbol{p}} \\ & \left(\boldsymbol{G}^{-1}\right)^T F_0 \cdot B^T \left(B \cdot F_0^T F_0 \cdot B^T\right)^+ \left[\left(B \cdot H \cdot B^T\right)^{-1}\right]^T. \end{aligned} \tag{3.121}$$

This suboptimal estimator has many desirable computation properties. The matrix G^{-1} can be precomputed and is sometimes rotationally invariant. G^{-1} is applied to a projection dataset which is then processed by a discretized Radon inversion algorithm such as filtered backprojection to form an image. The resulting image is then filtered by the inverse of the pixelized version of H.

3.6 Simulation studies

A series of simulations was performed on a cylindrical phantom, shown in figure 3.9. The phantom has non-zero constant intensity inside a circle with diameter $\sqrt{2}$ and is zero outside that circle. The cylindrical phantom studies were performed to compute regional bias and squared error between the estimate of the mean of the intensity and the true mean of the intensity. In these studies, a noiseless set of projections was created using the parallel beam tomograph model described in figure 2.5. The simulated projections for the cylindrical phantom were then scaled to a fixed image intensity. Two noisy sets of projections were created using either a normal or Poisson pseudo-random number generator with the mean and variance set to the value of the noiseless projections [46]. These three sets of projections were then reconstructed using the least squares estimator of section 3.2 using square pixels, natural pixels, and orthonormal natural pixels which are shown in figure 3.10. The continuous space representation of the cylindrical phantom using orthonormal natural pixels is shown in figure 3.11. The simulation and reconstructions for the noisy projections were repeated for 16 sample paths of the pseudo-random number generator. The results

presented are averages of all samples.

The continuous space representation of the square pixel truncated least squares estimate, equation 3.56 and equation 3.66, for the mean of the intensity of the cylindrical phantom as the truncation index or equivalently the number of generalized pixels reconstructed is varied is shown in figure 3.12. Figure 3.13 shows the continuous space representation of the truncated least squares estimate, equation 3.68, for the mean of the intensity of the cylindrical phantom using orthonormal natural pixels as the truncation index is varied. The images in both figure 3.12 and figure 3.13 were reconstructed using the are averages of 16 reconstructions of projection data with variance parameter equal to 16. The images are ordered from left to right and top to bottom to correspond with the standard non-decreasing order of singular values; the image at the top left corresponds to the largest singular value and the image at the bottom right corresponds to the smallest singular value. Each image has been scaled to give the largest contrast within the image; therefore, information about relative intensity between images has been lost.

Figure 3.14 shows the observed bias,

$$Eb - B^T\hat{c}, \tag{3.122}$$

for the reconstructed images of figure 3.12 and figure 3.13. Since the original projection model has little spatial variance, the normalized bias is small for both pixelizations. However, the normalized bias for the orthonormal natural pixel least squares estimate is about an order of magnitude smaller than the square pixel estimate. For

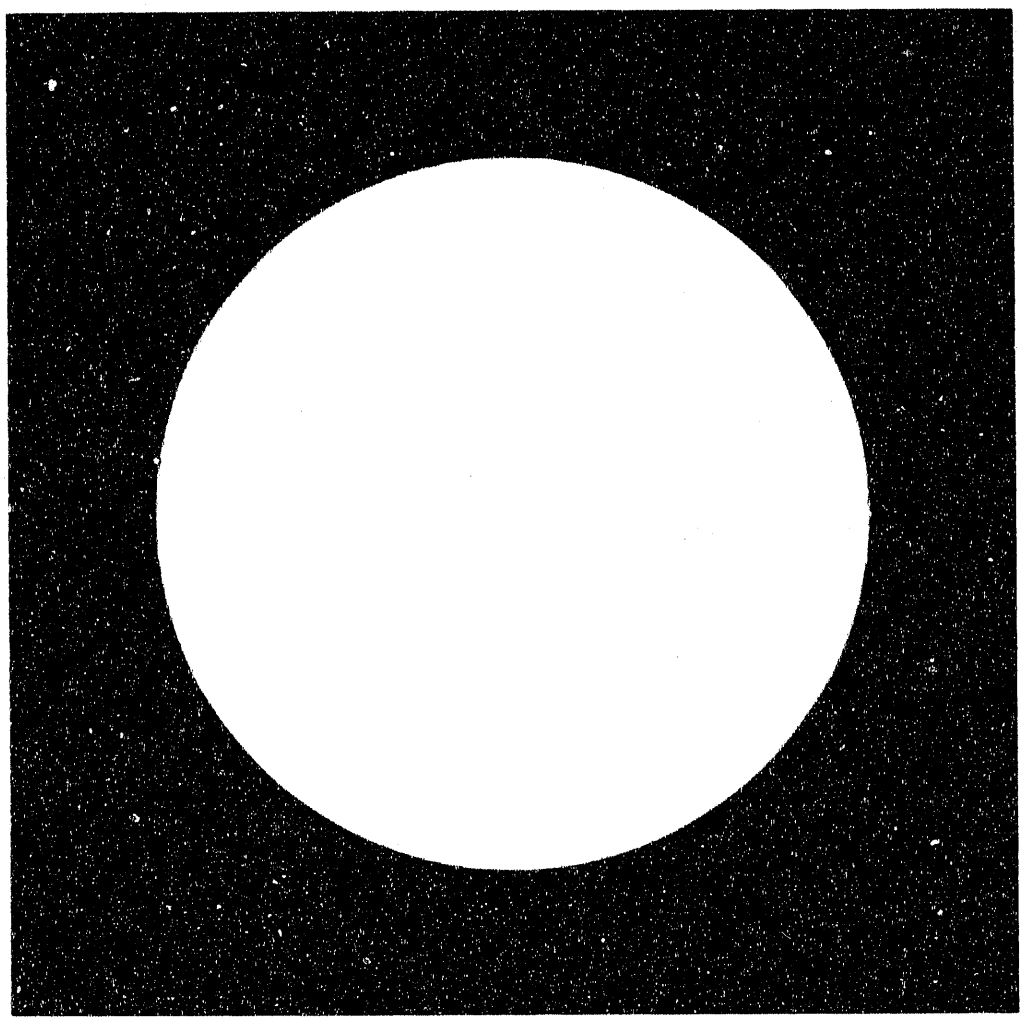


Figure 3.9: Cylindrical phantom used for simulations.

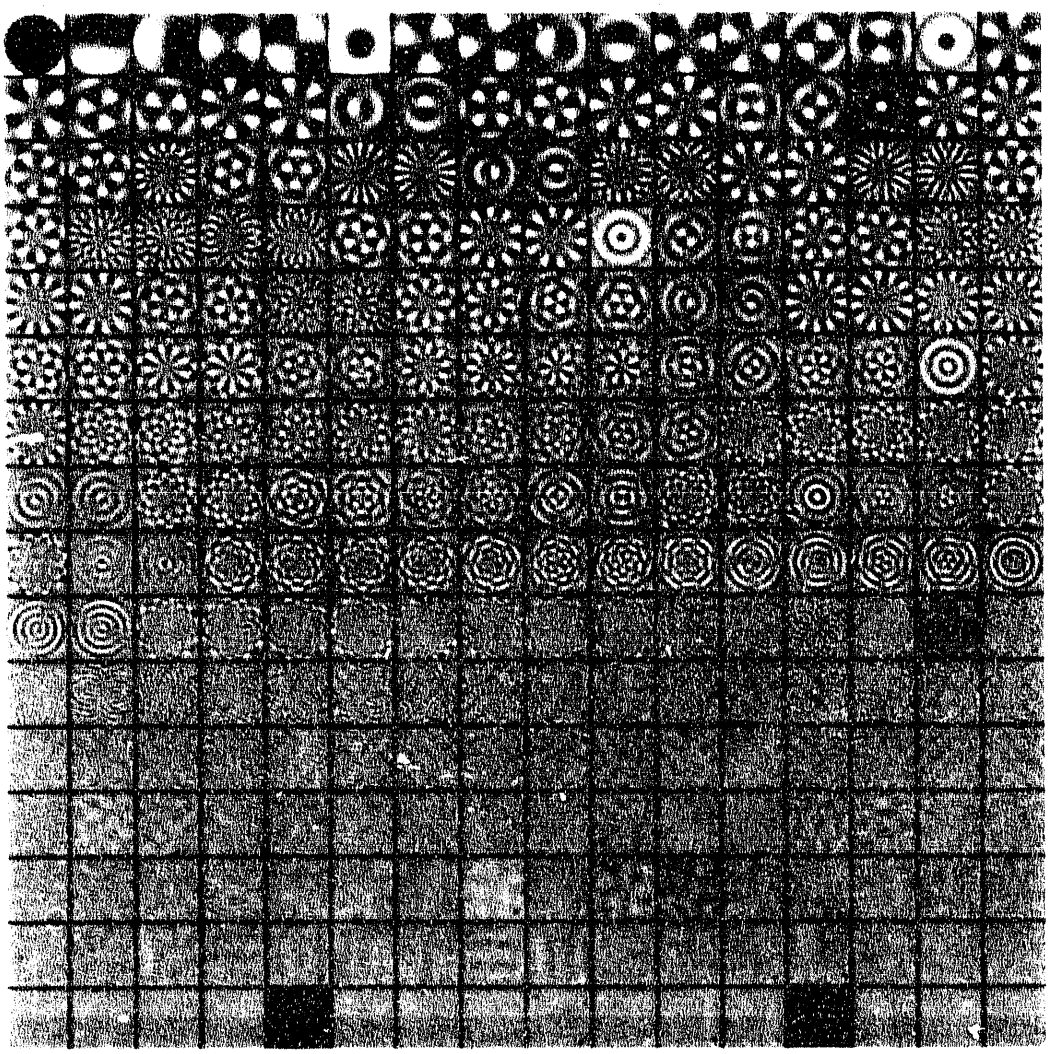


Figure 3.10: New orthonormal pixel basis for a parallel beam tomographic system with 16 equally spaced projection angles and 16 projection bins at each angle. The pixels have been arranged to correspond to a non-increasing ordering of the the singular values from left to right and top to bottom.

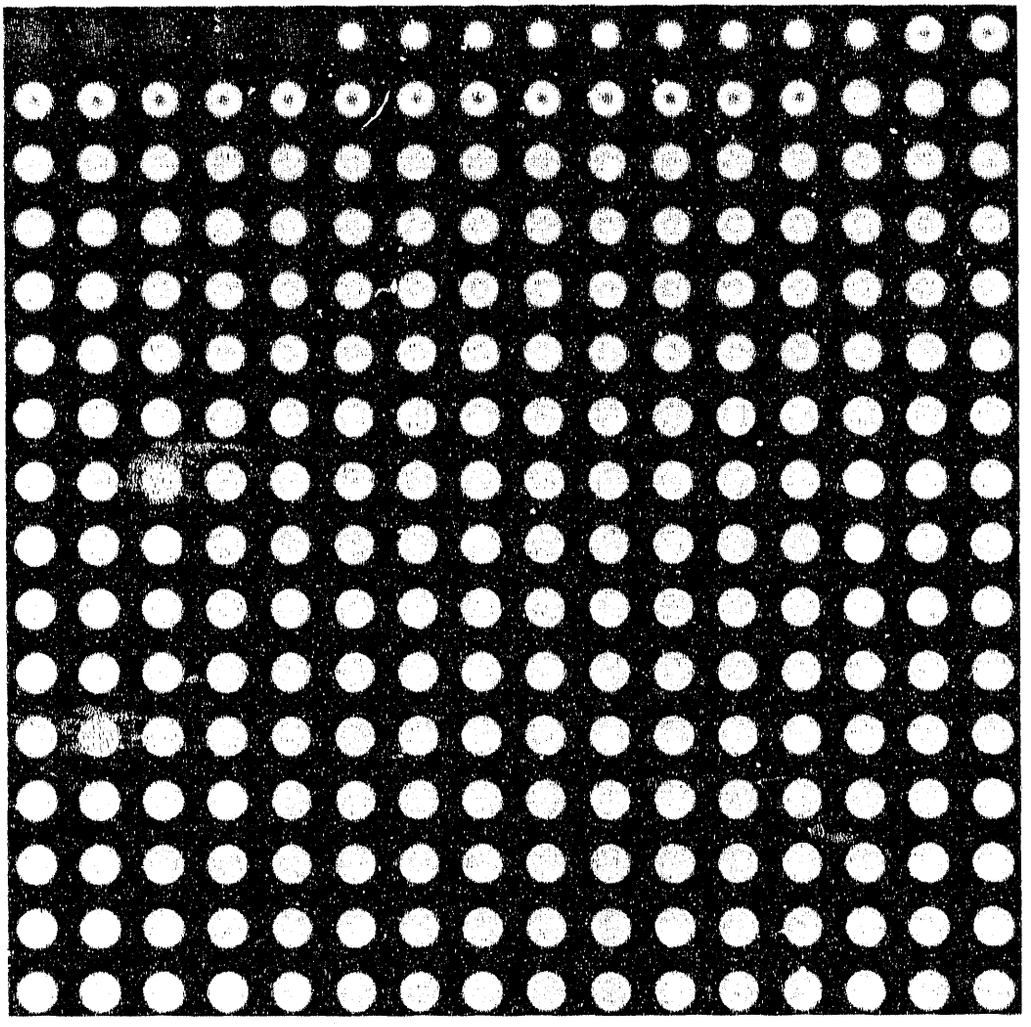


Figure 3.11: Representation of a cylindrical phantom using the orthonormal natural pixels shown in figure 3.10 as more pixels are used to represent the object.

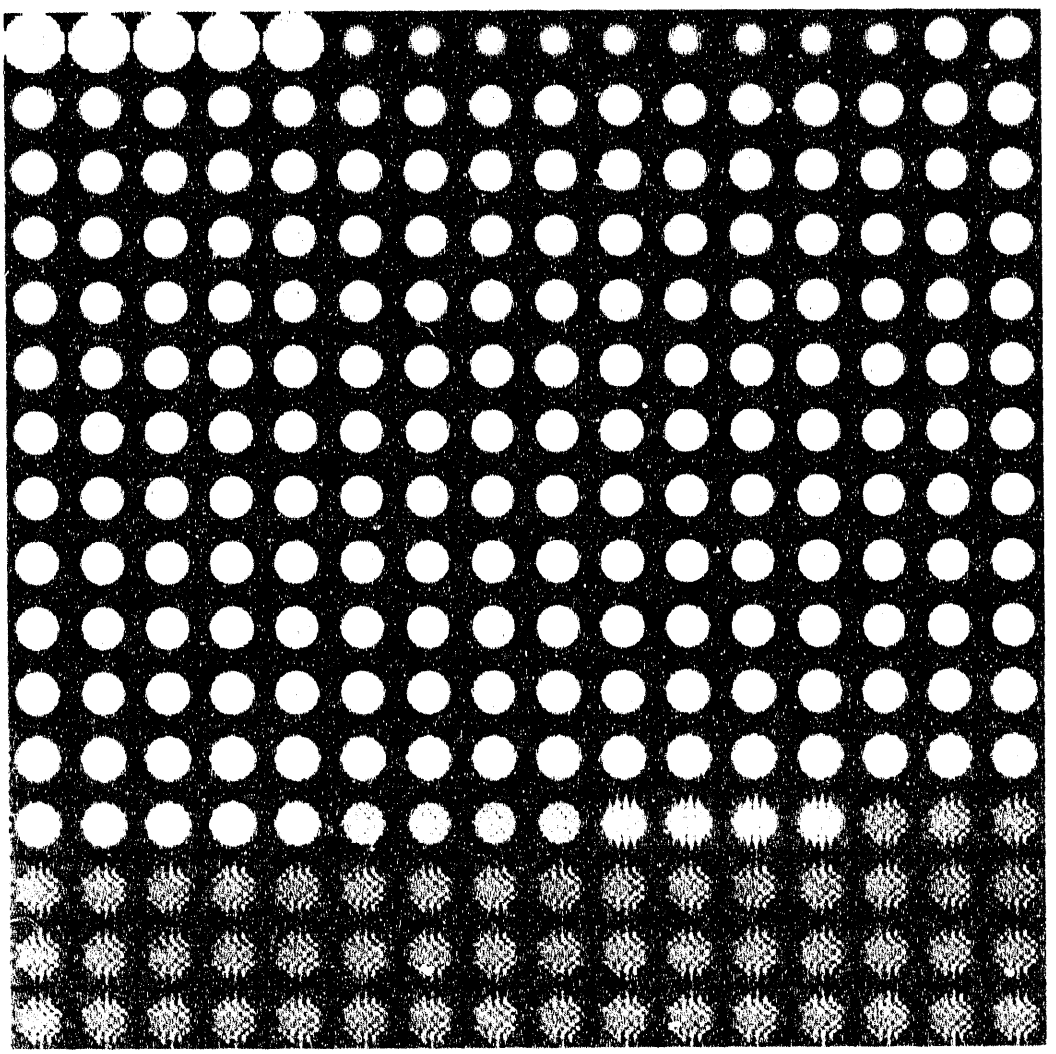


Figure 3.12: Square pixel least squares reconstruction from noisy projections of cylindrical object. The number of generalized pixels used in the reconstruction increases left to right and top to bottom.

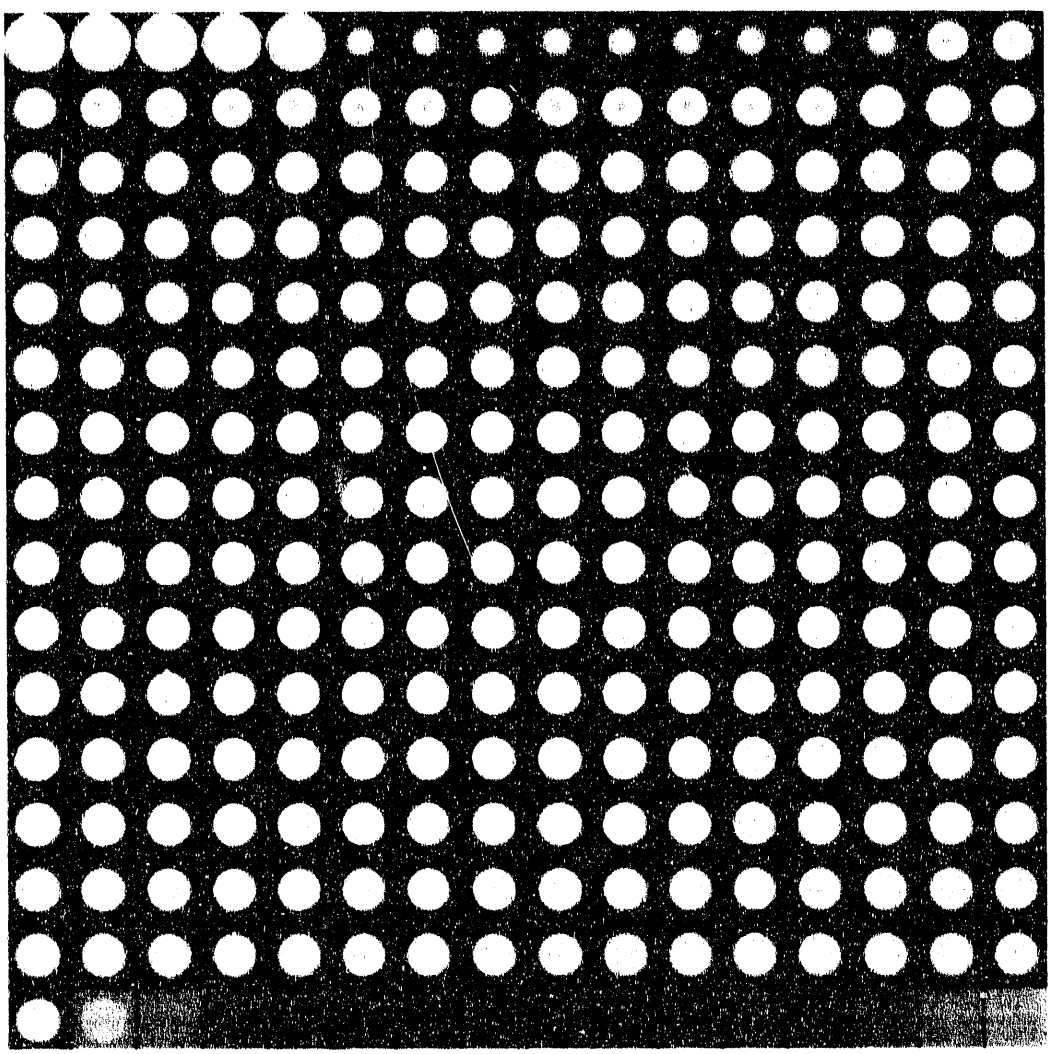


Figure 3.13: Orthonormal natural pixel least squares reconstruction from noisy projections of cylindrical object. The number of generalized pixels used in the reconstruction increases left to right and top to bottom.

tomography systems that have more spatial variance, the observed bias is generally more pronounced.

The effects of using the diagonal singular value filter matrix defined by equation 3.66 on the projection sum of squared errors,

$$\left\| E\boldsymbol{p} - F \cdot B^T \hat{\boldsymbol{c}} \right\|_2^2, \tag{3.123}$$

are shown in figure 3.15 and on the object integral of squared errors,

$$\left\| Eb - B^T \hat{\boldsymbol{c}} \right\|_2^2, \tag{3.124}$$

are shown in figure 3.16. The solid line corresponds to a noiseless simulation; the other three curves are for simulations that used a projection dataset sampled from a multivariate normal distribution with variance parameter, as defined in equation 4.8, equal to 1 (dotted line), 4 (long dashed line), and 16 (short dashed line). In this context, the variance parameter is equivalent to a signal to noise ratio for the projection measurements. Each curve was normalized by the total expected object intensity squared to compare simulations with different object intensities.

The projection error decreases with the inclusion of each orthonormal pixel basis function independent of noise level. However, for square pixels, the projection error will increase with the inclusion of more generalized pixel basis functions with the most rapid increase at high noise levels, i.e., small variance parameter. For the noiseless simulation, the projection error for square pixels does not become zero because there is systematic error in the representation of the cylindrical object in the square

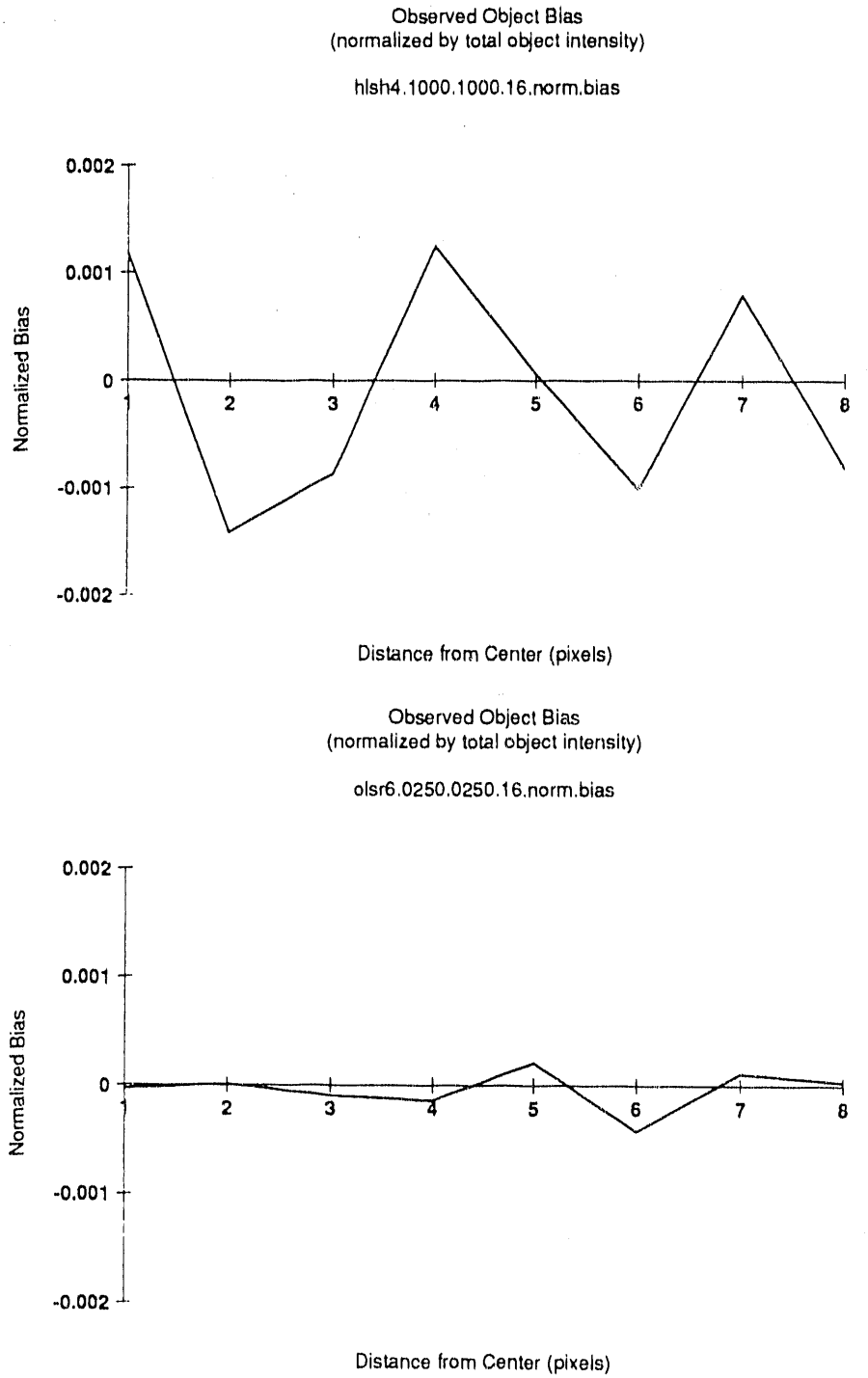


Figure 3.14: Observed bias of least squares estimate of cylindrical object versus distance from center using upper) square pixels and lower) orthonormal natural pixels.

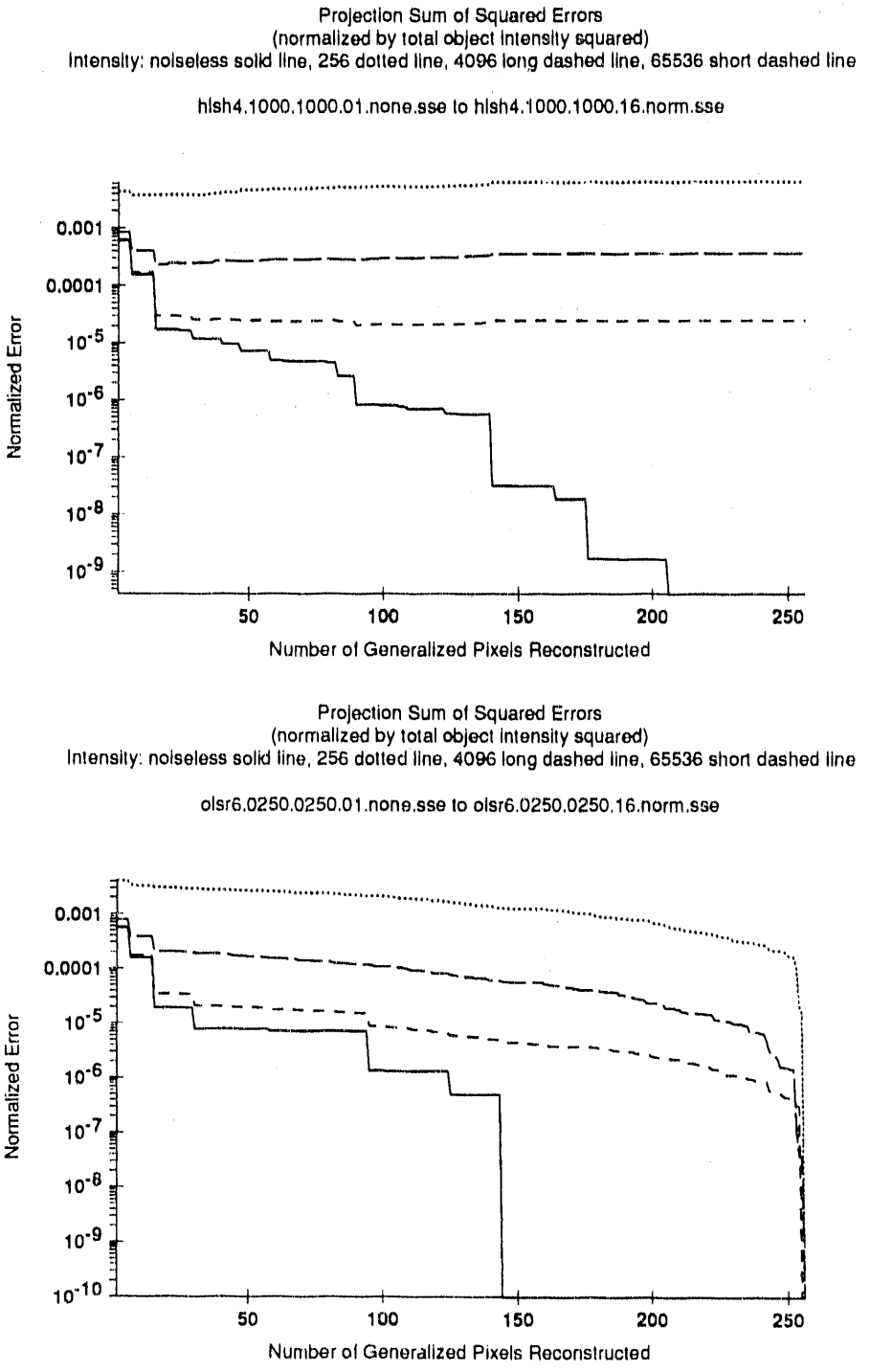


Figure 3.15: Sum of squared differences between the projection of the least squares estimate and the projection of cylindrical object versus number of generalized pixels reconstructed as the expected intensity is varied using upper) square pixels and lower) orthonormal natural pixels.

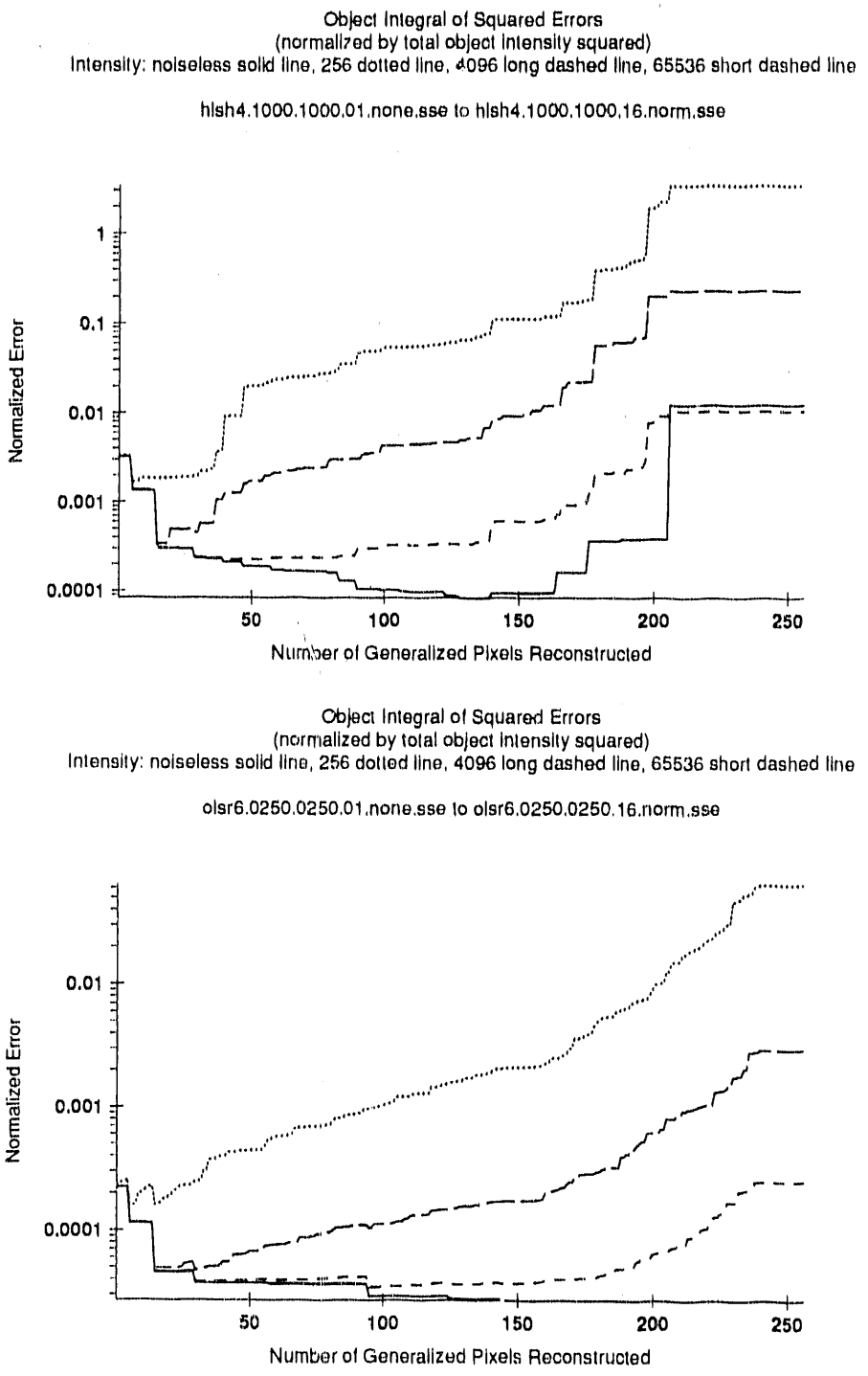


Figure 3.16: Integral of squared differences between the least squares estimate and cylindrical object versus number of generalized pixels reconstructed as the expected intensity is varied using upper) square pixels and lower) orthonormal natural pixels.

pixel basis. For display purposes only, a lower limit of 10^{-10} was imposed on the projection sum of squared errors for orthonormal natural pixels in figure 3.16. The projection sum of squared errors was 10^{-28} when 256 orthonormal natural pixels were reconstructed; i.e., there is very little systematic reprojection error using orthonormal natural pixels.

The average of the object integral of squared errors decreases and then increases for both pixelizations in the noisy simulations as more generalized pixels are included in the reconstruction. The increase in object squared error is due to statistical noise being added to the reconstructed image while no new information about the object is being added. An additional systematic error is present in the square pixel reconstruction and since this error is not stochastic, it will also be present in noiseless simulations. When using square pixels, the object error decreases and then begins to increase rapidly as the truncation index is increased in the noiseless simulation. By using orthonormal natural pixels, the systematic error due to the mathematical model is eliminated as evidenced by the non-increasing object integral of squared errors for the noiseless simulation. With noise present, reconstructions using either pixelization undergo a rapid increase in object integral of squared error beyond a particular truncation index that corresponds to fixed resolution. A detailed discussion of the dramatic increase in the object error and its relationship to statistical fluctuations in the measured projection data is in chapter 4.

When datasets sampled from a Poisson distribution were used, similar results were observed. Typically, the number of generalized pixels needed to observed a rapid increase in object error is somewhat less and the projection sum of squared

errors is almost always greater.

3.7 Experimental studies

An example of the identification procedure used for a positron tomograph follows. Attenuation in the tissue due to photon absorption was measured by performing a transmission experiment. In this experiment, a positron emitter is orbited around the object and a set of projections was measured with the position of the source known [47]. A projection dataset was acquired without the object present, e.g, in air, to compute detector efficiency. A reconstruction of the attenuation coefficients was performed as described by Huesman *et al* [12]. It was assumed that attenuation of photons within the object varies slowly such that attenuation can be considered constant across a projection ray and therefore a line integral approximation was justified. Also, the statistical quality of these measurements was assumed to be such that the fluctuations of F do not need to be considered during emission reconstruction.

Crystal penetration was studied under the assumption of separability as described in section 2.1.1 [2]. Stochastic and deterministic simulations were used to determine the projection blurring matrix $\boldsymbol{G}$ of equation 2.22. A lower triangular circulant block form for $\boldsymbol{G}$ was obtained by arranging the elements of the projection vector, $\boldsymbol{p}$, according to the distance from the center of the tomograph and by ordering the elements within each block according to angle. $\boldsymbol{G}$ was inverted using a fast algorithm that uses both the lower triangular and the circulant block properties. This inverse was applied to an attenuation and detector efficiency corrected emission dataset of the 37

point hot spot phantom, shown in figure 1.1, acquired using the Donner 600-Crystal Positron Tomograph [47]. The resulting projection dataset was reconstructed using the backprojection of filtered projections algorithm and the reconstructed image is shown in the lower part of figure 1.2. Points that are on the outer edge are clearly improved over those shown in the upper image of figure 1.2 which have not been corrected for crystal penetration.

Also visible on the penetration compensated image are artifacts due to the correction procedure. These artifacts arise from the fact that in the stochastic simulation of the projection blurring matrix, $\boldsymbol{G}$, the elements of $\boldsymbol{G}$ are computed as average transition probabilities between two projection bins and the measured projections of a point source will have different transition probabilities than the average. The method appears to amplify statistical noise a great deal in order to gain a modest improvement in radial resolution. Further work to characterize the statistical correlations between pixel estimates is under way.

Chapter 4

Optimization

The generalized pixel estimators in chapter 3 were derived for a tomograph with known system response, F. It was shown that the fluctuations of these estimators due to random variations in the measurements can be characterized in terms of the covariance between estimates. In this chapter, a technique to characterize the effects angular and lateral sampling density have on reconstructed image resolution and noise is developed and used to examine an example system. These results lead to a method to optimize sampling strategies given bounds for noise and resolution. The optimization procedure uses a criterion that is a function of the covariance matrix.

4.1 Covariance

The least squares, normal maximum likelihood, and Poisson maximum likelihood generalized pixel estimators for the mean of the unknown distribution, b, have asymptotically a covariance matrix

$$\Sigma_{\hat{\mathbf{c}}} = \bar{p}\left(B \cdot F^T F \cdot B^T\right)^+ . \tag{4.1}$$

if the measurements are assumed to be independent and homoscedastic with variance $\bar{p}$.

To bound the statistical error associated with the reconstruction process, the L_2 norm of the covariance matrix is computed using induced norms. To compute the induced norm, a unit vector $\mathbf{c}$ is multiplied by the covariance matrix and the norm of the resulting vector is calculated. The length of the unit vector is scaled up or down and the largest scale factor is the induced norm. Symbolically, the operations are

$$\|\Sigma_{\hat{\mathbf{c}}}\|_2 \equiv \max_{\|\mathbf{c}\|_2=1} \|\Sigma_{\hat{\mathbf{c}}}\mathbf{c}\|_2 . \tag{4.2}$$

Substitution of equation 4.1 into the definition of induced norm yields

$$\|\Sigma_{\hat{\mathbf{c}}}\|_2 = \max_{\|\mathbf{c}\|_2=1} \left\| \bar{p}\left(B \cdot F^T F \cdot B^T\right)^+ \mathbf{c} \right\|_2 \tag{4.3}$$

which for the orthonormal natural pixel basis, $B3$ of equation 3.11 is

$$\|\Sigma_{\hat{\mathbf{c}}}\|_2 = \max_{\|\mathbf{c}\|_2=1} \left\| \bar{p}\left\{ \left[\left(S \cdot S^T\right)^+\right]^{\frac{1}{2}} S \cdot V^T \cdot F^T F \cdot V \cdot S^T \left[\left(S \cdot S^T\right)^+\right]^{\frac{1}{2}} \right\}^+ \mathbf{c} \right\|_2 \tag{4.4}$$

After simplifying the terms of the pseudo-inverse by substituting the singular value decomposition of the projection formation operator,

$$\|\Sigma_{\hat{c}}\|_2 = \max_{\|c\|_2=1} \left\| \bar{p} \left(S \cdot S^T \right)^+ c \right\|_2 . \tag{4.5}$$

The induced norm of a diagonal matrix is equal to the maximum of the absolute value of the matrix elements; therefore,

$$\|\Sigma_{\hat{c}}\|_2 = \max_i \bar{p} \frac{1}{(S \cdot S^T)_{ii}} \tag{4.6}$$

and since the maximum of a reciprocal is the reciprocal of the minimum

$$\|\Sigma_{\hat{c}}\|_2 = \frac{\bar{p}}{\min_i (S \cdot S^T)_{ii}} . \tag{4.7}$$

From the induced norm, it can be seen that the bound on the noise amplification during reconstruction is inversely proportional to the square of the smallest singular value of the projection formation operator. Thus, it is necessary to find the square of the singular values of F to compute the error bound. This differs from the deterministic error propagation approach where the noise amplification is the ratio of the largest singular value to the smallest singular value; i.e., the condition number [14] [18].

4.2 Simulation studies

A parallel beam tomographic system with equally spaced projection angles and lateral projection bins similar to the one in figure 2.5 was simulated to examine the effects

of angular and lateral sampling density on reconstruction. The number of angles sampled, Θ, was varied from 1 to 80 while keeping the number of lateral sampling bins, K, fixed at 16 in one set of simulations. A family of plots of the norm of the covariance matrix versus the number of generalized pixels reconstructed or equivalently the truncation index, J, is shown in figure 4.1. Recall that the truncation index defined in equation 3.66 is the index of the smallest singular value included in the reconstruction; thus, according to equation 4.7, the norm of the covariance matrix will be equal to the inverse of the singular value of the projection normal matrix at the truncation index. To compare the simulations with a different number of angular samples, the variance parameter, $\bar{p}$, was varied such that all the simulations are noise equivalent; i.e.,

$$\mathrm{Tr}\left(\boldsymbol{\Sigma_p}\right) \;=\; \Theta K \bar{p} \;=\; \text{constant}. \tag{4.8}$$

Therefore, $\bar{p}$ is inversely proportional to the total number of projection measurements, ΘK, and the covariance norm plots have been divided by the value of ΘK used in the simulation.

When the number of projection angles is 16 or greater the moving average of the norm of the covariance matrix increases almost linearly with a slope of approximately 0.6 on a log-log plot as the truncation index increases until there is a more rapid increase at truncation index 150. In other simulations, not shown here, that had 8 and 24 lateral sampling bins, a similar increase in the moving average of the norm of the covariance matrix with approximate slope 0.6 and then a rapid increase is observed

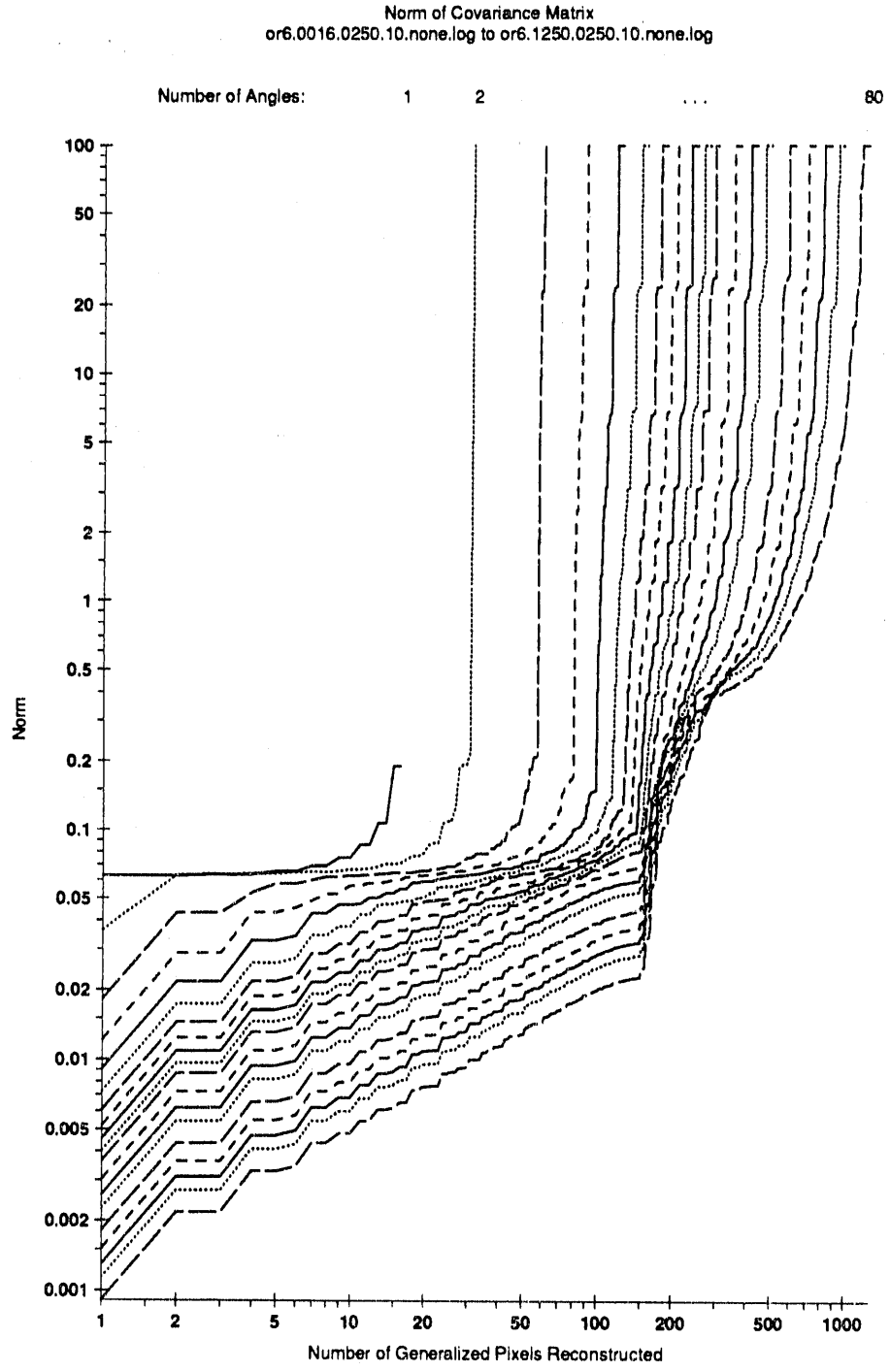


Figure 4.1: Norm of covariance matrix versus number of generalized pixels reconstructed for a set of parallel beam tomographic systems with $\Theta = \{$ 1, 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 24, 28, 32, 40, 48, 56, 64, 80 $\}$ equally spaced projection angles and $K = 16$ projection bins at each angle.

when the number of projection angles is greater than or equal to the number of lateral projection bins. Note that norm of the covariance matrix changes most significantly over only a three order of magnitude range implying that numerical error from the robust technique used to compute the singular values should be negligible.

There is a correspondence between the point where the moving average of the norm begins to rapidly increase and the total number of projections measurements. The analysis used to obtain equation 2.21 where the number of linearly independent projection measurements was found to be $\Theta(K-1)+1$ was based upon the zeroth spatial moment of the mean of the object distribution. By considering the higher order spatial moments of the mean of the object distribution using a technique similar to that proposed by Ein-Gal *et al* [48], there will be $\Theta-1$ degrees of freedom lost for the zeroth moment when each angle is paired with one angle, $\Theta-2$ degrees of freedom are lost for the first moment when each angle is mated with two angles, and so on up to the $K-1$ moment. Thus, there are approximately

$$\Theta K - \sum_{k=0}^{K-1} (\Theta - k - 1) = \frac{K^2+K}{2} \tag{4.9}$$

measurements that are independent in the zero through $K-1$ moments. This analysis relies upon continuous lateral sampling and is therefore only a lower bound on the number of independent measurements for discrete lateral sampling.

The important feature of figure 4.1 is for reconstruction spatial resolutions coarser than a well defined resolution threshold, increasing the number of angles sampled will decrease the noise in images reconstructed from data acquired by a system that has

more angular samples than the number of lateral samples. For resolutions finer than that resolution threshold, very little improvement in resolution is obtained without a large increase image noise. For $K = 16$, the lower bound from equation 4.9 is 136 generalized pixels and the number observed from simulations is 150 generalized pixels.

Huesman [49] was one of the first to realize the implications of finite angular and lateral sampling on statistical reconstruction noise. He performed a detailed study of these effects using a projection model based on parallel line integrals sampling a circular grid of square pixels. His analysis suggests the number of projection angles should be 1.5 D/d and the number of lateral samples should be 2.0 D/d to maintain an appropriate bound on reconstruction noise amplification. D is the linear dimension of the reconstruction region and d is the linear dimension of the cells into which the reconstruction region is subdivided (resolution length); therefore, the number of square pixels should be $\frac{\pi}{12}\Theta K$. For $K = 16$, Huesman would predict 67 square pixels. This bound is much more conservative than the bound from the moment analysis or the bound from the simulations of figure 4.1. It is possible that while trying to study only statistical errors, Huesman was also experiencing significant deterministic errors due to line integral model sampling of square pixels and the numerical instability of the matrix inversion technique being used. These deterministic errors may have influenced his choice for the angular and lateral sampling bounds.

Figure 4.2 shows a family of plots of the norm of the covariance matrix versus the truncation index, J, for a set of simulations where the number of lateral sampling bins was varied from 1 to 80 while keeping the number of angles sampled fixed at 16. When the number of laterally sampled bins is less than or equal to the number of

angles sampled, a sharp increase in the moving average of the norm of the covariance matrix is again observed above the index predicted by equation 4.9; e.g., for $K = 8$, moment analysis yields 36 generalized pixels and the value from simulation is 38 generalized pixels. When the number of laterally sampled bins is greater than the number of angles sampled, there is a less abrupt change in the slope of the covariance norm moving average.

By increasing the number of lateral projection bins, resolution and reconstructed image noise are always improved. However, for any angular and lateral sampling density, there is a resolution threshold beyond which improvements in resolution are only achieved with a large increase in image noise. Furthermore, the number of generalized pixels corresponding to that resolution threshold are far fewer than the total number of projection measurements.

To elaborate the utility of this method to optimize sampling, consider a parallel beam tomographic system with equally spaced angular and lateral sampling that has the number of angular samples constrained to be $\Theta = 16$. The desired resolution for a reconstructed image is 100 pixels. From figure 4.2, the number of lateral sampling bins should be $K = 14$ to obtain a reasonable bound on reconstructed image noise.

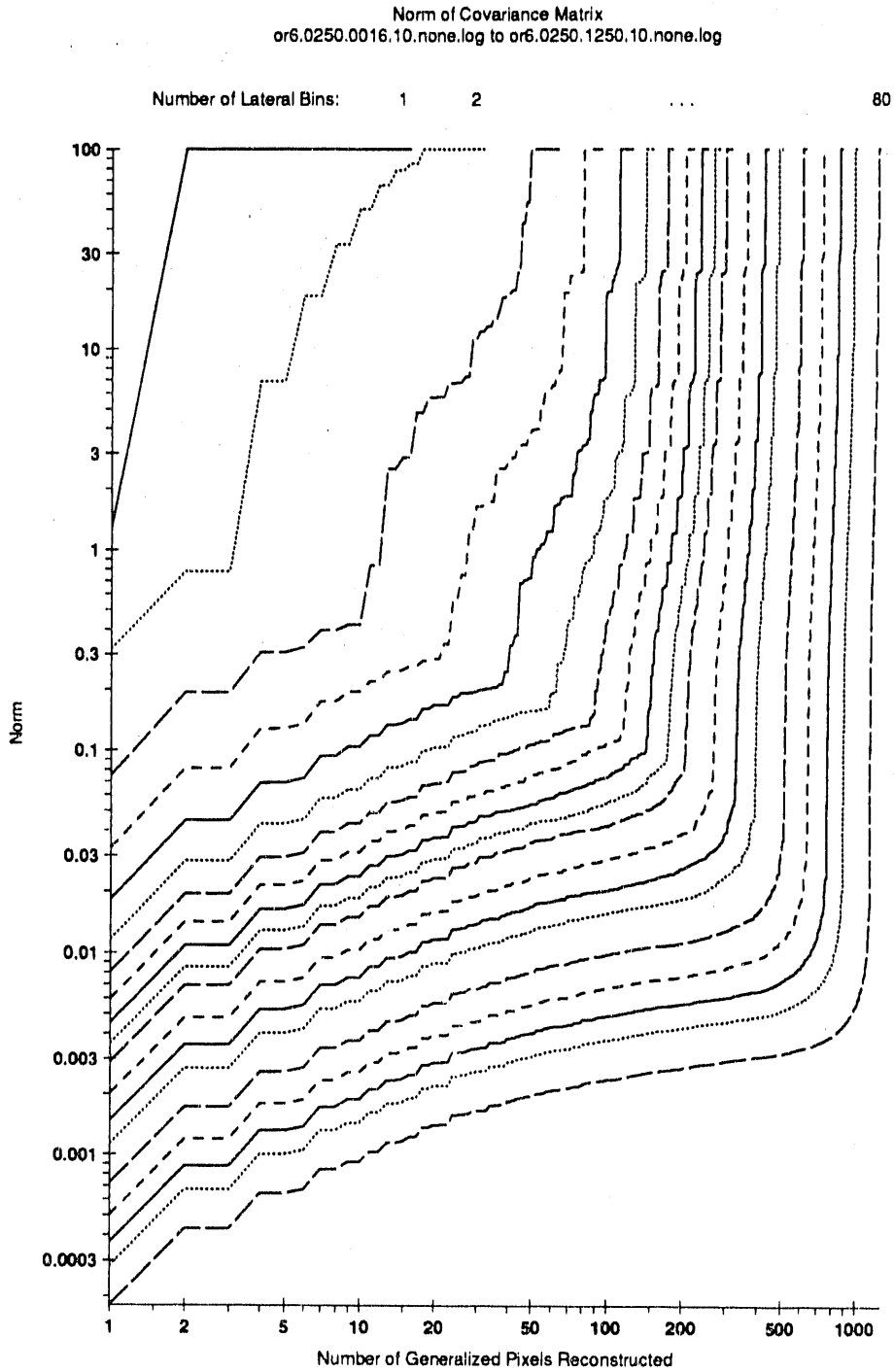


Figure 4.2: Norm of covariance matrix versus number of generalized pixels reconstructed for a set of parallel beam tomographic systems with $\Theta = 16$ equally spaced projection angles and $K = \{$ 1, 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 24, 28, 32, 40, 48, 56, 64, 80 $\}$ projection bins at each angle.

Chapter 5

Implementation of Algorithms

5.1 Orthonormal Least Squares

The continuous space representation of the truncated minimum L_2 norm least squares estimator for the mean value of the orthonormal natural pixel image is from equation 3.68

$$\hat{b} = \sum_{j=0}^{J-1} V \cdot S_j^T \left(S \cdot S^T\right)^+ \boldsymbol{U}^T \underline{\boldsymbol{p}}. \tag{5.1}$$

This image is converted to a square pixel representation by performing the operations of equation 3.4 using the Heavyside basis operator, $B1$, of equation 3.9. The resulting square pixel representation is

$$\hat{c} = \sum_{j=0}^{J-1} B1 \cdot V \cdot S_j^T \left(S \cdot S^T\right)^+ \boldsymbol{U}^T \underline{\boldsymbol{p}} \tag{5.2}$$

and after substituting equation 3.12 into equation 5.2,

$$\hat{c} = \sum_{j=0}^{J-1} B1 \cdot F^T \boldsymbol{U} \left(S \cdot S^T \right)^+_{jj} \boldsymbol{U}^T \underline{\boldsymbol{p}} \tag{5.3}$$

In general, the computation of the projection normal matrix requires $\Theta(\Theta^2 K^2)$ integral evaluations[1] [2]. An $\Omega(\Theta^3 K^3)$ singular value or eigenvalue decomposition routine is used to compute $\boldsymbol{U}$ and $S \cdot S^T$. After the singular value decomposition of the projection normal matrix is known, evaluation of equation 5.3 uses $O(\Theta^2 K^2)$ operations.

For a rotationally invariant system, described in section 2.1.2, the evaluation of the projection normal matrix requires only $\Theta(\Theta K^2)$ integral evaluations and its singular value decomposition requires $\Omega(\Theta K^3)$ operations using the block circulant singular value decomposition algorithm described in section 5.2. The rotationally invariant

[1] Let $n, n_0 \in \mathbf{N}$ and $\epsilon \in \mathbf{R}, \epsilon > 0$. Also, $f, g : \mathbf{N} \rightarrow \mathbf{R}$. Then, define [51]

1. Upper bound

$$O(f(n)) \equiv \{g(n) : g(n) \leq \epsilon f(n) \, \forall \, n > n_0\}$$

2. Lower bound

$$\Omega(f(n)) \equiv \{g(n) : g(n) \geq \epsilon f(n) \, \forall \, n > n_0\}$$

3. Combined bound

$$\Theta(f(n)) \equiv O(f(n)) \cap \Omega(f(n))$$

4. Asymptotic

$$f(n) \sim g(n) \Leftrightarrow \lim_{n \rightarrow \infty} \frac{f(n)}{g(n)} = 1$$

[2] Using the symbol Θ for the number of projection angles measured and the combined bound function $\Theta(\cdot)$ is somewhat confusing but parenthesis distinguish between the two uses.

orthonormal least squares estimator is computationally tractable on current computer systems. The separable estimators discussed in section 3.5 also have similar computational properties.

5.1.1 Implementation

The C programming language [50] function call

$$\text{orthols}\ (\hat{\mathbf{c}}, F \cdot B1^T, B1 \cdot F^T, \text{svd}(A), \left(S \cdot S^T\right)^{\frac{1}{2}}_{JJ}, integration\_factor)$$

will compute the estimator of equation 5.3. The source code for a sequential implementation is given in section B.1 and a block diagram of the algorithm is shown in figure 5.1. The first argument is a vector of square pixel image amplitudes that will be computed by the function. The second and third arguments are, respectively, the projector and backprojector functions which are compatible with the RECLBL [12] reconstruction software. The fourth argument is a routine that will return the singular value decomposition of the projection normal matrix, A; an example C language routine is given in section B.2. The fifth argument is the minimum singular value to be used in the reconstruction and the last argument is an integration factor as defined by RECLBL.

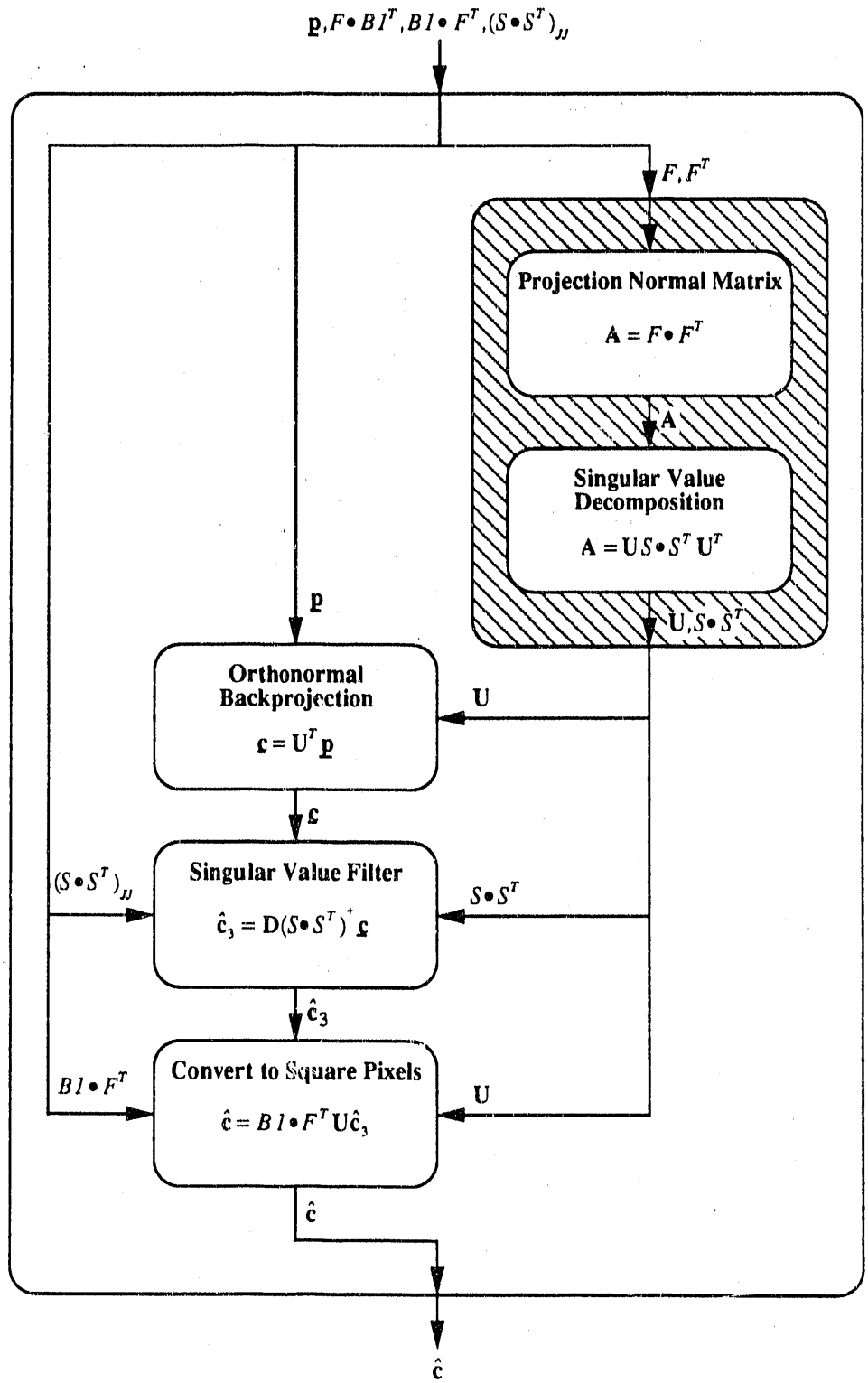


Figure 5.1: Block diagram of orthonormal natural pixel least squares estimation algorithm.

5.2 Block Circulant Singular Value Decomposition

A special class of matrices have the block circulant structure shown in equation 5.4. There are $M \times M$ blocks each of dimension $m \times n$. This form of matrix arises quite frequently when a function is invariant under rotation. As an example for the rest of this section, the case where $M = m = n = 64$ shall be used because it is representative of typical SPECT reconstructions.

$$\boldsymbol{A} = \begin{bmatrix} \boldsymbol{A}_0 & \boldsymbol{A}_1 & \boldsymbol{A}_2 & .. & \boldsymbol{A}_{M-2} & \boldsymbol{A}_{M-1} \\ \boldsymbol{A}_{M-1} & \boldsymbol{A}_0 & \boldsymbol{A}_1 & .. & \boldsymbol{A}_{M-3} & \boldsymbol{A}_{M-2} \\ \boldsymbol{A}_{M-2} & \boldsymbol{A}_{M-1} & \boldsymbol{A}_0 & .. & \boldsymbol{A}_{M-4} & \boldsymbol{A}_{M-3} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ \boldsymbol{A}_2 & \boldsymbol{A}_3 & \boldsymbol{A}_4 & .. & \boldsymbol{A}_0 & \boldsymbol{A}_1 \\ \boldsymbol{A}_1 & \boldsymbol{A}_2 & \boldsymbol{A}_3 & .. & \boldsymbol{A}_{M-1} & \boldsymbol{A}_0 \end{bmatrix} \tag{5.4}$$

A $\Theta(mnM \log M)$ fast Fourier transform (FFT) technique [15] [52] and an $\Omega(M \min(m,n)mn)$ singular value decomposition (SVD) algorithm are used to compute the factorization [53]

$$\boldsymbol{A} = (\mathcal{F}_M \otimes \boldsymbol{I}_m)^\dagger \, \boldsymbol{D} \, (\mathcal{F}_M \otimes \boldsymbol{I}_n) \tag{5.5}$$

$$= (\mathcal{F}_M \otimes \boldsymbol{I}_m)^\dagger \, \boldsymbol{U_D S_D V_D^\dagger} \, (\mathcal{F}_M \otimes \boldsymbol{I}_n) \tag{5.6}$$

$$= \boldsymbol{U S_D V^\dagger} \tag{5.7}$$

$$= \boldsymbol{U S V^\dagger} \tag{5.8}$$

where $\mathcal{F}_M$ is a normalized $M \times M$ discrete Fourier operator matrix, $\boldsymbol{I}_m$ is an $m \times m$ identity matrix, and $\boldsymbol{I}_n$ is an $n \times n$ identity matrix. $\boldsymbol{U}$ and $\boldsymbol{V}$ are unitary matrices whose columns are respectively the left and right singular vectors of $\boldsymbol{A}$.[3] $\boldsymbol{S}$ is a generalized diagonal matrix containing the singular values of $\boldsymbol{A}$. The operator † is conjugate transpose and $\otimes$ is the outer product operation. A proof is given in section A.2.

5.2.1 Implementation

The C language function call

$$\text{zbcsvdc}\ (\boldsymbol{A}, M, m, n, \boldsymbol{S}, \boldsymbol{U_D}, \boldsymbol{V_D}, \mathit{compute\_uv}, \mathit{status})$$

is used to compute the singular value decomposition of $\boldsymbol{A}$ that is shown in equation 5.6. The source code for a sequential implementation is given in section B.3.

Each of the mn discrete Fourier transforms of equation 5.6 can be computed independently; i.e., each sum does not need the result or input of another sum. The SVD of the blocks of $\boldsymbol{D}$ also do not have input/output dependencies with other blocks and can be computed without explicit synchronization. Therefore, parallel processing implementations of the block circulant singular value decomposition algorithm are possible.

Two parallel versions of the BCSVD algorithm were implemented and tested on a Cray-2 supercomputer using macrotasks [53]. The $\Theta(M \log M)$ grain size of FFT

[3]The matrix $\boldsymbol{A}$ of equation 5.4 is more general than the one defined by equation 2.26 because the matrix in equation 5.4 has rectangular blocks and complex elements. Also, the symbols $\boldsymbol{U}$, $\boldsymbol{V}$, and $\boldsymbol{S}$ have different meanings than in equation 2.7.

tasks is extremely small. For the example, it takes about 0.45 ms [54] [55]. This is comparable to the 0.31 ms necessary to synchronize with a server process and is much smaller than the 2.63 ms necessary to create a new process. It is thus advantageous to increase the grain size of FFT tasks by computing m FFTs per task. The resulting granularity of $\Theta(mM \log M)$ is about 29 ms. The task granularity of an SVD process is $\Omega(\min(m,n)mn)$ which is 428 ms for the example problem.

A prescheduled algorithm was impler ented by creating one process for each of the n FFT tasks and another processes for each of the M SVD tasks. The parent task starts k processes with either an FFT or an SVD task. All of the k processes run to completion before another k processes are started. This method is very easy to implement because all synchronization is implicit in the fork and join like paradigm [56].

To overcome the process creation overhead, a self-scheduling algorithm was constructed [57]. This method is more complex than the prescheduled algorithm but has a smaller time overhead. It requires explicit synchronization between server processes and a task manager. k server processes are created and each waits for a start signal after initial setup of local state information. After receiving the start signal from the task manager, a server checks what part of the matrix it is to work on next. When finished the server sends a ready signal to the hibernating manager. The manager then reassigns each of the server processes until the task queue is empty.

5.2.2 Results

Figure 5.2 shows the computation time for different sizes of input matrices. The speedup of the algorithm, shown in figure 5.3, increases as the size of M, m, and n are increased. The prescheduled algorithm is faster for very small matrix sizes because the self-scheduled algorithm server processes have a larger startup overhead than a process started by the prescheduled algorithm. The self-scheduled algorithm is faster for medium sized problems that have small grain sizes but the prescheduled algorithm again approaches the speedup of self-scheduling as the problem size increases.

The efficiency, shown in figure 5.4, does not approach unity as quickly as expected. This might be attributed to the timesharing scheduling algorithm used by the CTSS operating system and not to synchronization overhead because the overhead, shown in table 5.1, is less than 1.0% for M, m, and n larger than 64 [58] [59]. It was not possible to verify this conjecture by using the machine without other users present.

$M = m = n$	prescheduled		self-scheduled	
	overhead (ms)	% overhead	overhead (ms)	% overhead
4	25.7		29.9	
8	46.8	42.77	32.3	5.09
16	80.2	10.89	37.3	1.29
32	155.7	2.66	47.2	0.32
64	306.8	0.61	67.1	0.07

Table 5.1: Synchronization overhead versus problem size.

The process creation time was found to be 2.63 ms. Task synchronization in the self-scheduling algorithm was 0.31 ms. A typical procedure call was measured to take 4.7 μs. Self-scheduling has less time overhead than prescheduling but is still 66 times

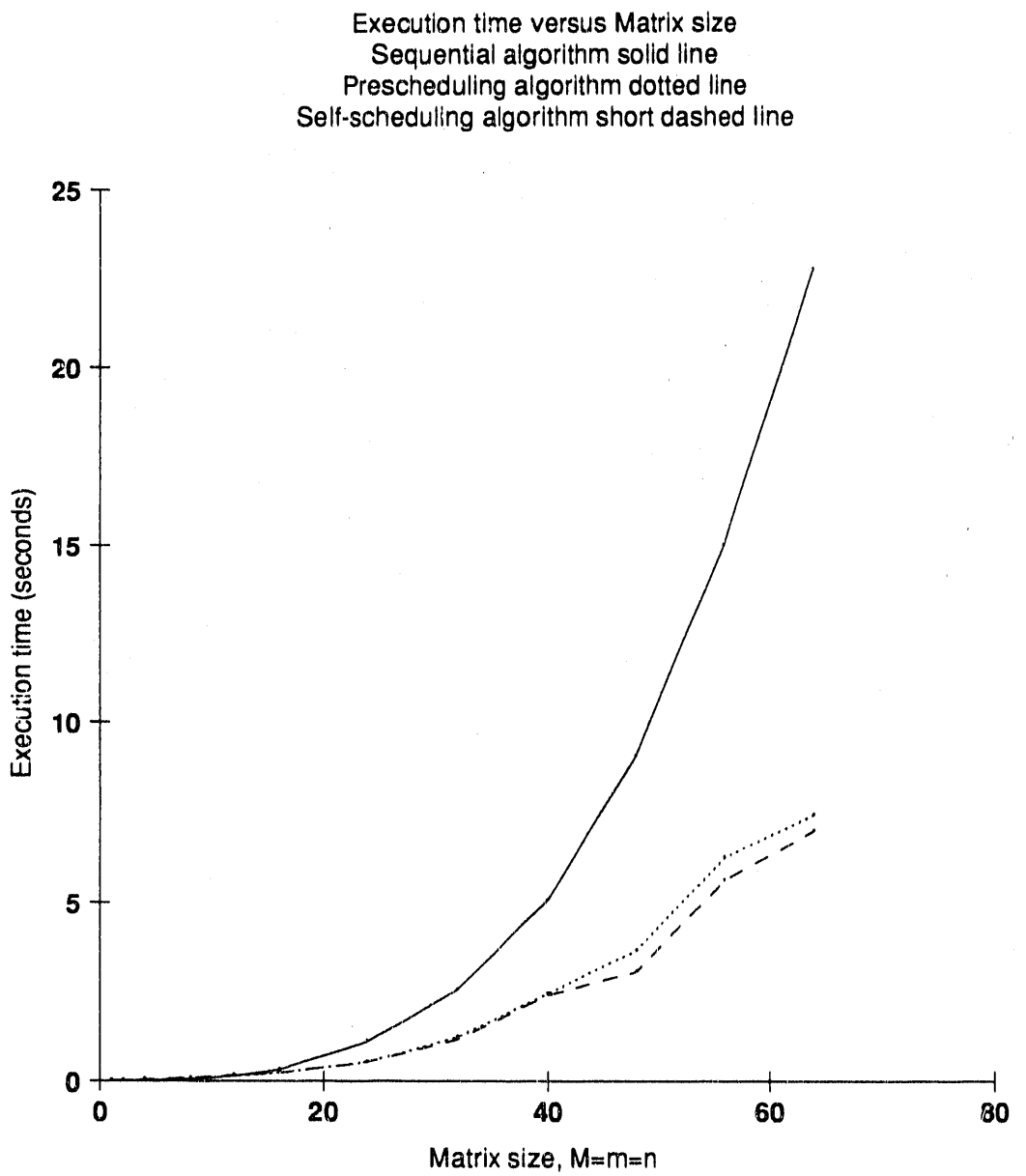


Figure 5.2: Computation time versus problem size with four tasks and four processors available to service tasks.

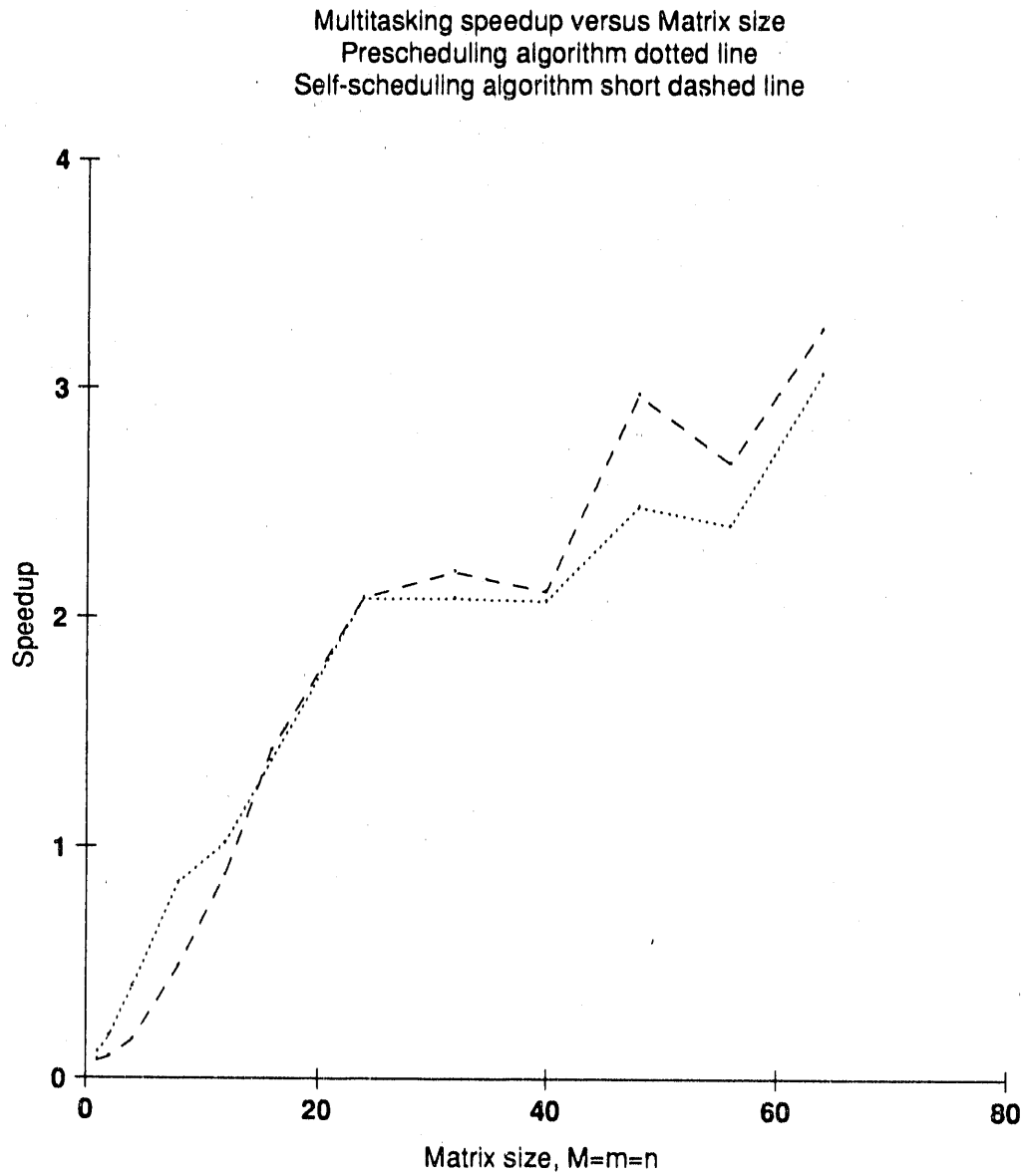


Figure 5.3: Speedup versus problem size with four tasks and four processors available to service tasks.

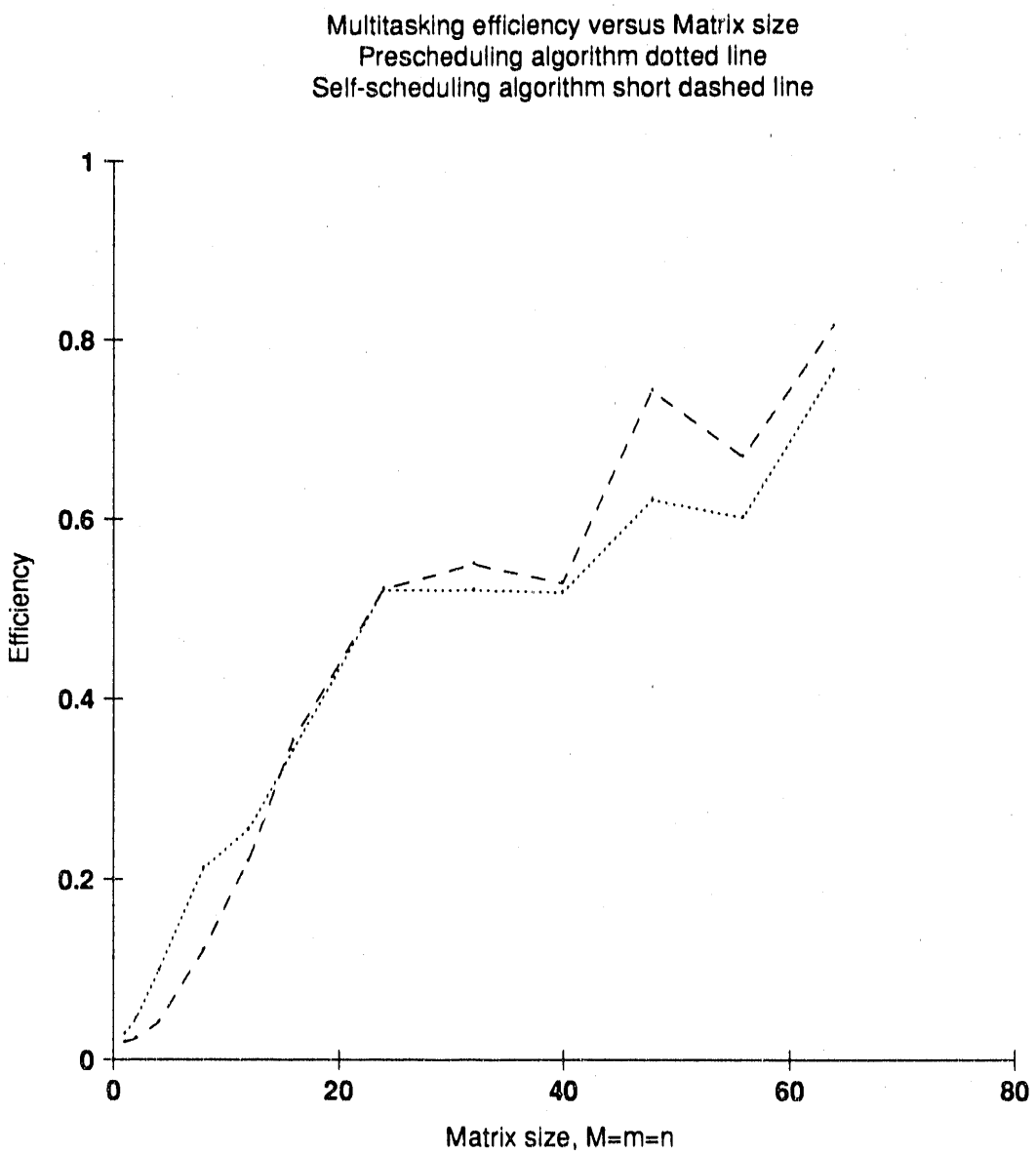


Figure 5.4: Efficiency versus problem size with four tasks and four processors available to service tasks.

more expensive than a procedure invocation.

Data memory usage and overhead is shown in table 5.2. Very little memory is necessary for the synchronization of tasks. Each of the processes needs some local working storage for computing FFTs and SVDs. Code memory usage and overhead is shown in table 5.3. The code space sharing was small due to a problem in the Fortran compiler that made code replication necessary.

$M = m = n$	sequential	prescheduled		self-scheduled	
	usage (kB)	usage (kB)	% overhead	usage (kB)	% overhead
8	112	409	265.2	475	324.1
16	240	533	130.4	604	151.7
32	1648	1946	18.1	1948	18.2
64	12400	12698	2.4	12888	3.9

Table 5.2: Data memory usage and overhead versus problem size with four tasks.

k	sequential	prescheduled		self-scheduled	
	usage (kB)	usage (kB)	% overhead	usage (kB)	% overhead
1	404	450	11.4	447	10.6
2	404	489	21.0	492	21.8
3	404	530	31.1	537	32.9
4	404	570	41.1	582	44.1

Table 5.3: Code memory usage and overhead versus number of active tasks.

Dynamic memory allocation costs are basically independent of the block size being allocated for small blocks. The cost depends almost entirely on the number of blocks being allocated. Each block takes approximately 0.68 ms to allocate. The server processes of the self-scheduling algorithm avoid this overhead by reusing their local storage during each activation. The prescheduling algorithm originally allocated

local storage blocks within each child process. This was deemed to be unsatisfactory and another parameter with working storage was passed to each child to avoid the overhead of dynamic memory allocation.

The BCSVD algorithm provides orders of magnitude speedup by utilizing the circulant structure of matrices. A further speedup was obtained using macrotasking. This does not reduce central processing unit charges because time on all processors is billed to the job [59]. However, a substantial savings in memory charges is achieved because the program memory residency time is reduced by the multiprocessor speedup [60] [61]. For typical problems M, m, and n are approximately 256. This requires approximately 800 megabytes of memory which can be quite costly to use.

Self-scheduling is useful when the task granularity is small. As the task granularity increases, prescheduling overhead becomes less important. Prescheduling is much easier to implement and debug. There are no explicit synchronizations to consider since the operating system handles the process allocation and scheduling. The parent only has to wait for the operating system to signal that the child has finished. Self-scheduling needs explicit synchronization with the server tasks and is therefore more difficult to implement and debug.

The Fortran compiler does not allocate local variables on the stack properly. It puts some local variables into static storage. Thus, code sharing is not possible for the Fortran subroutines. Each process must have a separate copy of the code and local data space. This was done by creating copies of the subroutines and giving each copy a unique name space by appending the process number to the name of the subroutine and all of its descendants.

Chapter 6

Summary and Conclusion

The work for my dissertation has focused on novel configuration space models (CSM) for processing multidimensional signals acquired using modern tomography systems that have an anisotropic or spatially variant response function. A motivation for the project is accurate estimation of radiotracer distributions from emission tomograph data. Accurate estimates are necessary for use in quantitative models of physiologically important processes such as the *in vivo* density of neuroreceptors. My research of algorithms and architectures for reconstructing tomographic data has three distinct parts: identification, estimation, and optimization.

An identification procedure is performed where the response function of the tomograph is found using a combination of deterministic and stochastic simulations and measured point response data. Because the model assumes the detection process has a discrete domain and the original distribution has a continuous domain, the model is easily adapted to include a variety of physical effects found in positron tomography

(PET), single photon emission computed tomography (SPECT), and nuclear magnetic resonance imaging (NMRI). For PET, the response function model can include radioactive decay, positron range, non-collinearity of photons, sampling geometry, attenuation, inter-crystal scatter, crystal penetration, and detection efficiency. Several of these phenomena lead to position dependent resolution.

An estimator for the unknown spatial distribution is formulated using the known response function and the statistical characteristics of the acquired data. Generalized pixel least squares, normal maximum likelihood, and Poisson maximum likelihood estimators and the corresponding covariance have been formulated to compensate for a spatially variant response. Reconstruction algorithms based on the spatially variant model have reduced bias when compared to spatially invariant methods. While bias is reduced by these estimators, undesirable statistical and systematic fluctuations due to pixelization effects can result. To reduce these fluctuations, an algorithm that uses a orthonormal pixel basis decomposition of the unknown spatial distribution was developed. The pixel basis functions are chosen from the right singular functions of the system response operator and therefore are fundamental to a particular tomography system. With this pixel basis, systematic error is zero and statistical error can be reduced by filtering the singular values of the impulse response function as evidenced by analytical calculations and simulation studies. Additionally, these simulation studies suggest the existence of a resolution threshold beyond which improvements in resolution are achieved only with a large increase in image noise. The convergence rate and the computation time for each iteration of iterative algorithms can also be adjusted by weighting the singular values without adding systematic error which is in contrast

to square and polar pixel multigrid approaches. However, some estimator bias will usually result for this suboptimal estimator. Computation burden is sometimes reduced when using the orthonormal natural pixel basis because symmetries in the data acquisition process are preserved. For some systems, further computational efficiency can be achieved using suboptimal separable estimators.

The third part of the work is to optimize the system so better estimates of the unknown spatial distribution will result. The criterion used is to minimize the norm of the covariance matrix while keeping the estimator unbiased. This is typically done by simulating a change in one of the parameters used to identify the system response function and evaluating the criterion; closed form solutions are not usually possible. In the orthonormal pixel basis described previously, the norm is readily computed from the singular values of the projection normal matrix. For emission tomography, the system parameters that are most easily changed are the angular and lateral sampling density. The important result for parallel sampling with equally spaced angles and bins is that for spatial resolutions coarser than a well defined resolution threshold, increasing the number of angles sampled will decrease the noise in images reconstructed from data acquired by a system that has more angular samples than the number of lateral samples. For resolutions finer than the resolution threshold, very little improvement in resolution is obtained without a large increase image noise. By increasing the number of lateral projection bins, resolution and reconstructed image noise are always improved. However, for any system configuration, there exists a resolution threshold beyond which improvements in resolution are achieved only with a large increase in image noise. Furthermore, the number of generalized pixels

corresponding to that resolution threshold are far fewer than the total number of projection measurements. In fact from moment and singular value analysis, the number of generalized pixels is approximately equal to one-half the number of projection measurements.

The identification, estimation, and optimization methods described above all require the solution of large systems of linear and non-linear equations. For example, the Donner 600-Crystal Positron Tomograph takes 120,200 projection measurements and the resulting linear system is 120,200 $\times$ 120,200. The computational complexity of the CSM has led to the use of a distributed computing environment in which workstations are used to analyze results from our identification, estimation, and optimization algorithms running on supercomputers. Several of the algorithms have been implemented using large grain parallel processing and also remote procedure calls.

The block circulant singular value decomposition (BCSVD) algorithm uses discrete Fourier transforms to rotate the blocks of a block circulant matrix into block diagonal form. Each block on the diagonal is then factored using a general singular value decomposition (SVD) algorithm. The BCSVD algorithm provides orders of magnitude speedup over general SVD algorithms. For a $64^2 \times 64^2$ block circulant matrix, computation time decreased from 12 hours to 23 seconds on a Cray-2. Because the BCSVD algorithm is easy to partition, a further speedup can be achieved using parallel processing. The orthogonality properties of multidimensional fast Fourier transforms (FFT) allows the FFT portion of the algorithm to partition into macrotasks. The SVD of the blocks of the block diagonal matrix can be computed independently and a macrotask can be assigned to each SVD. A multiprocessor speedup of

3.06 was achieved for prescheduling and for self-scheduling a multiprocessor speedup of 3.25 was observed using four processors on a Cray-2. Relative time overhead was 0.5% for the prescheduled algorithm and 0.07% for the self-scheduled algorithm. Relative memory overhead was 4% for both cases. Self-scheduling is useful when the task granularity is small. As the task granularity increases, prescheduling overhead becomes less important. The prescheduled algorithm is satisfactory for most emission tomography problems because all the dimensions of the matrices are greater than 64 and the task granularity will therefore be large when compared to the synchronization overhead.

Multitasking the block circulant singular value decomposition algorithm decreases overall computation costs by reducing the time large sections of memory are in use. Little or no gain comes from reduced central processing unit charges since processing time on all processors is charged to a job.

The data acquisition model presented in this work provides a mathematical framework to incorporate detailed knowledge about the response function of a tomography system and the statistical properties of the signals acquired using that system. Using this mathematical model, it is easy to represent systems of varying complexity; e.g., simple spatially invariant systems, systems with spatially variant response that have symmetries, and the most general linear case, a spatially variant system without symmetry. The novel singular value decomposition of the projection formation operator used in the data acquisition model is a powerful mathematical description of a tomography system and is fundamental to the identification, estimation, and

Projection formation

$$\boldsymbol{p} = F \cdot b$$

Projection formation singular value decomposition

$$F = \boldsymbol{U} S \cdot V^T$$

Projection normal matrix

$$\begin{aligned} \boldsymbol{A} &\equiv F \cdot F^T \\ &= \boldsymbol{U} S \cdot S^T \boldsymbol{U}^T \end{aligned}$$

Generalized pixel representation

$$b \approx B^T \boldsymbol{c}$$

Orthonormal natural pixel basis

$$B3 \equiv \left[\left(S \cdot S^T \right)^+ \right]^{\frac{1}{2}} S \cdot V^T$$

Generalized pixel least squares estimation

$$\begin{aligned} \hat{\boldsymbol{c}} &= \left(S \cdot V^T \cdot B^T \right)^+ \boldsymbol{U}^T \underline{\boldsymbol{p}} \\ \boldsymbol{\Sigma}_{\hat{\boldsymbol{c}}} &= \left(S \cdot V^T \cdot B^T \right)^+ \boldsymbol{U}^T \boldsymbol{\Sigma}_{\boldsymbol{p}} \boldsymbol{U} \left(B \cdot V \cdot S^T \right)^+ \\ \hat{b} &= B^T \left(S \cdot V^T \cdot B^T \right)^+ \boldsymbol{U}^T \underline{\boldsymbol{p}} \end{aligned}$$

Orthonormal natural pixel filtered singular value least squares estimation

$$\begin{aligned} \hat{\boldsymbol{c}} &= \boldsymbol{D} \left[\left(S \cdot S^T \right)^+ \right]^{\frac{1}{2}} \boldsymbol{U}^T \underline{\boldsymbol{p}} \\ \boldsymbol{\Sigma}_{\hat{\boldsymbol{c}}} &= \boldsymbol{D} \left[\left(S \cdot S^T \right)^+ \right]^{\frac{1}{2}} \boldsymbol{U}^T \boldsymbol{\Sigma}_{\boldsymbol{p}} \boldsymbol{U} \left[\left(S \cdot S^T \right)^+ \right]^{\frac{1}{2}} \boldsymbol{D} \\ \hat{b} &= V \cdot S^T \boldsymbol{D} \left(S \cdot S^T \right)^+ \boldsymbol{U}^T \underline{\boldsymbol{p}} \end{aligned}$$

Variance bound

$$\| \boldsymbol{\Sigma}_{\hat{\boldsymbol{c}}} \|_2 = \frac{\bar{p}}{\min_i \left(S \cdot S^T \right)_{ii}}$$

Block circulant singular value decomposition

$$\boldsymbol{A} = \left(\mathcal{F}_M \otimes \boldsymbol{I}_m \right)^\dagger \boldsymbol{U}_{\boldsymbol{D}} \boldsymbol{S}_{\boldsymbol{D}} \boldsymbol{V}_{\boldsymbol{D}}^\dagger \left(\mathcal{F}_M \otimes \boldsymbol{I}_n \right)$$

Table 6.1: Summary of major mathematical results.

optimization methods developed here. While the results presented for several representative tomography systems are not inconsequential, the mathematical techniques, which are summarized in table 6.1, used to obtain these results are the primary contribution from this work. These techniques should find the most utility in modeling the spatial sampling of each unique tomograph. Extensions of the model to include sampling in three or more spatial dimensions as well as time should be straightforward; but, in practice, higher dimensional applications may be limited by computational tractability without using special computing technology.

Because the data acquisition process is represented as a linear map from a continuous domain object space to a discrete domain observation space, it is a more physically realistic model of many systems than approximations using continuous-continuous maps or discrete-discrete maps. Thus, the validity of many results that were obtained using these approximations, e.g., angular and lateral sampling density in emission tomography, may need to be reexamined using the new, more robust techniques presented in this work. While the verification of old results is worthwhile, it is the unanswered questions such as the efficacy of iterative algorithms and stopping rules, the formulation of Bayesian estimators that use basis functions in the null space of the projection formation operator, the representation of object functions that are convex cones, and the efficiency of algorithm implementations that provide challenging new research opportunities for the application of the mathematical methods resulting from this thesis.

Bibliography

[1] Budinger TF, GT Gullberg, and RH Huesman. Emission computed tomography. In Herman GT, editor, *Image Reconstruction from Projections*, pages 147–246, Springer-Verlag, New York, NY, 1979.

[2] Huesman RH, EM Salmeron, and JR Baker. Compensation for crystal penetration in high resolution positron tomography. *IEEE Trans Nucl Sci*, **NS-36**:1100–1107, 1989.

[3] Radon J. Über die bestimmung von funktionen durch ihre integralwerte längs gewisser mannigfaltigkeiter. *Berichte Sächsische Akademie der Wissenschaften*, **69**:262–267, 1917.

[4] Cormack AM. Representation of a function by its line integrals, with some radiological applications. *J Appl Phys*, **34**(9):2722–2727, 1963.

[5] Cormack AM. Representation of a function by its line integrals, with some radiological applications II. *J Appl Phys*, **35**(10):2908–2913, 1964.

[6] Gullberg GT and TF Budinger. The use of filtering methods to compensate for constant attenuation in single-photon emission computed tomography. 1981.

[7] Tretiak O and C Metz. The exponential Radon transform. *SIAM J Appl Math*, **39**(2):341–354, 1980.

[8] O Y. An ECAT reconstruction method which corrects for attenuation and detector response. *IEEE Trans Nucl Sci*, **NS-30**(1):632–635, 1983.

[9] Ramachandran GN and AV Lakshminarayanan. Three-dimensional reconstruction from radiographs and electron micrographs: application of convolutions instead of Fourier transforms. *Proc Nat Acad Sci*, **68**(9):2236–2240, 1971.

[10] Chang L. A method for attenuation correction in radionuclide computed tomography. *IEEE Trans Nucl Sci*, **NS-25**(1):638–643, 1978.

[11] Deans SR. *The Radon Transform and some of its Applications.* John Wiley & Sons, New York, 1983.

[12] Huesman RH, GT Gullberg, WL Greenberg, and TF Budinger. *User's Manual: Donner Algorithms for Reconstruction Tomography.* Technical Report PUB-214, LBL, 1977.

[13] Hsieh RC and WG Wee. On methods of three-dimensional reconstruction from a set of radioisotope scintigrams. *IEEE Trans Systems, Man, Cybernetics*, **SMC-6**(12):854–862, 1976.

[14] Wood SL. A system theoretic approach to image reconstruction. May 1978. Ph.D. Thesis.

[15] Brigham EO. *The Fast Fourier Transform.* Prentice-Hall, Englewood Cliffs, NJ, 1974.

[16] Marr RB. On the reconstruction of a function on a circular domain from a sampling of its line integrals. *J Math Anal Appl*, **45**:357–374, 1974.

[17] Budinger TF and GT Gullberg. Three-dimensional reconstruction in nuclear medicine emission imaging. *IEEE Trans Nucl Sci*, **NS-21**:2–20, 1974.

[18] Strang G. *Linear Algebra and Its Applications.* Academic Press, Orlando, FL, 1980.

[19] Goitein M. Three-dimensional density reconstruction from a series of two-dimensional projections. *Nucl Instrum Meth*, **101**:509–518, 1972.

[20] Gordon R, R Bender, and GT Herman. Algebraic reconstruction techniques (ART) for three-dimensional electron microscopy and x-ray photography. *J Theor Biol*, **29**:471–481, 1970.

[21] Shepp LA and Y Vardi. Maximum likelihood reconstruction in positron emission tomography. *IEEE Trans Med Imag*, **1**:910–913, 1982.

[22] Lange K and RE Carson. EM reconstruction algorithms for emission and transmission tomography. *J CAT*, **8**:302–316, 1984.

[23] Censor Y. Finite series expansion reconstruction methods. *Proc IEEE*, **71**(3):409–419, 1983.

[24] Floyd CE, RJ Jaszczak, and RE Coleman. Image resampling on a cylindrical sector grid. *IEEE Trans Med Imaging*, **MI-5**(3):128–131, 1986.

[25] Oppenheim BE. More accurate algorithms for iterative 3-dimensional reconstruction. *IEEE Trans Nucl Sci*, **NS-21**:72–77, 1974.

[26] Trussell HJ, H Orun-Ozturk, and MR Civanlar. Errors in reprojection methods in computerized tomography. *IEEE Trans Med Imaging*, **MI-6**:220–227, 1987.

[27] Buonocore MH, WR Brody, and A Macovski. A natural pixel decomposition for two dimensional image reconstruction. *IEEE Trans Biomed Eng*, **BME-28**(2):69–78, 1981.

[28] Snyder DL and MI Miller. The use of sieves to stabilize images produced with the EM algorithm for emission tomography. *IEEE Trans Nucl Sci*, **NS-32**:3864–3872, 1985.

[29] Snyder DL, MI Miller, LJ Thomas, and DG Politte. Noise and edge artifacts in maximum-likelihood reconstructions for emission tomography. *IEEE Trans Med Imaging*, **MI-6**:223–238, 1987.

[30] Dempster AP, NM Laird, and DB Rubin. Maximum likelihood from incomplete data via the EM algorithm. *J Roy Stat Soc*, **39**:1–38, 1977.

[31] Lewitt RM and G Muehllehner. Accelerated iterative reconstruction for positron emission tomography based on the EM algorithm for maximum likelihood estimation. *IEEE Trans Med Imaging*, **MI-5**(1):16–22, 1986.

[32] Hanson KM. Bayesian and related methods in image reconstruction from incomplete data. In Stark H, editor, *Image Recovery: Theory and Application*, pages 79–123, Academic Press, Orlando, 1987.

[33] Levitan E and GT Herman. A maximum *a posteriori* probability expectation algorithm for image reconstruction in emission tomography. *IEEE Trans Med Imaging*, **MI-6**:185–192, 1987.

[34] Leahy R, TJ Hebert, and R Lee. Applications of Markov random fields in medical imaging. *Prog Clin Biol Res*, **363**:1–14, 1991.

[35] Rangayyan R, AP Dhawan, and R Gordon. Algorithms for limited-view computed tomography: An annotated bibliography and a challenge. *Appl Opt*, **24**(23):4000–4012, 1985.

[36] Andrews HC and BC Hunt. *Digital Image Restoration*. Prentice-Hall, Englewood Cliffs, NJ, 1977.

[37] Baker JR and TF Budinger. Advanced models for medical imaging. In *High Speed Computing: Scientific Applications and Algorithm Design*, pages 221–226, Urbana, IL, 1987.

[38] Luenberger DG. Optimization by vector space methods. In Howard RA, editor, *Decision and Control*, John Wiley and Sons, Inc., New York, 1969.

[39] Huesman RH and BM Mazoyer. Kinetic data analysis with a noisy input function. *Phys Med Bio*, **32**(12):1569–1579, 1987.

[40] Moore EH. *Bull Amer Math Soc*, **26**:394–395, 1920.

[41] Penrose R. A generalized inverse for matrices. *Cambridge Philosophical Soc*, **51**:406–413, 1955.

[42] Nassi M, WR Brody, BP Medoff, and A Macovski. Iterative reconstruction-reprojection: an algorithm for limited data cardiac-computed tomography. *IEEE Trans Biomed Eng*, **BME-29**(5):333–341, 1982.

[43] Kim JH, KY Kwak, SB Park, and ZH Cho. Projection space iteration reconstruction-reprojection. *IEEE Trans Med Imaging*, **MI-4**(3):139–143, 1985.

[44] Bickel PJ and KA Doksum. Mathematical statistics: Basic ideas and selected topics. In Lehmann EL, editor, *Holden-Day Series in Probability and Statistics*, Holden-Day, San Francisco, 1977.

[45] Haber SF, SE Derenzo, and DC Uber. Application of mathematical removal of positron range blurring in positron emission tomography. *IEEE Trans Nucl Sci*, **NS-37**(3):1293–1299, 1990.

[46] Knuth DE. *The Art of Computer Programming*. Volume 2, Addison Wesley, Reading, MA, 1981.

[47] Derenzo SE, RH Huesman, JL Cahoon, A Geyer, DC Uber, T Vuletich, and TF Budinger. Initial results from the Donner 600-crystal tomograph. *IEEE Trans Nucl Sci*, **NS-34**(1):321–325, 1987.

[48] Ein-Gal M, D Rosenfeld, and A Macovski. The consistency of the shadow: An approach to preprocessing in computerized tomography. In Gordon R, editor, *Image Processing for 2-D and 3-D Reconstruction from Projections: Theory and Practice in Medicine and the Physical Sciences*, pages WB5.1–WB5.4, Optical Society of America, Stanford, CA, 1975.

[49] Huesman RH. The effects of a finite number of projection angles and finite lateral sampling of projections on the propagation of statistical errors in transverse section reconstruction. *Physics in Medicine and Biology*, **22**(3):511–521, 1977.

[50] Kernighan BW and DM Ritchie. *The C Programming Language*. Prentice Hall, Englewood Cliffs, 1978.

[51] Knuth DE. *The Art of Computer Programming*. Volume 1, Addison Wesley, Reading, MA, 1981.

[52] Bracewell RN. Fourier techniques in two dimensions. In Price JR, editor, *Fourier Techniques and Applications*, pages 45–71, Plenum Press, New York, 1985.

[53] Baker JR. Macrotasking the singular value decomposition of block circulant matrices on the Cray-2. In *Proceedings of Supercomputing '89*, pages 243–247, Reno, NV, 1989.

[54] Buneman O. Vector FFT for the Cray-2. *NMFECC Buffer*, **10**(11):10–11, 1986.

[55] Despain AM. Very fast Fourier transform algorithms for hardware implementation. *IEEE Transactions on Computers*, **C-28**(5):333–341, 1979.

[56] Mundie DA and DA Fisher. Parallel processing in Ada. *Computer*, **19**(8):20–25, 1986.

[57] Quinn MJ. *Designing Efficient Algorithms for Parallel Computers*. McGraw-Hill, New York, NY, 1987.

[58] Mirin A. Parallelization of a 3-d MHD code, part I: methodology and results. *NMFECC Buffer*, **11**(7):14–16, 1987.

[59] Mirin A. Parallelization of a 3-d MHD code, part II: analysis of multiprocessing efficiency on the Cray-2. *NMFECC Buffer*, **11**(8):11–13, 1987.

[60] Patton PC. Multiprocessors: architectures and applications. *Computer*, **18**(6):29–40, 1985.

[61] Gelernter D. Domesticating parallelism. *Computer*, **19**(8):12–16, 1986.

Appendix A

Singular Value Decomposition

A.1 Projection Formation Singular Value Decomposition

The projection normal matrix is symmetric and positive semidefinite because for an arbitrary projection vector, $\boldsymbol{p}$,

$$\begin{aligned}
\boldsymbol{p}^T\boldsymbol{A}\boldsymbol{p} &= \sum_{\theta'=0}^{\Theta-1}\sum_{k'=0}^{K-1} p_{\theta' k'} \sum_{\theta=0}^{\Theta-1}\sum_{k=0}^{K-1} A_{\theta' k' \theta k} p_{\theta k} && \text{(A.1)}\\
&= \sum_{\theta'=0}^{\Theta-1}\sum_{k'=0}^{K-1} p_{\theta' k'} \sum_{\theta=0}^{\Theta-1}\sum_{k=0}^{K-1} \int_{\mathbb{R}} dy \int_{\mathbb{R}} dx\, f_{\theta' k'}(x,y)\, f_{\theta k}(x,y)\, p_{\theta k} && \text{(A.2)}\\
&= \int_{\mathbb{R}} dy \int_{\mathbb{R}} dx \sum_{\theta'=0}^{\Theta-1}\sum_{k'=0}^{K-1} f_{\theta' k'}(x,y)\, p_{\theta' k'} \sum_{\theta=0}^{\Theta-1}\sum_{k=0}^{K-1} f_{\theta k}(x,y)\, p_{\theta k} && \text{(A.3)}\\
&= \int_{\mathbb{R}} dy \int_{\mathbb{R}} dx \left[\sum_{\theta=0}^{\Theta-1}\sum_{k=0}^{K-1} f_{\theta k}(x,y)\, p_{\theta k}\right]^2 && \text{(A.4)}\\
&\geq 0. && \text{(A.5)}
\end{aligned}$$

Since $\boldsymbol{A}$ is positive semidefinite, the eigenvalues, $S \cdot S^T$, are non-negative. Let the eigenvalues be ordered such that

$$\begin{aligned}\left(S \cdot S^T\right)_{0,0} &\geq \cdots \geq \left(S \cdot S^T\right)_{r-1,r-1} \\ &\geq 0 = \left(S \cdot S^T\right)_{r,r} = \cdots = \left(S \cdot S^T\right)_{\Theta K-1,\Theta K-1}. \end{aligned} \tag{A.6}$$

Also, let $\boldsymbol{U}_1$ be the submatrix of orthonormal eigenvectors corresponding to the non-zero eigenvalues, $\left(S \cdot S^T\right)_{0,0}, \ldots \left(S \cdot S^T\right)_{r-1,r-1}$, and $\boldsymbol{U}_2$ be the submatrix of orthonormal eigenvectors corresponding to the zero eigenvalues, $\left(S \cdot S^T\right)_{r,r}, \ldots \left(S \cdot S^T\right)_{\Theta K-1,\Theta K-1}$, such that

$$\boldsymbol{U} = \begin{bmatrix} \boldsymbol{U}_1 & \boldsymbol{U}_2 \end{bmatrix} \text{ and} \tag{A.7}$$

$$S \cdot S^T = \begin{bmatrix} S_1 \cdot S_1^T & 0 \\ 0 & S_2 \cdot S_2^T \end{bmatrix} \tag{A.8}$$

$$= \begin{bmatrix} S_1 \cdot S_1^T & 0 \\ 0 & 0 \end{bmatrix}. \tag{A.9}$$

Therefore,

$$\boldsymbol{U}_1^T A \boldsymbol{U}_1 = S_1 \cdot S_1^T, \tag{A.10}$$

$$\boldsymbol{U}_2^T A \boldsymbol{U}_2 = 0 \text{ and consequently} \tag{A.11}$$

$$\boldsymbol{U}_2^T F = 0. \tag{A.12}$$

Choose V_1 so

$$S \cdot V_1^T = \boldsymbol{U}_1^T F \tag{A.13}$$

and choose V_2 such that

$$V = \begin{bmatrix} V_1 & V_2 \end{bmatrix} \text{ and} \tag{A.14}$$

$$V \cdot V^T = \begin{bmatrix} V_1 \cdot V_1^T & 0 \\ 0 & V_2 \cdot V_2^T \end{bmatrix} \tag{A.15}$$

$$= \delta(x' - x)\delta(y' - y). \tag{A.16}$$

$$\boldsymbol{U}^T F \cdot V = \begin{bmatrix} \boldsymbol{U}_1 & \boldsymbol{U}_2 \end{bmatrix}^T F \cdot \begin{bmatrix} V_1 & V_2 \end{bmatrix} \tag{A.17}$$

$$= \begin{bmatrix} \boldsymbol{U}_1^T \\ \boldsymbol{U}_2^T \end{bmatrix} F \cdot \begin{bmatrix} V_1 & V_2 \end{bmatrix} \tag{A.18}$$

$$= \begin{bmatrix} \boldsymbol{U}_1^T F \cdot V_1 & \boldsymbol{U}_1^T F \cdot V_2 \\ \boldsymbol{U}_2^T F \cdot V_1 & \boldsymbol{U}_2^T F \cdot V_2 \end{bmatrix} \tag{A.19}$$

$$= \begin{bmatrix} S_1 \cdot V_1^T \cdot V_1 & S_1 \cdot V_1^T \cdot V_2 \\ 0 \cdot V_1 & 0 \cdot V_2 \end{bmatrix} \tag{A.20}$$

$$= \begin{bmatrix} S_1 & 0 \\ 0 & 0 \end{bmatrix} \tag{A.21}$$

$$= S. \tag{A.22}$$

Finally, the singular value decomposition of F is

$$F = US \cdot V^T. \tag{A.23}$$

A.2 Block Circulant Singular Value Decomposition

Let $A_{p,q}^{K,L}$ be the p,q element of the K,L block of the matrix $\boldsymbol{A}$ in equation 5.4. Since $\boldsymbol{A}$ is block circulant,

$$A_{p,q}^{K,L} = A_{p,q}^{0,L-K+M} \tag{A.24}$$

$$= A_{p,q}^{L} \tag{A.25}$$

and it can be decomposed as follows

$$A_{p,q}^{K,L} = \frac{1}{\sqrt{M}} \sum_{u=0}^{M-1} e^{-\imath u K \frac{2\pi}{M}} \frac{1}{\sqrt{M}} \sum_{v=0}^{M-1} e^{\imath v L \frac{2\pi}{M}} \frac{1}{\sqrt{M}} \sum_{K'=0}^{M-1} e^{\imath u K' \frac{2\pi}{M}} \frac{1}{\sqrt{M}} \sum_{L'=0}^{M-1} e^{-\imath v L' \frac{2\pi}{M}} A_{p,q}^{K',L'} \tag{A.26}$$

$$= \frac{1}{\sqrt{M}} \sum_{u=0}^{M-1} e^{-\imath u K \frac{2\pi}{M}} \frac{1}{\sqrt{M}} \sum_{v=0}^{M-1} e^{\imath v L \frac{2\pi}{M}} \frac{1}{\sqrt{M}} \sum_{K'=0}^{M-1} e^{\imath u K' \frac{2\pi}{M}} \frac{1}{\sqrt{M}} \sum_{L'=0}^{M-1} e^{-\imath v L' \frac{2\pi}{M}} A_{p,q}^{0,L'-K'+M} \tag{A.27}$$

$$= \frac{1}{\sqrt{M}} \sum_{u=0}^{M-1} e^{-\imath u K \frac{2\pi}{M}} \frac{1}{\sqrt{M}} \sum_{v=0}^{M-1} e^{\imath v L \frac{2\pi}{M}} \frac{1}{\sqrt{M}} \sum_{K'=0}^{M-1} e^{\imath u K' \frac{2\pi}{M}} \frac{1}{\sqrt{M}} \sum_{L'=K'+M}^{2M+K'-1} e^{-\imath v (L'+K'-M) \frac{2\pi}{M}} A_{p,q}^{0,L'} \tag{A.28}$$

$$= \frac{1}{\sqrt{M}} \sum_{u=0}^{M-1} e^{-\imath u K \frac{2\pi}{M}} \frac{1}{\sqrt{M}} \sum_{v=0}^{M-1} e^{\imath v L \frac{2\pi}{M}} \frac{1}{\sqrt{M}}$$
$$\sum_{K'=0}^{M-1} e^{\imath (u-v) K' \frac{2\pi}{M}} \frac{1}{\sqrt{M}} \sum_{L'=0}^{M-1} e^{-\imath v L' \frac{2\pi}{M}} A_{p,q}^{0,L'} \tag{A.29}$$

$$= \frac{1}{\sqrt{M}} \sum_{u=0}^{M-1} e^{-\imath u K \frac{2\pi}{M}} \frac{1}{\sqrt{M}} \sum_{v=0}^{M-1} e^{\imath v L \frac{2\pi}{M}}$$
$$\sqrt{M} \delta^{u,v} \frac{1}{\sqrt{M}} \sum_{L'=0}^{M-1} e^{-\imath v L' \frac{2\pi}{M}} A_{p,q}^{0,L'} \tag{A.30}$$

$$= \frac{1}{\sqrt{M}} \sum_{u=0}^{M-1} e^{-\imath u K \frac{2\pi}{M}} \frac{1}{\sqrt{M}} \sum_{v=0}^{M-1} e^{\imath v L \frac{2\pi}{M}} \delta^{u,v} D_{p,q}^{u,v} \tag{A.31}$$

$$= \frac{1}{\sqrt{M}} \sum_{u=0}^{M-1} e^{-\imath u K \frac{2\pi}{M}} \frac{1}{\sqrt{M}} \sum_{v=0}^{M-1} e^{\imath v L \frac{2\pi}{M}} \delta^{u,v} D_{p,q}^{0,u-v}. \tag{A.32}$$

Thus, in matrix form, the block circulant decomposition of $\boldsymbol{A}$ is

$$\boldsymbol{A} = (\mathcal{F}_M \otimes \boldsymbol{I}_m)^\dagger \, \boldsymbol{U_D} \boldsymbol{S_D} \boldsymbol{V_D^\dagger} \, (\mathcal{F}_M \otimes \boldsymbol{I}_n). \tag{A.33}$$

Appendix B

Software and Documentation

B.1 orthols.c

```
/*
** Header:      @(#) orthols.c 1.5 91/10/29 16:12:21 baker penguin UCB/LBL
**
** Name:        orthols
**
** Purpose:     The subroutine orthols reconstructs the array image
**              using the orthonormal natural pixels least squares method.
**
** Input parameters:
**
**    prj       entry point of projector
**    bck       entry point of backprojector
**    fftsvd    entry point of singular value decomposition routine
**    min_sing  minimum singular value to be used in the solution
**    int_fac   integration factor
**
** Output parameters:
**
**    image     orthonormal least squares solution
**              image[parm.ipar.ndimu][parm.ipar.ndimu]
**
** Dependencies:
**
**    libreclbl.a
**      cemesg  error message routine
**      clgtxt  print text banner
**      memst   memory management
**      rchek   initialization check
**      orthobck
**              orthonormal natural pixels backprojection
**      orthosquare
```

```
**              orthonormal natural pixels <-> square pixels conversion
**
*/


#include <stdio.h>
#include <errno.h>
#include <sys/types.h>
#include "error.h"

#include "math.h"
#include "reclbl.h"

int
orthols (image, prj, bck, fftsvd, min_sing, int_fac)

    double              *image;
    int                 (*prj)();
    int                 (*bck)();
    int                 (*fftsvd)();
    double              *min_sing;
    int                 *int_fac;

{
    static char         SccsId[] = "@(#) orthols.c      1.5 91/10/29 16:12:21
                                    baker penguin UCB/LBL";

    static int          name[]  = { 'E', 'N', 'D', ' ',
                                    'O', 'R', 'T', 'H', 'O', 'L', 'S'
                                  };

    double              *proj, *proj_pointer;
    double              *error, *error_pointer;
    double              *ortho_image, *identity;
    double              *u, *sst;

    double              min_sst;
    double              sum;

    int                 pix_index, proj_index, angle_index;
    int                 number_of_proj, size;
    int                 maxfw, length;

    int                 status;

    int                 one = 1, two = 2;


    /*
    ** Check that setup has been called.
    */
    if (setchk() < 0)
        cemesg(&one, name[4], &one);

    length = 7;
    clgtxt(&name[4], &length);

    printf("\n\n\n\n           PARAMETERS FOR SUBROUTINE ORTHOLS\n\n");
```

```
printf("                    DESCRIPTION\n");
printf("MINSING - %10.3e  MINIMUM SINGULAR VALUE\n",
       *min_sing);
printf("INTFAC  -    %7d  INTEGRATION FACTOR\n",
       *int_fac);

/*
** Check the backprojector and projector to see if they match.
*/
rchek(bck, prj, &one);

/*
** Compute maximum storage size needed.
*/
number_of_proj = trgcom.nang * trgcom.kdimu;
size           = number_of_proj * number_of_proj;

/*
** Allocate dynamic memory
*/
if ((u           = (double *) malloc(size           * sizeof(double))
    ) == NULL)
    Perror("unable to allocate u");
if ((sst         = (double *) malloc(number_of_proj * sizeof(double))
    ) == NULL)
    Perror("unable to allocate sst");
if ((proj        = (double *) malloc(number_of_proj * sizeof(double))
    ) == NULL)
    Perror("unable to allocate proj");
if ((error       = (double *) malloc(number_of_proj * sizeof(double))
    ) == NULL)
    Perror("unable to allocate error");
if ((ortho_image = (double *) malloc(number_of_proj * sizeof(double))
    ) == NULL)
    Perror("unable to allocate ortho_image");
if ((identity    = (double *) malloc(number_of_proj * sizeof(double))
    ) == NULL)
    Perror("unable to allocate identity");

/*
** Get the right singular vectors and singular values of
** the projection normal matrix.
*/
status = (*fftsvd)(u, sst, prj, bck, int_fac);

/*
** Eliminate singular values less than the minimum.
*/
min_sst = MAX(0.0, *min_sing) * *min_sing;
for (proj_index = 0; proj_index < number_of_proj; proj_index++)
{
    if (sst[proj_index] < min_sst)
        sst[proj_index] = 0.0;
}

/*
** Get projection data and errors.
*/
```

```
proj_pointer  = proj;
error_pointer = error;
for (angle_index = 1; angle_index <= trgcom.nang; angle_index++)
{
    getum(&angle_index, proj_pointer, error_pointer);
    proj_pointer  += trgcom.kdimu;
    error_pointer += trgcom.kdimu;
}

/*
** Backproject from projections to orthogonal natural pixels.
** Normalization is incorporated in the singular value filter.
*/
for (pix_index = 0; pix_index < number_of_proj; pix_index++)
    identity[pix_index] = 1.0;
orthobck(ortho_image, proj, u, identity);

/*
** Singular value filter.  Singular values smaller than the
** minimum have already been zeroed.
*/
for (pix_index = 0; pix_index < number_of_proj; pix_index++)
{
    if (sst[pix_index] > 0.0)
        ortho_image[pix_index] /= sst[pix_index];
    else
        ortho_image[pix_index] = 0.0;
}
if (prtcom.tprint[3] == TRUE)
    for (pix_index = 0; pix_index < number_of_proj; pix_index++)
    {
        if (sst[pix_index] > 0.0)
            printf("sst[%4d]^+ = %18.10e\n", pix_index, 1.0/sst[pix_index]);
        else
            printf("sst[%4d]^+ = %18.10e\n", pix_index, 0.0);
    }

/*
** Convert from orthogonal natural to square pixels.
** Normalization is incorporated in the singular value filter.
*/
orthosquare(ortho_image, image, prj, bck, u, identity, &one);

/*
** Free dynamic memory.
*/
free(u);
free(sst);
free(proj);
free(error);
free(ortho_image);

length = -1;
memst(&maxfw, &length);
printf("\n\n          MAXIMUM SIZE OF BLANK COMMON THUS FAR=%7d ", maxfw);
printf("FLOATING POINT WORDS.\n");

/*
```

```
    ** Output trailer.
    */
    length = 11;
    clgtxt(&name[0], &length);

    /*
    ** Return SVD status.
    */
    return(status);
}
```

B.2 fftsvd.c

```
/*
** Header:      @(#) fftsvd.c 1.5 91/10/29 16:08:26 baker penguin UCB/LBL
**
** Name:        fftsvd
**
** Purpose:     Compute the singular value decomposition of the
**              projection normal matrix generated using the backprojector
**              bck and the projector prj.
**
** Input parameters:
**
**    prj       entry point of projector
**    bck       entry point of backprojector
**    int_fac   integration factor
**
** Output parameters:
**
**    u         left singular vectors of projection normal matrix
**              u[parm.ipar.nang][parm.ipar.kdimu]
**               [parm.ipar.nang][parm.ipar.kdimu]
**    sst       singular values of projection normal matrix
**              sst[parm.ipar.nang][parm.ipar.kdimu]
**
** Dependencies:
**
**    libreclbl.a
**      setup
**      fftgen
**    liblinpack.a
**      dsvdc
**
*/

#include <stdio.h>
#include <errno.h>
#include "error.h"

#include "math.h"
#include "linpack.h"
#include "reclbl.h"
#include "ftf.h"
```

```
int
fftsvd (u, sst, prj, bck, int_fac)

    double              *u;
    double              *sst;
    int                 (*prj)();
    int                 (*bck)();
    int                 *int_fac;

{
    static char         SccsId[] = "@(#) fftsvd.c       1.5 91/10/29 16:08:26
                                    baker penguin UCB/LBL";

    double              *fft;

    double              *v;
    double              *e, *work;
    double              *angles;

    double              phantom_pwid, image_pwid;

    int                 fft_size, f_rows, f_cols;
    int                 ft_rows;

    Template            *template;

    int                 phantom_pixels, image_pixels;
    int                 phantom_dim, image_dim;
    int                 number_of_bins;
    int                 angle_index;
    int                 bin_index;
    int                 offset;

    int                 compute_uv;

    int                 status;
    int                 one = 1, zero = 0;


    /*
    ** Compute maximum storage sizes.
    */
    image_dim      = ptrcom.ndimu;
    image_pwid     = ptrcom.pwid;
    image_pixels   = image_dim * image_dim;

    phantom_dim    = image_dim   * *int_fac;
    phantom_pwid   = image_pwid  / *int_fac;
    phantom_pixels = phantom_dim * phantom_dim;

    number_of_bins = trgcom.nang * trgcom.kdimu;

    f_rows   = number_of_bins;
    f_cols   = phantom_pixels;
    fft_size = f_rows * f_rows;

    /*
```

```
** Compute projection usage template.
*/
if ((template         = (Template *) malloc(trgcom.nang * sizeof(Template))
    ) == NULL)
    Perror("unable to allocate template");
if ((template[0].data = (int *)      malloc(number_of_bins * sizeof(int))
    ) == NULL)
    Perror("unable to allocate template data");

offset = 0;
for (angle_index = 0; angle_index < trgcom.nang; angle_index++)
{
    template[angle_index].data  = template[0].data + offset;
    template[angle_index].count = trgcom.kdimu;
    for (bin_index = 0; bin_index < trgcom.kdimu; bin_index++)
        template[angle_index].data[bin_index] = bin_index;
    offset += trgcom.kdimu;
}

/*
** Allocate dynamic memory.
*/
if ((fft    = (double *) malloc(fft_size    * sizeof(double))) == NULL)
    Perror("unable to allocate fft");
if ((angles = (double *) malloc(trgcom.nang * sizeof(double))) == NULL)
    Perror("unable to allocate angles");

/*
** Compute projection normal matrix, F . F^t.
*/
parm.ipar.ndimu = phantom_dim;
parm.par.pwid   = phantom_pwid;
setup(&parm.ipar, &parm.par, angles);

fftgen(fft, template, prj, bck);

parm.ipar.ndimu = image_dim;
parm.par.pwid   = image_pwid;
setup(&parm.ipar, &parm.par, angles);

/*
** Allocate storage for SVD.
*/
if ((e    = (double *) malloc(f_rows * sizeof(double))) == NULL)
    Perror("unable to allocate e");
if ((work = (double *) malloc(f_rows * sizeof(double))) == NULL)
    Perror("unable to allocate work");

/*
** Compute right singular vectors and singular values.
*/
compute_uv = 01;
v          = fft;
dsvdc(fft, &f_rows, &f_rows, &f_rows,
      sst, e,
      v, &f_rows,
      u, &f_rows,
      work, &compute_uv, &status
```

```
        );

    /*
    ** Free dynamic memory.
    */
    free(template[0].data);
    free(template);
    free(fft);
    free(e);
    free(work);

    /*
    ** Return SVD status.
    */
    return(status);
}
```

B.3 zbcsvdc.c

```
/*
** Header:      @(#) zbcsvdc.c 1.2 91/11/01 17:39:16 baker penguin UCB/LBL
**
** Name:        zbcsvdc
**
** Purpose:     Compute the singular value decomposition
**              of a complex square block circulant matrix,
**
**                      H *
**              a = Fusv F ,
**
**              where F_kl = I x exp(-i kl 2PI/block_count)
**              and I = identity(block_m).
**
** Input parameters:
**
**    block_count
**              number of blocks in a column of the BC form
**    block_m   number of rows in a block
**    block_n   number of columns in a block
**    a         complex BC matrix described by a column of
**              block_count blocks consisting of
**              block_m x block_n elements,
**              i.e., a[block_count][block_m][block_n]
**    compute_uv
**                                | 00 compute s
**              if compute_uv =   | 01 compute s, v
**                                | 10 compute s, u
**                                | 11 compute s, u, v
**
** Output parameters:
**
**    s         singular values of a
**              s[block_count][MAX(block_m,block_n)]
**    u         Fourier transformed left singular vectors of a
```

```
**              u[block_count][block_m][block_m]
**    v         Fourier transformed right singular vectors of a
**              v[block_count][block_n][block_n]
**    status    number of invalid singular values
**
** Dependencies:
**
**    libfft
**      zfftki  initialize complex 1D base-n FFT
**      zfftkf  forward complex 1D base-n FFT
**      zfftkb  inverse complex 1D base-n FFT
**    liblinpack
**      zsvdc   complex singular value decomposition
**      ztranc  complex conjugate transpose
**
*/

#include <stdio.h>
#include <errno.h>
#include "error.h"

#include "math.h"
#include "fft.h"
#include "linpack.h"

int
zbcsvdc (a, block_count, block_m, block_n, s, u, v, compute_uv, status)

    COMPLEX             *a;
    int                 *block_count;
    int                 *block_m;
    int                 *block_n;
    COMPLEX             *s;
    COMPLEX             *u;
    COMPLEX             *v;
    int                 *compute_uv;
    int                 *status;

{
    static char         SccsId[] = "@(#) zbcsvdc.c      1.2 91/11/01 17:39:16
                                    baker penguin UCB/LBL";

    COMPLEX             *a_pointer;
    COMPLEX             *u_pointer;
    COMPLEX             *v_pointer;
    COMPLEX             *s_pointer;
    COMPLEX             *e, *e_pointer;
    COMPLEX             *work;
    double              *wsave;

    int                 a_size, block_size;
    int                 compute_u, compute_v, compute_vu;

    int                 internal_status;

    int                 i;
```

```
/*
** Determine which singular vectors to compute.
*/
compute_u  = *compute_uv / 10;
compute_v  = *compute_uv % 1;
compute_vu = compute_v * 10 + compute_u;

/*
** Compute submatrix and total sizes.
*/
block_size = *block_m * *block_n;
a_size     = *block_count * block_size;

/*
** Allocate the FFT working storage.
*/
wsave = (double *) malloc((4 * *block_count + 15) * sizeof(double));
if (wsave == NULL)
    Perror("unable to allocate FFT working storage");

/*
** Initialize FFT.
*/
zfftki(block_count, &block_size, wsave);

/*
** Transform a to block diagonal form.
*/
a_pointer = a;
for (i = 0; i < block_size; i++)
{
    zfftkf(block_count, &block_size, a_pointer, wsave);
    a_pointer++;
}

/*
** Allocate storage for SVD routines.
*/
if ((work = (COMPLEX *) malloc(MAX(*block_m, *block_n) * sizeof(COMPLEX))
    ) == NULL)
    Perror("unable to allocate SVD working storage");
if ((e = (COMPLEX *) malloc(MAX(*block_m, *block_n) * sizeof(COMPLEX))
    ) == NULL)
    Perror("unable to allocate SVD working storage");

/*
** Compute the SVD of each block on the diagonal.
*/
a_pointer =  a;
u_pointer  = u;
v_pointer  = v;
s_pointer  = s;
*status    = 0;
for (i = 0; i < *block_count; i++)
{
    zsvdc(a_pointer, block_n, block_n, block_m,
          s_pointer, e,
          v_pointer, block_n,
```

```
                u_pointer, block_m,
                work, &compute_vu, &internal_status);
            if (compute_u == 1)
                ztranc(u_pointer, block_m, block_m, u_pointer);
            if (compute_v == 1)
                ztranc(v_pointer, block_n, block_n, v_pointer);
            a_pointer += block_size;
            u_pointer += block_size;
            v_pointer += block_size;
            s_pointer += MAX(*block_m, *block_n);
            *status   += internal_status;
        }

        /*
        ** Free up some storage.
        */
        free(wsave);
        free(work);
        free(e);

        return(0);
}
```

END

DATE
FILMED
5/11/92