# THE BLIND DECONVOLUTION OF LINEARLY BLURRED IMAGES

# THE BLIND DECONVOLUTION OF LINEARLY BLURRED IMAGES USING NON-PARAMETRIC STABILIZING FUNCTIONS

By

JAMES R. HARE, B. Eng.

A Thesis
Submitted to the School of Graduate Studies
in Partial Fulfilment of the Requirements
for the Degree
Master of Engineering

McMaster University

MASTER OF ENGINEERING (2000) McMaster University

(Electrical and Computer Engineering) Hamilton, Ontario

TITLE: The Blind Deconvolution of Linearly Blurred Images using Non-Parametric Stabilizing Functions

AUTHOR: James R. Hare

B. Eng. (McMaster University)

SUPERVISOR: Dr. James P. Reilly

NUMBER OF PAGES: x, 131

# Abstract

An iterative solution to the problem of blind image deconvolution is presented whereby a previous image estimate is explicitly used in the new image estimation process. The previous image is pre-filtered using an adaptive, non-parametric stabilizing function that is updated based on a current error estimate. This function is experimentally shown to dramatically benefit the convergence rate for the *a priori* restoration case. Noise propagation from one iteration to the next is reduced by the use of a second, regularizing operator, resulting in a hybrid iteration technique. Further, error terms are developed that shed new light on the error propagation properties of this method and others by quantifying the extent of noise and regularization error propagation. Optimal non-parametric, frequency adaptive stabilizing and regularization functions are then derived based on this error analysis.

# Acknowledgements

I wish to offer my sincerest gratitude to Dr. James P. Reilly for all of his insightful guidance, invaluable suggestions, and most of all for his seemingly unlimited patience during the course of this work. It has been wonderful working with Dr. Reilly, and an experience that I will remember always.

I would also like to acknowledge the financial support of our industrial sponsor, Pipetronix Ltd., as well as the Government of Canada through the Natural Sciences and Engineering Research Council.

Finally, and most importantly, I want to thank my wife Maria and our sons Alexander, Daniel and Nicholas for patiently enduring through those times, particularly in the past few months, when I really should have been there.

jrh

# Contents

**Abstract** iii

**Acknowledgements** iv

**1 Introduction** 1

1.1 Background . . . . . 2

1.1.1 Why Image Restoration is Difficult . . . . . 2

1.1.2 Regularization . . . . . 3

1.2 Restoration Methods . . . . . 4

1.2.1 Stochastic Methods . . . . . 4

1.2.2 Algebraic Methods . . . . . 6

1.2.3 Iterative Deconvolution Methods . . . . . 6

1.2.4 Multichannel Approaches using Wavelet Theory . . . . . 7

1.3 Scope of Thesis . . . . . 8

1.4 Outline of Thesis . . . . . 10

**2 Classical Image Restoration** 12

2.1 Image Formation and Degradation Model . . . . . 13

2.1.1 Continuous Model . . . . . 13

2.1.2 Discrete Model . . . . . 14

2.2 Common Point Spread Functions . . . . . 15
2.2.1 Linear Motion Blur . . . . . 16
2.2.2 Uniform Out-of-Focus Blur . . . . . 17
2.2.3 Atmospheric Turbulence Blur . . . . . 17
2.2.4 Uniform 2-D Blur . . . . . 17
2.3 Block-Toeplitz and Block-Circulant Matrices . . . . . 18
2.4 The Eigensystem of the Blurring Matrix . . . . . 19
2.5 Least Squares Approach . . . . . 20
2.6 Regularized Image Restoration . . . . . 22
2.6.1 Tikhonov Regularization . . . . . 23
2.6.2 Miller's Method . . . . . 25
2.7 Synopsis of Classical Restoration Approaches . . . . . 26

**3 Iterative Image Restoration** **28**
3.1 Linear Convergence Iterative Technique . . . . . 29
3.2 Re-blurring Methods . . . . . 30
3.3 Iterative Tikhonov-Miller Method . . . . . 31
3.4 Higher Order Convergence Methods . . . . . 32
3.5 Nonlinear Constraints . . . . . 34
3.6 Projections on to Convex Sets . . . . . 35
3.7 Synopsis of Iterative Restoration Techniques . . . . . 37

**4 A Hybrid Algorithm** **39**
4.1 Direct Multi-Image Deconvolution . . . . . 41
4.1.1 Error Analysis . . . . . 43
4.2 Hybrid Formulation . . . . . 46
4.2.1 Alternate Error Formulation . . . . . 47
4.2.2 Optimal Stability and Regularization Operators . . . . . 48

4.3 Stablilty Operator Effect on Convergence . . . . . 50
4.4 The Stability Operator: Effect on Asymptotic Error . . . . . 55
4.5 A Priori Deconvolution Results . . . . . 56
4.6 Discussion . . . . . 65

**5 Blind Deconvolution** **66**
5.1 *A Priori* Blur Estimation . . . . . 67
5.2 Joint PSF/Image Estimation Methods . . . . . 68
5.2.1 ARMA Methods . . . . . 68
5.2.2 Non-Parametric Deterministic Constraint Methods . . . . . 69
5.2.3 High Order Statistics Based Methods . . . . . 70
5.2.4 Unification of Image Estimation Methods . . . . . 70
5.3 A Blind Iterative Deconvolution Method . . . . . 73
5.3.1 Proposed Image and PSF Estimators . . . . . 73
5.3.2 Regularization Operator for the PSF Estimator . . . . . 75
5.3.3 *A Priori* Blur Estimation using GCV . . . . . 76
5.3.4 *A Priori* Blur Estimation Results . . . . . 78
5.4 Proposed Blind Iterative Algorithm . . . . . 80
5.4.1 Blind Deconvolution Results . . . . . 82
5.5 Discussion . . . . . 84

**6 Conclusions** **87**

**A Eigenvalue Analysis for 2-D Systems** **90**
A.1 Lexicographic Ordering . . . . . 90
A.2 Block Toeplitz Matrices . . . . . 91
A.3 The Eigensystem for the Blurring Matrix . . . . . 94

**B Error Terms** **97**

B.1 Error Term Expansion . . . . . 97

B.2 Alternate Error Formulation . . . . . 98

**Bibliography** **101**

**Program Listing** **109**

# List of Tables

5.1 PSF and noise variance estimation, 7 by 7 uniform out-of-focus blur. . 79
5.2 PSF and noise variance estimation, 9 pixel linear motion blur. . . . . 79
5.3 Results for Blind Procedure: Original and Estimated PSF . . . . . 83

# List of Figures


1.1 Image Degradation Process . . . . . 3

2.1 Intersection of two ellipsoids (after Katsaggelos *et al*) . . . . . 27

4.1 Two-Image Degradation System . . . . . 40

4.2 Proposed Iterative Process . . . . . 42

4.3 Effect of Stability Operator - ISNR (Lena) BSNR = 40 dB . . . . . 51

4.4 Effect of Stability Operator - Pictoral Results (Lena) BSNR = 40 dB 52

4.5 Effect of Stability Operator - ISNR (Cameraman) BSNR = 40 dB . . 53

4.6 Effect of Stability Operator - Pictoral Results (Cameraman) BSNR = 40 dB . . . . . 54

4.7 Deconvolution results: 9 Pixel Horizontal Blur. BSNR = 40 dB . . . 57

4.8 Deconvolution results: 9 Pixel Horizontal Blur. BSNR = 30 dB . . . 58

4.9 Deconvolution results: 9 Pixel Horizontal Blur. BSNR = 15 dB . . . 59

4.10 Deconvolution results: 7 x 7 Uniform Blur. BSNR = 30 dB . . . . . 60

4.11 Deconvolution results: 7 x 7 Uniform Blur. BSNR = 15 dB . . . . . 61

4.12 Deconvolution results: 9 Pixel Horizontal Blur. BSNR = 30 dB . . . 62

4.13 Deconvolution results: 9 Pixel Horizontal Blur. BSNR = 15 dB . . . 63

4.14 Deconvolution results: 7 x 7 Uniform Blur. BSNR = 30 dB . . . . . 64

5.1 Blind Iterative Procedure . . . . . 74

5.2 ISNR and Convergence Comparison (Lena) BSNR = 30 dB . . . . . 84

5.3 Blind Hybrid vs Iterative TM - Pictoral Results (Lena) BSNR = 30 dB 85

# Chapter 1

# Introduction

The process of deconvolution, or inverse filtering, is part of a wide range of applications in the diverse field of image restoration. Image restoration began in the late 1950's and early 1960's as an effort by scientists in the United States and former Soviet Union space programs to de-blur photographic images of the earth and solar system. This effort was intensified, particularly after the "space race" ended in 1969 with the first manned space flight to the moon when the focus of space programs changed to unmanned missions like the Ranger, Lunar Orbitor and Mariner initiatives. This focus has continued to this decade, recently receiving wide spread media recognition publicizing the problems with the main mirror of the Hubble Space Telescope.

Although astronomical imaging is still one of the primary applications of image restoration, many other applications are being developed, for instance in the medical imaging, law enforcement and forensic science fields to name but a few. Other prominent applications are the restoration of aging photographs and motion pictures, and the removal of artifacts resultant to the application of block based compression techniques.

The uses for image restoration mentioned here cannot begin to do complete justice to the possibilities that actually exist in practice. Hence the need for better and more

accurate image restoration algorithms.

## 1.1 Background

Digital image restoration deals with the study of methods to recover the original scene from degraded versions of that scene. Techniques are oriented towards modelling a degradation process, then using that knowledge to undo the effects of those degradations. Some common degrading effects on images are random observation noise, photographic out-of-focus effects, motion blur, and atmospheric turbulence blurring. This field is generally considered distinct from the area of image enhancement, which is purely designed to manipulate an image without making use of degradation models. The primary objective of image enhancement is solely to produce results that are pleasing to the observer without the primary importance set on image accuracy that is prominent in the image restoration field.

Figure 1.1 shows a schematic of the most common image degradation system. For simplicity sake, the additive noise is modeled to be zero mean, Gaussian independent and identically distributed (iid) noise. Some methods that will not be covered in this work model the noise as signal dependant, which often leads to non-linear solutions. An example would be work presented in [12]. The degrading function or point spread function (PSF), designated in Figure 1.1 as $\mathbf{H}$, is usually considered to be a spatially invariant discrete operator. This leads to the degradation process modeled as two-dimensional linear convolution.

### 1.1.1 Why Image Restoration is Difficult

Generally speaking, image recording processes tend to loose image information. The single most prominent attribute for all image degradations (see Section 2.1) is that of a band limiting process. This constitutes a "many-to-one" transformation, and

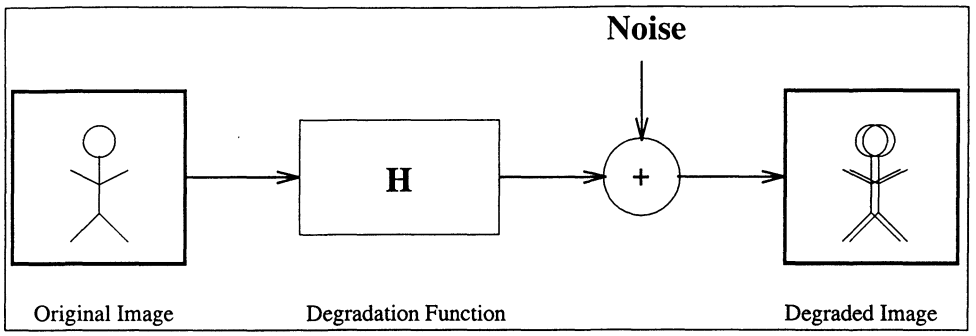


Figure 1.1: Image Degradation Process

introduces ambiguity when one attempts to reverse the process to regain the original image. As will be discussed in Chapter 2, a singular value decomposition of any band limiting blur operator **H** will have zeros or near-zero singular values. This makes direct inversion of the operator **H** impossible unless prior information is introduced into the solution method.

### 1.1.2 Regularization

The most common method of dealing with the ill-posed nature of the inverse filtering process is *regularized least squares*, or simply *regularization.* The most common method of regularization is referred to as *Tikhonov regularization*[56].

Regularization seeks to improve the image estimate by introducing a bias into the solution. Quantifying the optimal trade-off that is created by this bias is in effect the main challenge in implementing a regularized solution. There are many methods that employ regularized least squares in their solutions, some of which which will be covered in Chapter 2. A good review of methods for choosing a regularization parameter and the relationships between each other can be found in [10].

## 1.2 Restoration Methods

The scope of digital image restoration methods to be covered in this thesis can be divided into two main groups; stochastic methods and deterministic methods. These two groups can again be divided into blind and nonblind methods. The blind techniques can be further sub-divided into two groups; *a priori* blur estimation techniques where the PSF is determined using the degraded image, and *a posteriori* restoration where the PSF and image are jointly estimated. Generally, *a priori* blur estimation techniques are less complicated but limited in scope due to constraints on the PSF model.

### 1.2.1 Stochastic Methods

#### Bayesian Methods

Bayesian methods assume a structure to the *a posteriori* probability density function (PDF) of the original image given the degraded image. To accomplish this, the PDFs of both the image and additive noise are both required. For the maximum *a posteriori* (MAP) estimator, the image and noise are both considered to be multivariate Gaussian with zero mean. The noise is also assumed to be iid.

Associated with the MAP estimator is the maximum likelihood estimator (ML or MLE). Here the image is considered to be a non-random vector, maximizing the likelihood function. Originally, these estimators did not account for noise in the problem formulation, and so were not robust when used for noise contaminated images. Later versions include noise in the model, and are developed for the blind case. Most use the expectation maximization algorithm (EM) to find the ML solution[27][35][36][39].

### Wiener Filtering

The Wiener filter is the linear minimum variance estimator. This filter is designed to minimize the mean-squared error between the original and restored images. This filter is equivalent to the MAP estimator when the multivariate Gaussian assumption is used.

### Kalman Filtering

Kalman filtering uses state-space equations in a recursive dynamic system. The Kalman filtering process consists of two stages; a prediction step and an update step. The state dimension may be quite large for two-dimensional blurs, so the use of a Kalman filter is computationally expensive, and therefore not as practical as other methods. Sub-optimal but efficient alternatives to the standard Kalman filter have been developed, in particular the reduced update Kalman filter (RUKF)[61][62] and the reduced order model Kalman filter (ROMKF)[1].

### Generalized Cross Validation

The method of generalized cross validation (GCV) uses the "leave one out" principle to test a solution's applicability to the regularized least squares criterion. This method is flexible, and is used to estimate regularization parameters and PSF parameters in a blind procedure [47][48].

### 1.2.2 Algebraic Methods

#### Tikhonov Miller Regularization

This is a class of methods for finding a solution to a linear system of equations where the observation vector is contaminated by noise. This class of methods is to solve ill-posed problems by introducing *a priori* information, usually in the form of a smoothness constraint. Differences in these related methods are usually in the problem formulation, resulting in slightly different regularization parameters[6][10][15][16][17][18][20] [44][50].

### 1.2.3 Iterative Deconvolution Methods

#### Successive Approximations

Related to the algebraic, regularized methods are iterative ones that are based on successive approximations of the direct inversion operator by using an appropriate expansion. These methods when unconstrained converge to the direct solution. Their main use in practice, however involves constraints that can be applied after each iteration, thus reducing the ambiguity that is involved with ill-posed problems [8][24][25][22][23][29][30][34][37][51][55].

These iterative techniques can be modified to improve convergence rates, but the application of constraints between iterations is no longer possible[26][33][46].

#### Projections on to Convex Sets

Another popular iterative method is called alternating projections on to convex sets (POCS). This is a method where known properties of the original image are defined as convex sets. A projection operator can be formulated that finds the closest point on the convex set in the least squares sense, given the originating vector is outside of

that set. This method is covered in [21][24][53] and [65].

#### Blind Deconvolution Methods

The literature covering blind deconvolution is diverse, and includes many different approaches. Some of the most common are reviewed in [32]. In this article, blind deconvolution methods are categorized as:

- Zero sheet separation methods.
- A priori blur estimation.
- ARMA parameter estimation methods.
- Nonparametric, deterministic image constraints restoration.
- Nonparametric methods based on higher oder statistics.

The following references represent a cross section of the current literature on blind deconvolution [2][9][13][31][38][41][45][52][58][60][63].

### 1.2.4 Multichannel Approaches using Wavelet Theory

This approach decomposes a single channel image into multiple channels or sub-bands using wavelet theory. The idea is to treat each separate channel as a linear space-invariant problem, thus allowing for some non-stationarity for the image as a whole. Most schemes apply a known technique to individual channels, for example using linear minimum mean square error (LMMSE) filters[11][28][4] or a Bayesian paradigm[40]. The authors of [4] also report work on using the EM algorithm and constrained least squares within each channel. A multiscale Kalman pre-filter is used in [5]. These methods incorporate both *within* and *between* channel correlations to form a solution. All of the aforementioned are not blind solutions, requiring that the

PSF be known *a priori*. A blind multiresolution method is developed in [64], which uses DFT zeros to estimate the PSF. Some methods for designing complex valued *single side-band* filters have also been developed [49]. With this type of subband filter design, it may be possible to de-couple the relationships between individual channels of the decomposed image.

## 1.3 Scope of Thesis

This research work has a focus on two-dimensional regularized least squares restoration with a major emphasis on blind iterative deterministic methods. To reach that goal, methods with restricted scope will be first analysed to shed light on the specific properties of image restoration. This analysis will start with the degradation system model, how it is applied to direct regularized least squares methods, and then the analysis moves on to iterative methods that converge to the direct solution. The expansion of these ideas to the blind case will then be covered. This work will deal exclusively with monochromatic images, as for practical reasons it is common practice to assume that the RGB components of colour image pixels are independent, ignoring the mutual relations that exist between them.

The history behind image restoration has followed a similar type of development. Since the introduction of digital image restoration until quite recently, most methods assume the point spread function to be known. These methods, known collectively as *a priori* restoration have reached a degree of maturity, since many yield similar results. The problem of joint image and blur identification (blind deconvolution or *a posteriori* identification and restoration) is not as mature, and was therefore a major goal set at the outset of this research work. The extension of this work from the single channel to multichannel case will not be covered, but is assumed to be the next logical step for continuation of this work.

The problem at hand is divided into four parts.

1. Firstly, the underlying essence of the image restoration is identified by reviewing "classical" techniques. The ill-posed nature of the image restoration problem will be shown by analysing the eigensystem of the blurring operator, and some of the most common *regularized least squares* methods will be compared.

2. These techniques will then be cast as iterative algorithms based on the method of successive approximations. The use of nonlinear constraints will be shown as an effective way to incorporate prior knowledge into the restoration process.

3. Using this background, a new iterative procedure using error-adaptive non-parametric regularization and stability functions will be presented. Although developed as an *a priori* method, analysis assumes an error in the PSF operator thus alowing for expansion to a blind algorithm. An *a priori* version of this algorithm is also presented in [19]. Experimental results comparing this new method with the iterative Tikhonov-Miller method will be presented, as well as experimental proof of the increased rate of convergence attributed to the addition of a stability function.

4. This *a priori* method is extended to the blind case by introducing an estimator for the PSF. An operator that is dual to the regularizing function is developed, and a blind iterative algorithm presented. Experimental results are also presented.

The thesis concludes with closing comments and suggestions for future development of these ideas.

## 1.4 Outline of Thesis

Each of the following four chapters form a background for the next, after which a final chapter concludes the thesis with comments and suggestions for future research.

Chapter 2, "Classical Image Restoration Approaches" outlines some of the most common deterministic methods for image restoration by inverse filtering. The image degradation model used throughout this work will be specified, along with elaborations on the difficulties that arise with image deconvolution. Finally, regularized least squares methods will be introduced with an emphasis on direct or algebraic methods.

Chapter 3, "Iterative Image Restoration" presents some of the most common iterative techniques for digital image restoration, as well as describing some of the advantages of iterative methods over direct methods. The technique of applying prior knowledge by the use of non-linear constraints will be discussed, as well as the application of constraints by the use of alternating projections onto convex sets (POCS). Iterative algorithms with higher order convergence rates are also discussed.

Chapter 4, "A Hybrid Algorithm for Iterative Deconvolution" presents an adaptive iterative algorithm that uses nonparametric stability and regularization operators. Error terms will be developed, and subsequently used to estimate optimal operators. Experimental results using this algorithm are presented, as well as experimental proof of increased convergence rate due to the inclusion of a stability operator.

Chapter 5, "Blind Deconvolution" extends the *a priori* methods of the previous chapters to the blind case, where the PSF is not explicitly known. Blind identification methods are roughly divided into two groups; methods that determine the PSF directly from the degraded image before the image restoration process begins and those that jointly determine the PSF and image as the algorithm progresses. Several blind procedures will be discussed, including maximum likelihood estimation (MLE)

and generalized cross validation (GCV). A blind iterative algorithm based on the material presented in Chapter 4 will then be presented with corresponding experimental results.

Finally, Chapter 6 draws some conclusions, making some suggestions for future research.

# Chapter 2

# Classical Image Restoration Approaches

Although methods for restoring images are diverse, a great majority use the same image formation model[3][33]. Initially, this image degradation model will be described, outlining the different types of degradation that are common to the image formation process. As we will see, only a certain number of these degradation sources can be considered dominant, making it possible to simplify the image degradation process to that of a discrete, linear, space invariant system. Some common degradation functions, or *point spread functions* (PSF) will be presented, along with justification for their parametric simplifications. This degradation system will be then re-formulated in a *matrix-vector* form where certain matrix structure properties will be exploited to transform the problem into the discrete Fourier space where dramatic computational efficiencies can be attained. Using further assumptions on the structure of the matrix-vector formulation, this Fourier space transformation can be shown to be an eigenvalue analysis problem, thus shedding light on some important properties of this image degradation system. Once the common point spread functions are analysed, the ill-posed nature of image restoration by deconvolution is revealed.

## 2.1 Image Formation and Degradation Model

### 2.1.1 Continuous Model

A three-dimensional object can be recorded on to a two-dimensional image plane by a recording device such as a camera. This image can be subject to many types of degradation in the imaging process including diffraction effects, atmospheric turbulence, motion, and out-of-focus blurs. The main effect of all of these degradations is that of a bandwidth reduction function. Further, these degradations can change for different physical areas of the image, making the degradation space-variant in nature. Finally, the sensor response of the recording system may be non-linear, further degrading the image.

The case of restoring two-dimensional images from their blurred or degraded counterparts will govern the scope of this work. For the sake of computational efficiency, assumptions that simplify the degradation model will be made. Firstly, the incorporation of nonlinear responses into the degradation model results in an overly complicated restoration algorithm that may well be impossible to use in image restoration in general [33]. Hence we will restrict analysis of the image formation model to that of a linear degradation system characterized by a two-dimensional point spread function (PSF). Further to this, since it is overly complex and burdensome from a computational viewpoint to consider a distinct PSF for each coordinate of an image, we will again restrict further discussion to a space invariant or stationary PSF. These assumptions lead to the most commonly adopted degradation system, that of linear, two-dimensional convolution.

$$r(x,y) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} h(x-s,y-t)s(s,t)dsdt$$

$$= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} h(s,t)s(x-s,y-t)dsdt$$
$$= h(x,y) * s(x,y) \tag{2.1}$$

where $r(x,y)$ is a received or recorded image, $h(x,y)$ is a point spread function, $s(x,y)$ is the underlying or desired two-dimensional image, and $*$ denotes linear two-dimensional convolution.

In addition to the deterministic degradations described thus far, images are invariably further degraded by additive stochastic observation noise, revising 2.1 to:

$$r(x,y) = h(x,y) * s(x,y) + n(x,y) \tag{2.2}$$

where $n(x,y)$ is assumed to be zero-mean white Gaussian noise with a variance $\sigma_n^2$.

### 2.1.2 Discrete Model

The discrete form of the equation (2.2) is achieved by sampling the received image $r(x,y)$ on a uniform 2-D lattice of dimension $M \times N$. This is done only after appropriately band-limiting $r(x,y)$ using an anti-aliasing filter. At this point it is important to note that the original continuous image, under practical situations, cannot be perfectly recovered after sampling since the conditions to do so cannot be satisfied in practice[33]. These sets of conditions for perfect reconstruction are as follows:

1. The original image $s(x,y)$ and the PSF $h(x,y)$ must both be bandlimited and sampled at Nyquist rate. The anti-aliasing filter must also have all pass characteristics for all frequencies less than the maximum bandwidth for both $s(x,y)$ and $h(x,y)$.

2. If $s(x,y)$ and $h(x,y)$ are not bandlimited, the anti-aliasing filter must be an *ideal* 2-D low-pass filter with a cut-off frequency equal to Nyquist frequency.

The first set of conditions, although attainable are rarely met in practice as images are seldom bandlimited. The second set of conditions cannot be met in practice due to the requirement of an ideal anti-aliasing filter.

Incorporating all of these assumptions, we can now say that the degradation system for most imaging processes can be approximately expressed in discrete form as two-dimensional linear convolutive blurring with additive white Gaussian noise:

$$r(i,j) = \sum_{k=1}^{M}\sum_{l=1}^{N} h(i-k, j-l)s(k,l) + n(i,j) \tag{2.3}$$

where $r(i,j)$ is a sampled pixel value in an $M \times N$ recorded image, $h(i,j)$ is the sampled unit impulse response of the point-spread function (PSF) for the image recording system, $s(i,j)$ is the sampled underlying or desired signal, and $n(i,j)$ is additive white Gaussian noise. The assumption here and throughout this work is that all images are appropriately band limited before sampling at a sufficient rate to prevent aliasing. Since the same ideas expressed here can be used for square as well as rectangular images without loss of generality, for simplicity of notation images will now be assumed square of dimension $N \times N$. Equation (2.3) is now re-written as:

$$\begin{aligned} r(i,j) &= \sum_{k=1}^{N}\sum_{l=1}^{N} h(i-k, j-l)s(k,l) + n(i,j) \\ &= h(i,j) * s(i,j) + n(i,j) \end{aligned} \tag{2.4}$$

where $*$ denotes two-dimensional linear convolution.

## 2.2 Common Point Spread Functions

Many of the most common blurs or point spread functions can be expressed parametrically, with the set of parameters determining the extent of the blur. Although continuous in nature, point spread functions will be presented in a discrete format

here as they will be used in that context. As previously discussed in Section 2.1.2, the exact relation between the discrete and continuous models cannot be quantified due to aliasing. As established before, the assumption of a sufficiently high sampling rate to prevent aliasing error is made here.

Since the underlying physical phenomena in the image degradation process is well understood, certain prior assumptions can be used to constrain point spread functions. In all PSF model cases, both the underlying and recorded image are positive, real-valued signals. From the image degradation model (2.4) we see that the PSF must be positive and real-valued as well.

$$h(i,j) \geq 0 \quad \forall \; i,j \tag{2.5}$$

An assumption of energy preservation will also be made, since the degradations in the imaging system are passive in nature. Consequently, any discrete PSF will be constrained such that:

$$\sum_{i,j \in S_h} h(i,j) = 1.0 \tag{2.6}$$

where $S_h$ is the area of support for $h(i,j)$, which for practical reasons will be considered finite in extent. Finally, all blur models considered will be space-invariant.

The following sections will outline some of the most frequently encountered PSF models for practical situations.

### 2.2.1 Linear Motion Blur

This type of blur is caused by a relative motion between the imaging device, e.g. camera, and the object being recorded. There may many forms of relative motion like translation, rotation, change of scale or a combination of these, but by far the most important case is that of translation. To simplify this model, a constant velocity

is assumed in the horizontal direction only.

$$h(i) = \begin{cases} \frac{1}{L} & if -\frac{L}{2} \leq i \leq \frac{L}{2} \\ 0 & otherwise \end{cases} \tag{2.7}$$

where the single parameter $L$ is the length of motion in pixels.

### 2.2.2 Uniform Out-of-Focus Blur

A simplified out-of-focus model can be derived using a geometrical approach[54] that results in a uniform intensity distribution in the area of support for the PSF. This area of support, which is circular with radius $R$, is given by:

$$h(i,j) = \begin{cases} \frac{1}{\pi R^2} & if \sqrt{i^2 + j^2} \leq R \\ 0 & otherwise \end{cases} \tag{2.8}$$

### 2.2.3 Atmospheric Turbulence Blur

Atmospheric turbulence blur can be simplified by approximating it with a Gaussian function[42].

$$h(i,j) = Kexp\left(-\frac{i^2 + j^2}{2\sigma_G^2}\right) \tag{2.9}$$

The parameter $\sigma_G$ determines the severity of the blur, and the constant $K$ is chosen such that (2.6) is satisfied. The above function as given does not have finite support as assumed previously, so the PSF must be truncated appropriately.

### 2.2.4 Uniform 2-D Blur

This model is similar to (2.8), except that the area of support is not considered circular. This type of PSF model can be used when a separable function will facilitate deconvolution. Also in this case, it may be empirically verified that the extent of

degradation is not as severe for similar areas of support.

$$h(i,j) = \begin{cases} \frac{1}{L^2} & if -\frac{L}{2} \leq i \leq \frac{L}{2}, -\frac{L}{2} \leq j \leq \frac{L}{2} \\ 0 & otherwise \end{cases} \tag{2.10}$$

We conclude this section by noting that all of the above PSF functions, with the exception of the atmospheric blur PSF, are characterized by a regular pattern of zeros in the frequency domain. Although the atmospheric blur PSF does not explicitly have zeros in the DFT domain, it does have many near or numerically zero values. The significance of this fact will be shown in Section 2.4, where the eigensystem of the blurring operation will be discussed.

## 2.3 Block-Toeplitz and Block-Circulant Matrices

It is convenient to express (2.4) in a matrix-vector form. Here the received and underlying images, $r(i,j)$ and $s(i,j)$ respectively, are lexicographically ordered into vectors of size $N^2 \times 1$. The point spread function is transformed into a $N^2 \times N^2$ blurring matrix $\mathbf{H}$ which yields:

$$\mathbf{r} = \mathbf{H}\mathbf{s} + \mathbf{n} \tag{2.11}$$

where $\mathbf{r}$, $\mathbf{s}$ and $\mathbf{n}$ are columnized or lexicographically ordered versions of the received, desired, and noise images respectively and $\mathbf{H}$ is an $N^2 \times N^2$ convolutive blurr operator.

The convolutive blur operator $\mathbf{H}$ is block-Toeplitz in structure, and will be square if $\mathbf{r}$ and $\mathbf{s}$ are of the same dimension[1]. A block-Toeplitz matrix can be replaced by a block-circulant one with appropriate zero-padding for $\mathbf{h}_i$ and $\mathbf{s}$ since these matrix

[1]This will be true even when the images are $M \times N$ instead of the assumed $N \times N$

types are similarly structured[3][33]. Refer to Appendix A for a more detailed discussion on the justification for making a block-circulant assumption for the blurring matrix $\mathbf{H}$.

## 2.4 The Eigensystem of the Blurring Matrix

Assuming that $\mathbf{H}$ is block-circulant, and that the recorded image $r(i,j)$ is sufficiently zero padded such that the results of linear and circular convolution are the same, it is well known that $\mathbf{H}$ can be diagonalized using the discrete Fourier transform[14]:

$$\mathbf{H} = \mathbf{W}\mathbf{D}\mathbf{W}^H \tag{2.12}$$

where $\mathbf{D}$ is a complex valued diagonal matrix with nonzero elements consisting of the 2-dimensional discrete Fourier transform (DFT) coefficients of the the PSF $h(i,j)$, $\mathbf{W}$ is a matrix with columns of eigenvectors which are the normalized complex exponential basis functions for the DFT, and the superscript $H$ denotes the Hermitian transpose (See Appendix A for further details on the eigendecomposition of block-circulant matrices). The eigenvector matrix $\mathbf{W}$ is such that a two-dimensional discrete Fourier transform can be achieved by the operation:

$$\mathcal{F}\{\mathbf{x}\} = N\mathbf{W}^H\mathbf{x}$$

where $\mathcal{F}\{.\}$ denotes the discrete Fourier transform operator. Returning to the matrix-vector degradation model of (2.11), and noting that $\mathbf{W}^{-1} = \mathbf{W}^H$ since $\mathbf{W}$ is unitary:

$$\begin{aligned} \mathbf{r} &= \mathbf{H}\mathbf{s} + \mathbf{n} \\ &= \mathbf{W}\mathbf{D}\mathbf{W}^H\mathbf{s} + \mathbf{n} \\ \mathbf{W}^H\mathbf{r} &= \mathbf{D}\mathbf{W}^H\mathbf{s} + \mathbf{W}^H\mathbf{n} \end{aligned}$$

$$\begin{aligned} \mathcal{F}\{\mathbf{r}\} &= \mathbf{D}\mathcal{F}\{\mathbf{s}\} + \mathcal{F}\{\mathbf{n}\} \\ \mathbf{r} &= \mathcal{F}^{-1}\{\mathbf{D}\mathcal{F}\{\mathbf{s}\}\} + \mathbf{n} \end{aligned} \tag{2.13}$$

where again $\mathcal{F}\{.\}$ denotes the discrete Fourier transform operator. Recall that $\mathbf{r}$ and $\mathbf{s}$ are column vectors that are the received and underlying images in lexicographic order. Since $\mathbf{D}$ is a diagonal matrix with $\mathcal{F}\{\mathbf{h}\}$ across the diagonal, the right hand side of the last line of (2.13) is simply the inverse Fourier transform of the $N^2$ scalar multiplications of the corresponding DFT coefficeints for $h(i,j)$ and $r(i,j)$. Hence this $N^2 \times N^2$ system can be decomposed into $N^2$ independent scalar relations:

$$\mathcal{R}(u,v) = \mathcal{H}(u,v)\mathcal{S}(u,v) + \mathcal{N}(u,v) \quad \forall \ u,v \tag{2.14}$$

where $\mathcal{R}(u,v)$ $\mathcal{H}(u,v)$ and $\mathcal{S}(u,v)$ are the 2-D discrete Fourier transform coefficients at frequency index $(u,v)$. Hence the image degradation model can be expressed in both the spatial and frequency domains using the block-circulant blurring matrix assumption.

To conclude this section, we suffice to say that the discrete Fourier transform coefficients of the discrete point spread function $h(i,j)$ are eigenvalues of the block-Toeplitz blurring matrix $\mathbf{H}$ due to the block-circulant assumption. As a result, the DFT spectrum of $h(i,j)$ can be an invaluable analysis tool for the image restoration problem.

## 2.5 Least Squares Approach

The system (2.11) can be directly solved by finding an estimate desired signal $\hat{\mathbf{s}}$ that satisfies a least squares fit to the data by minimizing the noise norm, given that $\mathbf{H}$ is known and follows one of the models of Section 2.2.

$$\hat{\mathbf{s}} = \arg\min_{\mathbf{s}} \|\mathbf{n}\|_2^2 = \arg\min_{\mathbf{s}} \|\mathbf{r} - \mathbf{H}\hat{\mathbf{s}}\|_2^2 \tag{2.15}$$

This leads to the generalized inverse filter:

$$\begin{aligned}(\mathbf{H}^T\mathbf{H})\hat{\mathbf{s}} &= \mathbf{H}^T\mathbf{r} \\ \hat{\mathbf{s}} &= (\mathbf{H}^T\mathbf{H})^{-1}\mathbf{H}^T\mathbf{r} \end{aligned} \tag{2.16}$$

Using the same assumptions and approach outlined in Section 2.4, the solution can be transformed to the frequency domain[3]. This diagonalization approach can be reformulated as:

$$\begin{aligned}\mathbf{H}^T\mathbf{H}\hat{\mathbf{s}} &= \mathbf{H}^T\mathbf{r} \\ \mathbf{W}\mathbf{D}^{\mathbf{H}}\mathbf{W}^{\mathbf{H}}\mathbf{W}\mathbf{D}\mathbf{W}^{\mathbf{H}}\hat{\mathbf{s}} &= \mathbf{W}\mathbf{D}^{\mathbf{H}}\mathbf{W}^{\mathbf{H}}\mathbf{r} \\ \mathbf{D}^{\mathbf{H}}\mathbf{D}\mathbf{W}^{\mathbf{H}}\hat{\mathbf{s}} &= \mathbf{D}^{\mathbf{H}}\mathbf{W}^{\mathbf{H}}\mathbf{r} \\ \mathbf{D}^{\mathbf{H}}\mathbf{D}\mathcal{F}\{\hat{\mathbf{s}}\} &= \mathbf{D}^{\mathbf{H}}\mathcal{F}\{\mathbf{r}\} \\ \mathcal{H}^*(u,v)\mathcal{H}(u,v)\hat{\mathcal{S}}(u,v) &= \mathcal{H}^*(u,v)\mathcal{R}(u,v) \quad \forall\ u,v \\ \hat{\mathcal{S}}(u,v) &= \frac{\mathcal{H}^*(u,v)\mathcal{R}(u,v)}{|\mathcal{H}(u,v)|^2} \quad \forall\ u,v \end{aligned} \tag{2.17}$$

where again $\hat{\mathcal{S}}$,$\mathcal{R}$ and $\mathcal{H}$ are the 2-D discrete Fourier transforms of $\hat{s}(i,j)$, $r(i,j)$ and $h(i,j)$ respectively, and * denotes complex conjugation. The main issues to overcome in this solution, which are evident on inspection of the last line of (2.17), and in view of the fact $|\mathcal{H}(u,v)|$ is small in some regions for the PSF's under consideration, are:

- high frequency additive noise amplification, particularly if the PSF is inherently low pass with small or zero DFT coefficients in the higher frequency bands.
- retrieval of lost information. If the columnspace of the convolutive operator $\mathbf{H}$ does not span the entire signal space, it represents a many-to-one transformation.

The noise amplification problem cited in the first item above is especially prominent due to the energy preservation constraint on the PSF given in (2.6), as:

$$\left| \frac{\mathcal{H}(u,v)}{|\mathcal{H}(u,v)|^2} \right| \geq 1.0 \qquad \forall\ u,v \tag{2.18}$$

since

$$|\mathcal{H}(u,v)| \leq 1.0 \qquad \forall\ u,v \tag{2.19}$$

In fact, the equality relations above are only true for one coefficient of $\mathcal{H}$, specifically at the frequency indices $\mathcal{H}(0,0)$.

## 2.6 Regularized Image Restoration

Regularization theory was introduced as a solution to *discrete ill-posed* problems. Strictly speaking though, the term "ill-posed" describes continuous systems with an infinite dimensional space[18]. Despite this, certain finite dimensional discrete systems have properties similar to that of continuous ill-posed problems like Fredholm integral equations of the first kind[37]. A problem is considered ill-posed if an arbitrarily small perturbation in a data set could cause an arbitrarily large perturbation in the solution.

A class of discrete, ill-posed problems can be characterized by a linear system of equations:

$$\mathbf{Ax} = \mathbf{b}, \quad \mathbf{A} \in \Re^{n \times n} \tag{2.20}$$

where only the vector $\mathbf{b}$ contains additive observation noise. This is usually cast as a possibly over-determined linear least squares problem to minimise the residual norm:

$$\min_x \|\mathbf{Ax} - \mathbf{b}\|_2^2, \quad \mathbf{A} \in \Re^{m \times n}, \quad m \geq n \tag{2.21}$$

The above (2.20) and (2.21) are said to be discrete ill posed problems if the following are satisfied[18]:

1. the singular values of $\mathbf{A}$ decay gradually to zero

2. the ratio between the smallest and largest non-zero singular values is large

The difficulty in solving problems like 2.20 and 2.21 under these conditions is that the system of equations is unstable due to the small singular values. It is therefore necessary to incorporate prior information to stabilize the problem, thus determining a meaningful solution.

In comparing (2.11) to (2.20) and in view of the low-pass characteristic of the PSF's we deal with, we see that image restoration is an ill-posed problem.

### 2.6.1 Tikhonov Regularization

The best known form of regularization is *Tikhonov* regularization. Essentially, prior information is incorporated in the problem by the use of a side constraint or stabilizing functional. This constraint embodies some known feature of the solution, which in the case of image restoration is usually smoothness. Other common constraints include maximum power and maximum entropy. The form of constraint usually chosen is one that facilitates the mathematical analysis of the problem, that is:

$$\|\mathbf{Cs}\|_2^2 \tag{2.22}$$

where $\mathbf{C}$ is a real-valued block-circulant matrix of size $N^2 \times N^2$ that sets a semi-norm on the solution[15]. The resultant objective function is:

$$\min_{\mathbf{s}} \left\{ \|\mathbf{Hs} - \mathbf{r}\|_2^2 + \lambda \|\mathbf{Cs}\|_2^2 \right\} \tag{2.23}$$

where $\lambda$, referred to as the *regularization parameter*, is a positive scalar constant that controls the size or norm of the solution vector $\mathbf{s}$. The matrix $\mathbf{C}$, referred to as the *regularization operator* on the other hand determines the nature of the side constraint. In example, if $\mathbf{C}$ is a high pass operator, the higher frequency bands of the solution vector will be suppressed, thus enforcing a smoothness constraint.

The regularization parameter $\lambda$ controls the weight given to the minimization of the side constraint relative to the minimization of the residual norm, and therefore controls the trade-off between the first and second terms of 2.23. With larger values of $\lambda$, which is the case of more regularization, the restored image exhibits ringing effects. On the other hand, with smaller values of $\lambda$, the effect of noise amplification is more pronounced. There are various methods to estimate the value for $\lambda$, many of which are analysed in [10].

The regularization operator $\mathbf{C}$ is often a first or second derivative operator. Indeed, a common choice for this operator is the high pass two-dimensional Laplacian operator[3][33]. Another choice for regularization operator is the identity matrix, in which case the Tikhonov problem is said to be in standard form[15]. More examples for the regularization operator $\mathbf{C}$ are given in [25].

The resulting solution takes the form:

$$\begin{aligned} \left(\mathbf{H}^{\mathbf{T}}\mathbf{H} + \lambda \mathbf{C}^{\mathbf{T}}\mathbf{C}\right)\hat{\mathbf{s}} &= \mathbf{H}^{\mathbf{T}}\mathbf{r} \\ \hat{\mathbf{s}} &= \left(\mathbf{H}^{\mathbf{T}}\mathbf{H} + \lambda \mathbf{C}^{\mathbf{T}}\mathbf{C}\right)^{-1}\mathbf{H}^{\mathbf{T}}\mathbf{r} \end{aligned} \tag{2.24}$$

which, in the frequency domain due to block-circulant assumptions made on $\mathbf{H}$ and $\mathbf{C}$ can be written as:

$$\hat{\mathcal{S}}(u,v) = \frac{\mathcal{H}^*(u,v)\mathcal{R}(u,v)}{|\mathcal{H}(u,v)|^2 + \lambda\,|\mathcal{C}(u,v)|^2} \quad \forall\ u,v \tag{2.25}$$

### Constrained Least Squares Approaches

Identical results to (2.24) can be obtained in various ways. In example, a quadratic constraint can be applied to the least-squares problem by setting the problem as:

$$\min_x \|\mathbf{Hs} - \mathbf{r}\|_2^2 \quad \text{subject to} \quad \|\mathbf{Cs}\|_2^2 \leq E^2 \tag{2.26}$$

then using the Lagrange multiplier formulation:

$$\mathcal{L}(\mathbf{s}, \lambda) = \|\mathbf{Hs} - \mathbf{r}\|_2^2 + \lambda \left( \|\mathbf{Cs}\|_2^2 - E^2 \right) \tag{2.27}$$

where $\lambda$ is the Lagrangian multiplier and $E$ is a scalar constant that sets a specific limit to the norm $\|\mathbf{Cs}\|_2^2$.

Alternative approaches to the CLS problem are[24]:

$$\min_x \|\mathbf{Hs} - \mathbf{r}\|_2^2 \quad \text{subject to} \quad \|\mathbf{Cs}\|_2^2 = E^2 \tag{2.28}$$

where the equality of side constraint of (2.26) is strictly enforced:

$$\min_x \|\mathbf{Cs}\|_2^2 \quad \text{subject to} \quad \|\mathbf{Hs} - \mathbf{r}\|_2^2 = \epsilon^2 \tag{2.29}$$

where the roles of each side of (2.28) are reversed.

As previously mentioned, the resulting solution to the above (2.26), (2.28) and (2.29) are the same as (2.24). Differences come in the exact value chosen for the regularization parameter $\lambda$.

Yet another approach to constrained least squares is set as[6]:

$$\min_x \|\mathbf{Hs} - \mathbf{r}\|_2^2 \quad \text{subject to} \quad \|\mathbf{Cs} - \mathbf{r}\|_2^2 = \epsilon^2 \tag{2.30}$$

Again, the method of undetermined Lagrange multipliers can be used to obtain the solution:

$$\hat{\mathbf{s}} = \left( \mathbf{H^T H} + \lambda \mathbf{C^T C} \right)^{-1} (\mathbf{H} + \lambda \mathbf{C})^{\mathbf{T}} \mathbf{r}. \tag{2.31}$$

Although (2.31) differs from (2.24), it is mentioned here due to its similarity to the proposed solution presented in Chapter 4, in particular equation (4.2).

### 2.6.2 Miller's Method

In the Miller regularization approach[44], the restoration problem is set in a similar way to those in Section 2.6.1. Here, a vector is sought that satisfies two constraints

simultaneously, which are:

$$\|\mathbf{Cs}\|_2^2 \leq E^2$$

and

$$\|\mathbf{Hs} - \mathbf{r}\|_2^2 \leq \epsilon^2$$

Miller's approach combines the two constraints into one quadrature formula that results in:

$$M(\alpha, \mathbf{s}) = \|\mathbf{Hs} - \mathbf{r}\|_2^2 + \lambda \|\mathbf{Cs}\|_2^2 \leq 2\epsilon^2 \tag{2.32}$$

where the parameter $\lambda$ is set to $\left(\frac{\epsilon}{E}\right)^2$. Setting the gradient of (2.32) to zero and solving for $\mathbf{s}$ again results in (2.24).

## 2.7 Synopsis of Classical Restoration Approaches

The above approaches lead to the very same result of (2.24), with subtle differences in each approach attributed to the values assigned to the regularization parameter $\lambda$, and the regularization operator $\mathbf{C}$. Given this similarity, many researchers refer to all these related methods as "Tikhonov-Miller" regularization.

To adequately analyse the subtle differences between these regularization methods, as well as compare them to the generalized inverse (pseudo-inverse) filter, a set theoretic approach to analysis has been taken in [24] and [25]. In these articles, regularized solutions must reside in the intersection of two ellipsoids representing convex sets, namely:

$$Q_{\mathbf{s}\|\mathbf{r}} = \left\{\mathbf{s} : \|\mathbf{r} - \mathbf{Hs}\|_2^2 \leq \epsilon^2\right\} \tag{2.33}$$

where $\epsilon$ is an estimate of the noise norm, and

$$Q_{\mathbf{s}} = \left\{\mathbf{s} : \|\mathbf{Cs}\|_2^2 \leq E^2\right\} \tag{2.34}$$

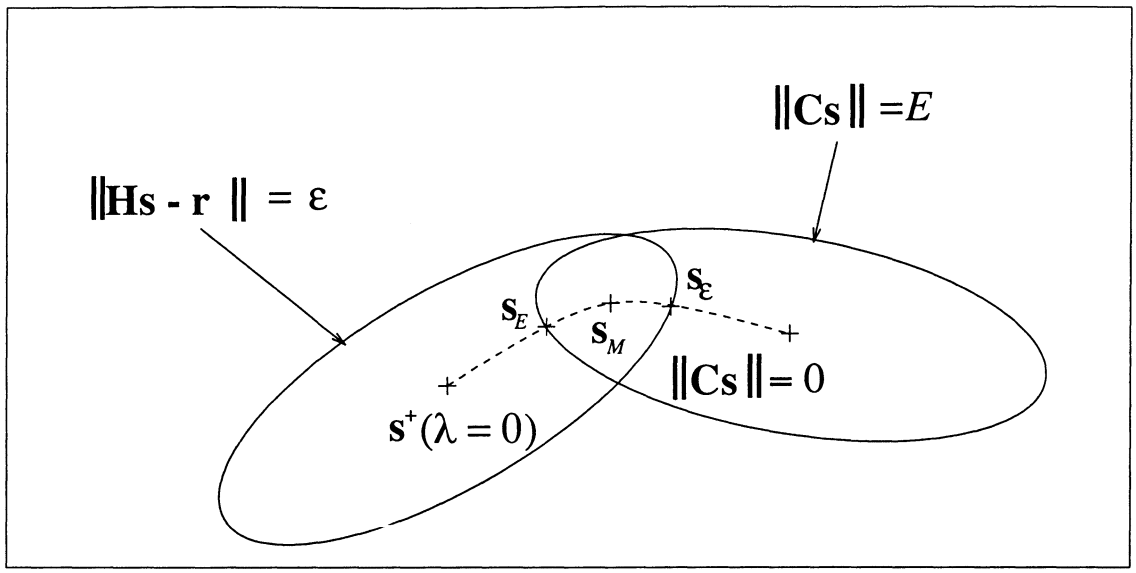


Figure 2.1: Intersection of two ellipsoids (after Katsaggelos *et al*)

where $E$ is a prescribed constant chosen to ensure that the two ellipsoids (sets) intersect. The pseudo-inverse solution, which is at the center of ellipsoid $Q_{\mathbf{s}||\mathbf{r}}$, is a special case for regularized image restoration with $\lambda$ set to zero, that is to say with no regularization. Figure 2.1 shows these two ellipsoids and the possible solutions where $\mathbf{s}_M$ indicates the Miller solution to (2.32), $\mathbf{s}_E$ the CLS solution to (2.28) and $\mathbf{s}_\epsilon$ the dual CLS solution to (2.29). The pseudo-inverse solution is designated as $\mathbf{s}^+$.

# Chapter 3

# Iterative Image Restoration

In the previous chapter, several classical direct algebraic methods were introduced, outlining some of the difficulties in solving the image restoration problem. The *ill-conditioned* nature of this operation was shown, as well as the many-to-one property of some of the most common point spread functions. Here we will introduce some important iterative versions of these methods. Iterative methods are popular as they use a series expansion to evaluate the blurring matrix inverse, thus avoiding direct matrix inversion. Most importantly though, they allow for the application of constraints at each iteration that, when properly applied, can allow convergence to a unique solution.

A simple iterative method that converges linearly to the direct inverse filter will be introduced, as it offers a simplified analysis for the convergence properties of many of the other methods. This method, known by various names, is limited in its use due to constraints on the PSF $h(i,j)$ that must be applied to ensure convergence. Another method referred to as "re-blurring" will be shown to be more universal in nature, with less restrictions on the PSF model. This re-blurring method will be shown to be equivalent to the generalized inverse or pseudo-inverse solution (2.16) presented in Section 2.5. The inclusion of regularization is then discussed, with the iterative

Tikhonov-Miller method presented. These methods offer a linear convergence rate, but variations with higher order convergence rates are possible. Finally, the use of prior knowledge about the target solution by applying nonlinear constraints will be discussed.

## 3.1 Linear Convergence Iterative Technique

This method is known as the Van Clittert[59], Landweber[37] or Bially[7] iteration. Starting with the degradation model (2.11), and ignoring the additive stochastic noise, the blurring process can be rewritten as:

$$\beta \mathbf{H}\mathbf{s} = \beta \mathbf{r} \tag{3.1}$$

where $\beta$ is either a constant or linear operator[46][51]. If we let the operator matrix $\mathbf{T} = (\mathbf{I} - \beta\mathbf{H})$, then we can evaluate $(\mathbf{I} - \mathbf{T})^{-1}$ using the Neuman series expansion[46]:

$$(\mathbf{I} - \mathbf{T})^{-1} = \sum_{n=0}^{\infty} \mathbf{T}^n \tag{3.2}$$

therefore

$$(\mathbf{I} - (\mathbf{I} - \beta\mathbf{H}))^{-1} = \sum_{n=0}^{\infty} (\mathbf{I} - \beta\mathbf{H})^n$$

$$(\beta\mathbf{H})^{-1} = \sum_{n=0}^{\infty} (\mathbf{I} - \beta\mathbf{H})^n \tag{3.3}$$

which is true if $\mathbf{H}$ represents a continuous function and $\|\mathbf{I} - \beta\mathbf{H}\| < 1$. If (3.3) is applied to both sides of (3.1), we then have a new solution to the inverse problem:

$$\mathbf{s} = \sum_{n=0}^{\infty} (\mathbf{I} - \beta\mathbf{H})^n \beta\mathbf{r} \tag{3.4}$$

which is the direct inverse solution when $\beta$ is appropriately chosen. If the $k$th approximation of $\mathbf{s}$ is defined as:

$$\mathbf{s}_k = \sum_{n=0}^{k-1} (\mathbf{I} - \beta\mathbf{H})^n \beta\mathbf{r} \tag{3.5}$$

then the next iteration is:

$$\begin{aligned}
\mathbf{s}_{k+1} &= \sum_{n=0}^{k} (\mathbf{I} - \beta\mathbf{H})^n \beta\mathbf{r} \\
&= \left[\mathbf{I} + (\mathbf{I} - \beta\mathbf{H}) + (\mathbf{I} - \beta\mathbf{H})^2 + \ldots + (\mathbf{I} - \beta\mathbf{H})^k\right] \beta\mathbf{r} \\
&= \mathbf{I}\beta\mathbf{r} + (\mathbf{I} - \beta\mathbf{H}) \left[\mathbf{I} + (\mathbf{I} - \beta\mathbf{H}) + (\mathbf{I} - \beta\mathbf{H})^2 + \ldots + (\mathbf{I} - \beta\mathbf{H})^{k-1}\right] \beta\mathbf{r} \\
&= \beta\mathbf{r} + (\mathbf{I} - \beta\mathbf{H})\, \mathbf{s}_k \\
\mathbf{s}_{k+1} &= \mathbf{s}_k + \beta\,(\mathbf{r} - \mathbf{H}\mathbf{s}_k)
\end{aligned} \tag{3.6}$$

This standard iterative solution can also be derived by making (3.1) a fixed point equation and using the method of successive approximations[33][51].

For this iteration to converge to $\mathbf{H}^{-1}\mathbf{r}$ the real part of the eigenvalues of $\mathbf{H}$ must be positive[8][33][51]. However, convergence is not preferable given the characteristics of most practical blurs and would result in an unusable image estimate. Interestingly enough, it has been shown that truncating the iterations at a certain value of $k$ is in itself a form of regularization[33]. Unfortunately, this iteration has limited use due to the positivity constraint on the real part of the complex eigenvalues, as this excludes the linear motion and uniform out-of-focus blurs introduced in Section 2.2.

## 3.2 Re-blurring Methods

One way to ensure that the real parts of all eigenvalues of a blurring function are positive is to pre-filter it with an abcissa reversed version of itself[21][29]. To see this, we realize the DFT coefficients are the eigenvalues of the blurring matrix $\mathbf{H}$; hence:

$$h(i,j) * h(-i,-j) \Leftrightarrow \mathcal{H}\mathcal{H}^* = |\mathcal{H}|^2 \tag{3.7}$$

so the eigenvalues are indeed positive. We then convolve both the PSF and the received image by the abcissa-reversed PSF to ensure convergence. This is the same

as replacing the PSF with its autocorrelation. The relation of (3.1) is then revised to:

$$\beta \mathbf{H^T H s} = \beta \mathbf{H^T r} \tag{3.8}$$

which, when $\beta = 1$ is the least squares solution introduced in Section 2.5. The corresponding successive approximation iteration for this, or a re-blurred equivalent to (3.6) is:

$$\mathbf{s}_{k+1} = \mathbf{s}_k + \beta \left( \mathbf{H^T r} - \mathbf{H^T H s_k} \right) \tag{3.9}$$

For (3.9) to converge, it is required that[3][8][24][33]:

$$\left| 1 - \beta \left| \mathcal{H}(u,v) \right|^2 \right| < 1 \quad \forall \ u, v \tag{3.10}$$

where $\mathcal{H}(u,v)$ are equivalently the the eigenvalues and DFT coefficients of the blurring matrix $\mathbf{H}$. From this, a range on $\beta$ can also be set:

$$0 < \beta < \frac{2}{\left| \mathcal{H}_{\max} \right|^2} \tag{3.11}$$

where $\mathcal{H}_{\max}$ is the maximum norm DFT coefficient $\mathcal{H}(u,v)$. Imposing the energy preservation constraint of (2.6) and the positivity constraint of (2.5) would then make $|\mathcal{H}_{\max}| = 1$, therefore:

$$0 < \beta < 2 \tag{3.12}$$

## 3.3 Iterative Tikhonov-Miller Method

To derive the iterative Tikhonov-Miller algorithm, the method of steepest descent can be used[33]. This is accomplished by minimizing (2.23) which we will designate as $\Phi(\mathbf{s})$.

$$\begin{aligned} \Phi(\mathbf{s}) &= \|\mathbf{Hs} - \mathbf{r}\|_2^2 + \lambda \|\mathbf{Cs}\|_2^2 \\ &= \mathbf{s^T H^T H s} - 2\mathbf{s^T H^T r} + \mathbf{r^T r} + \lambda \mathbf{s^T C^T C s} \end{aligned} \tag{3.13}$$

hence:

$$\begin{aligned} \nabla_{\mathbf{s}}\Phi(\mathbf{s}) &= 2\left[\mathbf{H^T H s} - \mathbf{H^T r} + \lambda \mathbf{C^T C s}\right] \\ &= 2\left[\left(\mathbf{H^T H} + \lambda \mathbf{C^T C}\right)\mathbf{s} - \mathbf{H^T r}\right] \end{aligned} \tag{3.14}$$

We then use the steepest descent iteration to formulate the $k$th estimate, where $\mathbf{d}_k$ shown below is the "downhill" direction vector and $\beta$ the positive scaling factor.

$$\begin{aligned} \mathbf{s_{k+1}} &= \mathbf{s_k} + \beta \mathbf{d_k} \\ &= \mathbf{s_k} - \frac{1}{2}\beta \nabla_{\mathbf{s}}\Phi(\mathbf{s})\Big|_{\mathbf{s}_k} \\ &= \mathbf{s}_k - \beta\left(\left(\mathbf{H^T H} + \lambda \mathbf{C^T C}\right)\mathbf{s_k} - \mathbf{H^T r}\right) \\ &= \mathbf{s}_k + \beta\left(\mathbf{H^T r} - \mathbf{s_k}\left(\mathbf{H^T H} + \lambda \mathbf{C^T C}\right)\right) \end{aligned} \tag{3.15}$$

Recalling that the DFT coefficients of a block-circulant matrix are its eigenvalues, the iteration can be shown to converge when[33]:

$$\left|1 - \beta\left(|\mathcal{H}(u,v)|^2 + \lambda\,|\mathcal{C}(u,v)|^2\right)\right| < 1 \quad \forall\ u, v \tag{3.16}$$

which can be rearranged to set bounds on $\beta$:

$$0 < \beta < \frac{2}{|\rho_{max}|} \tag{3.17}$$

where $|\rho_{max}|$ is the maximum eigenvalue for the term $\mathbf{H^T H} + \lambda \mathbf{C^T C}$.

## 3.4 Higher Order Convergence Methods

The standard iteration (3.6) along with the logical derivatives (3.9) and (3.15) exhibit linear or first-order convergence rates. The term-by-term implementation of the geometric series for $(\beta \mathbf{H})^{-1}$ given in (3.3) can be grouped, for instance into two groups of even and odd power terms:

$$\begin{aligned}(\beta\mathbf{H})^{-1} &= \sum_{n=0,2,4,\ldots} T^n + \sum_{n=1,3,5,\ldots} T^n \\ &= \sum_{n=0}^{\infty} T^{2n} + T\sum_{n=0}^{\infty} T^{2n} \\ &= (I+T)\sum_{n=0}^{\infty} T^{2n} \end{aligned} \tag{3.18}$$

Since the right hand side of (3.18) is again a geometric series, but this time in $T^{2n}$, it can be factored as in (3.18) again and again:

$$\begin{aligned}(\beta\mathbf{H})^{-1} &= (I+T)\left(I+T^2\right)\sum_{n=0}^{\infty} T^{4n} \\ &\vdots \\ &= (I+T)\left(I+T^2\right)\left(I+T^4\right)\ldots \\ &= \prod_{n=0}^{\infty}\left(I+T^{2^n}\right) \end{aligned} \tag{3.19}$$

In a way similar to the linear convergence case, an iterative algorithm can be derived for this expansion by using the first $k$ terms of the expansion (3.19) and defining an estimate image at iteration $k$ as:

$$\mathbf{s}_k = \left[\prod_{n=0}^{k-1}\left(I+T^{2^n}\right)\right]\beta\mathbf{r} \tag{3.20}$$

The estimate $\mathbf{s}_{k+1}$, when the iteration is initialized to $\mathbf{s}_0 = \beta\mathbf{r}$, is then related to $\mathbf{s}_k$ by the expression:

$$\mathbf{s}_{k+1} = \mathbf{s}_k + \mathbf{s}_k\left(I-\beta\mathbf{H}\right)^{2^k} \tag{3.21}$$

This second order convergence iteration can be extended to the $p$th order case as in [8][24][26][33][46][51]. The higher order or multi-step algorithms yield the same

results as their linear convergence counterparts, however the application of non-linear constraints between iterations (see Section 3.5) is not possible and therefore an open research topic[24][26].

## 3.5 Using Nonlinear Constraints with an Iterative Restoration Algorithm

We conclude the topic of successive approximation iteration methods with a discussion on the application of nonlinear constraints. This is a popular and straightforward way to incorporate prior knowledge into an image restoration solution. There are two major ways that constraints are applied; projections on to convex sets (POCS)[24][53][65] and the use of nonlinear contstraint operators in fixed point iteration methods.

A nonlinear constraint operator $C$ : can be applied to an arbitrary signal $x$ such that the relation:

$$x = Cx \tag{3.22}$$

will be true if and only if the signal $x$ already satisfies the constraints embodied in $C$. This is a convenient way to express adherence to prior knowledge, and $C$ has the property that if any signal $x$ already satisfies the proposed constraints, it will be unchanged by $C$ as stated in (3.22). Examples of constraints that have this property are positivity, band limitation, and a specified area of support.

An approach to obtaining an iteration equation that makes use of a nonlinear constraint is to modify (3.6) introduced in Section 3.1. Here, we can replace the target signal $\mathbf{s}_k$ with the constrained signal $C\mathbf{s}_k$.

$$\begin{aligned} \mathbf{s}_{k+1} &= C\mathbf{s}_k + \beta\left(\mathbf{r} - \mathbf{H}C\mathbf{s}_k\right) \\ &= \beta\mathbf{r} + \left(\mathbf{I} - \beta\mathbf{H}\right)C\mathbf{s}_k \end{aligned}$$

$$= \beta \mathbf{r} + G\mathbf{s}_k \tag{3.23}$$

where $G = (\mathbf{I} - \beta\mathbf{H})C$. This follows the standard fixed-point iteration equations that follow the form:

$$x_{k+1} = Fx_k \tag{3.24}$$

where the fixed point operator $F$ is defined by:

$$F\mathbf{s}_k = \beta\mathbf{r} + (\mathbf{I} - \beta\mathbf{H})\, C\mathbf{s}_k \tag{3.25}$$

For convergence to a fixed point, the operator $F$ must be a contraction mapping, and must satisfy the following:

$$\|F\mathbf{s}_i - F\mathbf{s}_j\| \leq r \|\mathbf{s}_i - \mathbf{s}_j\|, \quad 0 \leq r < 1 \tag{3.26}$$

If the operator $F$ is indeed a contraction mapping in a given sub-space, then there is a unique fixed point in that sub-space such that $x = Fx$ and an iterative estimation scheme based on successive approximations will converge to that point for any starting signal choice in that subspace[51]. The iteration control parameter $\beta$ can be chosen such that (3.26) holds.

Application of these concepts to the iterative Tikhonov Miller iteration (3.15) results in:

$$\mathbf{s}_{k+1} = C\mathbf{s}_k + \beta\left(\mathbf{H^T r} - \left(\mathbf{H^T H} + \lambda \mathbf{C^T C}\right) C\mathbf{s}_k\right) \tag{3.27}$$

## 3.6 Projections on to Convex Sets

The method of alternating projections on to convex sets (POCS) is an alternative approach to applying prior knowledge into the restoration process. Here, desirable properties of a reconstructed signal are defined by two or more convex signal sets. If

the desired deconvolution solution can be adequately defined by several sets of signals, and given that those signal sets are convex with a nonempty intersection, alternately projecting intermediate solutions onto these convex sets will result in convergence to a final solution. Some convex signal sets frequently encountered in practice are band limited signals, bounded signals, signals with a given area, and signals that are constant over a given interval [21][53][65].

Geometrically, a set $A$ is said to be convex if the line connecting any two vectors $\vec{u}_1, \vec{u}_2 \in A$ are completely subsumed in $A$. If any portion of that line lies outside of $A$, then the set is not convex. Hence a projection onto a convex set $A$ of any vector $\vec{v} \notin A$ will be the unique element in $A$ that is closest to $\vec{v}$.

For example, these ideas can be applied to inverse filtering by considering the two convex sets

$$Q_{\mathbf{s}|\mathbf{r}} = \left\{\mathbf{s} \middle| \|\mathbf{r} - \mathbf{H}\mathbf{s}\|^2 \leq \epsilon^2\right\}$$

and

$$Q_{\mathbf{s}} = \left\{\mathbf{s} \middle| \|\mathbf{C}\mathbf{s}\|^2 \leq E^2\right\}$$

where $\epsilon^2$ is the measurement noise variance and $E^2$ is an energy limit set on the high pass version of the estimated signal $\hat{s}(i,j)$. The respective projections, $P_1\mathbf{s}$ and $P_2\mathbf{s}$ are defined by [24]:

$$P_1\mathbf{s} = \mathbf{s} + \lambda_1(\mathbf{I} + \lambda_1\mathbf{H}^T\mathbf{H})^{-1}\mathbf{H}^T(\mathbf{r} - \mathbf{H}\mathbf{s}) \tag{3.28}$$

$$P_2\mathbf{s} = \left[\mathbf{I} - \lambda_2(\mathbf{I} + \lambda_2\mathbf{C}^T\mathbf{C})^{-1}\mathbf{C}^T\mathbf{C}\right]\mathbf{s} \tag{3.29}$$

where $\lambda_1$ and $\lambda_2$ are iteration control parameters.

There are three possible outcomes in the application of POCS, depending on the way that the convex sets intersect[21].

1. Given two or more convex sets with a nonempty intersection, alternately projecting onto the sets will converge to a point included in that intersection. The

actual point of convergence will depend on the initialization point on the POCS process.

2. If any two convex sets do not intersect, the result is a *limit cycle*, or a "mean square" solution to the problem. The solutions will be the two points in each set that are the closest in the mean square sense to each other.

3. When any three or more convex sets do not intersect, the POCS process will break down. The result is a *greedy limit cycle* that is dependant on the order of projections. The resulting projections generally do not display any desirable signal properties.

The restoration of degraded images can in many cases be posed as a POCS problem. POCS is particularly useful for ill-posed deconvolution problems, where regularization is imposed by way of the possibly nonlinear convex constraints on the solution set.

## 3.7 Synopsis of Iterative Restoration Techniques

Iterative deconvolution techniques, although more time consuming than direct methods, are popular due to their flexibility. Not only can direct inversion be avoided, but most importantly *a priori* constraints can be applied at each step to ensure a solution that displays desirable image qualities. The application of constraints in effect removes the ambiguity imposed by the ill-posed nature of the problem.

There are two major methods of iterative deconvolution; the method of successive approximations and POCS. The method of successive approximations has a linear convergence rate, but can be formulated to display $p$th order convergence characteristics. This gain in efficiency however precludes the possibility of applying constraints between iterations, which will result in an undesirable image estimate. POCS is also

a popular and efficient way to regularize the image solution. If the convex sets are chosen such that their intersection is nonempty, the projections will converge to a fixed point that is dependent on the algorithm initialization.

# Chapter 4

# A Hybrid Algorithm for Iterative Deconvolution

The two preceding chapters dealt with some classical approaches to image restoration. Although there are many other methods of image restoration, these few approaches were introduced as they are closely related to the method proposed in this work. The successive approximation approach to iteratively solving the deconvolution inverse filter has been analysed in numerous publications[8][22][23][24][25][28][33][34][51]. These methods are based on minimizing the asymptotic error as the number of iterations becomes infinitely large. Also, the regularization parameter $\lambda$ and operator $\mathbf{C}$ are assumed to be constant in $k$. These facts lead to some important questions, namely:

1. If a method based on minimizing the transient error at each iteration were developed, how would the asymptotic error be affected?

2. Can this approach of analysing the transient error benefit iteration convergence properties?

In this chapter, an answer to these questions is provided with a new iterative algorithm based on minimizing the residual error norm for a two-image system. The

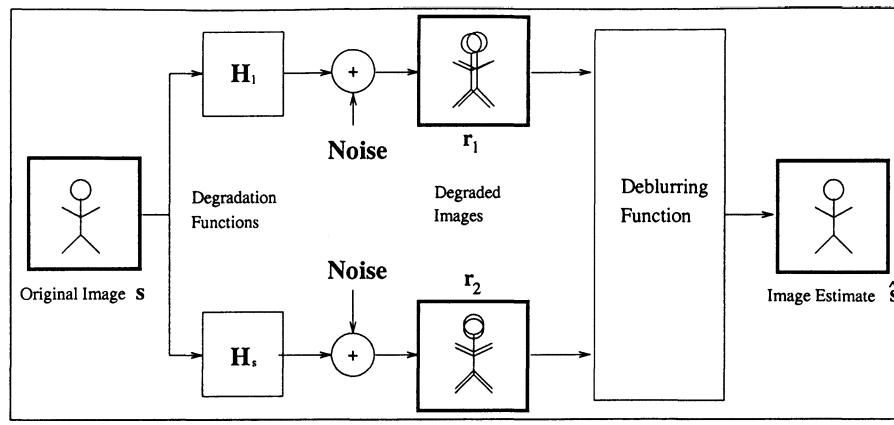


Figure 4.1: Two-Image Degradation System

idea of using two images is a natural extension of existing work based on the blind identification of one-dimensional signals[57]. This two image approach, depicted in Figure 4.1, will be used as a basis for estimating a current image at a given iteration using a filtered version of the previous estimate as the second degraded image. The two-signal or two-image approach was originally adopted to facilitate the design of an inverse filter, as the corresponding two-PSF blurring matrix will at the very least have a null space dimension as small as the single blurring matrix **H**, and will in most cases have a null space of lesser dimension. In the case of non-singular matrices as well, it can be said that the two-PSF blurring matrix will in most cases be better conditioned than the single PSF situation. Hence the notion of choosing a second blur operator that improves conditioning will be discussed. In fact, it will be argued that since the "second image" in this two image system is a previous estimate in an iterative process, this second blurring matrix should not have large eigenvalues that coincide with the small or zero eigenvalues of the actual (or estimated in the case of blind deconvolution) blurring matrix. The result is a hybrid *adaptive* iterative algorithm that uses a filtered version of a previous image estimate for stability purposes, as well as the incorporation of a non-parametric, frequency adaptive regularization operator. An

error analysis, based on the current PSF estimate and assuming a blind iterative procedure is developed. Both the regularization parameter and secondary "pseudo-PSF" are estimated at each iteration based on the minimization of the current transient error estimate. The beneficial effect of the additional stability operator will then be shown experimentally to have significantly improved convergence properties relative to previous methods.

## 4.1 Direct Multi-Image Deconvolution

One approach is to assume that two versions[1] of the received signal are available, $r_1(i,j)$ and $r_2(i,j)$. Then the degradation process of equation (2.11) can be reformulated in matrix-vector form as:

$$\begin{bmatrix} \mathbf{r}_1 \\ \mathbf{r}_2 \end{bmatrix} = \begin{bmatrix} \mathbf{H}_1 \\ \mathbf{H}_2 \end{bmatrix} \begin{bmatrix} \mathbf{s} \end{bmatrix} + \begin{bmatrix} \mathbf{n}_1 \\ \mathbf{n}_2 \end{bmatrix} \tag{4.1}$$

Using the same approach as before, we can again seek to minimize the objective function corresponding to (2.15) resulting in:

$$\hat{\mathbf{s}}_k = \left(\mathbf{H}_1^{\mathrm{T}}\mathbf{H}_1 + \mathbf{H}_2^{\mathrm{T}}\mathbf{H}_2\right)^{-1}\left(\mathbf{H}_1^{\mathrm{T}}\mathbf{r}_1 + \mathbf{H}_2^{\mathrm{T}}\mathbf{r}_2\right) \tag{4.2}$$

As before, the same diagonalization approach can be taken if both $\mathbf{H}_1$ and $\mathbf{H}_2$ are assumed to be block-circulant. Hence the frequency domain formulation[2]:

$$\hat{\mathcal{S}}(u,v) = \frac{\mathcal{H}_1^*(u,v)\mathcal{R}_1(u,v) + \mathcal{H}_2^*(u,v)\mathcal{R}_2(u,v)}{|\mathcal{H}_1(u,v)|^2 + |\mathcal{H}_2(u,v)|^2} \tag{4.3}$$

This solution is well-posed if $\mathcal{H}_1$ and $\mathcal{H}_2$ do not share common zeros, which will be true if the columnspace of the matrix

[1] Two images of the same source corresponding to distinct PSF's

[2] Note that for following discussions, the DFT coefficient indices $(u,v)$ will be omitted. Any scripted variables (e.g. $\mathcal{H}$ or $\mathcal{R}$) will be considered to be complex valued DFT coefficients.

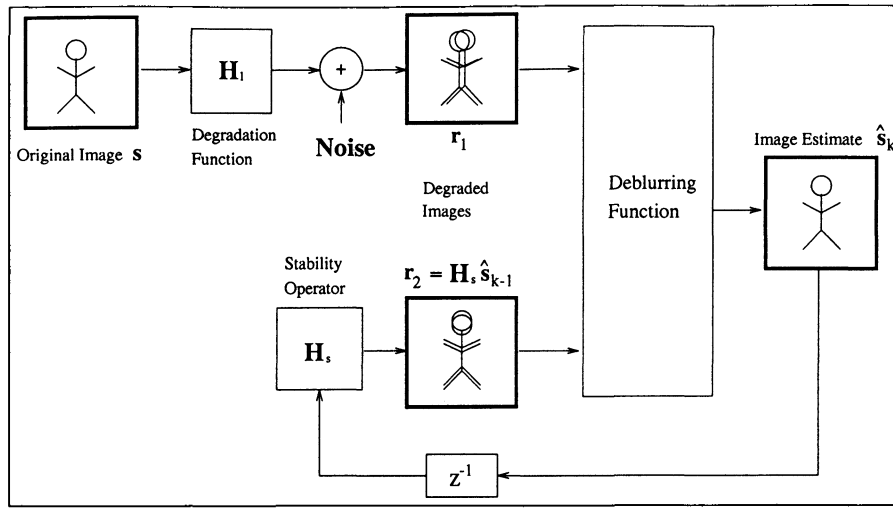


Figure 4.2: Proposed Iterative Process

$$\mathbf{H} = \begin{bmatrix} \mathbf{H}_1 \\ \mathbf{H}_2 \end{bmatrix}$$

is full rank. The difficulty in undertaking this approach is that two versions of the image must be available. Since an auxiliary image is often not available in practical applications, we propose a novel iterative approach. The image $\mathbf{r}_2$ is replaced with a filtered version of the image estimate $\hat{\mathbf{s}}_{k-1}$ at the previous iteration. Specifically, at iteration $k$ we set $\mathbf{r}_2^{(k)} = \mathbf{H}_s\hat{\mathbf{s}}_{k-1}$ as in Figure 4.2. The "pseudo"-PSF $\mathbf{H}_s$ is chosen to be a stability operator in a manner yet to be described. The DFT domain expression for the $k$th estimate of the image is then:

$$\hat{\mathcal{S}}_k = \frac{\mathcal{H}^*\mathcal{R} + \mathcal{H}_s^*\mathcal{H}_s\hat{\mathcal{S}}_{k-1}}{|\mathcal{H}|^2 + |\mathcal{H}_s|^2} \tag{4.4}$$

Obviously, no new information is available with this approach. However, we show later that there are significant advantages with respect to speed of convergence by including a "selective memory" of past image estimates.

### 4.1.1 Error Analysis

To gain some insights into the properties of this preliminary algorithm, expressions for the error terms in equation (4.4) can be developed. These expressions will be presented in the frequency domain, as the approximate relationship between the DFT coefficients of $\mathbf{H}$ and its eigenvalues due to the block-circulant assumption lead to some intuitive conclusions. The estimates for $\mathcal{S}$ and $\mathcal{H}$ at the *kth* iteration can be defined as:

$$\hat{\mathcal{S}}_k = \mathcal{S} + \mathcal{E}_{\hat{\mathcal{S}}_k}$$

$$\hat{\mathcal{H}}_k = \mathcal{H} + \mathcal{E}_{\hat{\mathcal{H}}_k}$$

We note here that although the PSF $\mathcal{H}$ is considered to be known, the PSF estimate $\hat{\mathcal{H}}_k$ is defined firstly to identify the effect of PSF error on the overall error, and secondly to facilitate the extension of this analysis to the blind case in Chapter 5. Here we also look at the stability operator $\mathcal{H}_s$, renaming the expression [3]

$$|\mathcal{H}_s|^2 = \mathcal{H}_s^* \mathcal{H}_s = \gamma \tag{4.5}$$

An error expression can then be obtained by making the above substitutions, then subtracting $\mathcal{S}$ from (4.4).

$$\mathcal{E}_{\hat{\mathcal{S}}_k} = \frac{\hat{\mathcal{H}}_k^* \mathcal{N} - \hat{\mathcal{H}}_k^* \mathcal{E}_{\hat{\mathcal{H}}_k} \mathcal{S} + \gamma_k \mathcal{E}_{\hat{\mathcal{S}}_{k-1}}}{\left|\hat{\mathcal{H}}_k\right|^2 + \gamma_k} \tag{4.6}$$

where $\mathcal{N}$ are the DFT coefficients of the additive observation noise $\mathbf{n}$. The total error for each DFT coefficient in the estimated image can therefore be broken down into three components; error due to additive observation noise, error due to the mis-estimation of the blur operator $\hat{\mathbf{h}}_k(i,j)$, and finally the error due to the mis-estimation

[3] Although not in scripted text or complex valued, the operator $\gamma$ as defined in (4.5) is still a function of frequency, i.e. $\gamma(u,v)$

in the unconstrained estimate $\hat{\mathbf{s}}_{k-1}(i,j)$. Noting that $\mathcal{E}_{\hat{\mathcal{S}}_{k-1}}$ can be expressed recursively with respect to the other two types of error in previous iterations, equation (4.6) can thus be re-written (See Appendix B for a more detailed development):

$$\mathcal{E}_{\hat{\mathcal{S}}_k} = \mathcal{N}\Sigma_{\mathcal{N}}^{(k)} - \mathcal{S}\Sigma_{\mathcal{H}}^{(k)} \tag{4.7}$$

where

$$\begin{aligned}\Sigma_{\mathcal{N}}^{(k)} &= \frac{\mathcal{H}_k^*}{\mathcal{Z}_k} + \sum_{i=1}^{k}\left[\prod_{j=i}^{k}\frac{\gamma_j}{\mathcal{Z}_j}\right]\frac{\mathcal{H}_{i-1}^*}{\mathcal{Z}_{i-1}} \\ &= \frac{\hat{\mathcal{H}}_k^* + \gamma_k\Sigma_{\mathcal{N}}^{(k-1)}}{\mathcal{Z}_k}\end{aligned} \tag{4.8}$$

$$\begin{aligned}\Sigma_{\mathcal{H}}^{(k)} &= \frac{\hat{\mathcal{H}}_k^*\mathcal{E}_{\hat{\mathcal{H}}_k}}{\mathcal{Z}_k} + \sum_{i=1}^{k}\left[\prod_{j=i}^{k}\frac{\gamma_j}{\mathcal{Z}_j}\right]\frac{\hat{\mathcal{H}}_{i-1}^*\mathcal{E}_{\hat{\mathcal{H}}_{i-1}}}{\mathcal{Z}_{i-1}} \\ &= \frac{\hat{\mathcal{H}}_k^*\mathcal{E}_{\hat{\mathcal{H}}_k} + \gamma_k\Sigma_{\mathcal{H}}^{(k-1)}}{\mathcal{Z}_k}\end{aligned} \tag{4.9}$$

and $\mathcal{Z}_i = \left|\hat{\mathcal{H}}_i\right|^2 + \gamma_i$. From (4.7), (4.8) and (4.9) we can see that the three original error propagation terms are controlled by the recursive terms $\Sigma_{\mathcal{N}}^{(k)}$ and $\Sigma_{\mathcal{H}}^{(k)}$. It will soon be shown that the use of $\gamma$ results in a minimized error term that only propagates the more useful information from past estimates without consequence to the overall or asymptotic error. It is also important to note that the extent or "memory" factor involved with this recursive error propagation is controlled by the stability operator $\gamma$, which can be chosen to select regions of $\hat{\mathcal{S}}_{k-1}$ where the error is low.

Hence, even though the use of a second image to develop an iterative procedure is *ad hoc* in nature, the introduction of the parameter $\gamma$ allows extra flexibility. Also, the corresponding error analysis reveals considerable insight. Given this, there are two issues that need to be addressed.

1. Does the introduction of $\gamma$ in (4.4) and (4.6) have any regularization influence?

2. How can $\gamma$ be used to speed convergence?

Point two is addressed later in this chapter. We now address regularization.

With respect to regularization, close inspection of (4.4) reveals a conflict. On the one hand, to reduce error due to noise amplification, the value of $\gamma$ must be larger when $\left|\hat{\mathcal{H}}_k\right|^2$ is small. However, these larger values for $\gamma$ will increase the last term in (4.6) for those coefficients. Further, from (4.8) these very regions where $\left|\hat{\mathcal{H}}_k\right|$ is small are the same regions where the error magnitude $\left|\mathcal{E}_{\hat{\mathcal{S}}_k}\right|$ is large. Therefore, we see from this intuitive argument that the use of $\gamma$ in (4.4) cannot perform regularization.

An approach to verify this intuitive analysis is to assume that $\gamma$ will be held constant with $k$. The image estimate (4.4) can then be expanded:

$$\begin{aligned}
\hat{\mathcal{S}}_k &= \frac{\mathcal{H}^*\mathcal{R} + \gamma\hat{\mathcal{S}}_{k-1}}{|\mathcal{H}|^2 + \gamma} \\
&= \frac{\mathcal{H}^*\mathcal{R} + \gamma\left[\dfrac{\mathcal{H}^*\mathcal{R} + \gamma\left[\dfrac{\cdots}{|\mathcal{H}|^2+\gamma}\right]}{|\mathcal{H}|^2+\gamma}\right]}{|\mathcal{H}|^2 + \gamma} \\
&= \frac{\mathcal{H}^*\mathcal{R}}{|\mathcal{H}|^2 + \gamma}\left[\sum_{i=0}^{k}\left(\frac{\gamma}{|\mathcal{H}|^2+\gamma}\right)^i\right] \\
&= \frac{\mathcal{H}^*\mathcal{R}}{|\mathcal{H}|^2 + \gamma}\left[\frac{1-\left(\dfrac{\gamma}{|\mathcal{H}|^2+\gamma}\right)^{k+1}}{1-\left(\dfrac{\gamma}{|\mathcal{H}|^2+\gamma}\right)}\right] \qquad (4.10)
\end{aligned}$$

$$\lim_{k\to\infty}\hat{\mathcal{S}}_k = \frac{\mathcal{H}^*\mathcal{R}}{|\mathcal{H}|^2} \qquad (4.11)$$

which we can see is undefined for any zero eigenvalues of $\mathbf{H}$. So we can see that the stability operator $\gamma$ does not regularize the inverse filtering solution in this case.

In fact, we can see that at the limit, this iterative estimation procedure converges to the generalized least squares or pseudo-inverse solution. Therefore, some form of regularization is required apart from the operator $\gamma$.

## 4.2 Hybrid Formulation

Pursuant to this discussion, we extend the previous treatment to reduce the effect of noise propagation error by introducing a regularization operator to augment the stability operator $\gamma$. Starting again with the two-image matrix-vector form of (4.1), we seek to minimize the functional:

$$\left\| \begin{bmatrix} \mathbf{H}_1 \\ \mathbf{H}_2 \end{bmatrix} \mathbf{s} - \begin{bmatrix} \mathbf{r}_1 \\ \mathbf{r}_2 \end{bmatrix} \right\|_2^2 + \lambda \left\| \mathbf{Cs} \right\|_2^2 \tag{4.12}$$

where $\mathbf{C}$ and $\lambda$ are the regularization operator and parameter respectively. The minimization of the above results in the following expression:

$$\left(\mathbf{H}_1^{\mathbf{T}}\mathbf{H}_1 + \mathbf{H}_2^{\mathbf{T}}\mathbf{H}_2 + \lambda \mathbf{C}^{\mathbf{T}}\mathbf{C}\right)\mathbf{s} = \mathbf{H}_1^{\mathbf{T}}\mathbf{r}_1 + \mathbf{H}_2^{\mathbf{T}}\mathbf{r}_2 \tag{4.13}$$

As before, a block-circulant assumption can be made for the two block-Toeplitz blur operators $\mathbf{H}_1$ and $\mathbf{H}_2$ as well as the regularization operator $\mathbf{C}$. Hence this expression can be re-written in the frequency domain.

$$\hat{\mathcal{S}} = \frac{\mathcal{H}_1^* \mathcal{R}_1 + \mathcal{H}_2^* \mathcal{R}_2}{|\mathcal{H}_1|^2 + |\mathcal{H}_2|^2 + \lambda |\mathcal{C}|^2} \tag{4.14}$$

Most existing approaches to this problem treat $\lambda$ and $\mathcal{C}$ as separate entities, often choosing $\mathcal{C}$ as a known, smooth form and fixing $\lambda$ based on some estimated prior knowledge. Here we propose to combine these two into one $N^2 \times N^2$ real valued array

$\rho$ such that[4]:

$$\rho = \lambda \mathcal{C}^* \mathcal{C} = \lambda \left| \mathcal{C} \right|^2 \tag{4.15}$$

Incorporating this into the error analysis and defining $\mathbf{r}_2^{(k)} = \mathbf{H}_s \hat{\mathbf{s}}_{k-1}$ and $\gamma$ by (4.5) in a manner discussed previously, (4.4) is now changed to:

$$\hat{\mathcal{S}}_k = \frac{\hat{\mathcal{H}}_k^* \mathcal{R} + \gamma_k \hat{\mathcal{S}}_{k-1}}{\left| \hat{\mathcal{H}}_k \right|^2 + \gamma_k + \rho_k} \tag{4.16}$$

where as before the subscript refers to an estimate at iteration $k$. Now the error formulation of (4.7) has one additional term to account for regularization error.

$$\mathcal{E}_{\hat{\mathcal{S}}_k} = \mathcal{N} \Sigma_{\mathcal{N}}^{(k)} - \mathcal{S} \Sigma_{\mathcal{H}}^{(k)} - \mathcal{S} \Sigma_{\rho}^{(k)} \tag{4.17}$$

where the regularization error propagation term is

$$\begin{aligned} \Sigma_{\rho}^{(k)} &= \frac{\rho_k}{\mathcal{Z}_k} + \sum_{i=1}^{k} \left[ \prod_{j=i}^{k} \frac{\gamma_j}{\mathcal{Z}_j} \right] \frac{\rho_{i-1}}{\mathcal{Z}_{i-1}} \\ &= \frac{\rho_k + \gamma_k \Sigma_{\rho}^{(k-1)}}{\mathcal{Z}_k} \end{aligned} \tag{4.18}$$

and

$$\mathcal{Z}_i = \left| \hat{\mathcal{H}}_i \right|^2 + \gamma_i + \rho_i$$

Note also that $\mathcal{Z}_i$ is changed accordingly for (4.8) and (4.9) as well.

### 4.2.1 Alternate Error Formulation

The error formulation (4.17) can also be re-written in terms of $\hat{\mathcal{S}}_k$ and a noise estimate $\hat{\mathcal{N}}$ as

[4] As with $\gamma$, the operator $\rho$ is a function of frequency i.e. $\rho(u, v)$

$$\mathcal{E}_{\hat{\mathcal{S}}_k} = \frac{\left(\hat{\mathcal{N}} + \hat{\mathcal{S}}_k \mathcal{E}_{\hat{\mathcal{H}}_k}\right) \Sigma_{\mathcal{N}}^{(k)} - \hat{\mathcal{S}}_k \Sigma_{\mathcal{H}}^{(k)} - \hat{\mathcal{S}}_k \Sigma_{\rho}^{(k)}}{1 - \left(\hat{\mathcal{H}}_k - \mathcal{E}_{\hat{\mathcal{H}}_k}\right) \Sigma_{\mathcal{N}}^{(k)} - \Sigma_{\mathcal{H}}^{(k)} - \Sigma_{\rho}^{(k)}} \tag{4.19}$$

where $\hat{\mathcal{S}}_k$ is as per (4.16), and

$$\hat{\mathcal{N}} = \mathcal{R} - \hat{\mathcal{H}}_k \hat{\mathcal{S}}_k$$

The obvious benefit of (4.19) is that the error, when the PSF is assumed to be known *a priori*, is now completely defined in terms of known quantities. Experimental evidence shows that the use of (4.19) indeed does yield better results despite the instability introduced by the denominator term.

### 4.2.2 Optimal Stability and Regularization Operators

To choose $\gamma_k$ and $\rho_k$, we minimize the squared norm of the estimation error, which by Parseval's theorem is the same a minimizing $\mathcal{E}^*_{\hat{\mathcal{S}}_k} \mathcal{E}_{\hat{\mathcal{S}}_k}$ or $\left|\mathcal{E}_{\hat{\mathcal{S}}_k}\right|^2$. Note that here we minimize the error for each frequency domain coefficient separately as opposed to other methods that choose a static regularization operator and minimize the expectation of the entire error with respect to a regularization parameter[10]. The partial derivative of $\left|\mathcal{E}_{\hat{\mathcal{S}}_k}\right|^2$ with respect to $\gamma_k$ is

$$\frac{\partial \left|\mathcal{E}_{\hat{\mathcal{S}}_k}\right|^2}{\partial \gamma_k} = \frac{\gamma_k \left(\mathcal{E}_{\hat{\mathcal{S}}_{k-1}} \mathcal{A}^* + \mathcal{E}^*_{\hat{\mathcal{S}}_{k-1}} \mathcal{A}\right) - (\mathcal{A}^* \mathcal{B} + \mathcal{A} \mathcal{B}^*)}{\left(\left|\hat{\mathcal{H}}_k\right|^2 + \gamma_k + \rho_k\right)^3} \tag{4.20}$$

where

$$\begin{aligned} \mathcal{A} &= \mathcal{S}\left(\rho_k + \hat{\mathcal{H}}_k^* \mathcal{E}_{\hat{\mathcal{H}}_k}\right) + \mathcal{E}_{\hat{\mathcal{S}}_{k-1}} \left(\left|\hat{\mathcal{H}}_k\right|^2 + \rho_k\right) - \hat{\mathcal{H}}_k^* \mathcal{N} \\ &= \rho_k \hat{\mathcal{S}}_{k-1} + \left|\hat{\mathcal{H}}_k\right|^2 \hat{\mathcal{S}}_{k-1} - \hat{\mathcal{H}}_k^* \mathcal{R} + \hat{\mathcal{H}}_k^* \mathcal{E}_{\hat{\mathcal{H}}_k} \mathcal{S} \end{aligned} \tag{4.21}$$

and

$$\mathcal{B} = \mathcal{S}\left(\rho_k + \hat{\mathcal{H}}_k^* \mathcal{E}_{\hat{\mathcal{H}}_k}\right) - \hat{\mathcal{H}}_k^* \mathcal{N} \tag{4.22}$$

Setting this partial derivative to zero and solving for $\gamma_k$ yields:

$$\gamma_k = \frac{\mathcal{A}\mathcal{B}^* + \mathcal{A}^*\mathcal{B}}{\mathcal{E}_{\hat{\mathcal{S}}_{k-1}}\mathcal{A}^* + \mathcal{E}^*_{\hat{\mathcal{S}}_{k-1}}\mathcal{A}} \tag{4.23}$$

Repeating the same process with respect to $\rho_k$ yields:

$$\frac{\partial \left|\mathcal{E}_{\hat{\mathcal{S}}_k}\right|^2}{\partial \rho_k} = \frac{\mathcal{X}\left(\mathcal{Y} - \rho_k \mathcal{S}\right)^* + \mathcal{X}^*\left(\mathcal{Y} - \rho_k \mathcal{S}\right)}{\left(\left|\hat{\mathcal{H}}_k\right|^2 + \gamma_k + \rho_k\right)^3} \tag{4.24}$$

where

$$\begin{aligned} \mathcal{X} &= \mathcal{S}\left(\left|\hat{\mathcal{H}}_k\right|^2 + \gamma_k - \hat{\mathcal{H}}^*_k \mathcal{E}_{\hat{\mathcal{H}}_k}\right) + \gamma_k \mathcal{E}_{\hat{\mathcal{S}}_{k-1}} + \hat{\mathcal{H}}^*_k \mathcal{N} \\ &= \hat{\mathcal{H}}^*_k \mathcal{R} + \gamma_k \hat{\mathcal{S}}_{k-1} - \hat{\mathcal{H}}^*_k \mathcal{E}_{\hat{\mathcal{H}}_k} \mathcal{S} \end{aligned} \tag{4.25}$$

and

$$\mathcal{Y} = \gamma_k \mathcal{E}_{\hat{\mathcal{S}}_{k-1}} + \hat{\mathcal{H}}^*_k \mathcal{N} - \hat{\mathcal{H}}^*_k \mathcal{E}_{\hat{\mathcal{H}}_k} \mathcal{S} \tag{4.26}$$

Hence the value for $\rho_k$ that minimizes $\left|\mathcal{E}_{\hat{\mathcal{S}}_k}\right|^2$ is:

$$\rho_k = \frac{\mathcal{X}\mathcal{Y}^* + \mathcal{X}^*\mathcal{Y}}{\mathcal{S}\mathcal{X}^* + \mathcal{S}^*\mathcal{X}} \tag{4.27}$$

Note that (4.23) and (4.27) are nonlinear equations in two unknowns that need to be jointly solved. Both of these expressions yield values that are real due to the conjugate structure of (4.23) and (4.27). This does not however guarantee that values for $\gamma_k$ and $\rho_k$ are positive. Hence, at each iteration $\gamma_k$ and $\rho_k$ are constrained such that any negative values as calculated per (4.23) or (4.27) are set to zero.

$$\gamma_k = \begin{cases} \gamma_k & : \quad \gamma_k \geq 0 \\ 0 & : \quad \gamma_k < 0 \end{cases} \tag{4.28}$$

$$\rho_k = \begin{cases} \rho_k & : \quad \rho_k \geq 0 \\ 0 & : \quad \rho_k < 0 \end{cases} \tag{4.29}$$

In summary, we now have an iterative image restoration method that uses both a regularization operator to reduce noise amplification, and a stability operator that allows an additional degree of freedom. Further to this, we note that since this iteration is computed solely using scalar operations on complex DFT coefficient values, the algorithm complexity is $O(N^2)$ per iteration where $N^2$ is the number of pixels in the image. We now experimentally show the benefit gained by the introduction of $\gamma$ in (4.16), by comparing convergence rates with and without its use.

## 4.3 Stablilty Operator Effect on Convergence

In this chapter a frequency-adaptive, iterative algorithm for image restoration using an inverse filter or deconvolution technique is described. The development of this iterative scheme is based on a two-image analysis approach, where the second degraded image is replaced by a filtered version of a previous image estimate. This filter, the basis for the stability operator $\gamma$, should be chosen such that DFT coefficient values with a high confidence level of correctness are reinforced, thus facilitating the process.

To provide some anecdotal proof for this notion, experiments were run on the gray-tone image referred to as "Lena". Figure 4.3 compares the convergence rate for two runs of the unconstrained algorithm (4.16), one with $\gamma$ set to zeros, and the other with $\gamma$ calculated as per equation (4.27). Figure 4.5 shows similar results for the gray-tone "cameraman" image. Figures 4.4 and 4.6 show pictorial results, with almost identical solutions for the stabilized and non-stabilized cases. The stabilized case though reached an almost maximum improved signal-to-noise ratio (ISNR) level within three to four iterations compared to the non-stabilized case which took over one-hundred iterations.

The images were degraded with a nine pixel horizontal motion blur as per (2.7) and then further degraded by additive white Gaussian noise to achieve the blurred

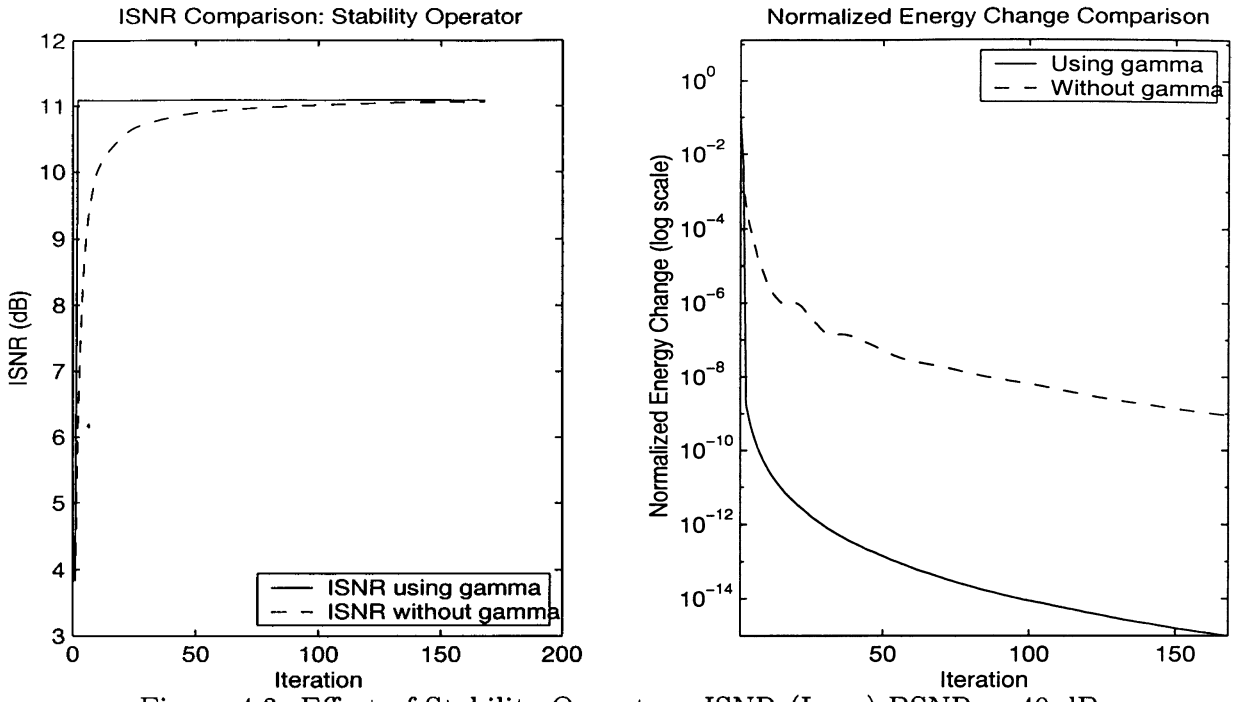


Figure 4.3: Effect of Stability Operator - ISNR (Lena) BSNR = 40 dB

signal to noise ratio (BSNR) indicated. The PSF was assumed to be known, and the noise variance was estimated using the generalized cross validation method to be described in Section 5.3.3. The normalized change in energy, used as a measure of convergence, is defined as:

$$\frac{\|\mathbf{s_{k+1}} - \mathbf{s_k}\|^2}{\|\mathbf{s_k}\|^2} \tag{4.30}$$

Also the ISNR and BSNR, both in dB, are respectively defined as:

$$ISNR = 10\log_{10}\left(\frac{\sum_{i,j}(s(i,j) - r(i,j))^2}{\sum_{i,j}(s(i,j) - \hat{s}(i,j))^2}\right)$$

$$BSNR = 10\log_{10}\left(\frac{var(\mathbf{Hs})}{\sigma_n^2}\right)$$

For both cases, the initial "guess" for the image $\hat{\mathcal{S}}_0$ was initialized to:

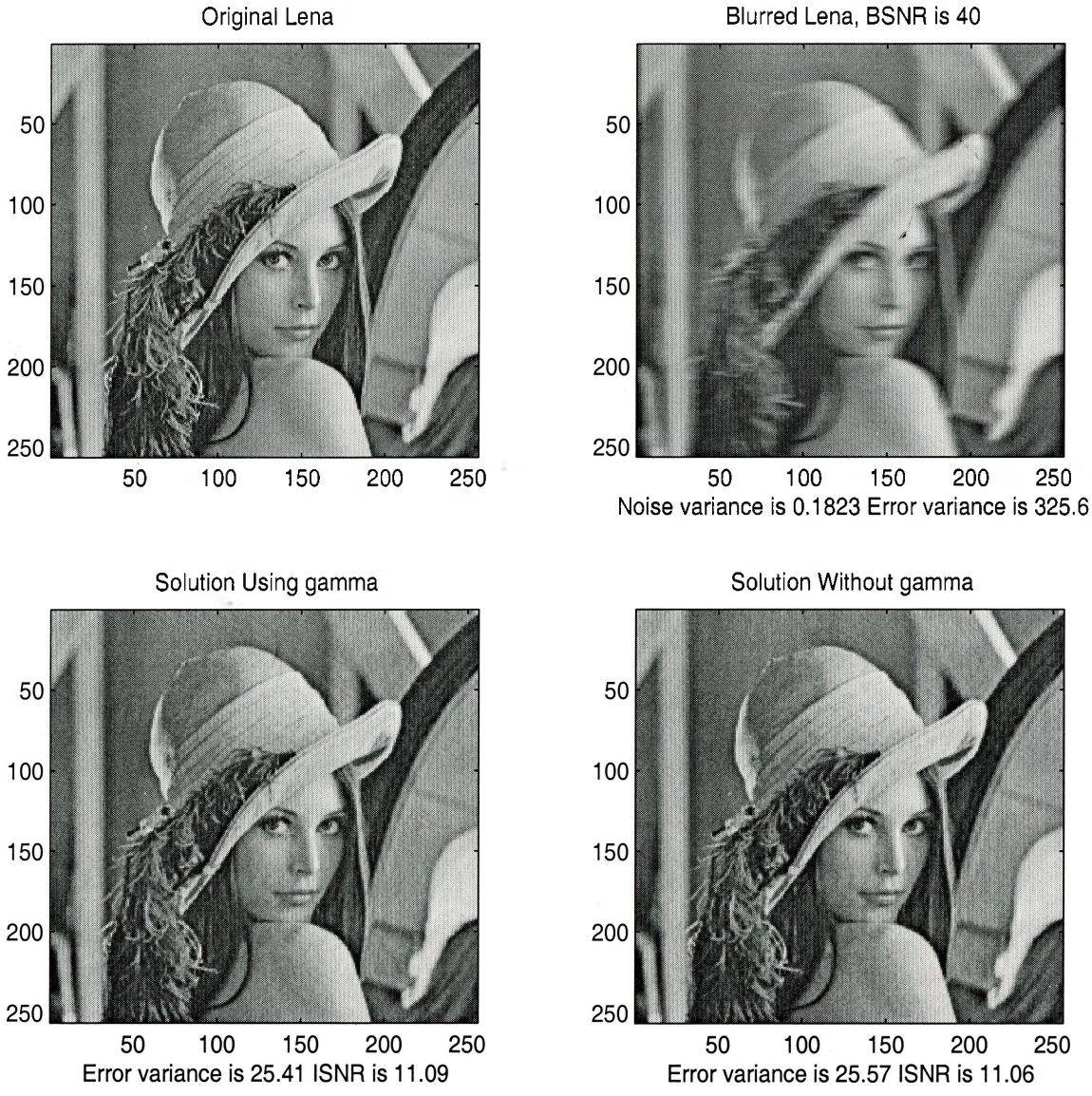


Figure 4.4: Effect of Stability Operator - Pictoral Results (Lena) BSNR = 40 dB

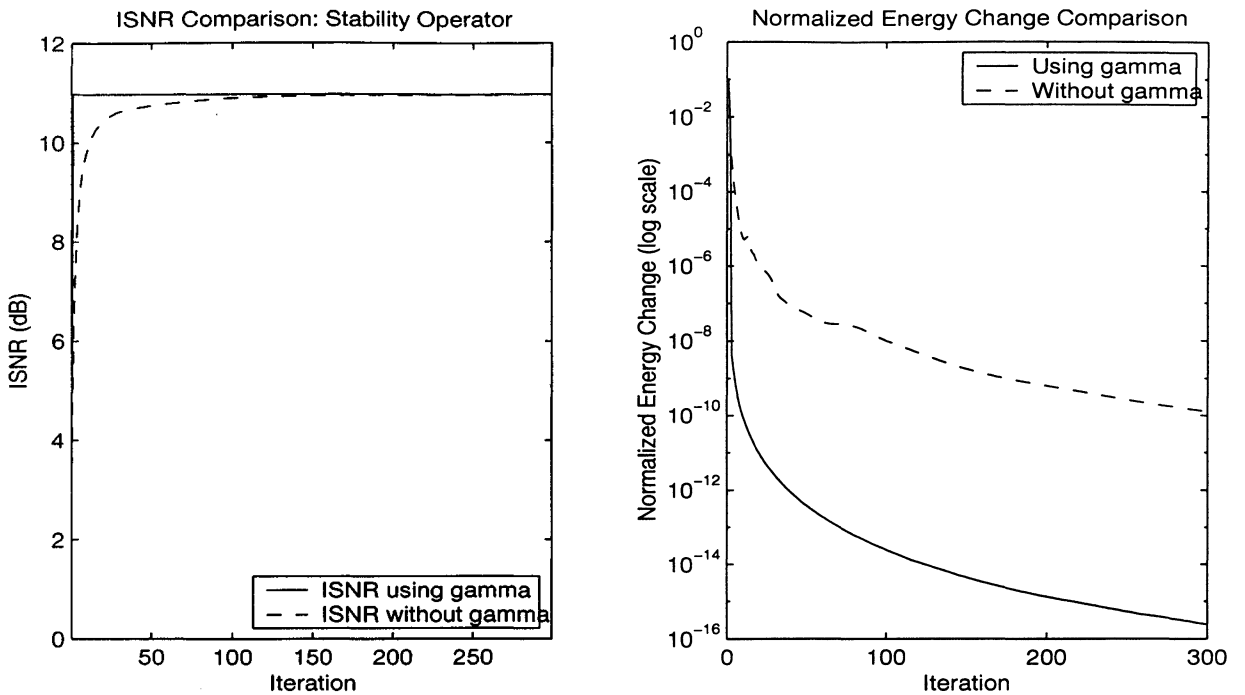


Figure 4.5: Effect of Stability Operator - ISNR (Cameraman) BSNR = 40 dB

$$\hat{\mathcal{S}}_0 = \frac{\mathcal{H}^* \mathcal{R}}{|\mathcal{H}|^2 + \rho_0}$$

where

$$\rho_0 = 0.5\left(1 - |\mathcal{H}|^2\right)$$

then $\rho_0$ was set to zeros. Before applying the image estimator (4.16), estimates for $\gamma$ and $\rho$ were calculated as follows:

- For the stabilized case, in order to reduce computational requirements, we use the following simplified procedure to determine $\gamma$ and $\rho$ rather than a multiple nonlinear equation solver as suggested earlier.
  - $\gamma_k$ was first calculated using (4.23) with $\rho_{k-1}$.
  - then $\rho_k$ was calculted using (4.27) with $\gamma_k$

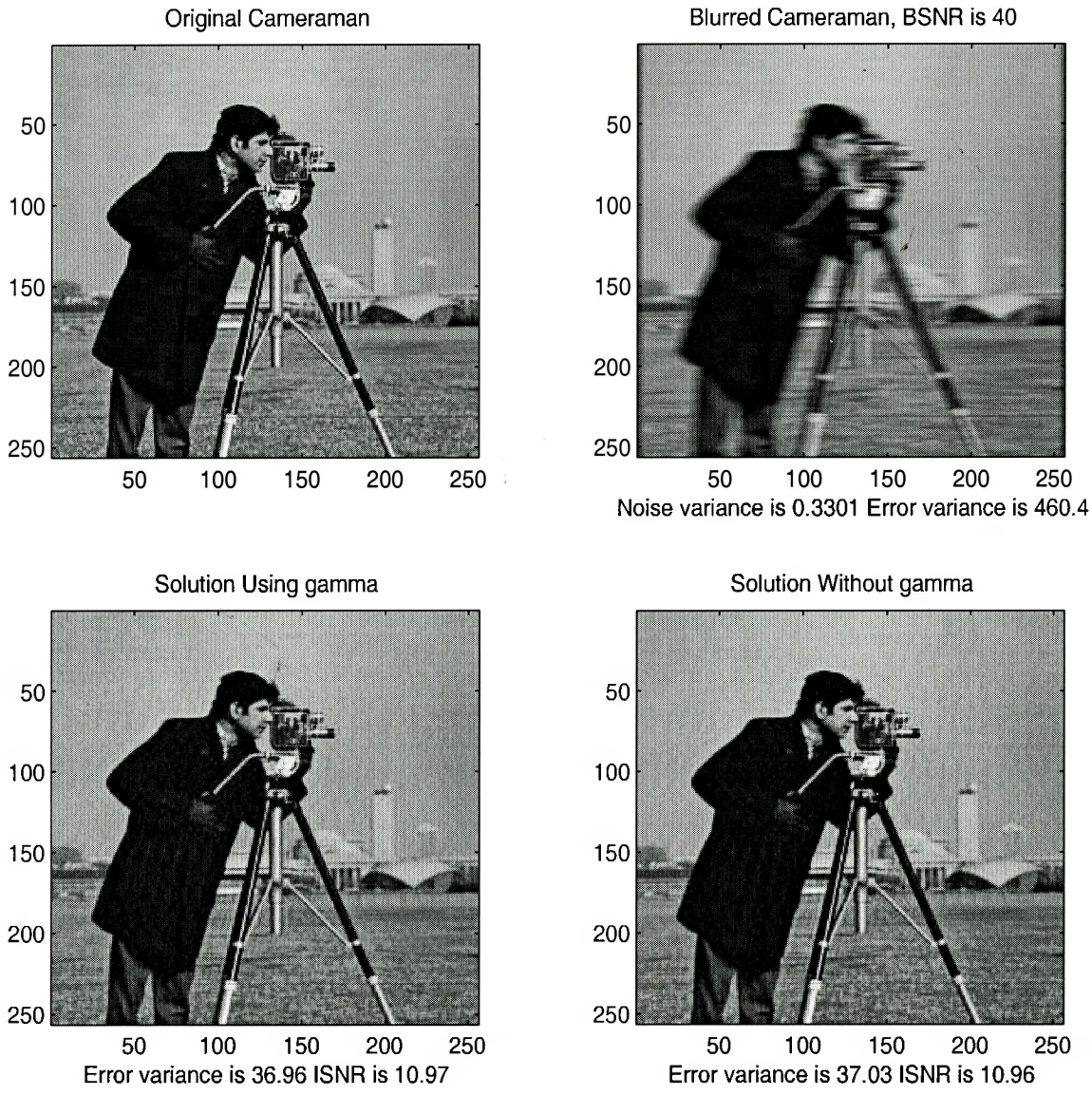


Figure 4.6: Effect of Stability Operator - Pictoral Results (Cameraman) BSNR = 40 dB

- For the unstabilized case, $\rho$ was calculated using (4.27) with $\gamma$ set to zeros.

## 4.4 The Stability Operator: Effect on Asymptotic Error

In this section we evaluate the effect of $\gamma$ on the asymptotic image error ($k \to \infty$). Asymptotically, $\rho$ and $\gamma$ are constant in $k$ and $\mathcal{E}_{\hat{\mathcal{H}}_k} = \mathbf{0}$, hence the error expansion of (B.3) is:

$$\mathcal{E}_{\hat{\mathcal{S}}_k} = \mathcal{N}\left[\frac{\mathcal{H}^*}{\mathcal{Z}}\sum_{i=0}^{k}\left(\frac{\gamma}{\mathcal{Z}}\right)^i\right] - \mathcal{S}\left[\frac{\rho}{\mathcal{Z}}\sum_{i=0}^{k}\left(\frac{\gamma}{\mathcal{Z}}\right)^i\right] \tag{4.31}$$

where $\mathcal{Z} = |\mathcal{H}|^2 + \gamma + \rho$. If we assume the term

$$\frac{\gamma}{\mathcal{Z}} < 1 \quad \forall \; u, v$$

then

$$\mathcal{E}_{\hat{\mathcal{S}}_k} = \mathcal{N}\left[\frac{\mathcal{H}^*}{\mathcal{Z}}\left[\frac{1-\left(\frac{\gamma}{\mathcal{Z}}\right)^{k+1}}{1-\left(\frac{\gamma}{\mathcal{Z}}\right)}\right]\right] - \mathcal{S}\left[\frac{\mathcal{H}^*}{\mathcal{Z}}\left[\frac{1-\left(\frac{\gamma}{\mathcal{Z}}\right)^{k+1}}{1-\left(\frac{\gamma}{\mathcal{Z}}\right)}\right]\right] \tag{4.32}$$

taking the limit of (4.32) yields:

$$\begin{aligned}\lim_{k\to\infty}\mathcal{E}_{\hat{\mathcal{S}}_k} &= \mathcal{N}\left[\frac{\mathcal{H}^*}{\mathcal{Z}\left(1-\frac{\gamma}{\mathcal{Z}}\right)}\right] - \mathcal{S}\left[\frac{\rho}{\mathcal{Z}\left(1-\frac{\gamma}{\mathcal{Z}}\right)}\right] \\ &= \frac{\mathcal{H}^*\mathcal{N} - \rho\mathcal{S}}{|\mathcal{H}|^2 + \rho}\end{aligned} \tag{4.33}$$

which is independent of $\gamma$. Further, this limiting error is the same as that for any direct regularized least squares method when $\rho = \lambda\,|\mathcal{C}|^2$. Hence the introduction of $\gamma$ does not affect the asymptotic error. However, by selecting $\gamma$ so that the *transient* error is minimized, significant improvements in convergence are obtained.

## 4.5 A Priori Deconvolution Results

Figures 4.7 to 4.14 show further pictorial results of the *a priori* (nonblind) algorithm presented in this chapter. Figures 4.7 through 4.11 show experimental results using the iteration outlined by (4.16) on the "Lena" image. Images for 4.7 to 4.9 were degraded using a 9-pixel horizontal blur and figures 4.10 to 4.11 were degraded by a 7-by-7 uniform blur. Both scenarios include additive noise at various BSNR levels.

Figures 4.12 and 4.14 show experimental results using the iteration outlined by (4.16) on the "Cameraman" image. Images were degraded using a 9-pixel horizontal blur and additive noise.

It is seen from the figures that the improvement as measured by ISNR offered by the proposed de-blurring process is significant in comparison to the iterative TM algorithm which is used as a benchmark. Further to the recorded ISNR improvements it can be readily seen that the prominent ringing effect evident in the TM case is significantly reduced for the proposed method. It should also be noted here that the iterative TM algorithm is conducted under *ideal* conditions, that is to say with regularization parameter $\lambda_{TM}$ set to the normally unknown value of

$$\lambda_{TM} = \frac{\sum_{u,v} |\mathcal{N}(u,v)|^2}{\sum_{u,v} |\mathcal{C}(u,v)\mathcal{S}(u,v)|^2} \tag{4.34}$$

where $\mathcal{C}$ is the Laplacian operator, $\mathcal{N}$ is the noise, and $\mathcal{S}$ is the real or underlying image. These results are also comparable to, and in most cases outperform other methods including another frequency adaptive iteration method presented in [22]. Other comparable "single channel" iterative deconvolution results are presented in [24][28][55].

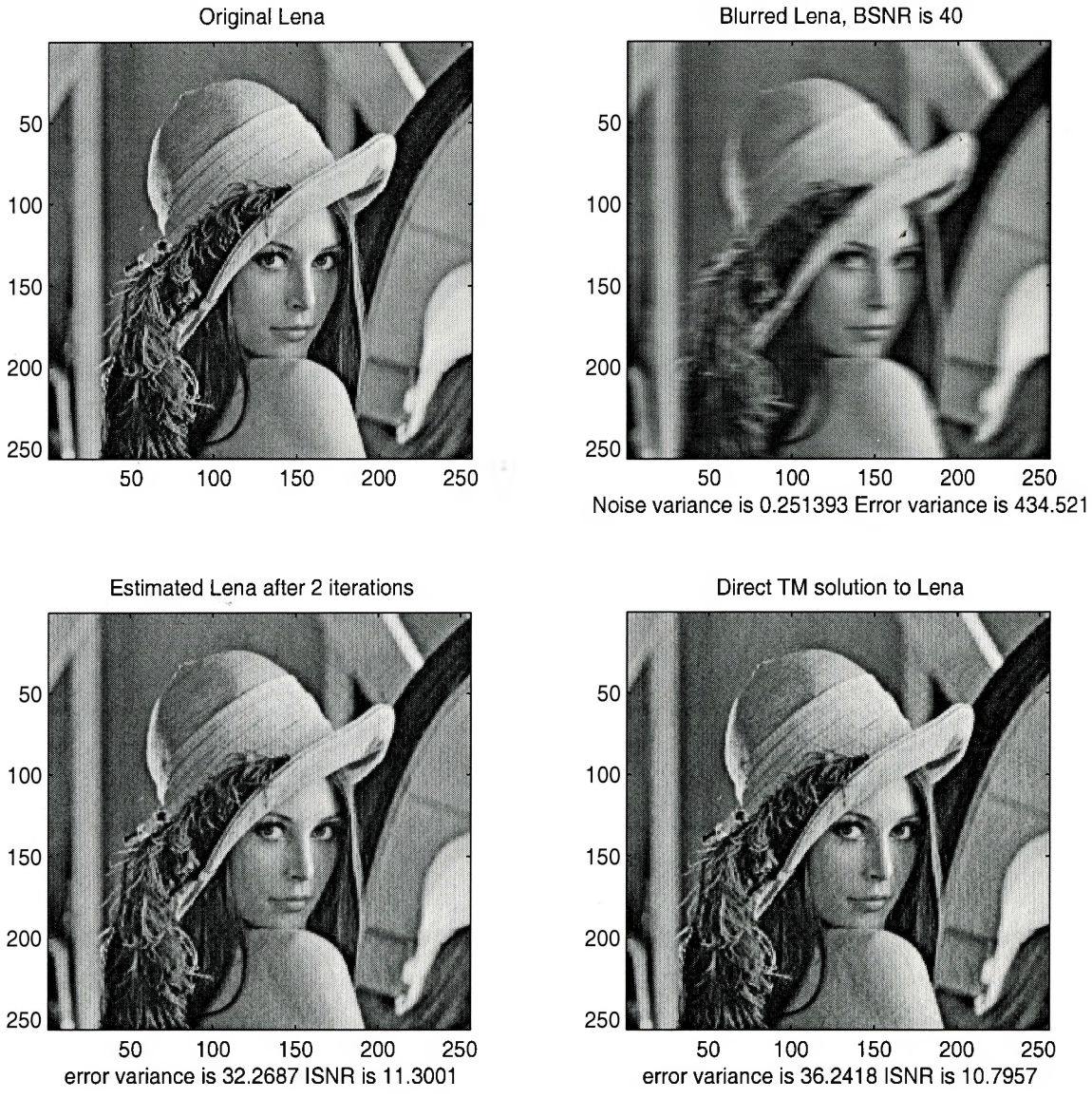


Figure 4.7: Deconvolution results: 9 Pixel Horizontal Blur. BSNR = 40 dB

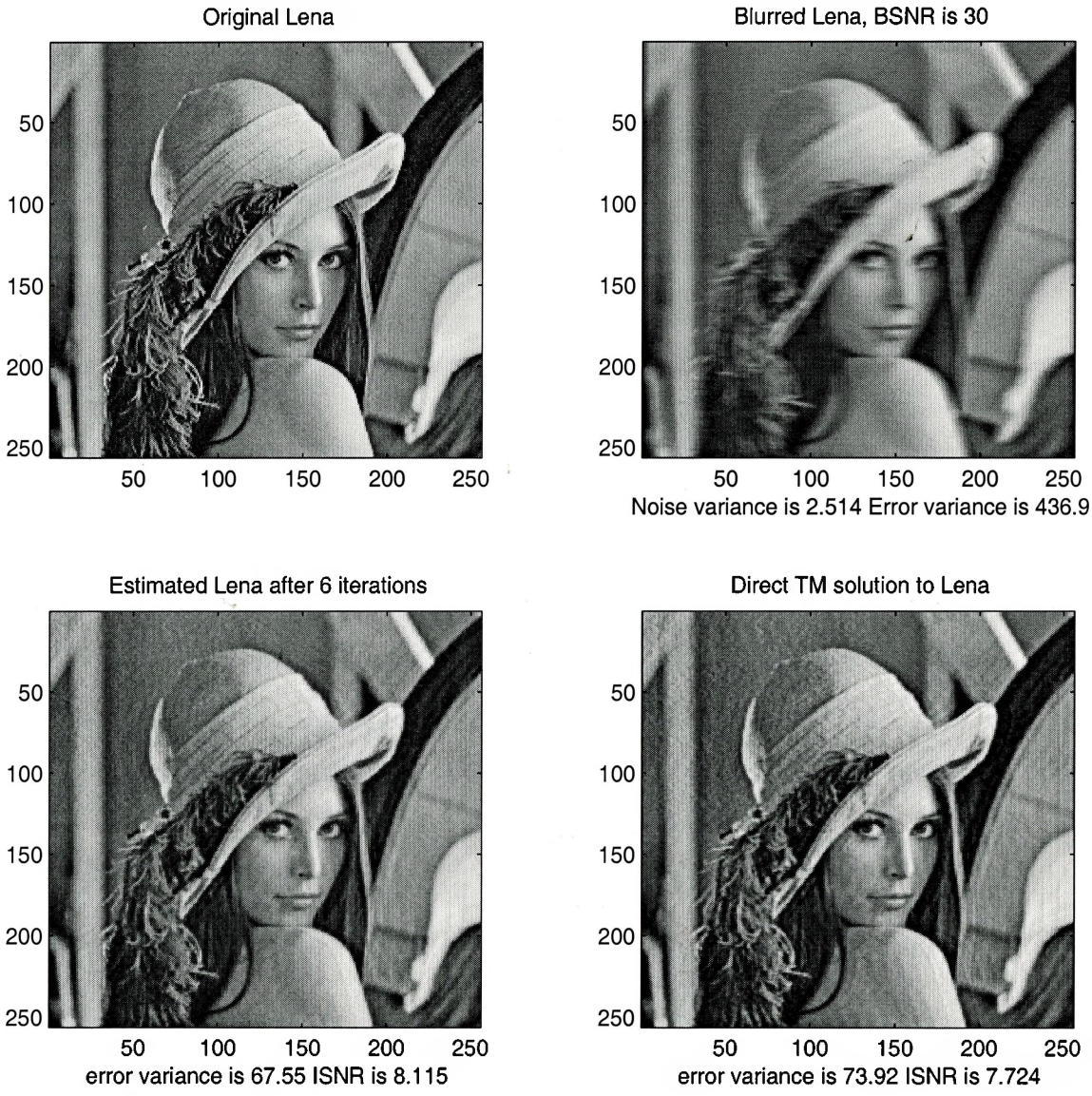


Figure 4.8: Deconvolution results: 9 Pixel Horizontal Blur. BSNR = 30 dB

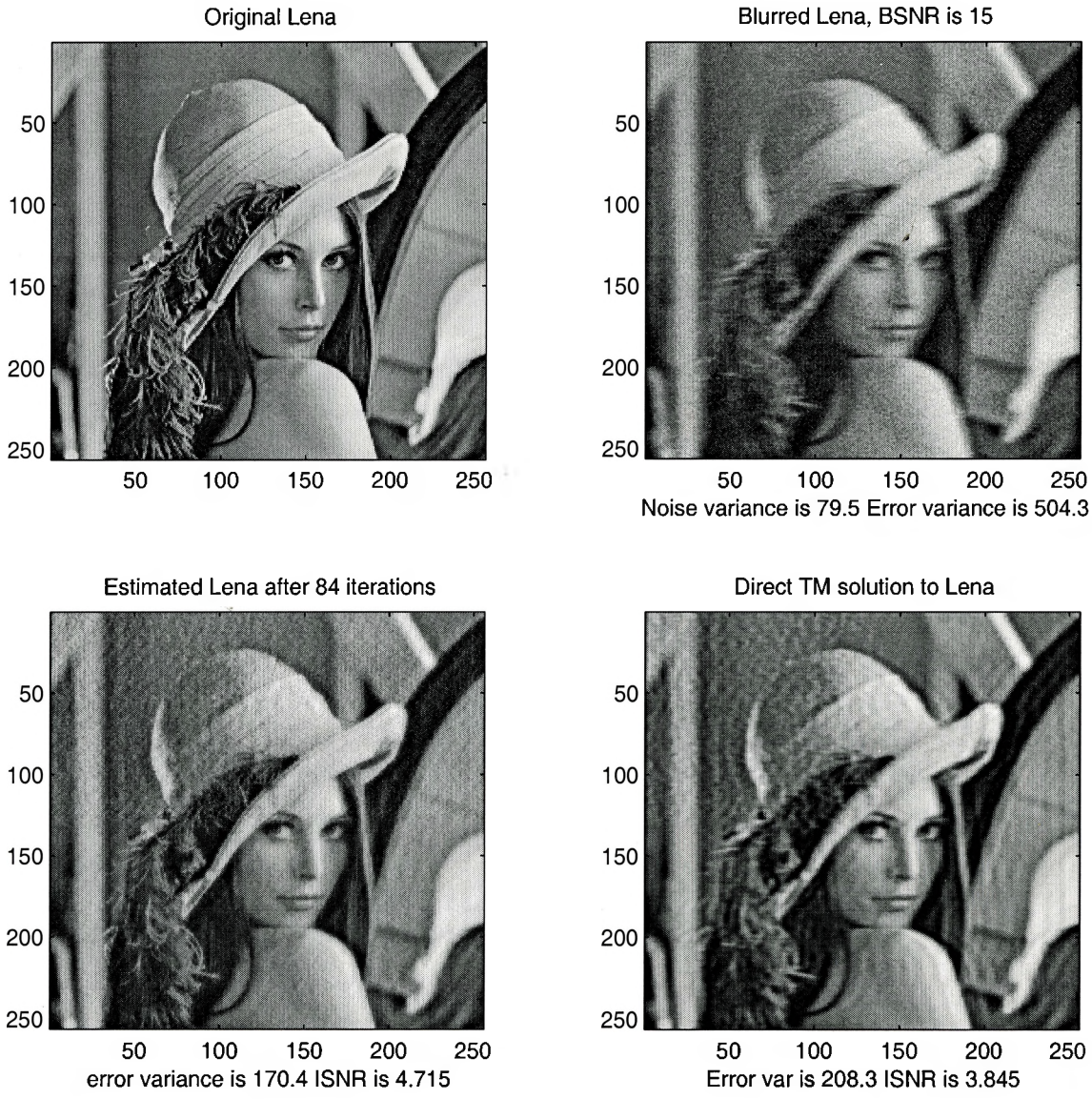


Figure 4.9: Deconvolution results: 9 Pixel Horizontal Blur. BSNR = 15 dB

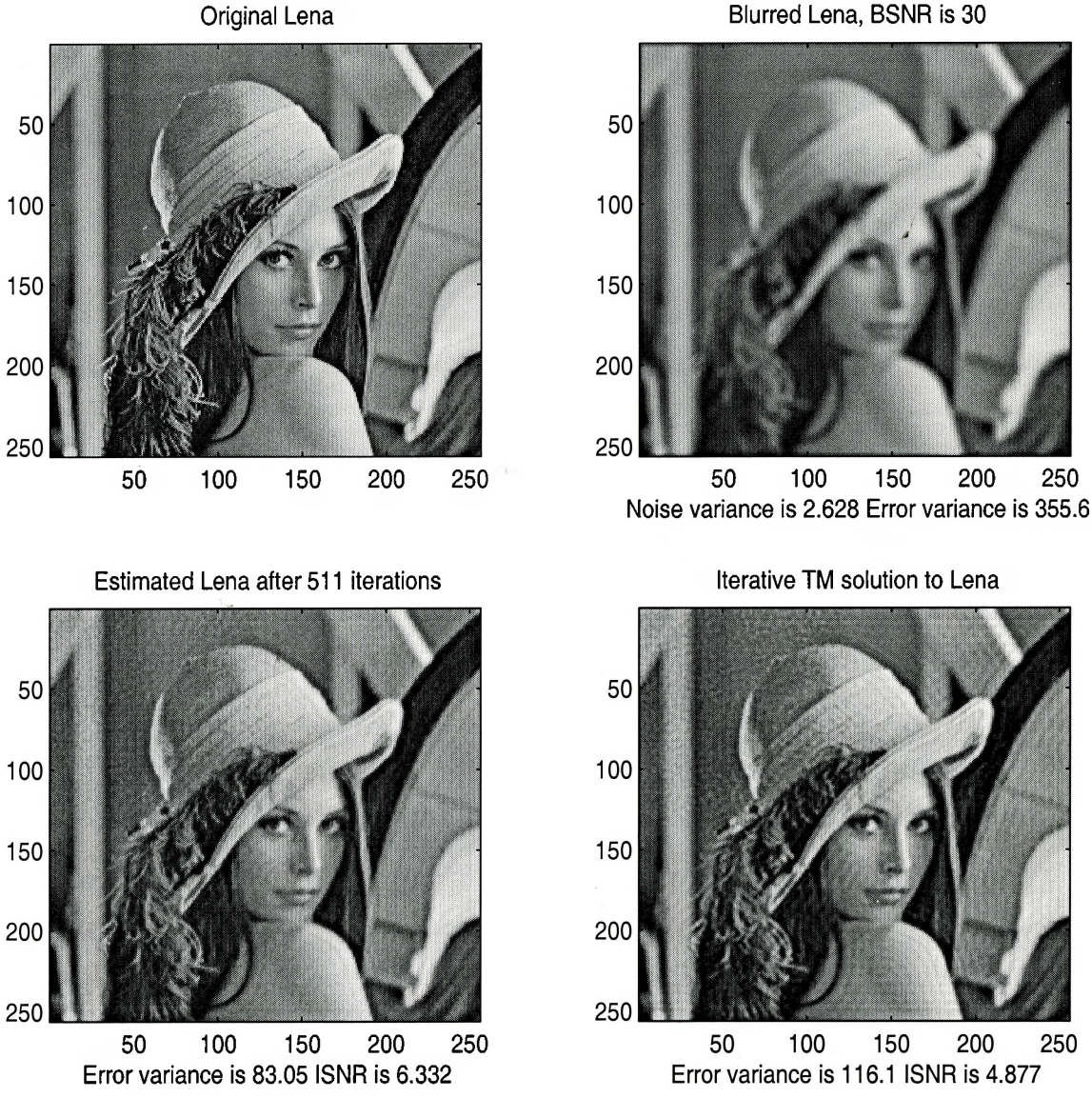


Figure 4.10: Deconvolution results: 7 x 7 Uniform Blur. BSNR = 30 dB

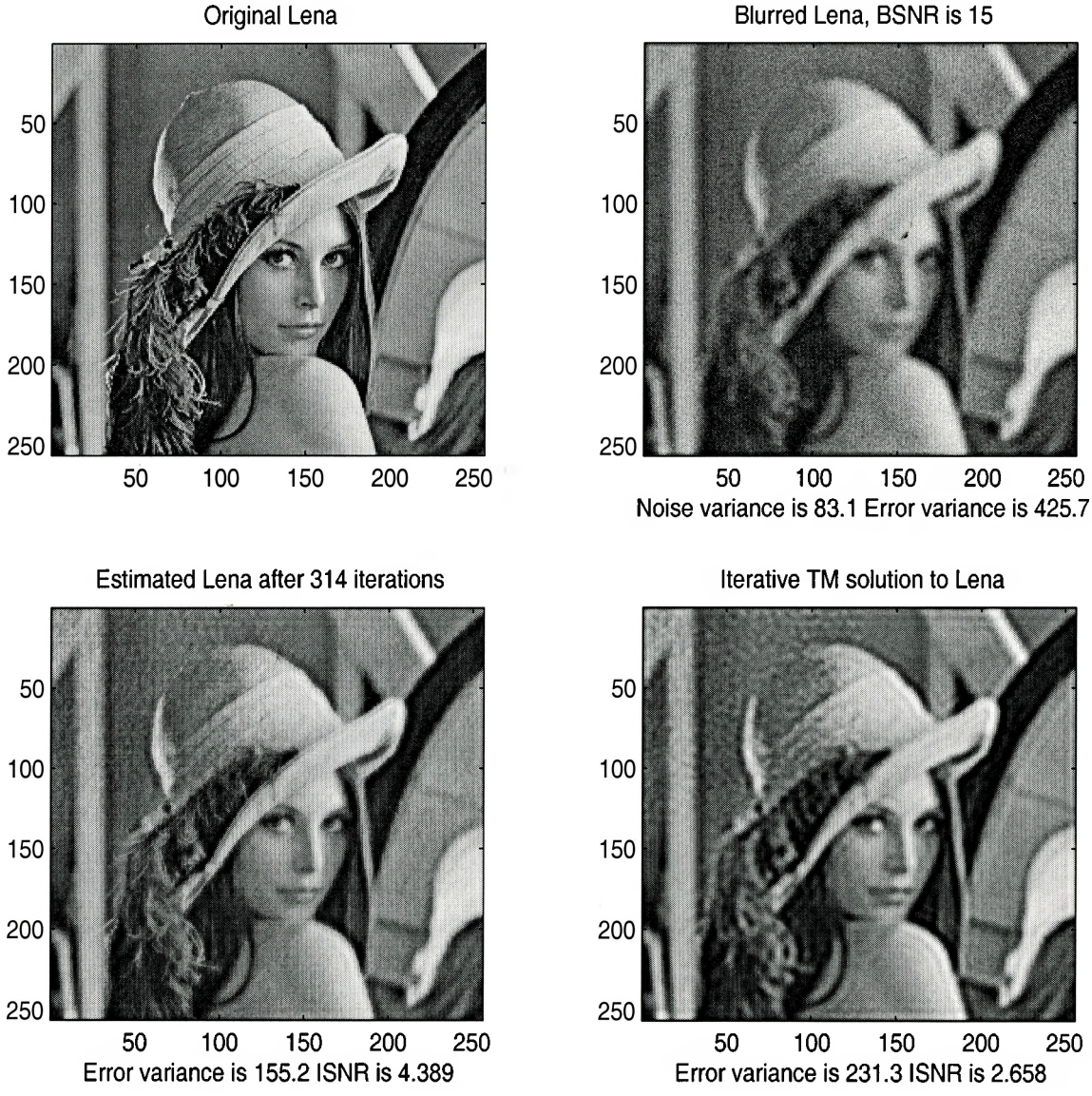


Figure 4.11: Deconvolution results: 7 x 7 Uniform Blur. BSNR = 15 dB

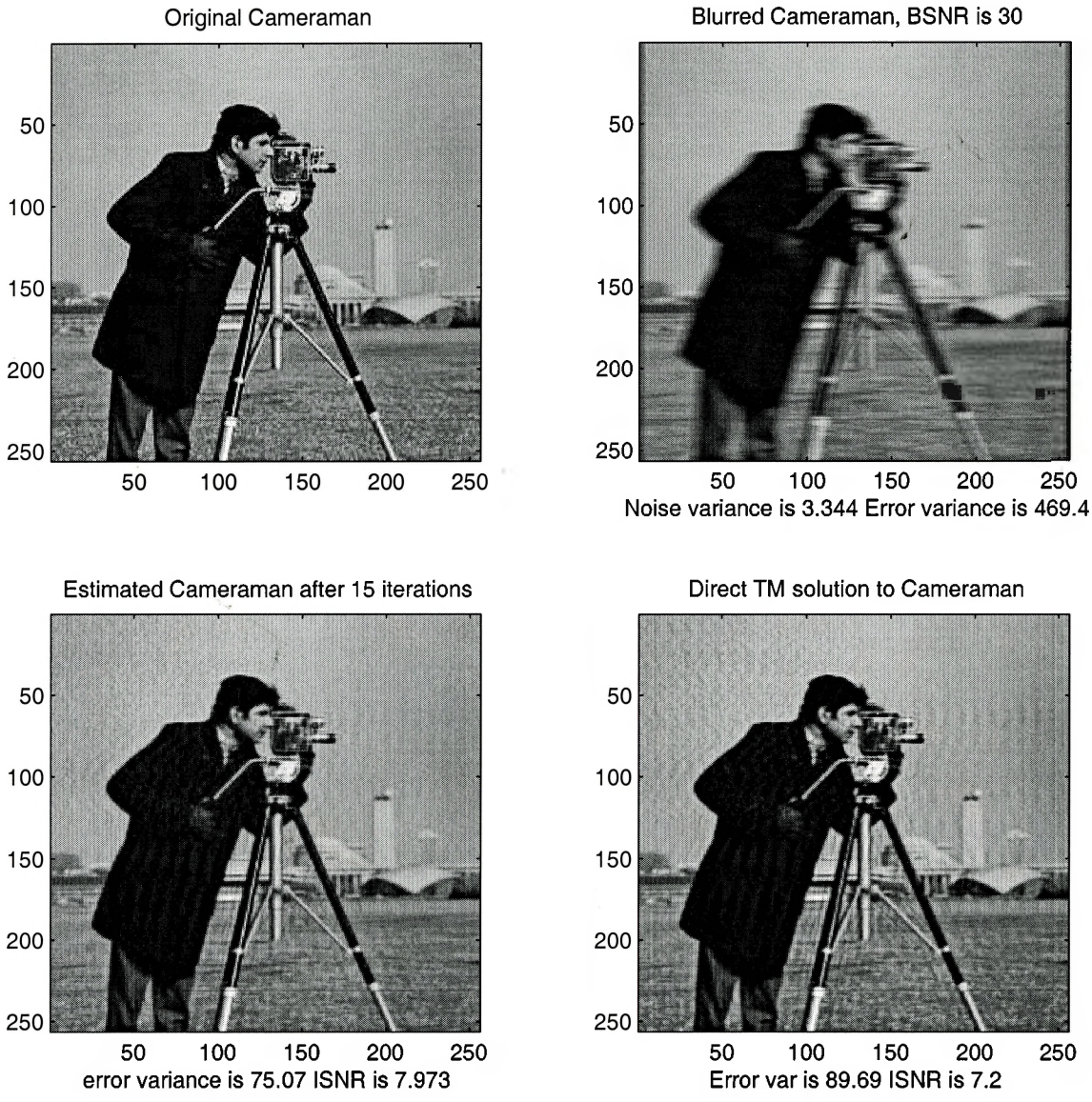


Figure 4.12: Deconvolution results: 9 Pixel Horizontal Blur. BSNR = 30 dB

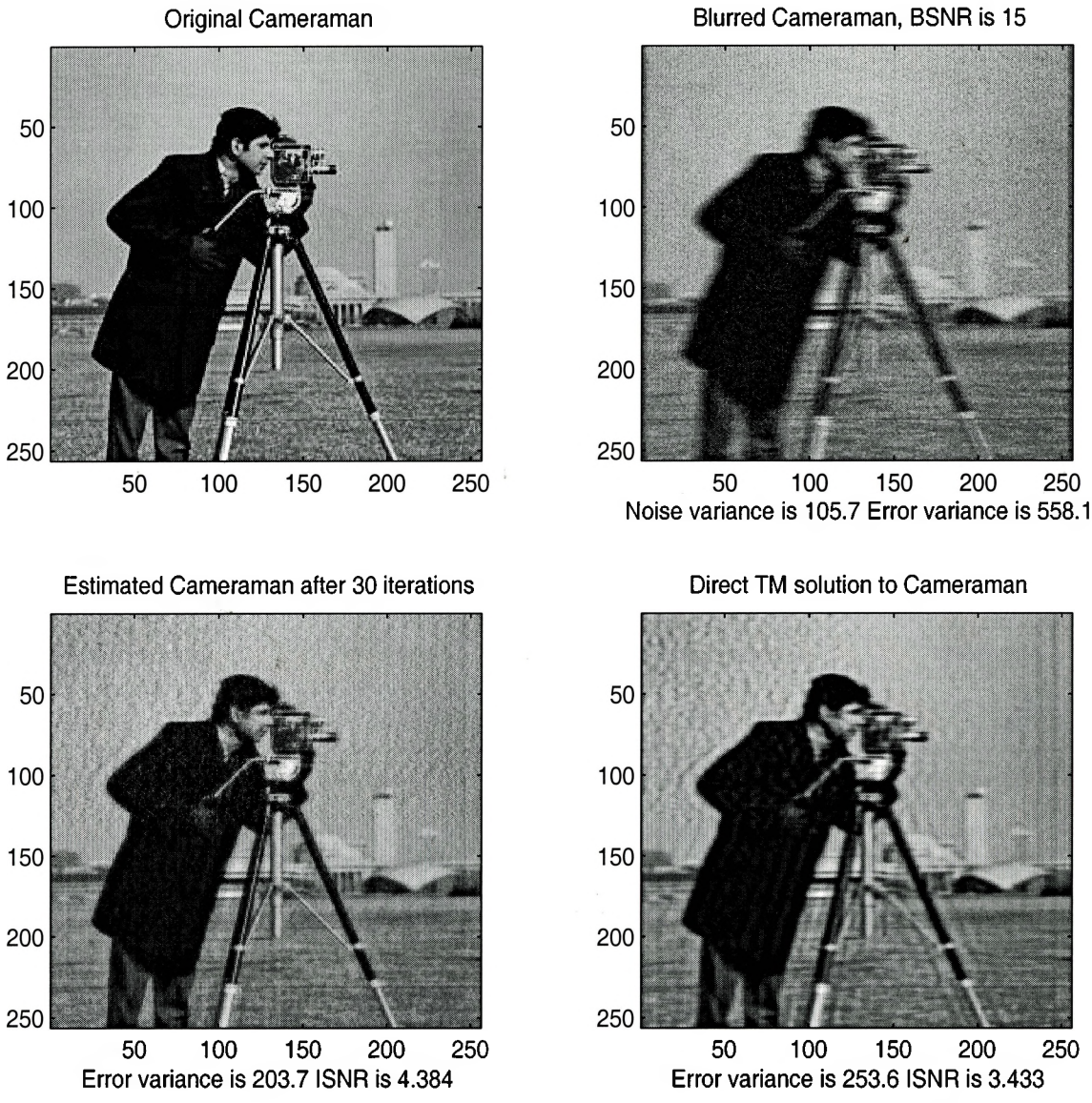


Figure 4.13: Deconvolution results: 9 Pixel Horizontal Blur. BSNR = 15 dB

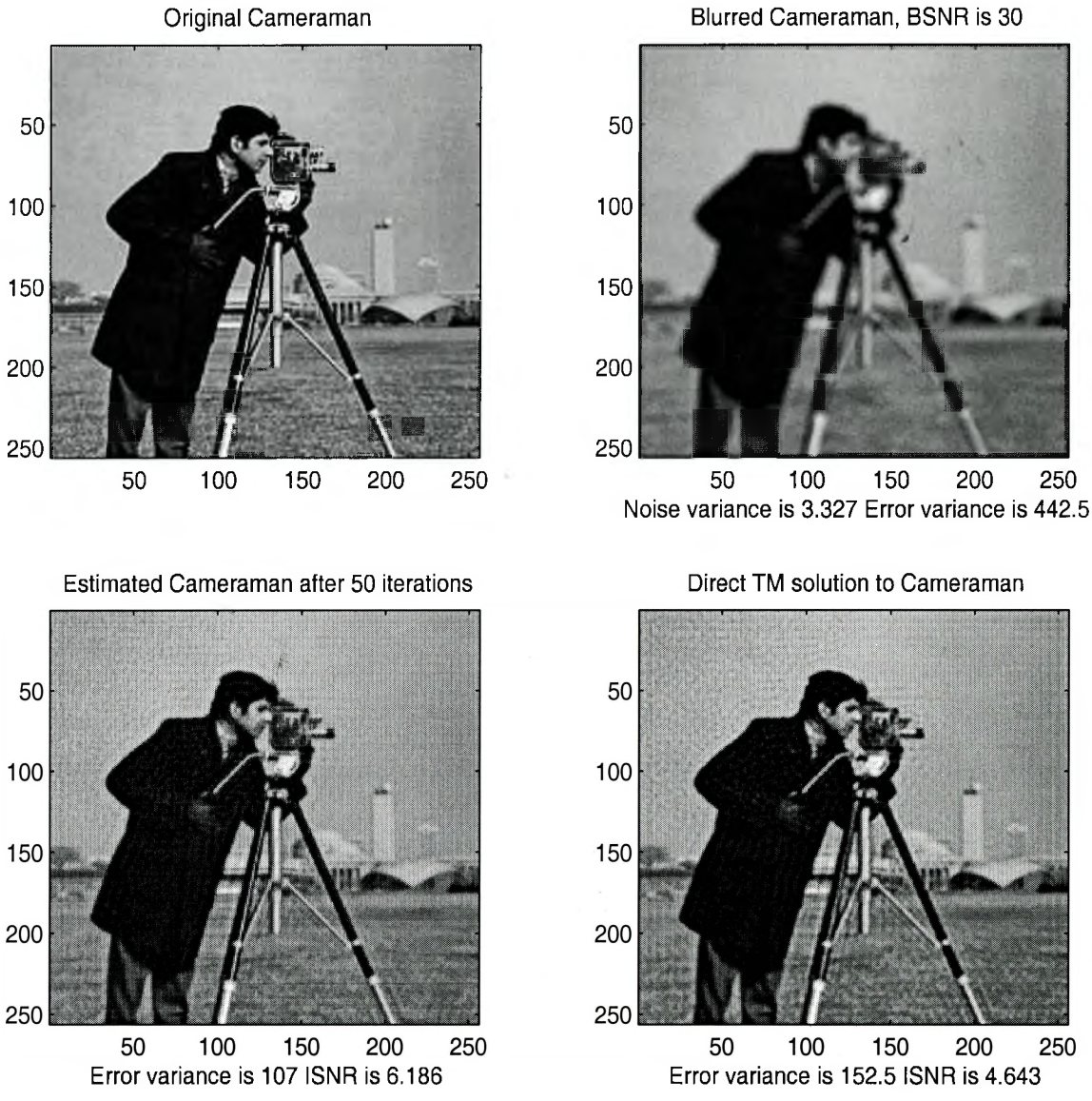


Figure 4.14: Deconvolution results: 7 x 7 Uniform Blur. BSNR = 30 dB

## 4.6 Discussion

In Section 4.2.2, optimal frequency-adaptive stability and regularization operators are derived. This optimization is facilitated by a novel error analysis such that $\gamma$ and $\rho$ can be adaptively estimated based on a past history that is encapsulated in the updated variables $\Sigma_{\mathcal{N}}^{(k)}$, $\Sigma_{\mathcal{H}}^{(k)}$ and $\Sigma_{\rho}^{(k)}$. In developing these error propagation terms in this way, more accurate error estimates, and therefore better updates of $\gamma$ and $\rho$ are possible. In other words, as the iteration continues, the estimates for $\mathcal{S}$ and $\mathcal{N}$ improve, and therefore our estimates for $\mathcal{E}_{\hat{\mathcal{S}}_k}$ and by extension $\gamma$ and $\rho$ improve as well.

The use of a stability operator $\gamma$ has been shown experimentally to improve the convergence rate in a dramatic fashion. The stability and regularization operators are optimized on a coefficient-by-coefficient basis in the DFT domain. This is unlike many methods that define a regularization operator that is static, and control the regularization process by a parameter that is a positive scalar quantity.

In the next chapter, these ideas will be applied to blind deconvolution where the PSF is not known *a priori*.

# Chapter 5

# Blind Deconvolution

In previous chapters, the issue of identifying the image blur or point spread function was not addressed. Indeed, most classical image restoration techniques require explicit prior knowledge of the PSF before any reasonable results can be achieved. The sad reality, however is that for most cases the PSF is not known *a priori.* Hence the need for *blur estimation* or *blind deconvolution* techniques. Many methods have been developed to estimate this function, for various types of image classes. In general, there are two main approaches to blind deconvolution[32]:

1. Identify the PSF separately for the image (*a priori blur identification*).
    - DFT "Null-search" Techniques
        - Zero Sheet Separation
        - Cepstrum Techniques
        - Bicepstrum Techniques
    - Point Source Techniques
2. Incorporate the PSF identification procedure with the restoration algorithm (*joint PSF/image estimation*).

- Maximum Likelihood Estimation (MLE)
- Generalized Cross Validation (GCV)
- Iterative Blind Deconvolution (IBD)
- Simulated Annealing (SA)
- Non-negativity and Support Constraints Recursive Inverse Filtering (NAS-RIF)
- Minimum Entropy Deconvolution (MED)

Initially, existing blind image deconvolution methods will be briefly discussed. Then, a simple *a priori* method will be explored as a solution to the initialization problem that many blind iterative image restoration techniques encounter. *A priori* methods can sometimes be accurate particularly if the type of PSF is known and easily parameterized. This result can then be used as a starting point for a more complex iterative technique, which will be presented and applied to the problem at hand. This revised technique will address the inherent instability and slow convergence of the IBD and other algorithms by using the stability operator $\gamma$ introduced in Chapter 4. A detailed description of the proposed blind iterative deconvolution algorithm will then be presented.

## 5.1 *A Priori* Blur Estimation

Most a priori blur identification methods rely on known features of the PSF and parameterization of the blur to reduce the complexity of the identification procedure. Many of the popular parametric PSF models were presented in Section 2.2. Techniques may look at estimating the PSF from known features of the true image like point sources on a uniform background or edges. Others invoke prior knowledge of the blur mechanism, using the frequency domain nulls or magnitude spectrum minima

of the blurred image to determine the PSF parameters. This approach is justified by the frequency domain realization of the image formation model (2.14), where the nulls of $\mathcal{H}$ will be those in common with $\mathcal{HS}$. Since additive noise tends to mask these nulls, the effectiveness of this approach depends a lot on the signal-to-noise ratio. Methods that estimate PSF parameters using frequency domain zeros include spectra magnitude cepstrum and bicepstrum techniques. These techniques tend to be relatively low in complexity, and are relatively easy to implement.

## 5.2 Joint PSF/Image Estimation Methods

The iterative or joint PSF estimation techniques are broader in scope and more difficult to use, on both computational and practical levels. Again, many methods must rely on parameterized models of the PSF to reduce computational complexity and facilitate convergence. The following sections deal with three major groups of blind iterative image restoration techniques; stochastic techniques that use an autoregressive image formation model and moving average blur model (ARMA), algebraic techniques that use non-expansive, nonlinear constraints in a recursive scheme, and techniques based on higher-order statistics.

### 5.2.1 ARMA Methods

A good example of a stochastic method is the use of maximum likelihood estimation (MLE) when applied to blind image restoration. This method uses an autoregressive moving average (ARMA) model for the blurred image, where the true image is assumed to be a 2-D autoregressive process, and the blur a 2-D moving average process. Many procedures use the expectation-maximization (EM) algorithm to achieve MLE image restoration[27][35][36][39]. Another example of a stochastic blur identification method that uses the ARMA blurred image model is generalized cross validation

(GCV), which can also be used for estimation of the regularization parameter $\lambda$ and regularization operator $\mathbf{C}$[47][48]. The ML and GCV methods thus far are the most successful for blind image processing[32]. The AR model however is not valid for images with abrupt nonstationarities such as edges, so restoration results tend to be over-smooth. Both the MLE and GCV methods have difficulty converging when the dimension of the optimization parameter space is too high, resulting in the algorithm becoming trapped in local minima. Also, the respective objective functions for both methods have a tendency to become insensitive to parameter changes when the total number of parameters is large. Both methods therefore require some other means to obtain an initial "guess" for PSF parameters before any standard optimization procedure can be used. Even then, convergence cannot be guaranteed therefore manual inspection of the final image estimate is required to verify results.

### 5.2.2 Non-Parametric Deterministic Constraint Methods

Non parametric methods such as iterative blind deconvolution (IBD)[2], simulated annealing (SA)[41][31] and non-negativity and support constraints recursive inverse filtering (NAS-RIF)[32] use deterministic constraints to achieve image restoration results. In general these methods do not assume parametric models for blurring mechanisms. The simulated annealing algorithm tends to be too computationally complex for most image restoration problems with an algorithm order per iteration of $O(M_s^4)$ where $M_s$ is the number of pixels in an image[1]. In comparison, The IBD algorithm has order $O(M_s \log_2(M_s))$ per iteration. The NAS-RIF algorithm is sensitive to additive observation noise, and is therefore only effective for high SNR image restoration applications. The IBD method, in general, tends to be unstable in its application[32][64] particularly at low SNR.

The IBD algorithm can be argued to be a blind application of projections onto

[1]In our case, this value $M_s$ is equal to $N^2$

convex sets (POCS) if the constraints chosen represent closed convex sets. Care must be taken in choosing each constraint; if three or more convex sets do not intersect, a *"greedy" limit cycle* will result[21]. As outlined in [21] and [65] it is also important to incorporate relaxation in the process.

### 5.2.3 High Order Statistics Based Methods

This class of methods account for the non-Gaussian nature of many images. Most apply an FIR inverse filter that is adaptively updated using the higher order statistics of the most recent image estimate. The FIR filter taps are optimized based on an *a priori* HOS model of the true image.

An example of a HOS based method is minimum entropy deconvolution. This method was developed for de-blurring images with "spikey" features such as seismic data, and as an extension to this any two-tone image[63][60].

### 5.2.4 Unification of Image Estimation Methods

Many image estimation techniques described herein, blind or otherwise, have the same characteristic form. The formal relationships between many methods have been studied in [33] where a generic form for these inverse filters is given by:

$$\hat{\mathbf{s}}_k = \left(\mathbf{H}^{\mathbf{T}}\mathbf{H} + \mathbf{M}\right)^{-1}\mathbf{H}^{\mathbf{T}}\mathbf{r} \tag{5.1}$$

where $\mathbf{H}$ can be the blurring matrix for a known blur, or alternately an estimate blur in a blind iterative process. For separate positive definite methods, the regularization term $\mathbf{M}$ takes on differing values. For example direct algebraic methods use:

$$\mathbf{M} = \lambda \mathbf{C}^{\mathbf{T}}\mathbf{C} \tag{5.2}$$

where the MLE method using EM has the form:

$$\mathbf{M} = \sigma_n^{2(k)} \mathbf{I} (\mathbf{\Lambda}_X^{(k)})^{-1} \tag{5.3}$$

The former form uses $\lambda$ and $\mathbf{C}$ in the various forms described in Chapter 2, where the latter form uses updated estimates of the observation noise variance $\sigma_n^2$ and the covariance matrix of the AR image formation system $\mathbf{\Lambda}_X$[27]. $\mathbf{I}$ is the identity matrix. These forms hold some similarity to the Wiener filter where the regularization term is:

$$\mathbf{M} = \mathbf{\Lambda_{nn}} \mathbf{\Lambda_{ss}}^{-1} \tag{5.4}$$

where $\mathbf{\Lambda_{nn}}$ and $\mathbf{\Lambda_{ss}}$ represent covariance matrices for the observation noise and image respectively. It is a common assumption that both the noise process and original image are multivariate Gaussian with zero mean, making (5.3) and (5.4) equivalent. M can also be chosen as:

$$\mathbf{M} = \lambda \mathbf{I} \tag{5.5}$$

which is the standard form for Tikhonov regularization.

The IBD process as presented in [32] also has the same form as (5.1), where the DFT domain expression for the $k$th estimate of the image $\hat{\mathcal{S}}_k$ is:

$$\hat{\mathcal{S}}_k = \frac{\hat{\mathcal{H}}_k^* \mathcal{R}}{\left|\hat{\mathcal{H}}_k\right|^2 + \alpha \left|\tilde{\mathcal{S}}_{k-1}\right|^{-2}} \tag{5.6}$$

where $\alpha$ is a positive scalar parameter representing the additive noise energy and the "tilde" identifies that element in the expression as a nonlinearly constrained variable, that is:

$$\tilde{\mathcal{S}}_k = C\left(\hat{\mathcal{S}}_k\right)$$

where $C$ represents a constraint operator.

The notion of using the inverse of an image estimate as a regularization operator is supported in [10] where it was proven that almost all estimators[2] for the regularization parameter $\lambda$ are equivalent when the regularization operator term $|C|^2$ is chosen to be $|\mathcal{S}|^{-2}$. Further to this, the estimated value for $\lambda$ was found to be the additive noise variance $\sigma_n^2$. This insight on the use of the image inverse as a form of regularization operator justifies the often-used Laplacian operator, which tends to whiten or de-correlate the image.

The distinction between the blind case as opposed to the *a priori* case is that with the former, an estimate or update of the blur operator must be obtained at each iteration in addition to the estimate of the image. Hence, by extension, the IBD estimator of Ayers and Dainty in [2] for the PSF $\mathcal{H}$ at iteration $k$ is:

$$\hat{\mathcal{H}}_k = \frac{\hat{\mathcal{S}}_k^* \mathcal{R}}{\left|\hat{\mathcal{S}}_k\right|^2 + \alpha \left|\tilde{\mathcal{H}}_{k-1}\right|^{-2}} \tag{5.7}$$

By comparison to (5.6), we see that the "roles" of the image and blur estimates are reversed. Interestingly, the MLE algorithm presented in [27] has the same form for PSF estimator with a different value for the "regularization" term:

$$\hat{\mathcal{H}}_k = \frac{\hat{\mathcal{S}}_k^* \mathcal{R}}{\left|\hat{\mathcal{S}}_k\right|^2 + N^2 \mathcal{S}_{S|r}^{(k)}} \tag{5.8}$$

Here the updated variable $\mathcal{S}_{S|r}^{(k)}$ represents the diagonalized values for the conditional covariance matrix of $\mathbf{s}$ given $\mathbf{r}$ and the $k$th estimate of the unknown parameter set. This covariance matrix is also assumed to be block-circulant as a simplification to analysis, enabling this Fourier domain implementation of the EM algorithm.

[2]The one exception found for this was the constrained least squares (CLS) estimate for $\lambda$.

## 5.3 A Blind Iterative Deconvolution Method

Figure 5.1 shows an overall schematic of the blind iterative procedure we are proposing. The *a priori* method presented in Chapter 4 is the innermost kernel of this procedure, which is essentially wrapped with the blur estimation procedure. A detailed algorithm corresponding to Figure 5.1 is given in Section 5.4.

### 5.3.1 Proposed Image and PSF Estimators

Instead of using one of the above (5.1) through (5.6) to estimate $\mathcal{S}$ at each iteration, the estimator (4.16) can be used with regularization and stability operators chosen as described in Chapter 4. The inclusion of a stability operator, as demonstrated in Section 4.3, will help the slow convergence rate that is characteristic of this type of iterative procedure.

For the blur estimator, we use the same logic as previously discussed with the IBD algorithm and switch the roles of $\mathcal{S}$ and $\mathcal{H}$ estimates in (4.16). The corresponding equation for $\hat{\mathcal{H}}_k$ is then:

$$\hat{\mathcal{H}}_{k+1} = \frac{\hat{\mathcal{S}}_k^* \mathcal{R} + \kappa_k \hat{\mathcal{H}}_k}{\left|\hat{\mathcal{S}}_k\right|^2 + \kappa_k + \delta_k} \tag{5.9}$$

where the *real* variables $\kappa_k$ and $\delta_k$ serve the same function as $\gamma_k$ and $\rho_k$ respectively. Error terms can be developed with optimal values for $\kappa$ and $\delta$ derived in the same manner as in Sections 4.1.1 and 4.2.

As introduced in Section 3.5, the application of constraints between iterations is a powerful method of removing the ambiguity possible with regularized least squares estimation methods. Empirically, it is easier to apply constraints to the PSF rather than the image since the physics behind the blurring process is better understood compared to that of image formation. Also, the PSF requires less parameters to estimate than does the image and in most cases, since images are rarely bandlimited, PSF

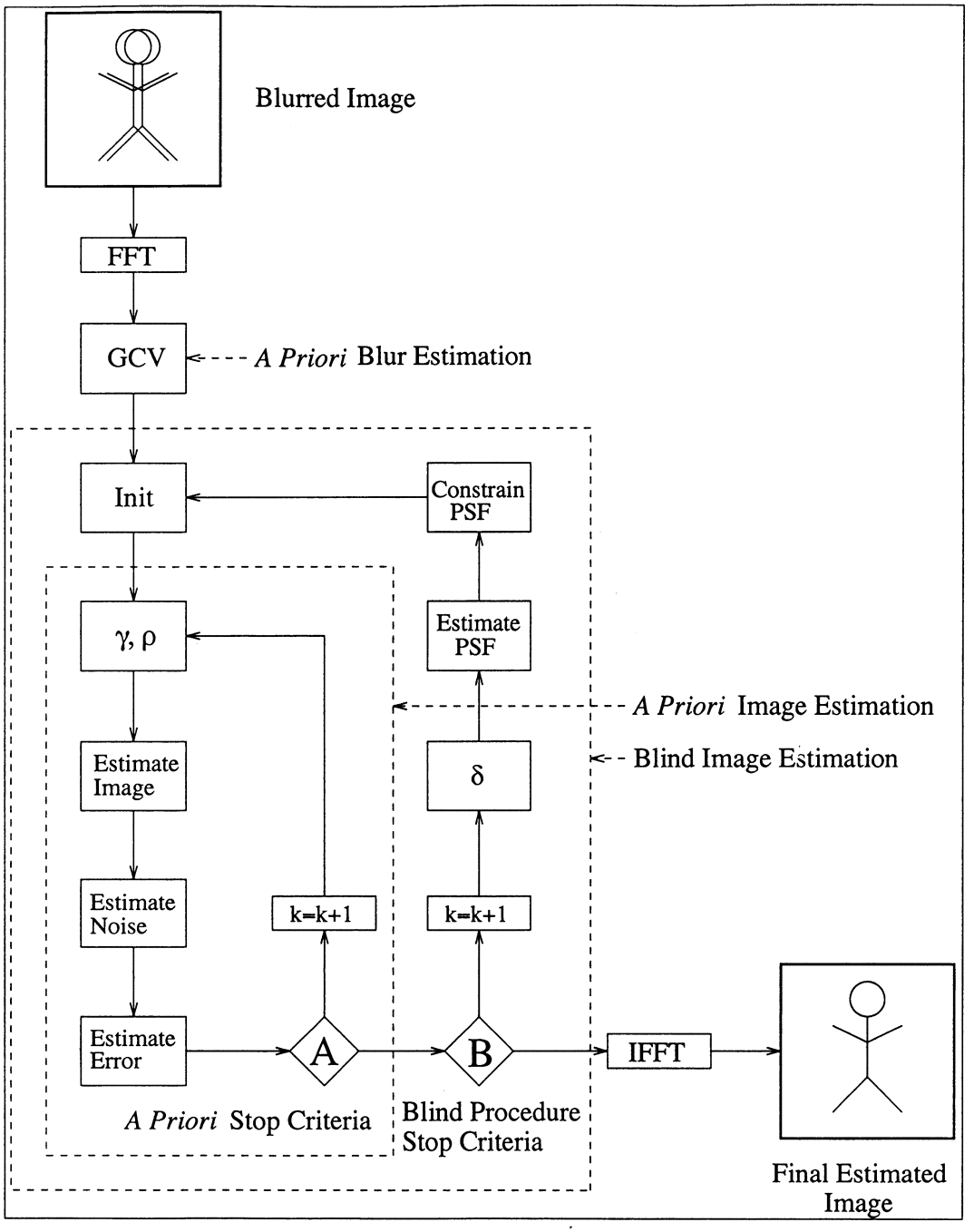


Figure 5.1: Blind Iterative Procedure

estimation is a better conditioned problem. Pursuant to this, the stability operator $\kappa$ will not be used in the PSF estimation process, so that nonlinear constraints such as positivity (2.5) and energy preservation (2.6) can be used as indicated in Figure 5.1. The resulting PSF estimator is then:

$$\hat{\mathcal{H}}_{k+1} = \frac{\hat{\mathcal{S}}_k^* \mathcal{R}}{\left|\hat{\mathcal{S}}_k\right|^2 + \delta_k} \tag{5.10}$$

### 5.3.2 Optimal Regularization Operator for the PSF Estimator

An error expression can be obtained in the same manner as Sections 4.1.1 and 4.2.

$$\mathcal{E}_{\hat{\mathcal{H}}_k} = \frac{\hat{\mathcal{S}}_k^* \mathcal{N} - \hat{\mathcal{S}}_k^* \mathcal{E}_{\hat{\mathcal{S}}_k} \mathcal{H}}{\left|\hat{\mathcal{S}}_k\right|^2 + \delta_k} \tag{5.11}$$

Repeating the same process for $\delta$ as was completed in Section 4.2.2 for $\rho_k$ yields:

$$\frac{\partial \left|\mathcal{E}_{\hat{\mathcal{H}}_k}\right|^2}{\partial \delta_k} = \frac{\mathcal{X}\left(\mathcal{Y} - \delta_k \mathcal{H}\right)^* + \mathcal{X}^*\left(\mathcal{Y} - \delta_k \mathcal{H}\right)}{\left(\left|\hat{\mathcal{S}}_k\right|^2 + \delta_k\right)^3} \tag{5.12}$$

where

$$\begin{aligned} \mathcal{X} &= \mathcal{H}\left(\left|\hat{\mathcal{S}}_k\right|^2 - \hat{\mathcal{S}}_k^* \mathcal{E}_{\hat{\mathcal{S}}_k}\right) + \hat{\mathcal{S}}_k^* \mathcal{N} \\ &= \hat{\mathcal{S}}_k^* \mathcal{R} \end{aligned} \tag{5.13}$$

and

$$\mathcal{Y} = \hat{\mathcal{S}}_k^* \mathcal{N} - \hat{\mathcal{S}}_k^* \mathcal{E}_{\hat{\mathcal{S}}_k} \mathcal{H} \tag{5.14}$$

Hence the value for $\delta_k$ that minimizes $\left|\mathcal{E}_{\hat{\mathcal{H}}_k}\right|^2$ is:

$$\delta_k = \frac{\mathcal{X}\mathcal{Y}^* + \mathcal{X}^*\mathcal{Y}}{\mathcal{H}\mathcal{X}^* + \mathcal{H}^*\mathcal{X}} \tag{5.15}$$

By virtue of the PSF eigenvalues in the denominator term of (5.15), the calculation for $\delta$ at those coefficients where $|\mathcal{H}|$ is small or zero is inherently unstable. Therefore safeguards at these coefficient indices must be adopted.

### 5.3.3 *A Priori* Blur Estimation using GCV

In this section, we describe the use of generalized cross validation as a means of determining an initial guess for the blur operator $\hat{\mathcal{H}}$ as shown in Figure 5.1. This estimate can then be applied to the iterative blind deconvolution process.

Generalized cross validation is a method whereby assumptions about the data can be tested to aid in the image restoration process. The basic idea behind GCV is that one portion of the data is used to obtain an estimate based on a particular assumption. The other portion of the data is then used to validate that assumption. This is sometimes called the "leave one out" method. This idea can be expanded by dividing the data into overlapping sets and imposing the assumption on all sets but one. This process is repeated with another set left out, until all sets have been left out in turn. In this way, all the data are used for both estimation and validation.

This process can be implemented by minimizing a function $V(\boldsymbol{\theta})$. By choosing different minimization parameters, GCV can be used to estimate the blur operator[48], or for determining the best regularization parameter $\lambda$ and regularization operator $\mathbf{C}$[47].

$V(\boldsymbol{\theta})$ using block-circulant assumptions on the blur operator $\mathbf{H}$, can be written as [47]:

$$V(\boldsymbol{\theta}) = N^2 \frac{\sum_{m=1}^{N}\sum_{n=1}^{N}\left[\left[\frac{\lambda|\mathcal{C}(m,n)|^2}{|\mathcal{H}(m,n)|^2+\lambda|\mathcal{C}(m,n)|^2}\right]^2 |\mathcal{R}(m,n)|^2\right]}{\left[\sum_{m=1}^{N}\sum_{n=1}^{N}\frac{\lambda|\mathcal{C}(m,n)|^2}{|\mathcal{H}(m,n)|^2+\lambda|\mathcal{C}(m,n)|^2}\right]^2} \tag{5.16}$$

where $\boldsymbol{\theta} = \begin{bmatrix} \lambda & \boldsymbol{\theta}_o \end{bmatrix}$ and $\boldsymbol{\theta}_o$ is a vector of parameters defining the PSF.

According to [10][32][47][48], the GCV function $V(\boldsymbol{\theta})$ may not be convex for the total range of $\boldsymbol{\theta}$, and hence have local minima. To prevent the possibility of incorrect PSF estimates due to local minima, a grid search can be completed on a target area of the blur parameter space. We assume the PSF can be described as $\mathbf{H}(\theta_1, \theta_2)$ where $(\theta_1, \theta_2) \in \Theta$ and $\Theta$ is the parameter space specifying $\mathbf{H}$. A description of the blur estimation procedure is as follows:

1. For a given $(\theta_1, \theta_2) \in \Theta$, calculate the best $\lambda$ according to the objective function (5.16) by first using a discrete search, then using the results of that search as an initial point for optimization of $V(\boldsymbol{\theta})$. The result of this optimization gives $\lambda_{GCV}$.

2. Using these values for $\lambda$ and $\mathbf{H}(\theta_1, \theta_2)$, use (5.16) directly to calculate $V(\boldsymbol{\theta})$.

3. Repeat 1 and 2 for all possible PSF's in the parameter space.

4. Choose $\hat{\mathcal{H}}_{GCV}$ as the PSF with the smallest value of $V(\boldsymbol{\theta})$ as the image blur operator

For calculating $\lambda_{GCV}$ and $\hat{\mathcal{H}}_{GCV}$, the Laplacian operator is used for $\mathcal{C}$. The corresponding $\lambda_{GCV}$ and $\hat{\mathcal{H}}_{GCV}$ are then used, with the Laplacian operator, to estimate the received image additive noise variance according to [10]:

$$\hat{\sigma}_n^2 = \left(\frac{1}{N^2}\right) \left\| \left(1 - \mathcal{H}\mathcal{A}(\lambda)\right)^{\frac{1}{2}} \mathcal{R} \right\|^2 \tag{5.17}$$

where

$$\mathcal{A}(\lambda) = \frac{\mathcal{H}^*}{|\mathcal{H}|^2 + \lambda |\mathcal{C}|^2}$$

### 5.3.4 *A Priori* Blur Estimation Results

Table 5.1 shows the results of a parametric grid search for the minimum GCV criterion. The "Lena" image was degraded using a seven-by-seven out-of-focus blur, and then further degraded with additive noise at various BSNR levels. This search is conducted over the following specification for $\Theta$:

$$\left.\begin{array}{rcccl} 1 & \leq & \theta_1 & \leq & 13 \\ 3 & \leq & \theta_2 & \leq & 13 \end{array}\right\} \theta_1,\ \theta_2 \ \in \ I_o \tag{5.18}$$

where $\theta_1$ is the maximum vertical dimension of the blur, $\theta_2$ is the maximum horizontal direction and $I_o$ denotes the set of odd integers. The blur is then in the shape of an ellipse with major axis dimensions $\theta_1$ by $\theta_2$. The PSF can be defined with respect to these parameters as:

$$h(i,j) = \begin{cases} \frac{1}{K} & \frac{i^2}{\theta_2^2} + \frac{j^2}{\theta_1^2} \leq \frac{1}{4} \\ 0 & otherwise \end{cases} \tag{5.19}$$

where $K$ is the number of non-zero pixels. As indicated in Table 5.1, this search works well except when the blurred signal-to-noise ratio becomes low. Estimates for the regularization parameter $\lambda$ are also not as accurate at low BSNR. Despite this, noise variance estimates are relatively good. Here, $\lambda_{TM}$ is calculated using (4.34).

Table 5.2 shows similar results searching the same parameter space only with the "Lena" image degraded with a nine pixel linear motion blur (i.e. $\theta_1 = 1$ and $\theta_2 = 9$). These results indicate similar trends as the out-of-focus blur, except estimates for the regularization parameter are underestimated at low BSNR levels. It is important to note that the value for $\lambda$ is only used in our case to estimate the noise variance and so this mis-estimation is not critical since noise variances are estimated reasonably well.

***A Priori* Blur Parameter Estimation**

| BSNR (dB) | $\hat{\theta}_1$ | $\hat{\theta}_2$ | $\lambda_{TM}$ | $\lambda_{GCV}$ | $\sigma_n$ | $\hat{\sigma}_n$ |
|---|---|---|---|---|---|---|
| 5 | 9 | 7 | 4.7811 | 4.9504 | 602.64 | 598.39 |
| 10 | 7 | 7 | 1.5023 | 3.0524 | 190.57 | 188.77 |
| 15 | 7 | 7 | 0.47943 | 0.83861 | 60.264 | 60.042 |
| 20 | 7 | 7 | 0.1498 | 0.20548 | 19.057 | 18.638 |
| 25 | 7 | 7 | 0.047423 | 0.059612 | 6.0264 | 5.8811 |
| 30 | 7 | 7 | 0.014983 | 0.0162 | 1.9057 | 1.8347 |
| 35 | 7 | 7 | 0.004696 | 0.0045861 | 0.60264 | 0.56239 |
| 40 | 7 | 7 | 0.0014998 | 0.0013916 | 0.19057 | 0.17672 |
| 45 | 7 | 7 | 0.00047383 | 0.00043127 | 0.060264 | 0.055024 |
| 50 | 7 | 7 | 0.00015136 | 0.00014102 | 0.019057 | 0.017844 |
| 55 | 7 | 7 | 4.7493e-05 | 4.4345e-05 | 0.0060264 | 0.0056312 |
| 60 | 7 | 7 | 1.5053e-05 | 1.425e-05 | 0.0019057 | 0.0018069 |

Table 5.1: PSF and noise variance estimation, 7 by 7 uniform out-of-focus blur.

***A Priori* Blur Parameter Estimation**

| BSNR (dB) | $\hat{\theta}_1$ | $\hat{\theta}_2$ | $\lambda_{TM}$ | $\lambda_{GCV}$ | $\sigma_n$ | $\hat{\sigma}_n$ |
|---|---|---|---|---|---|---|
| 5 | 1 | 11 | 4.5726 | 3.333 | 576.35 | 568.52 |
| 10 | 1 | 9 | 1.4368 | 1.0806 | 182.26 | 174.78 |
| 15 | 1 | 9 | 0.45852 | 0.40316 | 57.635 | 56.225 |
| 20 | 1 | 9 | 0.14327 | 0.15235 | 18.226 | 18.195 |
| 25 | 1 | 9 | 0.045354 | 0.053847 | 5.7635 | 5.972 |
| 30 | 1 | 9 | 0.014329 | 0.016124 | 1.8226 | 1.8773 |
| 35 | 1 | 9 | 0.0044912 | 0.0045961 | 0.57635 | 0.56282 |
| 40 | 1 | 9 | 0.0014344 | 0.001401 | 0.18226 | 0.17588 |
| 45 | 1 | 9 | 0.00045316 | 0.00043298 | 0.057635 | 0.054568 |
| 50 | 1 | 9 | 0.00014475 | 0.00013618 | 0.018226 | 0.017253 |
| 55 | 1 | 9 | 4.5422e-05 | 4.3319e-05 | 0.0057635 | 0.0054836 |
| 60 | 1 | 9 | 1.4396e-05 | 1.3344e-05 | 0.0018226 | 0.0016897 |

Table 5.2: PSF and noise variance estimation, 9 pixel linear motion blur.

## 5.4 Proposed Blind Iterative Algorithm

We now describe in more detail the proposed blind deconvolution algorithm of Figure 5.1.

1. ***A Priori* Blur Estimation**

   The purpose of this procedure is to find parametric PSF model parameters that best suit the degraded image data based on the GCV criterion (5.16) (see associated box in Figure 5.1).

   - Use method outlined in Section 5.3.3
   - Let parameter vector $\boldsymbol{\theta} = [\theta_1 \theta_2]^T$ be the blur vertical and horizontal dimensions respectively, and:

$$h(i,j) = \begin{cases} \frac{1}{K} & \frac{i^2}{\theta_2^2} + \frac{j^2}{\theta_1^2} \leq \frac{1}{4} \\ 0 & otherwise \end{cases}$$

   - Estimate noise variance $\sigma_n^2$ using (5.17).
   - Set area of support for PSF based on this model
   - Use the DFT of $h(i,j)$ as the initial PSF estimate $\tilde{\mathcal{H}}_0$ for the blind iterative process.

2. ***A Priori* Image Estimation**

   This part of the algorithm uses the current PSF estimate to evaluate an image estimate $\hat{\mathcal{H}}_k$ using the nonblind process described in Chapter 4. The algorithm iterates within the "*a priori* image estimation" block in Figure 5.1 until convergence is met.

   (a) Initialize key variables

   - $\rho_0 = c\left(1 - \left|\tilde{\mathcal{H}}_0\right|^2\right) \quad 0 < c \leq 1$

- $\hat{\mathcal{S}}_0 = \dfrac{\tilde{\mathcal{H}}_0^* \mathcal{R}}{\left|\tilde{\mathcal{H}}_0\right|^2 + \rho_0}$
- $\angle\left\{\tilde{\mathcal{N}}\right\} = \angle\left\{\mathcal{R} - \tilde{\mathcal{H}}_0 \hat{\mathcal{S}}_0\right\}$
- $\left|\tilde{\mathcal{N}}\right| = N^2 \sigma_n^2$
- $\hat{\mathcal{E}}_{\hat{\mathcal{S}}_0} = \dfrac{\tilde{\mathcal{H}}_0^* \tilde{\mathcal{N}} - \rho_0 \hat{\mathcal{S}}_0}{\left|\tilde{\mathcal{H}}_0\right|^2 + \rho_0}$
- $\gamma_0 = \mathbf{0}$, $\Sigma_{\mathcal{N}}^{(0)} = \frac{\tilde{\mathcal{H}}_0^*}{\mathcal{Z}_0}$, $\Sigma_{\rho}^{(0)} = \frac{\rho_0}{\mathcal{Z}_0}$ and $\Sigma_{\mathcal{H}}^{(0)} = \mathbf{0}$

(b) Increment iteration number $k$

(c) Calculate $\gamma_k$ and $\rho_k$ using (4.23) and (4.27).

(d) Use $\hat{\mathcal{H}}_k$, $\rho_k$ and $\gamma_k$ in equation (4.16) to estimate $\hat{\mathcal{S}}_k$

(e) Estimate noise

- $\angle\left\{\tilde{\mathcal{N}}\right\} = \angle\left\{\mathcal{R} - \tilde{\mathcal{H}}_0 \hat{\mathcal{S}}_0\right\}$
- $\left|\tilde{\mathcal{N}}\right| = N^2 \sigma_n^2$

(f) Calculate $\hat{\mathcal{E}}_{\hat{\mathcal{S}}_k}$ using (4.19).

(g) If *a priori* estimation stop criteria "A" are not met, go to 2(b). Otherwise, go to 3.

3. **Check blind procedure stop criteria**

   Here, we determine whether a new iteration for the blur operator $\hat{\mathcal{H}}$ is required using the most recent estimate of $\mathcal{S}$. If overall blind image estimation stop criteria "B" are not met, go to 4. Otherwise, go to 5.

4. **PSF Estimation**

   This process is part of the "blind image estimation" block of Figure 5.1. Within this part of the procedure, nonlinear constraints as discussed in Sections 2.2 and 5.3.1 are applied. Specifically, we apply a positivity constraint (2.5), an energy

preservation constraint (2.6), an area of support constraint, and a constraint such that the PSF pixel values are non-increasing from the origin.

(a) Estimate $\delta$ using (5.15)

(b) Estimate $\hat{\mathcal{H}}_k$ using (5.10)

(c) Use IDFT of $\hat{\mathcal{H}}_k$ and constrain in spatial domain

- Symmetry constraint
- $h(i,j) = \begin{cases} h(i,j) & h(i,j) \in S_h \\ 0 & otherwise \end{cases}$
  where $S_h$ is the area of support for $h(i,j)$.
- $\sum_{i,j} \tilde{h}_k(i,j) = 1$
- $\tilde{h}_k(i,j) \geq \tilde{h}_k(m,n) \quad \forall \quad |i| < |m|, |j| < |n|$

(d) Use DFT of constrained $\tilde{h}_k(i,j) \Rightarrow \tilde{\mathcal{H}}_k$ as next PSF estimate.

(e) Reduce *a priori* image estimation stop criteria "A". Go to 2(a)

5. **Terminate algorithm**

The final estimated image is the IDFT of the most recent $\hat{\mathcal{S}}_k$.

### 5.4.1 Blind Deconvolution Results

Figures 5.2 and 5.3 show the respective convergence comparison results and pictorial results of applying the blind iterative algorithm presented in Section 5.4 to the "Lena" image degraded by the symmetrical PSF with pixel values shown in the top half of Table 5.3 and additive white noise such that the BSNR is 30 dB. The final PSF, used for the last *a priori* image estimation cycle is shown in the bottom half of Table 5.3.

The final quality of the blindly restored images are superior to the benchmark restored images, both qualitatively and quantitatively. Not only is the final ISNR level of the blindly restored image much higher than that of the iterative Tikhonov-Miller

| Original PSF | | | | | | |
|---|---|---|---|---|---|---|
| 0 | 0 | 0.0259 | 0.0262 | 0.0259 | 0 | 0 |
| 0 | 0.0264 | 0.0272 | 0.0275 | 0.0272 | 0.0264 | 0 |
| 0.0259 | 0.0272 | 0.0281 | 0.0284 | 0.0281 | 0.0272 | 0.0259 |
| 0.0262 | 0.0275 | 0.0284 | 0.0287 | 0.0284 | 0.0275 | 0.0262 |
| 0.0259 | 0.0272 | 0.0281 | 0.0284 | 0.0281 | 0.0272 | 0.0259 |
| 0 | 0.0264 | 0.0272 | 0.0275 | 0.0272 | 0.0264 | 0 |
| 0 | 0 | 0.0259 | 0.0262 | 0.0259 | 0 | 0 |
| **Estimated PSF** | | | | | | |
| 0 | 0 | 0.0262 | 0.0266 | 0.0262 | 0 | 0 |
| 0 | 0.0263 | 0.0270 | 0.0274 | 0.0270 | 0.0263 | 0 |
| 0.0261 | 0.0271 | 0.0279 | 0.0283 | 0.0279 | 0.0271 | 0.0261 |
| 0.0264 | 0.0276 | 0.0284 | 0.0284 | 0.0284 | 0.0276 | 0.0264 |
| 0.0261 | 0.0271 | 0.0279 | 0.0283 | 0.0279 | 0.0271 | 0.0261 |
| 0 | 0.0263 | 0.0270 | 0.0274 | 0.0270 | 0.0263 | 0 |
| 0 | 0 | 0.0262 | 0.0266 | 0.0262 | 0 | 0 |

Table 5.3: Results for Blind Procedure: Original and Estimated PSF

method, it also does not exhibit the same ringing artifacts that are characteristic of many regularized least squares methods. Clearly, the blind algorithm has outperformed the nonblind iterative Tikhonov Miller algorithm. It is important to mention that for the benchmark run of the iterative Tikhonov-Miller method, not only were ideal values for the regularization parameter used (see Section 4.5), but the actual nonblind degrading PSF was also used and not an estimate.

As an indication of how the blind procedure compares to its nonblind counterpart, results using the known PSF for one cycle of the *a priori* algorithm results in an ISNR of 5.17 dB after 125 iterations compared to the blind results of 5.091 dB in 310 iterations.

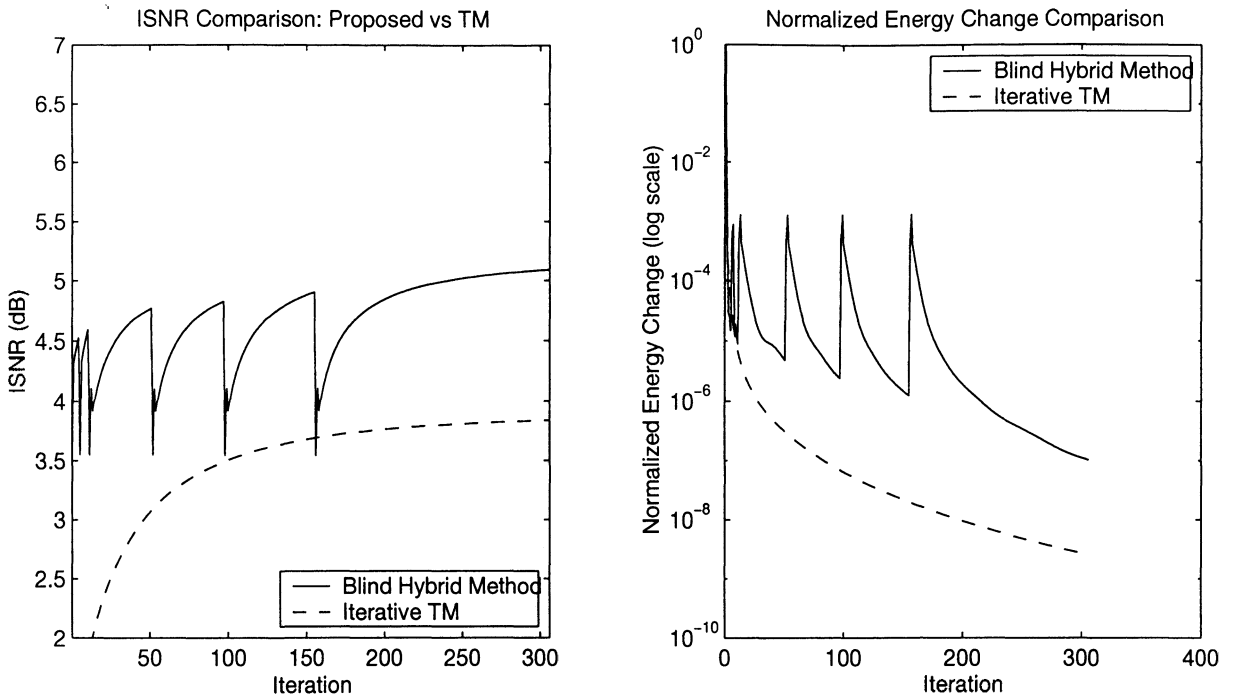


Figure 5.2: ISNR and Convergence Comparison (Lena) BSNR = 30 dB

## 5.5 Discussion

In this chapter, a blind iterative algorithm based on the material developed in Chapter 4 is presented. The accelerated convergence properties of the nonblind algorithm of Chapter 4 benefit the blind process by allowing multiple iterations of the image estimation part of the algorithm between each PSF estimation cycle. Each cycle of the nonblind procedure is apparent in the "sawtooth" pattern of the convergence comparison graphs of Figure 5.2. Convergence is achieved by lowering the termination criteria at each image estimation cycle.

The convergence properties of this algorithm require further study however. The error in the PSF (see equations (4.7) and (4.9)) propagates with each iteration, and will eventually cause divergence if left unchecked. This effect was prevented by setting a limit to the number of iterations within the "*a priori* image estimation" box in

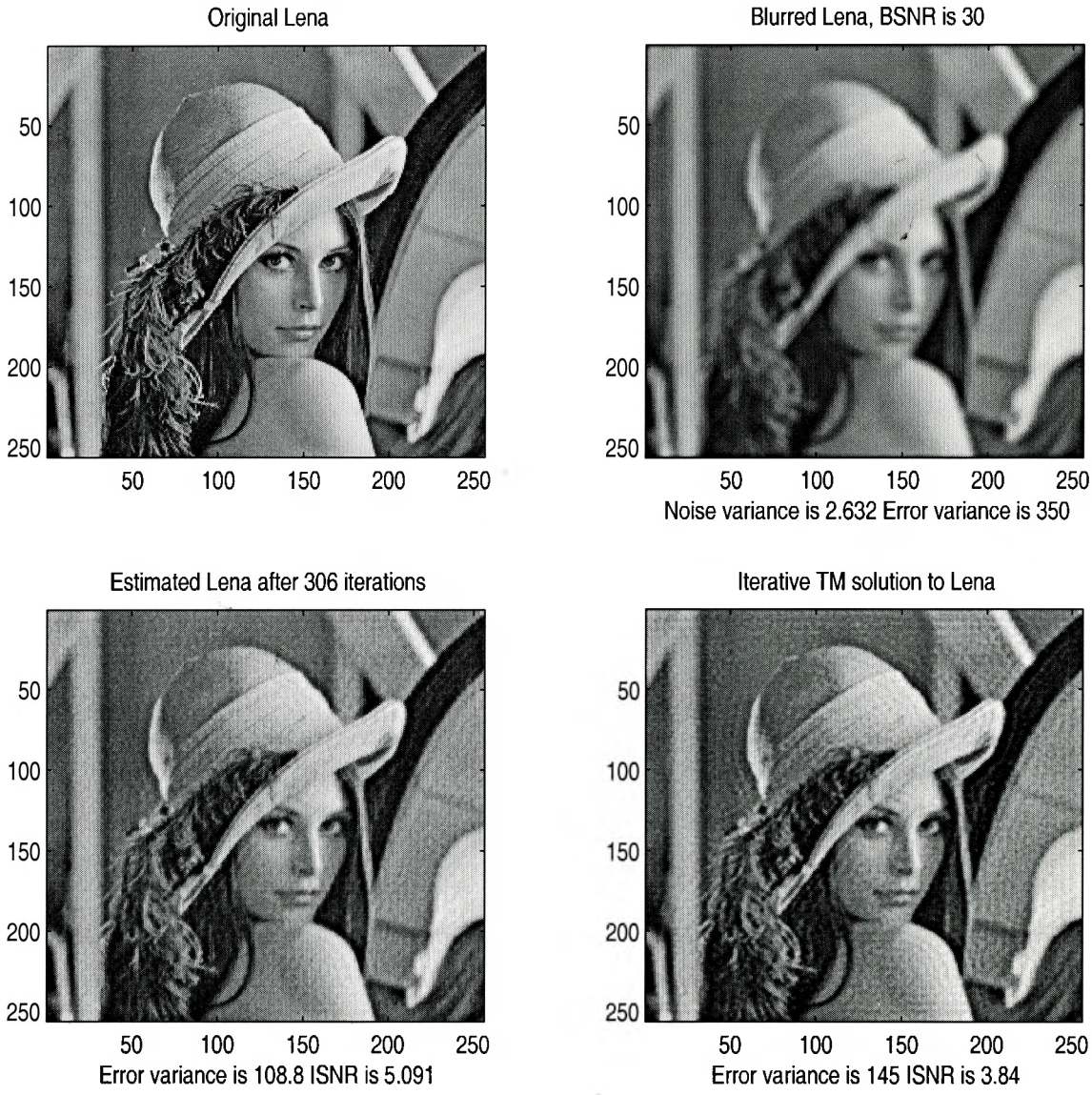


Figure 5.3: Blind Hybrid vs Iterative TM - Pictoral Results (Lena) BSNR = 30 dB

Figure 5.1 as well as a limit to the number of cycles this part of the algorithm was allowed to run. Relaxation techniques were also applied to each new image and PSF estimate.

An *a priori* blur estimation method using GCV is also presented. By using a highly parameterized model, an estimate for the PSF is determined for initialization. At this time an area of support for the PSF is determined, as well as an estimate for the additive noise variance.

# Chapter 6

# Conclusions

A new approach to iterative image deconvolution is proposed that highlights the nature of additive noise propagation, and its effect on a final image estimate. A new approach to the error analysis for this kind of iterative algorithm is developed, with error propagation terms $\Sigma_{\mathcal{N}}^{(k)}$, $\Sigma_{\rho}^{(k)}$ and $\Sigma_{\mathcal{H}}^{(k)}$ defined that "encapsulate" the past history of the iterative process with deference to noise, regularization, and PSF estimation errors respectively. Since these terms are independent of the original image and noise, they do not have cumulative error properties and therefore offer better transient error estimates as the iterative procedure progresses.

From this novel approach to error analysis, the idea of a stabilizing operator $\gamma$ to augment the regularizing operator $\rho$ is explored. Further, it is shown that the asymptotic error as the number of iterations $k \to \infty$ is independent of $\gamma$, converging to the regularized least squares solution. The additional degree of freedom offered by $\gamma$ manifests itself in a dramatically increased convergence rate.

Optimal stability and regularization operators are derived that change adaptively based on a current transient error estimate. Each DFT coefficient is optimized independently such that non-parametric, frequency adaptive regularization and stability operators result.

From the experimental results presented in Chapter 4, is is readily seen that the image estimation given by (4.16) outperforms the iterative Tikhonov-Miller method. The performance gains with this new algorithm are not only in the restored image accuracy as measured by ISNR, but restorations are qualitatively superior with less "ringing" when compared to iterative TM. These superior attributes are also achieved with a markedly faster convergence rate. It is also noted here as it was in Chapter 4 that the benchmark given by the iterative Tikhonov-Miller algorithm is conducted under *ideal* conditions, that is to say with regularization parameter $\lambda_{TM}$ set to a normally unknown value.

The *a priori* blur estimation method outlined in Section 5.3.3 works well until the blurred signal-to-noise ratio (BSNR) goes below 5 dB. Results shown in Chapter 5 are initialized using this method.

The blind deconvolution results shown in Chapter 5 show that the ideas of Chapter 4 can indeed be extended to the blind case with results that surpass the nonblind iterative TM method both qualitatively and quantitatively. Not only does this method outperform many existing methods, the results in Section 5.4.1 show that the final blind ISNR values are very close to those using the nonblind algorithm of Chapter 4 with a known PSF. The convergence properties of this blind method require more study, as the results obtained were the result of *ad hoc* convergence criteria.

Future research with respect to the new ideas presented herein can be divided into three parts:

1. Develop a procedure for the joint estimation of the stability and regularization operators $\gamma$ and $\rho$.

2. Development of a stable blind iterative algorithm that is not as dependent on initialization. Like many other blind methods such as MLE or GCV, the success of the algorithm presented in Chapter 5 is highly dependent on a good

initialization for the PSF. Making the estimation process more robust to initial PSF error is essential for application at practical levels.

3. Application of these ideas to multichannel restoration using wavelet theory. Many of the current multichannel restoration methods make use of known algorithms to be applied to each subband or channel with remarkable results. In a similar way, it may be possible to increase the performance of this estimation method by applying it to a multichannel framework. The claim of many researchers working in this area is that the multichannel approach models the nonstationarity of images to some degree. It is likely that this property, along with the frequency adaptive properties of our new technique, would result in further improvements.

In summary, the ideas on *a priori* or "nonblind" image restoration presented in this thesis add some new insights to a fairly mature field of research. When one considers the potential of extending these ideas to blind applications, the inherent value is far more substantial. This extension is indeed made in this work, thus fulfilling some of the potential these new ideas possess.

# Appendix A

# Block-Toeplitz Structure and Eigenvalue Analysis for 2-D Systems

In this appendix, the analysis details for the eigenvalues of the blurring matrix $\mathbf{H}$ are developed, including the lexicographic ordering of matrices, and the differences between block Toeplitz and block-circulant matrices.

## A.1 Lexicographic Ordering

Lexicographic ordering is a mapping of a two-dimensional (image) array into a one-dimensional vector. Essentially, each row of the image is transposed and stacked one atop of each other to form a vector. The received $M \times N$ image $\mathbf{r}$ for example would be:

$$\begin{aligned}\mathbf{r} = [\ & r(0,0), r(0,1), r(0,2), \ldots r(0,N-1), \\ & r(1,0), r(1,1), r(1,2), \ldots r(1,N-1), \ldots \\ & r(M-1,0), r(M-1,1) r(M-1,2) \ldots r(M-1,N-1)\ ]^T\end{aligned} \tag{A.1}$$

## A.2 Block Toeplitz Matrices

As outlined in Section 2.3, the linear two-dimensional convolution operation of (2.3),

$$r(i,j) = \sum_{k=1}^{M} \sum_{l=1}^{N} h(i-k, j-l) s(k,l) + n(i,j)$$

can be rewritten in matrix-vector form by lexicographically ordering the received and underlying images $r(i,j)$ and $s(i,j)$, and using an $MN \times MN$ convolution operator matrix $\mathbf{H}$ as in (2.11).

$$\mathbf{r} = \mathbf{H}\mathbf{s} + \mathbf{n}$$

Assuming the PSF $h(i,j)$ has a support area of $2L+1$ by $2L+1$ pixels with integer indexing from $-L$ to $L$ in each direction and $2L+1 \leq M$ or $N$, the convolution or blurring operator $\mathbf{H}$ has the structure:

$$\mathbf{H} = \begin{bmatrix} \mathbf{H}_0 & \mathbf{H}_{-1} & \mathbf{H}_{-2} & & \mathbf{H}_{-L} & & & & \\ \mathbf{H}_1 & \mathbf{H}_0 & \mathbf{H}_{-1} & \cdots & \mathbf{H}_{-L+1} & \mathbf{H}_{-L} & & \mathbf{0} & \\ \mathbf{H}_2 & \mathbf{H}_1 & \mathbf{H}_0 & & \mathbf{H}_{-L+2} & \mathbf{H}_{-L+1} & \mathbf{H}_{-L} & & \\ & \vdots & & & & & & \ddots & \\ \mathbf{H}_L & \mathbf{H}_{L-1} & \mathbf{H}_{L-2} & & & & & & \\ & \mathbf{H}_L & \mathbf{H}_{L-1} & \cdots & & & & & \\ & & \mathbf{H}_L & & & & & & \\ & & & \ddots & & & & & \\ & \mathbf{0} & & & & & & & \vdots \\ & & & & & & & \cdots & \mathbf{H}_0 \end{bmatrix} \tag{A.2}$$

where there are $M^2$ sub-matrices $\mathbf{H}_i$ which are of dimension $N \times N$ with the structure:

$$\mathbf{H}_i = \begin{bmatrix} h_{i,0} & h_{i,-1} & h_{i,-2} & & h_{i,-L} & & & & \\ h_{i,1} & h_{i,0} & h_{i,-1} & \cdots & h_{i,-L+1} & h_{i,-L} & & 0 & \\ h_{i,2} & h_{i,1} & h_{i,0} & & h_{i,-L+2} & h_{i,-L+1} & h_{i,-L} & & \\ & \vdots & & & & & & \ddots & \\ h_{i,L} & h_{i,L-1} & h_{i,L-2} & & & & & & \\ & h_{i,L} & h_{i,L-1} & \cdots & & & & & \\ & & h_{i,L} & & & & & & \\ & & & \ddots & & & & & \\ & 0 & & & & & & & \vdots \\ & & & & & & & \cdots & h_{i,0} \end{bmatrix} \tag{A.3}$$

where the PSF indices are indicated by subscripts, i.e. PSF element $h(i,j) = h_{i,j}$.

The eigenvectors and eigenvalues for large matrices like $\mathbf{H}$ are in general quite difficult to find. On the other hand, block-circulant matrices, which are similarly structured to block-Toeplitz matrices, are known to have discrete Fourier transform coefficients as eigenvalues, with normalized DFT bases as eigenvectors. Making this assumption, we can then revise the structure of $\mathbf{H}$ such that:

$$
\mathbf{H} = \begin{bmatrix}
\mathbf{H}_0 & \mathbf{H}_{-1} & \mathbf{H}_{-2} & & \mathbf{H}_{-L} & \mathbf{0} & \mathbf{0} & & \mathbf{H}_1 \\
\mathbf{H}_1 & \mathbf{H}_0 & \mathbf{H}_{-1} & \cdots & \mathbf{H}_{-L+1} & \mathbf{H}_{-L} & 0 & \cdots & \mathbf{H}_2 \\
\mathbf{H}_2 & \mathbf{H}_1 & \mathbf{H}_0 & & \mathbf{H}_{-L+2} & \mathbf{H}_{-L+1} & \mathbf{H}_{-L} & & \mathbf{H}_3 \\
 & \vdots & & & & \vdots & & \ddots & \vdots \\
\mathbf{H}_L & \mathbf{H}_{L-1} & \mathbf{H}_{L-2} & & & & & & \\
\mathbf{0} & \mathbf{H}_L & \mathbf{H}_{L-1} & \cdots & & & & & \\
\mathbf{0} & \mathbf{0} & \mathbf{H}_L & & & & & & \\
 & \vdots & & \ddots & & & & & \\
\mathbf{H}_{-L} & \mathbf{0} & \mathbf{0} & & & & & & \\
\mathbf{H}_{-L+1} & \mathbf{H}_{-L} & \mathbf{0} & & & & & & \\
\mathbf{H}_{-L+2} & \mathbf{H}_{-L+1} & \mathbf{H}_{-L} & & & & & & \\
 & \vdots & & \ddots & & & & & \vdots \\
\mathbf{H}_{-1} & \mathbf{H}_{-2} & \mathbf{H}_{-3} & \cdots & & & & \cdots & \mathbf{H}_0
\end{bmatrix} \tag{A.4}
$$

where each $\mathbf{H}_i =$

$$\begin{bmatrix}
h_{i,0} & h_{i,-1} & h_{i,-2} & & h_{i,-L} & 0 & 0 & & h_{i,1} \\
h_{i,1} & h_{i,0} & h_{i,-1} & \cdots & 0 & h_{i,L} & h_{i,L-1} & \cdots & h_{i,2} \\
h_{i,2} & h_{i,1} & h_{i,0} & & h_{i,-L+2} & h_{i,-L+1} & h_{i,-L} & & h_{i,3} \\
 & \vdots & & & & \vdots & & \ddots & \vdots \\
h_{i,L} & h_{i,L-1} & h_{i,L-2} & & & & & & \\
0 & h_{i,L} & h_{i,L-1} & \cdots & & & & & \\
0 & 0 & h_{i,L} & & & & & & \\
 & \vdots & & \ddots & & & & & \\
h_{i,-L} & 0 & 0 & & & & & & \\
h_{i,-L+1} & h_{i,-L} & 0 & & & & & & \\
h_{i,-L+2} & h_{i,-L+1} & h_{i,-L} & & & & & & \\
 & \vdots & & \ddots & & & & & \vdots \\
h_{i,-1} & h_{i,-2} & h_{i,-3} & \cdots & & & & \cdots & h_{i,0}
\end{bmatrix} \tag{A.5}$$

The block-circulant approximation here is equivalent to replacing a linear convolution with a circulant one. The errors introduced by the block-circulant approximation are minor, and affect mainly the image edges. This effect can be eliminated by zero padding the image (and correspondingly the PSF), making the linear and circular convolutions equivalent.

## A.3 The Eigensystem for the Blurring Matrix H

Eigenanalysis is a powerful tool for both the analysis and implementation of linear systems of equations. As mentioned in Section 2.4, the block-circulant matrix $\mathbf{H}$ can be diagonalized using its eigendecomposition:

$$\mathbf{H} = \mathbf{W}\mathbf{D}\mathbf{W}^H \tag{A.6}$$

For the following discussions, matrix element indices will have a double subscript to reflect the fact that we are dealing with lexicographically ordered images. Hence an element of the blurring matrix $\mathbf{H}$ can be identified by the notation:

$$X_{uv,mn} \quad \begin{cases} 0 \leq u < M \\ 0 \leq v < N \\ 0 \leq m < M \\ 0 \leq n < N \end{cases}$$

where the row index "$uv$" identifies the $v$th row of the $u$th sub-block in the vertical direction and the column index and "$mn$" identifies the $n$th column of the $m$th sub-block in the horizontal direction[1].

It is well known that the eigenvectors of a block-circulant matrix like $\mathbf{H}$ are complex-valued rotating exponentials:

$$W_{uv,mn} = \frac{1}{\sqrt{MN}} \exp\left(j2\pi\left(\frac{um}{M} + \frac{vn}{N}\right)\right) \tag{A.7}$$

where the term $(MN)^{-\frac{1}{2}}$ is to make each column of $\mathbf{W}$ unit norm. Due to this orthonormal relationship between the eigenvectors of $\mathbf{H}$, we can say that $\mathbf{W}$ is *unitary* and therefore:

$$\mathbf{W^H W = I}$$

hence

$$\mathbf{W^{-1} = W^H}$$

From this discussion we see that the DFT of the point spread function $h(m,n)$ is then:

$$\mathcal{H}(u,v) = \sum_{m=0}^{M-1}\sum_{n=0}^{N-1} h(m,n) \exp\left(-j2\pi\left(\frac{um}{M} + \frac{vn}{N}\right)\right)$$

[1]Recall that $\mathbf{H}$ is made up of $M^2$ $N \times N$ sub-matrices.

$$= \sqrt{MN}\ \mathbf{w}_{mn}^{H}\mathbf{h} \tag{A.8}$$

where $\mathbf{h}$ is the lexicographically ordered PSF $h(m,n)$ and $\mathbf{w}_{mn}$ is the $n$th column of the $m$th horizontal sub-block of the matrix $\mathbf{W}$.

# Appendix B

# Error Terms for the Hybrid Iterative Solution

## B.1 Error Term Expansion

The error term for the hybrid adaptive iterative image estimator (4.16) can be expressed as:

$$\mathcal{E}_{\hat{\mathcal{S}}_k} = \frac{\hat{\mathcal{H}}_k^* \mathcal{N} - \hat{\mathcal{H}}_k^* \mathcal{E}_{\hat{\mathcal{H}}_k} \mathcal{S} - \rho_k \mathcal{S} + \gamma_k \mathcal{E}_{\hat{\mathcal{S}}_{k-1}}}{\left|\hat{\mathcal{H}}_k\right|^2 + \gamma_k + \rho_k} \tag{B.1}$$

If we define the denominator term

$$\mathcal{Z}_i = \left|\hat{\mathcal{H}}_i\right|^2 + \gamma_i + \rho_i$$

then (B.1) can be expanded in the following way:

$$\mathcal{E}_{\hat{\mathcal{S}}_k} = \frac{\hat{\mathcal{H}}_k^* \mathcal{N} - \rho_k \mathcal{S} - \hat{\mathcal{H}}_k^* \mathcal{E}_{\hat{\mathcal{H}}_k} + \gamma_k \left[ \dfrac{\hat{\mathcal{H}}_{k-1}^* \mathcal{N} - \rho_{k-1} \mathcal{S} - \hat{\mathcal{H}}_{k-1}^* \mathcal{E}_{\hat{\mathcal{H}}_{k-1}} + \gamma_{k-1} \left[ \dfrac{\cdots}{\mathcal{Z}_{k-2}} \right]}{\mathcal{Z}_{k-1}} \right]}{\mathcal{Z}_k} \tag{B.2}$$

Combining similar terms, this expansion can be expressed in relation to the components of error attributed to additive noise, regularization, and the error in the PSF estimate respectively.

$$\begin{aligned}\mathcal{E}_{\hat{\mathcal{S}}_k} &= \mathcal{N}\left[\frac{\hat{\mathcal{H}}_k^*}{\mathcal{Z}_k}+\frac{\gamma_k\hat{\mathcal{H}}_{k-1}^*}{\mathcal{Z}_k\mathcal{Z}_{k-1}}+\frac{\gamma_k\gamma_{k-1}\hat{\mathcal{H}}_{k-2}^*}{\mathcal{Z}_k\mathcal{Z}_{k-1}\mathcal{Z}_{k-2}}+\cdots+\frac{\gamma_k\gamma_{k-1}\cdots\gamma_1\hat{\mathcal{H}}_0^*}{\mathcal{Z}_k\mathcal{Z}_{k-1}\cdots\mathcal{Z}_0}\right] \\ &\quad -\mathcal{S}\left[\frac{\rho_k}{\mathcal{Z}_k}+\frac{\gamma_k\rho_{k-1}}{\mathcal{Z}_k\mathcal{Z}_{k-1}}+\frac{\gamma_k\gamma_{k-1}\rho_{k-2}}{\mathcal{Z}_k\mathcal{Z}_{k-1}\mathcal{Z}_{k-2}}+\cdots+\frac{\gamma_k\gamma_{k-1}\cdots\gamma_1\rho_0}{\mathcal{Z}_k\mathcal{Z}_{k-1}\cdots\mathcal{Z}_0}\right] \\ &\quad -\mathcal{S}\left[\frac{\hat{\mathcal{H}}_k^*\mathcal{E}_{\hat{\mathcal{H}}_k}}{\mathcal{Z}_k}+\frac{\gamma_k\hat{\mathcal{H}}_{k-1}^*\mathcal{E}_{\hat{\mathcal{H}}_{k-1}}}{\mathcal{Z}_k\mathcal{Z}_{k-1}}+\frac{\gamma_k\gamma_{k-1}\hat{\mathcal{H}}_{k-2}^*\mathcal{E}_{\hat{\mathcal{H}}_{k-2}}}{\mathcal{Z}_k\mathcal{Z}_{k-1}\mathcal{Z}_{k-2}}+\cdots\right. \\ &\qquad \left.+\frac{\gamma_k\gamma_{k-1}\cdots\gamma_1\hat{\mathcal{H}}_0^*\mathcal{E}_{\hat{\mathcal{H}}_0}}{\mathcal{Z}_k\mathcal{Z}_{k-1}\cdots\mathcal{Z}_0}\right] \end{aligned} \tag{B.3}$$

We then return to the same expression presented in Section 4.1.1 as equation (4.7):

$$\mathcal{E}_{\hat{\mathcal{S}}_k} = \mathcal{N}\Sigma_{\mathcal{N}}^{(k)} - \mathcal{S}\Sigma_{\rho}^{(k)} - \mathcal{S}\Sigma_{\mathcal{H}}^{(k)}$$

where

$$\Sigma_{\mathcal{N}}^{(k)} = \frac{\mathcal{H}_k^*}{\mathcal{Z}_k} + \sum_{i=1}^{k}\left[\prod_{j=i}^{k}\frac{\gamma_j}{\mathcal{Z}_j}\right]\frac{\mathcal{H}_{i-1}^*}{\mathcal{Z}_{i-1}}$$

$$\Sigma_{\mathcal{H}}^{(k)} = \frac{\hat{\mathcal{H}}_k^*\mathcal{E}_{\hat{\mathcal{H}}_k}}{\mathcal{Z}_k} + \sum_{i=1}^{k}\left[\prod_{j=i}^{k}\frac{\gamma_j}{\mathcal{Z}_j}\right]\frac{\hat{\mathcal{H}}_{i-1}^*\mathcal{E}_{\hat{\mathcal{H}}_{i-1}}}{\mathcal{Z}_{i-1}}$$

and

$$\Sigma_{\rho}^{(k)} = \frac{\rho_k}{\mathcal{Z}_k} + \sum_{i=1}^{k}\left[\prod_{j=i}^{k}\frac{\gamma_j}{\mathcal{Z}_j}\right]\frac{\rho_{i-1}}{\mathcal{Z}_{i-1}}$$

## B.2 Alternate Error Formulation

In the case where it is advantageous to "reset" all error propagation terms in the iterative blind deconvolution process, the error formulation (4.7) can be re-written to

explicitly include the initial image estimate $\hat{\mathcal{S}}_0$. Note that this development assumes that there is no previous estimate to $\hat{\mathcal{S}}_0$, therefore the stability operator $\gamma_0$ is initialized to zeros and therefore:

$$\mathcal{E}_{\hat{\mathcal{S}}_0} = \frac{\hat{\mathcal{H}}^*\mathcal{N} - \rho_0\mathcal{S} - \hat{\mathcal{H}}_0^*\mathcal{E}_{\hat{\mathcal{H}}_0}\mathcal{S}}{\left|\hat{\mathcal{H}}_0\right|^2 + \rho_0} \tag{B.4}$$

Each numerator term for (B.4) can be removed from the original error estimate (4.7), thus adding another multiplicative term $\Pi_\gamma^{(k)}$ that embodies the error propagation of the initial estimate from one iteration to the next.

$$\mathcal{E}_{\hat{\mathcal{S}}_k} = \mathcal{N}\Sigma_{\mathcal{N}}^{(k)} - \mathcal{S}\Sigma_{\mathcal{H}}^{(k)} - \mathcal{S}\Sigma_{\rho}^{(k)} + \Pi_\gamma^{(k)}\mathcal{E}_{\hat{\mathcal{S}}_0} \tag{B.5}$$

where

$$\begin{aligned}
\Sigma_{\mathcal{N}}^{(k)} &= \frac{\mathcal{H}_k^*}{\mathcal{Z}_k} + \sum_{i=2}^{k}\left[\prod_{j=i}^{k}\frac{\gamma_j}{\mathcal{Z}_j}\right]\frac{\mathcal{H}_{i-1}^*}{\mathcal{Z}_{i-1}} \\
&= \frac{\hat{\mathcal{H}}_k^* + \gamma_k\Sigma_{\mathcal{N}}^{(k-1)}}{\mathcal{Z}_k} \qquad \text{(B.6)} \\
\Sigma_{\mathcal{H}}^{(k)} &= \frac{\hat{\mathcal{H}}_k^*\mathcal{E}_{\hat{\mathcal{H}}_k}}{\mathcal{Z}_k} + \sum_{i=2}^{k}\left[\prod_{j=i}^{k}\frac{\gamma_j}{\mathcal{Z}_j}\right]\frac{\hat{\mathcal{H}}_{i-1}^*\mathcal{E}_{\hat{\mathcal{H}}_{i-1}}}{\mathcal{Z}_{i-1}} \\
&= \frac{\hat{\mathcal{H}}_k^*\mathcal{E}_{\hat{\mathcal{H}}_k} + \gamma_k\Sigma_{\mathcal{H}}^{(k-1)}}{\mathcal{Z}_k} \qquad \text{(B.7)} \\
\Sigma_{\rho}^{(k)} &= \frac{\rho_k}{\mathcal{Z}_k} + \sum_{i=2}^{k}\left[\prod_{j=i}^{k}\frac{\gamma_j}{\mathcal{Z}_j}\right]\frac{\rho_{i-1}^*}{\mathcal{Z}_{i-1}} \\
&= \frac{\hat{\rho}_k + \gamma_k\Sigma_{\rho}^{(k-1)}}{\mathcal{Z}_k} \qquad \text{(B.8)} \\
\Pi_\gamma^{(k)} &= \prod_{i=1}^{k}\frac{\gamma_i}{\mathcal{Z}_i} \\
&= \frac{\gamma_k\Pi_\gamma^{(k-1)}}{\mathcal{Z}_k} \qquad \text{(B.9)}
\end{aligned}$$

and as before,

$$\mathcal{Z}_i = |\mathcal{H}_i|^2 + \gamma_i + \rho_i \tag{B.10}$$

Note that (B.7), (B.8) and (B.9) differ from their counterpart equations (4.8), (4.9) and (4.18) only by the summation starting indices. The first term for each summation (i.e. $i = 1$) is now included in the expression $\Pi_\gamma^{(k)} \mathcal{E}_{\hat{\mathcal{S}}_0}$. Essentially, this shows the effect of the initial guess and how it propagates based on the multiplicative term $\Pi_\gamma^{(k-1)}$. By examination of the first line of (B.9), we see that if

$$\frac{\gamma_i}{\mathcal{Z}_i} < 1 \quad \forall\, i \tag{B.11}$$

then the nonblind algorithm of Chapter 4 will converge independent of the initial image estimate $\hat{\mathcal{S}}_0$ and its error $\mathcal{E}_{\hat{\mathcal{S}}_0}$.

# Bibliography


[1] D.L. Anguin, H. Kaufman, "Image Restoration Using Reduced Order Models", *Signal Processing*, Vol. 16, pp. 21-28, 1989.

[2] G. R. Ayers, J. C. Dainty, "Iterative Blind Deconvolution Method and its Applications", *Optics Letters*, Vol 13, No. 7, pp. 547-549, July 1988.

[3] M.R. Banham, A.K. Katsaggelos, "Digital Image Restoration", *IEEE Signal Processing Magazine*, pp. 24-41, March 1997.

[4] M.R. Banham, N.P. Galatsanos, H.L. Gonzalez, A.K. Katsaggelos, "Multichannel Restoration of Single Channel Images Using a Wavelet-Based Subband Decomposition", *IEEE Transactions on Image Processing*, Vol. 3, No. 6, pp. 821-833, November 1994.

[5] M.R. Banham, A.K. Katsaggelos, "Spatially Adaptive Wavelet-Based Multiscale Image Restoration", *IEEE Transactions on Image Processing*, Vol. 5, No. 4, pp. 619-634, April 1996.

[6] T. Berger, J.O. Strömberg, T. Eltoft, "Adaptive Regularized Constrained Least Squares Image Restoration", *IEEE Transactions on Image Processing*, Vol. 8, No. 9, pp. 1191-1203, September 1999.

[7] H. Bially, "Iterative Behandlung Linearer Funktionalgleichungen", *Archive for Rational Mechanics and Analysis*, Vol. 4, pp. 166-176 July 1959.

[8] J. Biemond, R.L. Lagendijk, R.M. Mersereau, "Iterative Methods for Image Deblurring", *Proceedings of the IEEE*, Vol. 78, No. 5, pp. 856-883, May 1990.

[9] B.L.K. Davey, R.G. Lane, R.H.T. Bates, "Blind Deconvolution of Noisy Complex-Valued Image", *Optics Communications*, Vol. 69, No. 5,6, pp. 353-356, January 1989.

[10] N.P. Galatsanos, A.K. Katsaggelos, "Methods for Choosing the Regularization Parameter and Estimating the Noise Variance in Image Restoration and Their Relation", *IEEE Transactions on Image Processing*, Vol. 1, No. 3, pp. 322-336, July 1992.

[11] N.P. Galatsanos, R.T. Chin, "Digital Restoration of Multichannel Images", *IEEE Transactions on Acoustics, Speech, and Signal Processing*, Vol. 37, No. 3, pp. 415-421, March 1989.

[12] S. Geman, D. Geman, "Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. PAMI-6, No. 6, pp. 721-741, November 1984.

[13] D.C. Ghiglia, L.A. Romero, G.A. Mastin, "Systematic Approach to Two-Dimensional Blind Deconvolution by Zero-Sheet Separation", *Journal of the Optical Society of America*, Vol. 10, No. 5 pp. 1024-1036, May 1993.

[14] G.H. Golub, C.F. Van Loan, *Matrix Computations*, Third Ed., Baltimore MD: The Johns Hopkins University Press, 1996.

[15] G.H. Golub, P.C. Hansen, D.P. O'Leary, "Tikhonov Regularization and Total Least Squares", *SIAM Journal of Matrix Analysis and Applications*, Vol. 21, No. 1, pp. 185-194, January 2000.

[16] G.H. Golub, C.F. Van Loan, " An Analysis of the Total Least Squares Problem", *SIAM Journal of Numerical Analysis*, Vol. 17, No. 6, pp. 883-893, December 1980.

[17] G.H. Golub, U. von Matt, "Quadratically Constrained Least Squares and Quadratic Problems", *Numerische Mathematik*, Vol. 59, pp. 561-580, 1991.

[18] P. C. Hansen, "Regularization Tools, A Matlab Package for Analysis of Discrete Regularization Problems", *Numerical Algorithms 6*, pp. 1-35, 1994.

[19] J. R. Hare, J. P. Reilly, "The Deconvolution of Linearly Blurred Images using Non-Parametric Stabilizing Functions", Accepted for publication in the *Proceedings, 2000 International Conference on Image Processing (ICIP)*, Vancouver, Canada, September 2000.

[20] C.W. Helstrum, "Image Restoration by the Method of Least Squares", *Journal of the Optical Society of America*, Vol. 57, No. 3, pp. 297-303, March 1967.

[21] P. A. Jansson (Ed.), *Deconvolution of Images and Spectra*, Toronto, ON: Academic Press, 1997.

[22] M. G. Kang, A. K. Katsaggelos, "Frequency Domain Adaptive Iterative Image Restoration and Evaluaton of the Regularization Parameter", *Optical Engineering*, Vol. 33, No. 10, pp. 3222-3232, October 1994.

[23] M. G. Kang, A. K. Katsaggelos, "Simultaneous Iterative Image Restoration and Evaluation of the Regualrization Parameter", *IEEE Transactions on Signal Processing*, Vol. 40, No. 9, pp. 2329-2334, September 1992.

[24] A. K. Katsaggelos, "Iterative Image Restoration Algorithms", *Optical Engineering*, Vol. 28, No. 7, pp. 735-748, July 1989.

[25] A. K. Katsaggelos, J. Biemond, R. W. Schafer, R. M. Mersereau "A Regularized Iterative Image Restoration Algorithm", *IEEE Transactions on Image Processing*, Vol. 39, No. 4, pp. 914-929, April 1991.

[26] A. K. Katsaggelos, S. N. Efstratiadis, "A Class of Iterative Signal Restoration Algorithms", *IEEE Transactions on Acoustics, Speech, and Signal Processing*, Vol. 38, No. 5, pp. 778-786, May 1990.

[27] A. K. Katsaggelos, K.T. Lay, "Maximum Likelihood Blur Identification and Image Restoration Using the EM Algorithm", *IEEE Transactions on Signal Processing*, Vol. 39, No. 3, pp. 729-733, March 1991.

[28] A. K. Katsaggelos, K.T. Lay, N.P. Galatsanos, "A General Framework for Frequency Domain Multi-Channel Signal Processing", *IEEE Transactions on Image Processing*, Vol. 2, No. 3, pp. 714-420, July 1993.

[29] S. Kawata, Y. Ichioka "Iterative Image Restoration for Linearly Degraded Images. I. Basis", *Journal of the Optical Society of America*, Vol. 70, pp. 762-767, 1980.

[30] S. Kawata, Y. Ichioka "Iterative Image Restoration for Linearly Degraded Images. II. Reblurring Procedure", *Journal of the Optical Society of America*, Vol. 70, pp. 768-772, 1980.

[31] S. Kirkpatrick, C.D. Gelatt Jr., M.P. Vecchi, "Optimization by Simulated Annealing", *Science*, Vol. 220, pp. 671-680, 1983.

[32] D. Kundur, D. Hatzinakos, "Blind Image Deconvolution", *IEEE Signal Processing Magazine*, pp. 43-64, May 1996.

[33] R. L. Lagendijk, J. Biemond, *Iterative Identification and Restoration of Images*, Boston, Mass:Kluwer Academic Press, 1991.

[34] R. L. Lagendijk, J. Biemond, D. E. Boekee, "Regularized Iterative Image Restoration with Ringing Reduction", *IEEE Transactions on Acoustics, Speech, and Signal Processing*, Vol. 36, No. 12, pp. 1874-1887, December 1988.

[35] R. L. Lagendijk, A.M. Tekalp, J. Biemond, "Maximum Likelihood Image and Blur Identification: A Unifying Approach", *Optical Engineering*, Vol. 29, No. 5, pp. 422-435, May 1990.

[36] R. L. Lagendijk, J. Biemond, D. E. Boekee, "Identificaion and Restoration of Noisy Blurred Iamges Using the Expectation-Maximization Algorithm", *IEEE Transactions on Acoustics, Speech, and Signal Processing*, Vol. 38, No. 7, pp. 1180-1191, July 1990.

[37] L. Landweber, "An Iteration Formula for Fredholm Integral Equations of the First Kind", *American Journal of Mathematics*, Vol. 73, pp. 615-624, 1951.

[38] R.G. Lane, R.H.T. Bates, "Automatic Multidimensional Deconvolution", *Journal of the Optical Society of America*, Vol. 4, No. 1, pp. 180-188, 1987.

[39] K.T. Lay, A.K. Katsaggelos, "Image Identification and Restoration Based on the Expectation-Maximization Algorithm", *Optical Engineering*, Vol. 29, No. 5, pp. 436-445, May 1990.

[40] R. Molina, A. K. Katsaggelos, J. Abad "Bayesian Image Restoration Using a Wavelet-Based Subband Decomposition", *Proceedings, 1999 International Conference on Acoustics, Speech and Signal Processing*, pp 3257-3259, 1999.

[41] B.C. McCallum, "Blind Deconvolution by Simulated Annealing", *Optics Communications*, Vol. 75(2), pp. 101-105, February 1990.

[42] B. L. McGlamery, "Restoration of Turbulence Degraded Images", *Journal of the Optical Society of America*, Vol. 57, No. 3, pp. 293-297, 1967.

[43] R.P. Millane, P.J. Bones, H. Jiang, "Blind Deconvolution for Multidimensional Images", *Proceedings, 1994 International Conference on Acoustics, Speech and Signal Processing*, Adelaide, Australia, pp. V445-448, April 1994.

[44] K. Miller, "Least Squares Methods for Ill-Posed Problems with a Prescribed Limit", *SIAM Journal of Mathematical Analysis*, Vol. 1, No. 1, pp. 52-74, February 1970.

[45] N. Miura, K. Ohsawa, N. Baba, "Single-Frame Blind Deconvolution by Means of Frame Segmenation", *Optics Letters*, Vol. 19, No. 10, pp. 695-697, May 1994.

[46] C.E. Morris, M.A. Richards, M.H. Hayes, "Fast Reconstruction of Linearly Distorted Signals", *IEEE Transactions on Acoustics, Speech, and Signal Processing*, Vol. 36, No. 7, pp. 1017-1025, July 1988.

[47] S. J. Reeves, R. M. Mersereau, "Optimal Estimation of the Regularization Paramater and Stabilizing Functional for Regularized Image Restoration", *Optical Engineering*, Vol. 29, No. 5, pp. 446-454, May 1990.

[48] S. J. Reeves, R. M. Mersereau, "Blur Identification by the Method of Generalized Cross-Validation", *IEEE Transactions on Image Processing*, Vol. 1, No. 3, pp. 301-311, July 1992.

[49] J.P. Reilly, M. Siebert, M. Wilbur, N. Ahmadvand, "The Single-Sided Subband Decomposition: Application to the Decimation of Large Problems", Submitted for publication in *IEEE Transactions on Signal Processing.*

[50] A. Sabharwal, L. C. Potter, "Convexly Constrained Linear Inverse Problems: Iterative Least-Squares and Regularization", *IEEE Transactions on Signal Processing*, Vol. 46, No. 9, pp. 2345-2352, September 1998.

[51] R. W. Schafer, R. M. Mersereau, M. A. Richards, "Constrained Iterative Restoration Algorithms", *Proceedings of the IEEE*, Vol. 69, No. 4, pp. 432-450, April 1981.

[52] T.J. Schulz, "Multiframe Blind Deconvolution of Astronomical Images", *Journal of the Optical Society of America*, Vol. 10, No. 5, pp. 1064-1073, May 1993.

[53] M.I. Sezan, H. Stark, "Image Restoration by the Method of Convex Projections: Part 2 - Applications and Numerical Results", *IEEE Transactions on Medical Imaging*, Vol. MI-1, No. 2, pp. 95-101, October 1982.

[54] P. A. Stokseth, "Properties of a Defocussed Optical System", *Journal of the Optical Society of America*, Vol. 59, pp. 1314-1321, 1969.

[55] B. J. Sullivan, A. K. Katsaggelos, "New Termination Rule for Linear Iterative Restoration Algorithm" *Optical Engineering*, Vol. 29, No. 5, pp. 471-477, May 1990.

[56] A. Tikhonov, V. Arsenin, *Solution of Ill-Posed Problems*, New York:Wiley, 1977.

[57] L. Tong, S. Perreau, "Multichannel Blind Identification: From Subspace to Maximum Likelihood Methods", *Proceedings of the IEEE*, Vol. 86, No. 10, pp. 1951-1968, October 1998.

[58] F. Tsumuraya, M. Miura, N. Baba, "Iterative Blind Deconvolution Method using Lucy's Algorithm", *Astronomy and Astrophysics*, Vol. 282, pp. 699-708, 1994.

[59] P.H. van Clittert, "Zum Einfuluss der Spaltbreit auf die Intensitätsverteilung in Spektrallinien II", *Zeitschrift für Physik*, Vol. 69, pp. 298-308 1931.

[60] R.A. Wiggins, "Minimum Entropy Deconvolution", *Geoexploration*, Vol. 16, pp. 21-35, 1978.

[61] J.W. Woods, C.H. Radewan, "Kalman Filtering in Two Dimensions", *IEEE Transactions on Information Theory*, Vol. 23. No. 4, pp. 473-482, April 1977.

[62] J.W. Woods, V.K. Ingle, "Kalman Filtering in Two Dimensions: Further Results", *IEEE Transactions on Acoustics, Speech, and Signal Processing*, Vol. 29. No. 2, pp. 188-197, February 1981.

[63] H.-S. Wu, "Minimum Entropy Deconvolution for Restoration of Blurred Two-Tone Images", *Optics Letters*, Vol. 26, No. 15, pp. 1183-1184, July 1990.

[64] A.E. Yagle, "Multiresolution Blind Deconvolution of Symmetric Point-Spread Functions from Bioelectrical Potentials", *Proceedings, 2000 International Conference on Acoustics, Speech and Signal Processing*, Istanbul, Turkey, June 2000.

[65] D. C. Youla, H. Webb, "Image Restoration by the Method of Convex Projections: Part 1 - Theory", *IEEE Transactions on Medical Imaging*, Vol. MI-1, No. 2, pp. 81-94, October 1982.

# Program Listing

```
% script file DeConv.m
% to compare results for deconvolved images with
% the iterative Tikhonov-Miller algorithm

clear all

blind = logical(1);
pad = 1;                    % zero padding factor
real_ht = 7;                % PSF parameters
real_wd = 7;

max_other_count = 150;
max_num_change = 5;
min_change = 2e-5;

if blind
     lambda1 = 0.25;          % relaxation parameter
else
     lambda1 = 1;
end

do_lena = 1;

showfigures = 10;
BSNR = 30;
cd ../lena
if do_lena == 1
     rate = 2;
     row = round(300./rate);
     txt = 'Lena';
```

```
     lena = double(imread('lena.jpg','jpg'));
     lena = lena(1:rate:512,1:rate:512);
else
     txt = 'Cameraman';
     row = 60;
     lena = imread('camera.jpg','jpg');
     lena = double(lena(:,:,2));
end

btm = 0;
top = 255;

lena = lena - min(min(lena));
lena = round(lena .* 255 ./ max(max(lena)));

tM = size(lena,1);
tN = size(lena,2);

% generate point spread function h
L = 31;
lag = floor(L./2);
if blind
     h = g_PSF([real_ht real_wd],L).* m_PSF([real_ht real_wd],L);
     h = h ./ sum(sum(h));
else
     h = m_PSF([real_ht real_wd],L);
end

M = pad.*tM;
N = pad.*tN;

figure(1)
colormap(gray);
subplot(221),imagesc(lena,[btm top]);
title(sprintf('Original %s',txt));
% zero pad original image
tlena = zeros(M,N);
r_sta = tN.*(pad-1)./2+1;
r_stp = tN.*(pad-1)./2+tN;
c_sta = tM.*(pad-1)./2+1;
```

```
c_stp = tM.*(pad-1)./2+tM;
tlena(r_sta:r_stp,c_sta:c_stp) = lena;

c1 = zeros(size(h));
c2 = c1;
c3 = [0 -.25 0 ; -.25 1 -.25 ; 0 -.25 0];
% c3 is the Laplacian, c4 is used to make Laplacian zero phase
c4 = zeros(3,3);
c4(2,2) = 1;
c1(lag+1,lag+1) = 1;
c2(1,1) = 1;
C2 = fft2(c2,M,N);
C1 = fft2(c1,M,N);
C3 = fft2(c3,M,N);
C4 = fft2(c4,M,N);
%%%%%%%%%%%%%%%%%%%%%%%%%

H = fft2(h,M,N);
H = conj(C1).*H;
rH = H;
rh = h;

Laplacian = conj(C4).*C3;
clear C3 C4 c1 c2 c3 c4;

Lena = fft2(tlena,M,N);
R = fft2(tlena,M,N).*H;
r = real(ifft2(R));
r = r(c_sta:c_stp,c_sta:c_stp);
n_std = sqrt(std2(r).^2 ./ (10.^(BSNR./10)));
nois = zeros(size(lena));
nois = n_std .* randn(size(lena));
nois = nois - mean2(nois);
Nois = zeros(M,N);
Nois(r_sta:r_stp,c_sta:c_stp) = nois;
Nois = fft2(Nois,M,N);
Nois(1,1) = 0;

%whiten the noise in FFT domain
rPWR = sum(sum(abs(Nois).^2));
```

```
No(abs(Nois)==0)=eps;
Nois = sqrt(rPWR./(M.*N)).*sign(Nois);

Nois(1,1) = 0;
nois = real(ifft2(Nois));
nois = nois(r_sta:r_stp,c_sta:c_stp);

r = r + nois;
R = R + Nois;

figure(1);
subplot(222),imagesc(r,[btm top]);
title(sprintf('Blurred %s, BSNR is %0.4g',txt,BSNR));
xlabel(sprintf('Noise variance is %0.4g Error variance is %0.4g', ...
     n_std.^2,std2(lena-r).^2));
pause(1);

%%%%%%%% calculation of alpha for TM solution %%%%%%%%%%%%%%%%

alpha = sum(sum(abs(Nois).^2)) ./ sum(sum(abs(Laplacian.*Lena).^2));
tGamma = alpha.*abs(Laplacian).^2;

bestTM = conj(rH).*R ./ (abs(rH).^2 + tGamma);
besttm = real(ifft2(bestTM));
besttm = besttm(c_sta:c_stp,c_sta:c_stp);
bestTMISNR = 10.*log10(sum(sum(abs(lena-r).^2)) ./  ...
     sum(sum(abs(lena-besttm).^2)));

%%%%%%%%%%  set options for optimization %%%%%%%%%%%%%

options = optimset;
options.LevenbergMarquardt = 'on';
options.TolX = 1e-15;
options.TolFun = 1e-15;
options.LargeScale='on';
options.MaxFunEvals = 5000;

%%%%%%%%%%% a priori blur estimation %%%%%%%%%%%%%%%%%%%

start = logical(1);
```

```
if blind
     start_ht = 1;
     fin_ht = 13;
     start_wd = 3;
     fin_wd = 13;
else
     start_ht = real_ht;
     fin_ht = real_ht;
     start_wd = real_wd;
     fin_wd = real_wd;
end

for m = start_ht:2:fin_ht
     for n = start_wd:2:fin_wd
          eH = fft2(m_PSF([m,n],L),M,N);
          eH = conj(C1).*eH;

          V = zeros(1,length(-12:.5:2));
          ctr = 0;
          for q = -12:.5:2
               ctr = ctr+1;
               rho = exp(q) .* abs(Laplacian).^2;
               V(ctr) = checkGCV(R,eH,rho);
          end

          I = find(V == min(V));
          q = -12:.5:2;
          alph = exp(q(I(1)));

          alph = lsqnonlin('gcv_hp1',alph,1e-6,1, ...
               options,R,eH,Laplacian);
          rho = alph .* abs(Laplacian).^2;

          denom = abs(eH).^2 + rho;
          denom(abs(denom)==0) = eps;
          Z = rho./denom;

          nerr = sum(sum(abs(Z).^2 .* abs(R).^2)) ./ ...
               sum(sum(abs(Z))).^2;
```

```
          nerr = nerr./ N;

          if start
               err = nerr;
               start = logical(0);
               ht = m;
               wd = n;
               alph_gcv = alph;

          else
               if (nerr <= err)
                    disp([m n nerr alph alpha]);
                    err = nerr;
                    ht = m;
                    wd = n;
                    alph_gcv = alph;
               end
          end
     end
end

disp(sprintf('ht is %g and wd is %g',ht,wd));
h = m_PSF([ht wd],L);
H = fft2(h,M,N);
H = conj(C1).*H;


%%%%%%%%%%%%%%% noise variance estimation %%%%%%%%%%%%%%%
var = abs(H).^2 ./ (abs(H).^2 + alph_gcv.*abs(Laplacian).^2);
var = sqrt(1-var).*R;
var = (M.*N) ./ (tM.*tN) .* sum(sum(abs(var).^2)) ./(M.*N).^2;

PWR = var.*(tM.*tN).*(M.*N);
AOS = (m_PSF([ht wd],L) > 0);      % area of support for PSF

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

if blind == 1
     h = m_PSF([ht wd],L);
     H = fft2(h,M,N);
```

```
     H = conj(C1).*H;
else
     h = rh;
     H = rH;
end


distance = 100;
stop = 1e-15;
count = 0;
other_count = 0;

rho = 0.5 .* abs(1-abs(H).^2);
rho(1,1) = 0;
SE = conj(H).*R ./ (abs(H).^2+rho);

No = R-H.*SE;
No(1,1) = 0;
nPWR = sum(sum(abs(No).^2));

No = sqrt(PWR./(M.*N)).*sign(No);
No(1,1) = 0;

sumQ = conj(H) ./ (abs(H).^2 + rho);
sumRho = rho ./ (abs(H).^2 + rho);
Eh = zeros(size(H));
sumEh = zeros(size(H));

Ek = (conj(H).*No - rho.*SE) ./ (abs(H).^2+rho);
tSE = conj(rH).*R;

ISNR = [];
tISNR = [];
dist = [];
tdist = [];

Gamma = zeros(size(H));
NSE = SE;
nH = H;
beta = BSNR./30;
```

```
beta = min([beta 1.9]);
num_change = 0;

TR = R(1:pad:M,1:pad:N);
TSE = SE(1:pad:M,1:pad:N);
TH = H(1:pad:M,1:pad:N);
%%%%%%%%%%%%%%%%%%%% main image estimation loop %%%%%%%%%%%%%%%%%%%%
while ((distance > stop | tdistance > stop) | count < 2) & ...
 ~(num_change == max_num_change & other_count == max_other_count)

     count = count + 1;
     other_count = other_count + 1;
     if count == 1
          No = R-H.*SE;
          No(1,1) = 0;
          No = sqrt(PWR./(M.*N)).*sign(No);
          No(1,1) = 0;
     end
     if blind == 1
     %%%%%%%%%% PSF estimation loop (if blind) %%%%%%%%%%%%%%%%
          %%%%%%%%% sample H, SE, No, and Ek spectra %%%%%%%%%%
          TH = H(1:pad:M,1:pad:N);
          TSE = SE(1:pad:M,1:pad:N);
          TNo = No(1:pad:M,1:pad:N);
          TEk = Ek(1:pad:M,1:pad:N);

          clear get_H_rho;
          h_rho = get_H_rho(TR,TSE,TH,TNo,TEk);

          denom = abs(TSE).^2 + h_rho;
          nH = conj(TSE).*(TR) ./ denom;

          nh = real(ifft2(nH));
          nh = estimate(nh,L);
          nh = nh .* AOS;

          nh = nh ./ sum(sum(nh));

          %%%%% change PSF if nonblind procedure has converged %%%%%%
          if (distance<=min_change|other_count>=max_other_count) ...
```

```
& num_change < max_num_change

Hdist = [];
TH = nH;
Hdistance = 0;
Hstop = 1e-6;
Hcount = 0;
while Hdistance > Hstop
     Hcount = Hcount + 1;
     TNo = TR-TH.*TSE;
     TNo(1,1) = 0;
     TNo = sqrt(PWR./(M.*N)).*sign(TNo);
     TNo(1,1) = 0;

     h_rho = get_H_rho(TR,TSE,TH,TNo,TEk);

     denom = abs(TSE).^2 + h_rho;
     nH = conj(TSE).*(TR) ./ denom;

     nnh = real(ifft2(nH));
     nnh = estimate(nnh,L);
     nnh = nnh .* AOS;

     nnh = nnh ./ sum(sum(nnh));
     nh = nh + 0.25.*(nnh - nh);

     nH = fft2(nh,tM,tN);

     Hdistance = sum(sum(abs(nH - TH).^2)) ...
          ./ sum(sum(abs(nH).^2));
     Hdist = [Hdist Hdistance];
     TH = nH;
end

h = h + 0.25.*(nh - h);
H = conj(C1).*fft2(h,M,N);
num_change = num_change + 1;

Gamma = zeros(size(H));
```

```
            rho = 0.5.*(1-abs(H).^2);
            rho(1,1) = 0;

            %%%%%% reinitialize variables %%%%%%%%%%%%
            SE = conj(H).*R ./ (abs(H).^2 + rho);
            No = R-H.*SE;
            No(1,1) = 0;
            No = sqrt(PWR./(M.*N)).*sign(No);
            sumQ = conj(H) ./ (abs(H).^2+rho);
            sumRho = rho ./  (abs(H).^2+rho);

            Ek = (conj(H).*No - rho.*SE) ./ (abs(H).^2+rho);
            rho = zeros(size(H));
            other_count = 0;
            min_change = min_change./2;
      end
end % if blind==1
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

clear get_a_C
clear get_a_rho;
if (count == 1)
      Gamma = zeros(size(H));
      rho = zeros(size(H));
end

Gamma = get_a_gamma(R,H,Eh,Ek,No,SE,rho);
rho = get_a_rho(R,H,Eh,Ek,No,SE,Gamma);

tNSE = beta.*conj(rH).*R + ...
      tSE.*(1 - beta.*(abs(rH).^2 + tGamma));

denom = abs(H).^2 + rho + Gamma;
msk = (abs(denom)==0 | ~isfinite(denom));

if sum(sum(msk)) > 0
      disp(sprintf('%g zeros or non finite coeff in denom', ...
            (sum(sum(msk)))));
end
```

```
NSE = zeros(size(SE));
NSE(~msk) = (conj(H(~msk)).*(R(~msk)) + ...
     Gamma(~msk).*(SE(~msk))) ./ (denom(~msk));

NSE(msk) = SE(msk);
NSE(~isfinite(NSE)) = SE(~isfinite(NSE));

% update error propagation gain terms
sumQ(~msk) = (conj(H(~msk)) + ...
     Gamma(~msk).*sumQ(~msk))./denom(~msk);
sumRho(~msk) = (rho(~msk) + ...
     Gamma(~msk).*sumRho(~msk))./denom(~msk);
sumEh(~msk) = (conj(H(~msk)).*Eh(~msk) + ...
     Gamma(~msk).*sumEh(~msk))./ denom(~msk);

sumQ(msk) = 0;
sumRho(msk) = 0;
sumEh(msk) = 0;

%%%%%% estimate noise spectrum %%%%%%%%%%%%
No = R-H.*(NSE);
No(1,1) = 0;

No = sqrt(PWR./(M.*N)).*sign(No);
No(1,1) = 0;

%%%%%%%%% estimate transient error %%%%%%%%%%
Ek = zeros(size(H));

denom = 1 - (0.*H.*sumQ + sumRho +sumEh);
msk = abs(denom) > 0;
Ek = No.*sumQ - NSE.*(sumRho+sumEh);
Ek(msk) = Ek(msk) ./ denom(msk);
Ek(~msk | ~isfinite(Ek)) = 0;

disp(sprintf('max sumQ is %g and max sumRho is %g', ...
max(max(abs(sumQ))),max(max(abs(sumRho)))));
disp(sprintf('avg sumQ is %g and avg sumRho is %g', ...
mean2((sumQ)),mean2((sumRho))));
disp(sprintf('estimated noise variance is %g',var));
```

```
disp(' ');

se = real(ifft2(NSE));
tse = real(ifft2(tNSE));

se = se(c_sta:c_stp,c_sta:c_stp);
tse = tse(c_sta:c_stp,c_sta:c_stp);

tISNR = [tISNR 10.*log10(sum(sum(abs(lena-r).^2)) ...
     ./ sum(sum(abs(lena-tse).^2)))];
ISNR = [ISNR 10.*log10(sum(sum(abs(lena-r).^2)) ...
     ./ sum(sum(abs(lena-se).^2)))];
maxcount = find(ISNR==max(ISNR));
maxcount = maxcount(length(maxcount));

if maxcount == count
     bestSE = NSE;
     bestse = real(ifft2(bestSE));
     bestse = bestse(c_sta:c_stp,c_sta:c_stp);
     bestISNR = 10.*log10(sum(sum(abs(lena-r).^2)) ...
          ./ sum(sum(abs(lena-bestse).^2)));
end

distance = sum(sum(abs(NSE - SE).^2)) ...
     ./ sum(sum(abs(NSE).^2));
tdistance = sum(sum(abs(tNSE - tSE).^2)) ...
     ./ sum(sum(abs(tNSE).^2));
if distance <= stop
     disp('proposed procedure has converged');
end
if tdistance <= stop
     disp('iterative TM procedure has converged');
end
dist = [dist distance];
tdist = [tdist tdistance];

%%%%%%%% show images and graphs %%%%%%%%%%%%%%%%%%%%
if (rem(count,showfigures) == 0 | distance <= stop) | count < 20

     figure(1)
```

```
subplot(223),imagesc(se,[btm top]);
title(sprintf('Estimated %s after %g iterations', ...
     txt,count));

xlabel(sprintf('Error variance is %0.4g ISNR is %0.4g', ...
     std2(lena-se).^2,ISNR(count)));

subplot(224),imagesc(tse,[btm top]);
title(sprintf('Iterative TM solution to %s',txt));
xlabel(sprintf('Error variance is %0.4g ISNR is %0.4g', ...
     std2(lena-tse).^2,tISNR(count)));

figure(2)
clf
plot(lena(round(row),:),'r');
hold on

plot(se(round(row),:),'g');
title(sprintf('row %g',row));

figure(3)
clf
plot(real(H(1,:)),'b');
hold on
plot(real(Gamma(1,:)),'r');
plot(real(rho(1,:)),'k');
plot(alpha.*ones(1,N),'k-.');
plot(real(rH(1,:)),'m--');
plot(real(H(:,1)'),'g');
axis([1 M -1.5 1.5]);

figure(4)
clf
subplot(121),plot(ISNR(max([count-100000 1]):count),'r')
if count > 1
     subplot(121),axis([1 count floor(min(ISNR))-1 ...
          ceil(max(ISNR))+1]);
end

title('ISNR Comparison: Proposed vs TM');
```

```
        xlabel('Iteration');
        ylabel('ISNR (dB)');

        hold on
        subplot(122),semilogy(dist,'r');
        title('Normalized Energy Change Comparison');
        xlabel('Iteration')
        ylabel('Normalized Energy Change (log scale)');

        hold on

        subplot(121),plot(tISNR(max([count-100000 1]):count),'r--')
        legend('Blind Hybrid Method','Iterative TM',4);
        if count > 1
             subplot(121),axis([1 count floor(min(ISNR))-1  ...
                  ceil(max(ISNR))+1]);
        end

        subplot(122),semilogy(tdist,'r--');
        legend('Blind Hybrid Method','Iterative TM');

        if blind == 1
             figure(5)
             ctr = lag+1;
             clf
             subplot(311),imagesc(h(ctr-5:ctr+5,ctr-5:ctr+5));
             subplot(312),mesh(h);
             title('current PSF');
             subplot(313),mesh(nh);
             title('latest estimate of PSF or nh');
        end
        disp(sprintf('Maximum ISNR is %g at iteration %g', ...
             max(ISNR),maxcount));
        disp(sprintf('The average MS pixel error in h is %g  ...
             at iteration %g',mean2((rh(AOS)-h(AOS)).^2),count));
        pause(1);
   end

   %%%%% apply relaxation to result if blind %%%
```

```
      SE = SE + lambda1.*(NSE - SE);
      tSE = tNSE;
      %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
end

if exist('se') ~= 1
      se = real(ifft2(NSE));
      se = se(r_sta:r_stp,c_sta:c_stp);
end
```

```
function h = m_PSF(p,L);

% h = m_PSF(p,L);
%
% returns an elliptical PSF with uniform pixel values.
% p = [p1 p2] which are the vertical and horizontal major axis
% dimensions of the ellipse.
% a height parameter of 1 (i.e. p1 = 1) is a linear motion blur.

if length(p) == 1
     p = [1 p];
end

lag = floor(L./2);

m = (1:L)';
ctr = lag+1;
h1 = (m-ctr).^2 ./ (p(1)./2).^2;
h1 = h1(:,ones(1,L));
h2 = (m'-ctr).^2 ./ (p(2)./2).^2;
h2 = h2(ones(L,1),:);
h = (h1+h2)<1;
h = h + (h1+h2<=1);

h = h./(sum(sum(h)));
```

```
function h = g_PSF(p,L);

% h = g_PSF(p,L);
%
% returns a Gaussian PSF with std parameters p = [p1 p2]
% which are the vertical and horizontal standard deviations
% repectively.

if rem(L,2) == 0;
     L = L + 1;
end

lag = floor(L./2);
h = zeros(L,L);
sd1 = p(1);
if length(p) == 2
     sd2 = p(2);
else
     sd2 = p(1);
end

for m = -lag:lag
     for n = -lag:lag
          h(m+lag+1,n+lag+1) = exp(-(0.5).* ...
               ((m.^2./(sd1.^2))+(n.^2./(sd2.^2))));
     end
end

h = h./(sum(sum(h)));
```

```
function f = gcv_hp1(p,R,H,C);

% f = gcv_hp1(p,R,H,C);
%
% parameter estimation for regularization parameter lambda
% f is objective V(theta) for generalized cross validation

N = size(R,1);
M = N.^2;


alphC = p .* abs(C).^2;
denom = abs(H).^2 + alphC;
Z = alphC./denom;

f = M.* sum(sum(Z.^2 .* abs(R).^2)) ./ sum(sum(Z)).^2;
```

```
function Gamma = get_a_gamma(R,H,Eh,Ek,No,Sk,rho)

% Gamma = get_a_gamma(R,H,Eh,Ek,No,Sk,rho)
%
% function to estimate stability operator "gamma"

S = Sk;

A = abs(H).^2.*Sk - conj(H).*R + rho.*Sk + conj(H).*Eh.*S;
B = -conj(H).*No + rho.*(S) + conj(H).*Eh.*S;

denom = real(A.*conj(Ek) + conj(A).*Ek);
msk = abs(denom)~=0;
Gamma = zeros(size(H));
Gamma(msk) = real(A(msk).*conj(B(msk)) +  ...
     conj(A(msk)).*B(msk)) ./ denom(msk);
Gamma(~msk) = 0;
Gamma(1,1) = 0;

Gamma(Gamma<0) = 0;
```

```
function rho = get_a_rho(R,H,Eh,Ek,No,Sk,Gamma)

% rho = get_a_rho(R,H,Eh,Ek,No,Sk,Gamma);
%
% function to estimate regularization parameter "rho"


S = Sk;

X = conj(H).*R + Gamma.*Sk - conj(H).*Eh.*S;
Y = Gamma.*Ek + conj(H).*No - conj(H).*Eh.*S;

denom = real((S).*conj(X) + conj(S).*X);

msk = (abs(denom) ~= 0);
rho = zeros(size(H));
rho(msk) = real(conj(X(msk)).*Y(msk) +  ...
     X(msk).*conj(Y(msk))) ./ denom(msk);

rho(rho<0) = 0;
rho(1,1) = 0;
```

```
function rho = get_H_rho(R,Sk,H,No,Ek)

% rho = get_H_rho(R,Sk,H,No,Ek);
%
% function to estimate regularization parameter for PSF estimation

Ek = 0;

X = conj(Sk).*R;
Y = conj(Sk).*(No - Ek.*H);

denom = real((H).*conj(X) + conj(H).*X);

msk = (abs(denom) ~= 0);
rho = zeros(size(H));
rho(msk) = real(conj(X(msk)).*Y(msk) +  ...
     X(msk).*conj(Y(msk))) ./ denom(msk);

rho(rho<0) = 0;

mx = max(max(abs(Sk).^2));
rho(~msk) = mx;
```

```
function eh = estimate(h,L);

% eh = estimate(h,L);
%
% function to estimate PSF by
% averaging pixels to attain symmetry
% and enforcing the non-increasing pixel value
% constraint

N = size(h,1);

ctr = N./2 + 1;
lag = floor(L./2);

h = fftshift(h);

h = h(ctr-lag:ctr+lag,ctr-lag:ctr+lag);

h(h<0) = 0;
ctr = lag+1;

A = h(ctr,ctr);
B1 = h(ctr,1:lag);
B2 = fliplr(h(ctr,ctr+1:L));
C1 = h(1:lag,ctr);
C2 = flipud(h(ctr+1:L,ctr));

D1 = h(1:lag,1:lag);
D2 = fliplr(h(1:lag,ctr+1:L));
D3 = flipud(h(ctr+1:L,1:lag));
D4 = flipud(fliplr(h(ctr+1:L,ctr+1:L)));

% average each quadrant in PSF

B1 = (B1+B2)./2;
C1 = (C1+C2)./2;
D1 = (D1+D2+D3+D4)./4;

% enforce non-increasing with index constraint
D1(1,:) = zeros(1,lag);
```

```
D1(:,1) = zeros(lag,1);

if B1(lag) > A
     B1(lag) = A;
end
if C1(lag) > A;
     C1(lag) = A;
end
for m = lag:-1:2
     if B1(m-1) > B1(m)
          B1(m-1) = B1(m);
     end
     if C1(m-1) > C1(m)
          C1(m-1) = C1(m);
     end
     msk = D1(:,lag) > C1;
     D1(msk,lag) = C1(msk);
     msk = D1(lag,:) > B1;
     D1(lag,msk) = B1(msk);

     for n = lag:-1:2
          if D1(m,n-1) > D1(m,n)
               D1(m,n-1) = D1(m,n);
          end
     end
     for n = lag:-1:2
          if D1(n,m-1) > D1(n,m)
               D1(n,m-1) = D1(n,m);
          end
     end
end

eh = [D1 C1 fliplr(D1) ; B1 A fliplr(B1) ;  ...
flipud(D1) flipud(C1) flipud(fliplr(D1))];
```