

# Inverse Filtering by Signal Reconstruction from Phase

by

Megan M. Fuller

B.S. Electrical Engineering  
Brigham Young University, 2012

Submitted to the Department of Electrical Engineering and Computer Science

in partial fulfillment of the requirements for the degree of

Master of Science in Electrical Engineering and Computer Science

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2014

© Massachusetts Institute of Technology 2014. All rights reserved.

Author .....  
Department of Electrical Engineering and Computer Science  
April 24, 2014

Certified by .....  
Jae S. Lim  
Professor of Electrical Engineering  
Thesis Supervisor

Accepted by .....  
Professor Leslie A. Kolodziejcki  
Chairman, Department Committee on Graduate Students



# Inverse Filtering by Signal Reconstruction from Phase

by

Megan M. Fuller

Submitted to the Department of Electrical Engineering and Computer Science  
on April 24, 2014, in partial fulfillment of the  
requirements for the degree of  
Master of Science in Electrical Engineering and Computer Science

## Abstract

A common problem that arises in image processing is that of performing inverse filtering on an image that has been blurred. Methods for doing this have been developed, but require fairly accurate knowledge of the magnitude of the Fourier transform of the blurring function and are sensitive to noise in the blurred image. It is known that a typical image is defined completely by its region of support and a sufficient number of samples of the phase of its Fourier transform. We will investigate a new method of deblurring images based only on phase data. It will be shown that this method is much more robust in the presence of noise than existing methods and that, because no magnitude information is required, it is also more robust to an incorrect guess of the blurring filter. Methods of finding the region of support of the image will also be explored.

Thesis Supervisor: Jae S. Lim

Title: Professor of Electrical Engineering



## Acknowledgments

First, I would like to thank my advisor, Prof. Jae Lim, for giving me a place in his group. His support, patience, and insight have been invaluable, and he has helped me learn that most critical of research skills: How to ask questions.

I would also like to thank my labmates, Xun Cai and Lucas Nissenbaum, for their help and friendship, and express my gratitude to Cindy LeBlanc, who keeps the group running.

Finally, I am deeply indebted to my parents, who have always helped to cultivate my love of learning, and who have been such wonderful examples to me throughout my life.

Thank you all.



# Contents

<b>1</b>	<b>Introduction</b>	<b>13</b>
<b>2</b>	<b>Background</b>	<b>15</b>
2.1	Image Restoration . . . . .	15
2.2	Magnitude Retrieval . . . . .	19
<b>3</b>	<b>Performance in the Presence of Noise</b>	<b>23</b>
3.1	Effect of Noise on Phase . . . . .	23
3.2	Characterization of Phase-Based Deconvolution . . . . .	24
3.2.1	Experimental Setup . . . . .	24
3.2.2	Effect of the Number of Iterations . . . . .	27
3.2.3	Effect of the Size of the DFT . . . . .	30
3.2.4	Effect of the Starting Point . . . . .	32
3.3	Comparison to Other Methods . . . . .	35
3.3.1	Comparison to Inverse Filtering Techniques . . . . .	36
3.3.2	Comparison to Image Restoration Techniques . . . . .	36
3.4	Summary . . . . .	39
<b>4</b>	<b>Performance with Imperfect Knowledge of the Blurring Function</b>	<b>41</b>
4.1	Attributes of Blurring Functions . . . . .	41
4.2	Characterization of Phase-Based Deconvolution . . . . .	42
4.2.1	Experimental Setup . . . . .	42
4.2.2	Effect of the Number of Iterations . . . . .	44

4.2.3	Effect of the Size of the DFT . . . . .	49
4.2.4	Effect of the Starting Point . . . . .	49
4.3	Comparison to Other Methods . . . . .	54
4.3.1	Comparison to Inverse Filtering Techniques . . . . .	54
4.3.2	Comparison to Image Restoration Techniques . . . . .	57
4.4	Noise . . . . .	61
4.4.1	Characterization of Phase-Based Deconvolution . . . . .	61
4.4.2	Comparison to Other Methods . . . . .	62
4.5	Summary . . . . .	64
<b>5</b>	<b>Obtaining the Region of Support</b>	<b>67</b>
5.1	Description of Methods . . . . .	68
5.1.1	ROS from the Degraded Image . . . . .	68
5.1.2	ROS from Morphological Processing . . . . .	69
5.1.3	ROS with Markers . . . . .	70
5.2	Comparison of Methods . . . . .	71
5.2.1	No Noise, Perfect Knowledge of the Blurring Function . . . . .	72
5.2.2	Noise, Perfect Knowledge of the Blurring Function . . . . .	75
5.2.3	No Noise, Imperfect Knowledge of the Blurring Function . . . . .	78
5.3	Summary . . . . .	80
<b>6</b>	<b>Conclusions and Future Work</b>	<b>81</b>
<b>A</b>	<b>Photo Credits</b>	<b>83</b>

# List of Figures

3-1	Original images . . . . .	26
3-2	OS-MSE as a function of noise variance and number of iterations for the tire image . . . . .	28
3-3	Example of artifacts present in the restored image when the algorithm was run to convergence . . . . .	29
3-4	Illustration of the effect of number of iterations on artifacts when stopping short of convergence was optimal . . . . .	29
3-5	Illustration of the effect of number of iterations on artifacts when convergence was optimal . . . . .	30
3-6	OS-MSE as a function of noise variance and DFT size for the cameraman image . . . . .	31
3-7	Illustration of the effect of DFT size on visual quality . . . . .	32
3-8	OS-MSE of the liftingbody image when the iteration was initialized with the degraded image magnitude . . . . .	34
3-9	OS-MSE of the bag image when the iteration was initialized with the undegraded image magnitude . . . . .	35
3-10	Comparison of deconvolution methods . . . . .	37
3-11	Cameraman image deconvolved with several different methods . . . . .	38
3-12	Comparison of restoration methods . . . . .	39
3-13	Cameraman image restored with several different methods . . . . .	40
4-1	Fourier transform of an 11x11-point Gaussian filter with $\sigma = 5$ . . . . .	42
4-2	Comparison of the phases of two lowpass filters . . . . .	43

4-3	OS-MSE of the bag image when the deblurring function was an 11x11-point averager . . . . .	45
4-4	Illustration of the artifacts produced in the bag image as the phase-based reconstruction iteration converged . . . . .	46
4-5	OS-MSE of the cameraman image when the deblurring function was an 11x11-point averager . . . . .	46
4-6	Illustration of the artifacts produced in the cameraman image as the phase-based reconstruction iteration converged . . . . .	47
4-7	OS-MSE of the liftingbody image when the deblurring function was an 11x11-point averager . . . . .	47
4-8	OS-MSE of the liftingbody image when the deblurring function was a 9x9-point averager . . . . .	48
4-9	Illustration of the effect of DFT size on visual quality when the deblurring filter was an 11x11-point averager . . . . .	50
4-10	OS-MSE as a function of number of iterations, DFT size, and starting point (constant and degraded magnitude) for the bag image . . . . .	51
4-11	OS-MSE as a function of number of iterations, DFT size, and starting point (constant and original magnitude) for two images . . . . .	52
4-12	OS-MSE as a function of number of iterations and DFT size for the liftingbody image (starting point degraded image) . . . . .	53
4-13	Comparison of inverse filtering techniques when the deblurring filter was an 11x11-point averager . . . . .	55
4-14	Comparison of inverse filtering techniques when the deblurring filter was a 9x9-point averager . . . . .	56
4-15	Comparison of restoration techniques when the deblurring filter was an 11x11-point averager . . . . .	58
4-16	Cell image restored with several different methods when the deblurring filter was an 11x11-point averager . . . . .	59
4-17	Cameraman image restored with several different methods when the deblurring filter was an 11x11-point averager . . . . .	60

4-18	Comparison of restoration techniques when the deblurring filter was a 9x9-point averager . . . . .	60
4-19	Cameraman image restored with several different methods when the deblurring filter was a 9x9-point averager . . . . .	61
4-20	Comparison of phase-based deconvolution of the liftingbody image to existing methods (deblurring filter an 11x11-point averager) . . . . .	63
4-21	Liftingbody image restored with several different methods (deblurring filter an 11x11-point averager) . . . . .	64
4-22	Comparison of phase-based deconvolution of the liftingbody image to existing methods (deblurring filter a 9x9-point averager) . . . . .	65
4-23	Liftingbody image restored with several different methods (deblurring filter a 9x9-point averager) . . . . .	66
5-1	Original images used in the region of support experiments . . . . .	68
5-2	OS-MSE as a function of number of iterations and ROS error for the black moon image . . . . .	70
5-3	Illustration of the procedure used in marking a blurry image with an unknown region of support . . . . .	71
5-4	OS-MSE as a function of number of iterations and DFT size when the ROS was obtained by morphological processing . . . . .	73
5-5	Artifacts in “convergent” circle images . . . . .	73
5-6	OS-MSE as a function of number of iterations and DFT size when the ROS was obtained by the use of markers . . . . .	74
5-7	Artifacts in “convergent” images when markers were used to indicate the ROS . . . . .	75
5-8	Comparison of methods of determining the ROS of the black moon image in the presence of noise . . . . .	77
5-9	Comparison of artifacts produced by different methods of obtaining the ROS . . . . .	77

5-10 OS-MSE of the circle image when the ROS and the size of the blurring function were unknown . . . . .	79
5-11 OS-MSE of the the black moon image when the ROS and the size of the blurring function were unknown . . . . .	79

# Chapter 1

## Introduction

Two-dimensional deconvolution is a problem of great practical interest. Images taken with a digital camera can be blurred by a number of sources, and often this blurring can be modeled as the convolution of the original image with a blurring function. Therefore, to obtain the original image from the degraded one requires a deconvolution. Many methods of doing this have been developed, each with its own strengths and weaknesses.

Most of these methods rely on Fourier domain techniques. That is, they divide the Fourier transform of the degraded image by some function of the Fourier transform of the blurring function. This leads to two weaknesses in particular. First, because a blurring function is a lowpass signal, deconvolution in the frequency domain requires the high frequencies to be divided by very small numbers. Typical images are also lowpass signals, so this tends to amplify the noise in the image. Second, these methods rely on accurate knowledge of the blurring function.

These weaknesses can be addressed by exploiting the fact that a typical image can be represented by only the phase of its Fourier transform and its region of support. This implies that only the phase of the Fourier transform of the image needs to be corrected and allows deconvolution to be performed by subtraction rather than division, thus mitigating the noise amplification problem. It also only requires knowledge of the phase of the Fourier transform of the blurring function, which may be easier to estimate in some situations than both the phase and the magnitude.

In this thesis, we will explore the strengths and weaknesses of doing deconvolution based on phase. In Chapter 2, the necessary background information will be presented, including the details of the deconvolution problem, a discussion of several existing deconvolution methods, and the algorithm by which an image can be reconstructed from only the phase of its Fourier transform and its region of support.

In Chapter 3, we will examine the performance of phase-based deconvolution in the presence of noise. The effect of changing several parameters of the phase-based reconstruction algorithm will be explored and phase-based deconvolution will be compared to other methods.

In Chapter 4, we will look at the performance of phase-based deconvolution when the blurring function is not known perfectly. The situations in which it is easier to estimate the phase of the Fourier transform of the blurring function than the entire Fourier transform of the blurring function will be explained. Again, the effect of changing the parameters of the phase-based reconstruction algorithm will be explored, and phase-based deconvolution will be compared to other methods.

In Chapters 3 and 4, the region of support of the image will be assumed known. In Chapter 5, we will discuss and compare several methods for obtaining the region of support of the image if it is not known.

Finally, in Chapter 6, we will conclude the thesis and suggest possible future work.

# Chapter 2

## Background

This chapter will be divided into two parts. First, the image restoration problem will be explained, including a summary of some of the most common methods currently used to solve this problem. Then, a discussion of the magnitude retrieval problem will be presented. The way in which the magnitude retrieval problem relates to the image restoration problem is the topic of this thesis, and will be elaborated on in the next chapter.

### 2.1 Image Restoration

Image degradation is typically modeled as a degradation system followed by additive noise [2]. Mathematically, this can be written as

$$g(n_1, n_2) = f(n_1, n_2) * b(n_1, n_2) + v(n_1, n_2) \quad (2.1)$$

or, in the frequency domain,

$$G(\omega_1, \omega_2) = F(\omega_1, \omega_2)B(\omega_1, \omega_2) + V(\omega_1, \omega_2) \quad (2.2)$$

where  $f(n_1, n_2)$  is the original image,  $b(n_1, n_2)$  is the degradation system, often some sort of blurring filter,  $v(n_1, n_2)$  is additive noise, which is known only in a statisti-

cal way, and  $g(n_1, n_2)$  is the degraded image. Capital letters indicate the Fourier transforms of the respective sequences.

The goal of image restoration is to use  $g(n_1, n_2)$  and available information about  $f(n_1, n_2)$ ,  $b(n_1, n_2)$ , and  $v(n_1, n_2)$  to compute some  $\hat{f}(n_1, n_2)$  that is an approximation of  $f(n_1, n_2)$ . This notation will be used throughout.

There are many variations on Eq. 2.1. For example, the additive noise could actually be data dependent, as in the case of Poisson noise, which is common in astronomical images [3]. It may be that the noise is not additive, but multiplicative, which is seen in infrared radar images [9]. However, one of the most common types of noise is signal independent additive white Gaussian. This fits the model of Eq. 2.1 and can come, at least approximately, from many sources, including the read out noise of A/D converters [3].

The blurring term,  $b(n_1, n_2)$ , can also come from many sources. A few of the most common include lens misfocus, atmospheric turbulence, and uniform linear motion [9] [2]. Note that, in writing Eqs. 2.1 and 2.2, it was implicitly assumed that the blurring function was linear and space invariant. This is not always the case, but is approximately true for a large class of degradations and simplifies the restoration computations considerably [2]. Therefore, this assumption will remain in place.

A host of methods currently exist to solve the image restoration problem, each with particular strengths and weaknesses. A few of the most common will now be explained. This list is not meant to be exhaustive, but representative of the kinds of algorithms available.

*Inverse Filtering:* Inverse filtering involves the computation of

$$\hat{F}(\omega_1, \omega_2) = \frac{G(\omega_1, \omega_2)}{B(\omega_1, \omega_2)} \quad (2.3)$$

If  $B(\omega_1, \omega_2)$  is known perfectly and has no zeros, and if there is no noise, inverse filtering will produce exactly the correct answer. However, in the presence of noise, the noise can be amplified, and the result can become visually unacceptable [2]. Variations on inverse filtering include thresholded inverse filtering, where a maximum

value for  $\left|\frac{1}{B(\omega_1, \omega_2)}\right|$  is set to avoid this problem [9] and the use of the “pseudoinverse” filter, which handles the problem of zeros in  $B(\omega_1, \omega_2)$  by setting  $\hat{F}(\omega_1, \omega_2)$  equal to zero at those points [7]. There are also iterative spatial domain methods to compute the deconvolution, and these can be stopped short of convergence or used to impose additional constraints (for example, non-negativity) on the final result [11].

It is worth emphasizing here that inverse filtering and its variations are a somewhat different class of algorithm from those explained below. Inverse filtering solves the problem of blurring, but makes no attempt to account for noise. The methods explained below were all designed with a specific noise model in mind and can be considered simultaneously deblurring and de-noising algorithms.

*Wiener Filtering:* Wiener filtering consists of the computation of

$$H(\omega_1, \omega_2) = \frac{1}{B(\omega_1, \omega_2)} \frac{|B(\omega_1, \omega_2)|^2}{|B(\omega_1, \omega_2)|^2 + S_v(\omega_1, \omega_2)/S_f(\omega_1, \omega_2)} \quad (2.4)$$

where  $S_v(\omega_1, \omega_2)$  and  $S_f(\omega_1, \omega_2)$  are the power spectral densities of the noise and signal, respectively. The image is then restored by taking the inverse Fourier transform of  $\hat{F}(\omega_1, \omega_2) = H(\omega_1, \omega_2)G(\omega_1, \omega_2)$ . The Wiener filter is often called the linear least mean squared error filter because it is a linear system and minimizes  $E\{[f(n_1, n_2) - \hat{f}(n_1, n_2)]^2\}$  [2]. Whether this is the correct quantity to minimize will depend on the application.

Unlike inverse filtering, Wiener filtering will not tend to amplify the noise. This can be seen by looking at Eq. 2.4. At frequencies where the noise is large relative to the signal,  $H(\omega_1, \omega_2)$  will be small, and at frequencies where the noise is small relative to the signal,  $H(\omega_1, \omega_2)$  will be large. However, Wiener filtering relies on knowledge of the power spectral densities of the signal and noise, and these may not be known precisely, if at all. Therefore, in practice, the approximation of  $\frac{S_v(\omega_1, \omega_2)}{S_f(\omega_1, \omega_2)} = K$  where  $K$  is some constant is often used [2].

There are many variations on the Wiener filter, including the parametric Wiener filter, which can give the user more control over the amount of smoothing done, and adaptive Wiener filters, which can account for spatial variance in the statistics of the

noise or image. These can give better results at the cost of more fine-tuning by the user or, in the case of adaptive Wiener filters, more computation [9].

*Constrained Least Mean Squared Error Filter:* Also known as the regularized filter, the constrained least mean squared error filter is an important variation of the Wiener filter that gives the minimum mean squared error solution under a constraint of minimizing the second derivative of the restored image. In this case, the deblurring filter is given by

$$H(\omega_1, \omega_2) = \frac{1}{B(\omega_1, \omega_2)} \frac{|B(\omega_1, \omega_2)|^2}{|B(\omega_1, \omega_2)|^2 + \gamma|P(\omega_1, \omega_2)|^2} \quad (2.5)$$

where  $P(\omega_1, \omega_2)$  is the Laplacian operator, which serves as an approximation for the second derivative, and  $\gamma$  is a Lagrange multiplier. An optimum  $\gamma$  for given noise statistics can be found iteratively. Constraining the second derivative in this way tends to make the restored image smoother. This filter also has the advantage of not requiring any knowledge of the statistics of the signal or noise besides the mean and variance of the noise, which are used in the iteration to find  $\gamma$  [2].

*Richardson-Lucy Iteration:* The Richardson-Lucy iteration relies on a probabilistic interpretation of blurring. It treats the value of each pixel in  $f(n_1, n_2)$  as a count of the number of events that happened there. It assumes the blurring function is normalized, so that the total count of events in  $f(n_1, n_2)$  matches the total count of events in  $g(n_1, n_2)$ . Then, it uses Bayes's theorem to find the probability of an event at a particular  $f(n_1, n_2)$  given an event at a particular  $g(n_1, n_2)$  and given the blurring function. It can be shown that the maximum likelihood solution to this problem can be found with the following iteration. (Note that the one-dimensional case is given for simplicity. The extension to two dimensions is straightforward.)

$$f_0(i) = \frac{\sum_n f(n)}{I} \quad (2.6)$$

$$f_{r+1}(i) = f_r(i) \sum_{k=i}^c \frac{b(k-i+1)g(k)}{\sum_{j=a}^d b(k-j+1)f_r(j)} \quad (2.7)$$

where  $I$  is the size of  $f(n)$ ,  $J$  is the size of  $b(n)$ ,  $a = \max(1, k - J + 1)$ ,  $d = \min(k, I)$ ,

and  $c = i + J - 1$  [13].

Note that Eq. 2.7 is equivalent to

$$f_{r+1}(n) = f_r(n) \left( b(n) \otimes \frac{g(n)}{b(n) * f_r(n)} \right) \quad (2.8)$$

where  $\otimes$  indicates correlation and  $*$  indicates convolution [1].

The Richardson-Lucy iteration presents several advantages. It finds the maximum likelihood solution to the deconvolution problem subject to the constraint that the result be non-negative, and there is nothing in Eq. 2.7 which requires  $b(n_1, n_2)$  to be spatially invariant. Furthermore, bad pixels can be masked by simply not including them in the calculations [3]. However, if there is noise, the Richardson-Lucy iteration must be stopped short of convergence, or it will attempt to fit the noise too closely and produce unacceptable errors [10]. It is not a trivial matter to determine at what point the iteration should be stopped, and, depending on the features of the image, the optimal stopping point may be different for different regions [3].

There are many other image restoration methods. For example, several wavelet-based techniques have recently been developed [8] [12]. The purpose of this section is not to give an exhaustive list of all possible algorithms. As will be explained in more detail in the next chapter, phase-based deconvolution is strictly an inverse filtering method. It will be compared to some image restoration methods not to imply that it should be used as a restoration system, but to demonstrate how robust it is.

## 2.2 Magnitude Retrieval

The magnitude retrieval problem consists of reconstructing the magnitude of the Fourier transform of a signal given only the phase of the Fourier transform of that signal. Because any signal can be convolved with a zero-phase system and produce a different output with the same Fourier transform phase but a different magnitude, it is not, in general, possible to do this.

However, it can be shown that if the Z-transform of the signal does not have any

zeros in conjugate-reciprocal pairs or on the unit circle, and if the region of support of the signal is known (that is, the signal is zero outside of some known range), then the signal is uniquely defined by the phase of its Fourier transform to within a scale factor [4].

These conditions can be extended to the case of multi-dimensional signals, and, furthermore, it can be shown that almost all two-dimensional signals satisfy the Z-transform domain constraints [4]. In addition, the phase of the continuous Fourier transform need not be known. The phase of a sufficiently large discrete Fourier transform (DFT) is also enough to uniquely identify the signal. In this case, a sufficiently large DFT for an image that is  $N_1 \times N_2$  pixels would be at least  $(2N_1 - 1) \times (2N_2 - 1)$  points [4].

There is a closed-form solution to the problem of computing a spatial domain signal from the phase of its DFT. However, this requires the inversion of prohibitively large matrices when the signals are reasonably sized images. Fortunately, there is also an iterative algorithm which alternately imposes the spatial domain constraints and the frequency domain constraints [4]. Specifically, it is as follows, where  $x(n_1, n_2)$  is the signal to be reconstructed and  $X(k_1, k_2)$  is its DFT:

1. Start with an initial guess of the magnitude of  $X(k_1, k_2)$ . Call this  $Y'_0(k_1, k_2)$ . This can be anything—a constant, random values, the magnitude of the DFT of similar images, etc. The algorithm will converge from any starting point.
2. Compute  $Y_{r+1}(k_1, k_2) = |Y'_r(k_1, k_2)|e^{j\theta_X(k_1, k_2)}$ , where  $\theta_X(k_1, k_2)$  is the known phase of  $X(k_1, k_2)$ .
3. Compute  $y_{r+1}(n_1, n_2) = \text{IDFT}[Y_{r+1}(k_1, k_2)]$ .
4. Set  $y'_{r+1}(n_1, n_2) = y_{r+1}(n_1, n_2)$  inside the known region of support, and zero everywhere else.
5. Compute  $Y'_{r+1}(k_1, k_2) = \text{DFT}[y'_{r+1}(n_1, n_2)]$ .
6. Repeat steps 2-5 until convergence is reached.

This algorithm has been proven to converge to the correct solution, provided the uniqueness constraints are met [4]. This algorithm works with general two-dimensional signals. In the special case of images, when the values of the spatial domain signal are known to be positive, step 4 can be modified to set  $y'_{r+1}(n_1, n_2) = |y_{r+1}(n_1, n_2)|$  inside the region of support, and the algorithm will still converge [14].



# Chapter 3

## Performance in the Presence of Noise

### 3.1 Effect of Noise on Phase

The relationship between white Gaussian noise in the spatial domain and phase errors in the Fourier transform of a signal is quite complicated. Any particular sample of the Fourier transform is a function of all the spatial domain samples, leading to a large covariance matrix. Furthermore, in the case of real, finite-length signals, such as images, the real and imaginary parts of the Fourier transform are not independent, but related by the Hilbert transform. Finally, the phase is the arctangent of the imaginary part divided by the real part, which is a nonlinear operation. This makes it very difficult to perform any theoretical analysis of the phase of white noise, even before the signal has been added to it.

Furthermore, because the noise model being considered is additive, the phase of the signal and the phase of the noise do not interact in a straightforward way. Instead, the phase of the noisy signal is given by

$$\theta_G(\omega_1, \omega_2) = \tan^{-1} \left( \frac{\text{Im}\{F(\omega_1, \omega_2)\} + \text{Im}\{V(\omega_1, \omega_2)\}}{\text{Re}\{F(\omega_1, \omega_2)\} + \text{Re}\{V(\omega_1, \omega_2)\}} \right) \quad (3.1)$$

where  $G(\omega_1, \omega_2)$  is the Fourier transform of the noisy signal,  $\theta_G(\omega_1, \omega_2)$  is the phase

of  $G(\omega_1, \omega_2)$ ,  $F(\omega_1, \omega_2)$  is the Fourier transform of the original signal, and  $V(\omega_1, \omega_2)$  is the Fourier transform of the noise.

Qualitatively, this means that, where the real or imaginary parts of  $F(\omega_1, \omega_2)$  are large relative to  $V(\omega_1, \omega_2)$ , the effect of the noise on the phase will be small, and where the real or imaginary parts of the signal are small relative to the noise, the effect on the phase will be large.

## 3.2 Characterization of Phase-Based Deconvolution

For the same reasons it is difficult to perform a theoretical analysis of spatial domain noise on the Fourier transform of a signal, it is also difficult to analyze what noise in the phase of the Fourier transform of a signal will do to the reconstructed signal. However, noise in the phase does not seem to change the fact of convergence of the algorithm to some solution. What this convergent solution is depends on the size of the internal DFT used by the iteration and the starting point of the iteration. Additionally, the question remains whether the convergent solution is the best representation of the original image, or whether it would be better to stop the iteration at some intermediate point, as is the case with the Richardson-Lucy iteration.

In this section, we will first explain the experimental setup used to produce the reported results. We will then examine the effect of the number of iterations, the effect of the size of the internal DFT used in the iteration, and the effect of the starting point of the iteration.

### 3.2.1 Experimental Setup

These experiments were performed assuming the basic model of image degradation held. That is,

$$g(n_1, n_2) = f(n_1, n_2) * b(n_1, n_2) + v(n_1, n_2) \quad (3.2)$$

where many different images were used for  $f(n_1, n_2)$ ,  $b(n_1, n_2)$  was an 11x11-point Gaussian filter with  $\sigma = 5$ , and  $v(n_1, n_2)$  was zero-mean Gaussian white noise with differing variances. The images used are shown in Fig. 3-1. These cover a variety of image characteristics, and the results can be considered representative. Note that, while the original image aspect ratios have been preserved in Fig. 3-1, the original sizes have not. For example, the original cameraman image was 256x256 pixels, while the original liftingbody image was 512x512 pixels, even though these two appear to be the same size in Fig. 3-1. Also, Fig. 3-1a, 3-1c, and 3-1i are shown at 75% their true resolution for copyright reasons. (See Appendix A for image credits.)

In this chapter, the blurring function and the region of support were assumed known. The effect of loosening these assumptions will be addressed in subsequent chapters.

The quality metric used was a variation of mean squared error, computed as

$$e = \frac{1}{N_1 N_2} \sum_{n_1=0}^{N_1-1} \sum_{n_2=0}^{N_2-1} \left( f(n_1, n_2) - k \hat{f}(n_1, n_2) \right)^2 \quad (3.3)$$

where  $k$  is a scale factor computed to minimize the error, and is given by

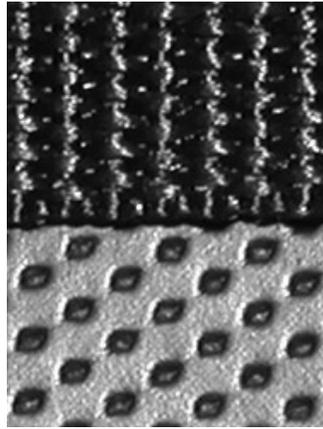
$$k = \frac{\sum_{n_1=0}^{N_1-1} \sum_{n_2=0}^{N_2-1} f(n_1, n_2) \hat{f}(n_1, n_2)}{\sum_{n_1=0}^{N_1-1} \sum_{n_2=0}^{N_2-1} \left( \hat{f}(n_1, n_2) \right)^2} \quad (3.4)$$

This metric, which will be referred to as optimal scaling mean squared error (OS-MSE), was chosen because phase-based reconstruction is only accurate to within a scale factor. It should be noted that the original image would not typically be available in practice, and so  $k$  would have to be estimated. It should also be noted that this error measure is not necessarily an absolute indicator of visual quality, and so sample images will be provided throughout to demonstrate the kinds of artifacts present in the restored images.

Unless stated otherwise, the initial guess of the magnitude of the restored image used by the phase-based reconstruction iteration was simply a constant. In some cases, the phase-based reconstruction iteration led to an intermediate spatial domain



(a) moon (358x537 pixels)



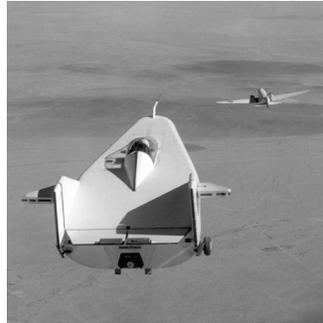
(b) bag (189x250 pixels)



(c) pout (240x291 pixels)



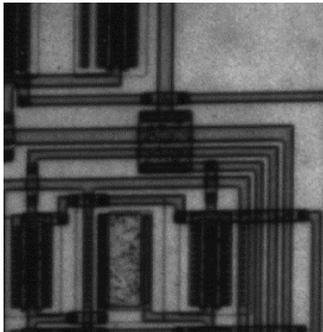
(d) cameraman (256x256 pixels)



(e) liftingbody (512x512 pixels)



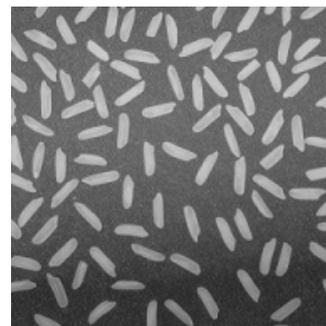
(f) tire (232x205 pixels)



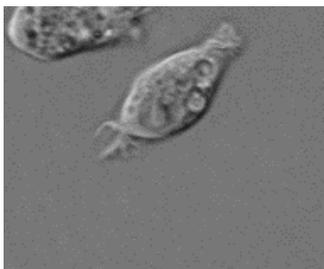
(g) circuit (272x280 pixels)



(h) westconcordorthophoto (364x366 pixels)



(i) rice (256x256 pixels)



(j) cell (191x159 pixels)



(k) coins (300x246 pixels)



(l) eight (308x242 pixels)

Figure 3-1: Original images

image with such small pixel values that the quantization of double precision floating-point numbers became a problem. To avoid this, a check was placed in the iteration to multiply the reconstructed image by a large constant if it ever became too small. Since phase-based reconstruction only gives the result to within a scale factor, this would not change the final result. The alternative would be to compute and scale by  $k$  at each iteration. This was not done to avoid introducing any dependence on the original  $f(n_1, n_2)$  into the iteration.

### 3.2.2 Effect of the Number of Iterations

To find the optimal number of iterations, the original image was degraded as described previously and the phase-based reconstruction algorithm was allowed to run for a certain number of iterations. The OS-MSE between the resulting restored image and the original was computed and saved. This was repeated for several different amounts of noise, for different DFT sizes, and for each image.

In almost every case, more iterations improved the quality of the restored image in the OS-MSE sense until convergence was reached. The results for the tire image with an internal DFT size of twice the image size, shown in Fig. 3-2, were typical. Convergence was found to occur between 500 and 1,000 iterations in general, with the notable exception being the moon image, which was still improving after even 10,000 iterations.

It should be emphasized that there is no theoretical guarantee of convergence in the presence of noise, but all the experiments in this chapter did indeed find some convergent solution except the moon image, which also seemed to be converging even though the experiments were not run long enough to actually reach convergence.

The few cases in which more iterations did not always lead to a lower OS-MSE were the bag, westconcordorthophoto, and coins images at low noise values and the cell image at high noise values. In the case of the bag, westconcordorthophoto, and coins images, the visual quality of the convergent solutions was similar to that of the pre-convergent solutions, and which was considered the better restoration would probably be a matter of application.

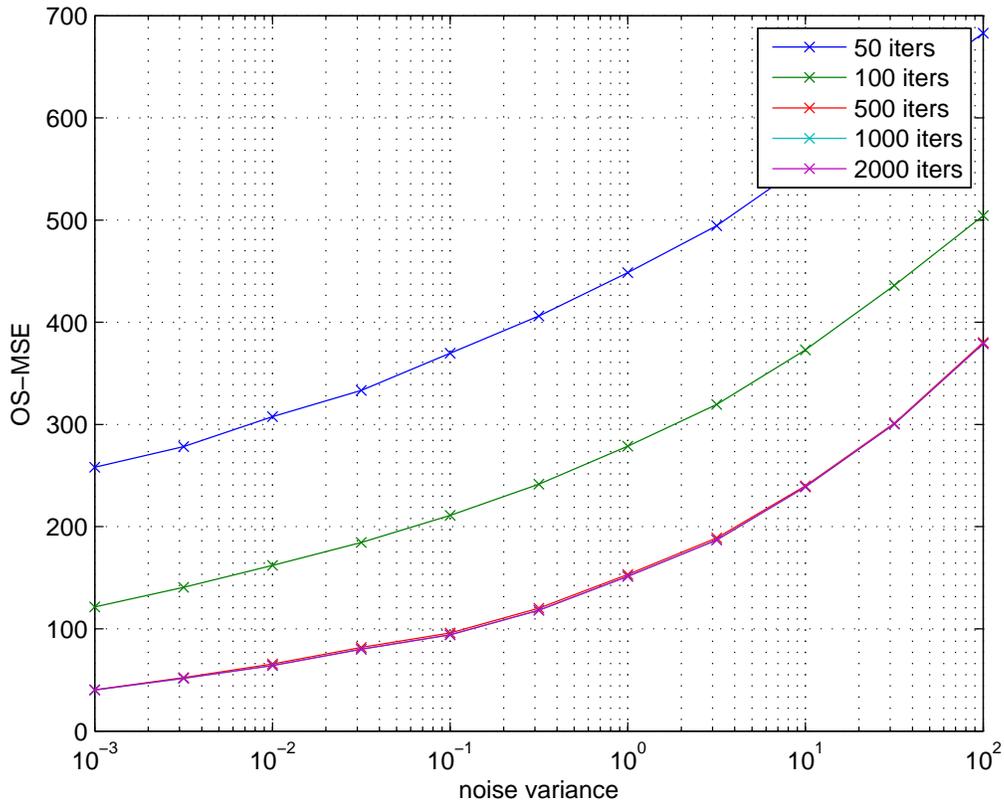
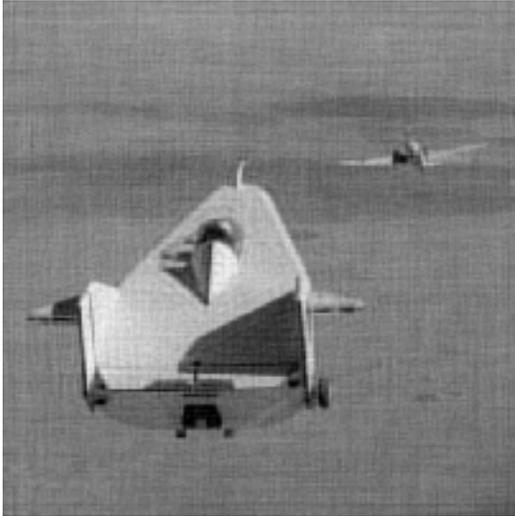


Figure 3-2: OS-MSE as a function of noise variance and number of iterations for the tire image. Internal DFT size was twice the image size.

There are several things to note about this analysis. First, the kind of artifacts that appeared in the convergent solution, namely horizontal and vertical lines, was similar across all images. A few examples are shown in Fig. 3-3. The pre-convergent solutions showed varying artifacts, but graininess was common. As convergence was reached, these artifacts tended to be diminished and the horizontal and vertical lines became better defined. In some cases, the convergent solution appeared somewhat less sharp than the pre-convergent solution. A few examples of this are shown in Figs. 3-4 and 3-5. In these and all subsequent restored images, only the region of support of the original image is shown—the degraded image was slightly larger because of the blurring.

From these results, it is reasonable to conclude that, in general, phase-based deconvolution performs best when run to convergence. Even when this is not the

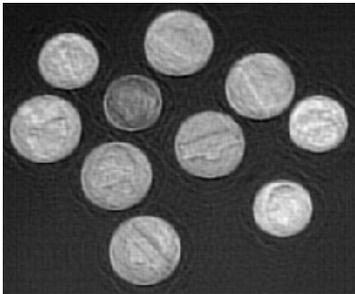


(a) liftingbody, restored

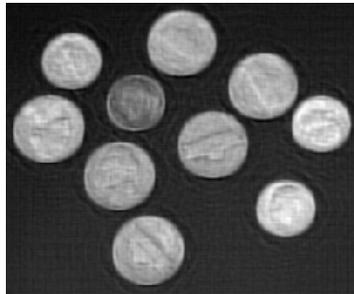


(b) cell, restored

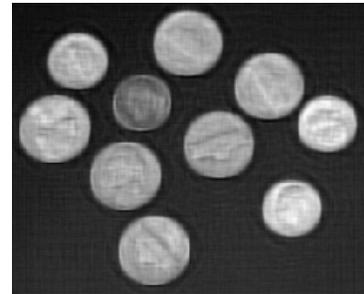
Figure 3-3: Example of artifacts present in the restored image when the algorithm was run to convergence. Noise variance was 1.



(a) 500 iterations



(b) 1000 iterations



(c) 2000 iterations

Figure 3-4: An illustration of the effect of number of iterations on artifacts when stopping short of convergence was optimal. As the number of iterations increases, the lines surrounding the coins decrease and the image becomes less sharp. In this case, the restored image has a slightly higher OS-MSE as the number of iterations increases, but the visual quality is not clearly worse. Noise variance was  $10^{-1.5}$ .

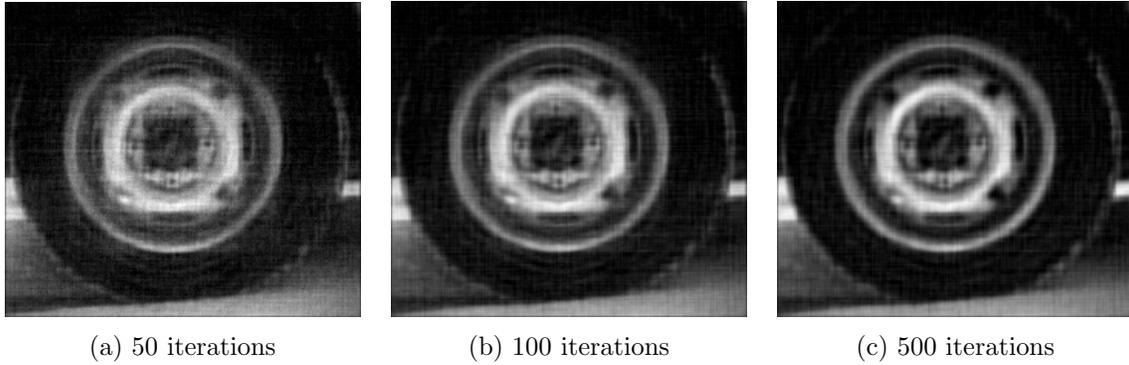


Figure 3-5: An illustration of the effect of number of iterations on artifacts when convergence was optimal. As the number of iterations increases, the graininess decreases and the horizontal and vertical lines become more pronounced. Noise variance was 1.

case in the OS-MSE sense, it may be in the visual quality sense, and any gains to be had by stopping short of convergence are small.

### 3.2.3 Effect of the Size of the DFT

In the absence of noise, there is one correct solution, and the phase-based reconstruction iteration will converge to that solution regardless of the size of the internal DFT, provided it is at least twice the image size, as required by the uniqueness theorem. In the presence of noise, however, a solution that satisfies both the spatial and frequency domain constraints simultaneously may not exist. In this case, convergence indicates a point has been found such that the spatial and frequency constraints are being violated circularly. That is, a frequency domain solution, when corrected with the given phase, produces a spatial domain solution, that, when corrected with the given region of support, has a Fourier transform equal to the original frequency domain solution, and the progress of the algorithm stops. If this is the situation, then the size of the DFT used by the phase-based reconstruction algorithm could change the solution to which the reconstruction algorithm converges.

This does seem to be the case in practice. The images were restored with DFT sizes of 2, 3, 4, and 5 times the corrupted image size. These tests produced no clear winner among the DFT sizes, though in general a DFT size of twice the image size did somewhat worse than the others, and a size of five times the image size often did

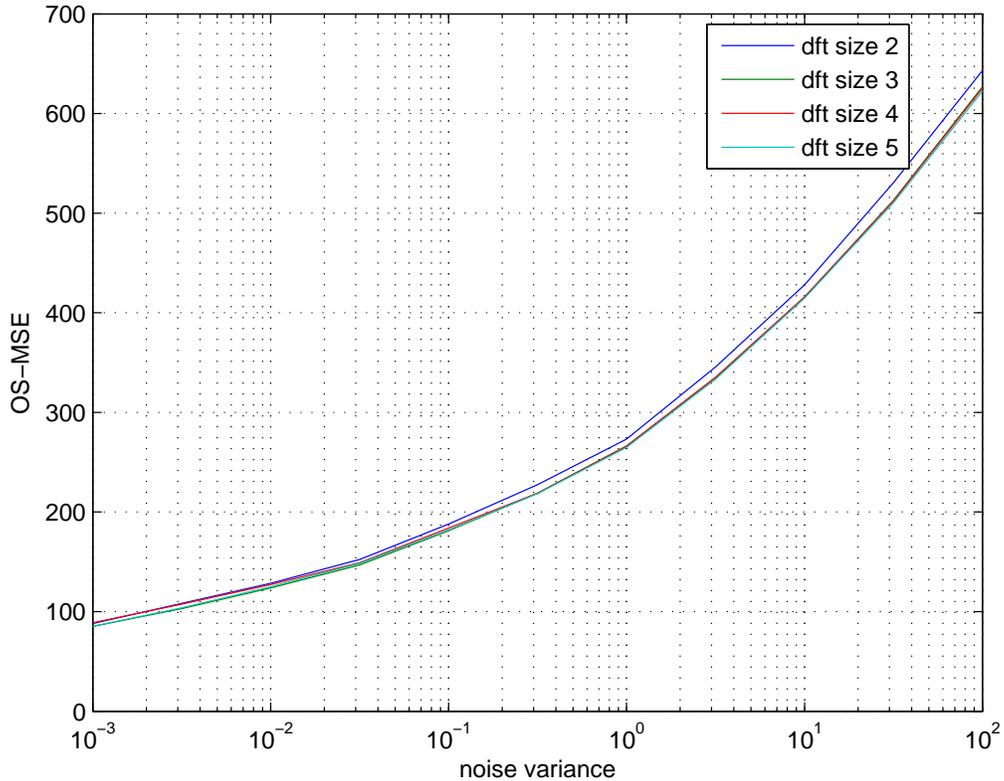


Figure 3-6: OS-MSE as a function of noise variance and DFT size for the cameraman image. Here “dft size 2” indicates the internal DFT size was twice the degraded image size, “dft size 3” indicates three times the degraded image size, and so forth. Results are the convergent solution.

somewhat better. However, the differences were only slight, and it seems there is not much to be gained by changing the DFT size.

The cameraman image was typical, and the results are shown in Figs. 3-6 and 3-7. Fig. 3-6 shows the OS-MSE values of the convergent solutions of each DFT size, and Fig. 3-7 shows the actual images. Careful inspection does indicate that a size of 3, 4, or 5 times the image size is somewhat better than twice the image size, but the gains are small. Nevertheless, phase-based deconvolution is computationally intensive regardless of the DFT size and will be most useful in situations where computational power is not an issue. This being the case, there is no reason not to use a larger DFT to capture whatever gains are available.



(a) DFT size: twice the image size



(b) DFT size: three times the image size



(c) DFT size: four times the image size



(d) DFT size: five times the image size

Figure 3-7: Illustration of the effect of DFT size on visual quality. Noise variance was 1.

### 3.2.4 Effect of the Starting Point

Although the previous experiments were performed with a constant as the initial guess of the image magnitude, there is no reason this must be the case. In the absence of noise, the algorithm will eventually converge to the correct solution from any non-zero starting point, and, in the presence of noise, the starting point may affect the solution to which the iteration converges. A constant is one convenient choice, but there are other natural options. For example, the starting point could be the magnitude of the degraded image.

Experiments performed from this starting point gave, in general, very similar convergent results to those produced by the previous experiments. Improvements

achieved by changing the starting point were slight, and which starting point was better was a function of both the image being restored and the noise level with which it was corrupted, with no clear pattern emerging.

The exception to this rule was the moon image, where the results obtained by starting with the degraded image were much better than those obtained by starting with a constant. This is likely because the phase-based reconstruction algorithm did not reach convergence from either starting point. It was frequently the case that the pre-convergent solutions obtained by starting with the degraded image were better than those obtained by starting with a constant, even though the convergent results were similar and the starting point did not seem to affect how many iterations were required for convergence.

The general trends that held when the starting point was a constant also held when the starting point was the degraded image. That is, the convergent solution was typically the best and an internal DFT size of at least three times the degraded image size frequently did better than an internal DFT size of only twice the degraded image size. There were, however, several situations in which these general rules did not hold. For example, in some of the images, at high noise values, fewer iterations became better than the convergent solution. This is likely because the phase was sufficiently noisy and the initial guess of the magnitude was sufficiently accurate that attempting phase-based reconstruction of the magnitude actually produced a magnitude that was more corrupted than the original. The liftingbody image was representative of this class, and results are shown in Fig. 3-8.

This was further confirmed by experiments in which the original, undegraded image magnitude was used as the starting point. In this case, fewer iterations always gave a better result. A particularly dramatic example of this was the bag image. The results are shown in Fig. 3-9. Again, this is reasonable—a correct initial magnitude can only be corrupted by a noisy phase during reconstruction. (Interestingly, the convergent solution was comparable to that obtained when starting with a constant or the degraded image.)

From this, it is reasonable to conclude that, if the phase-based reconstruction

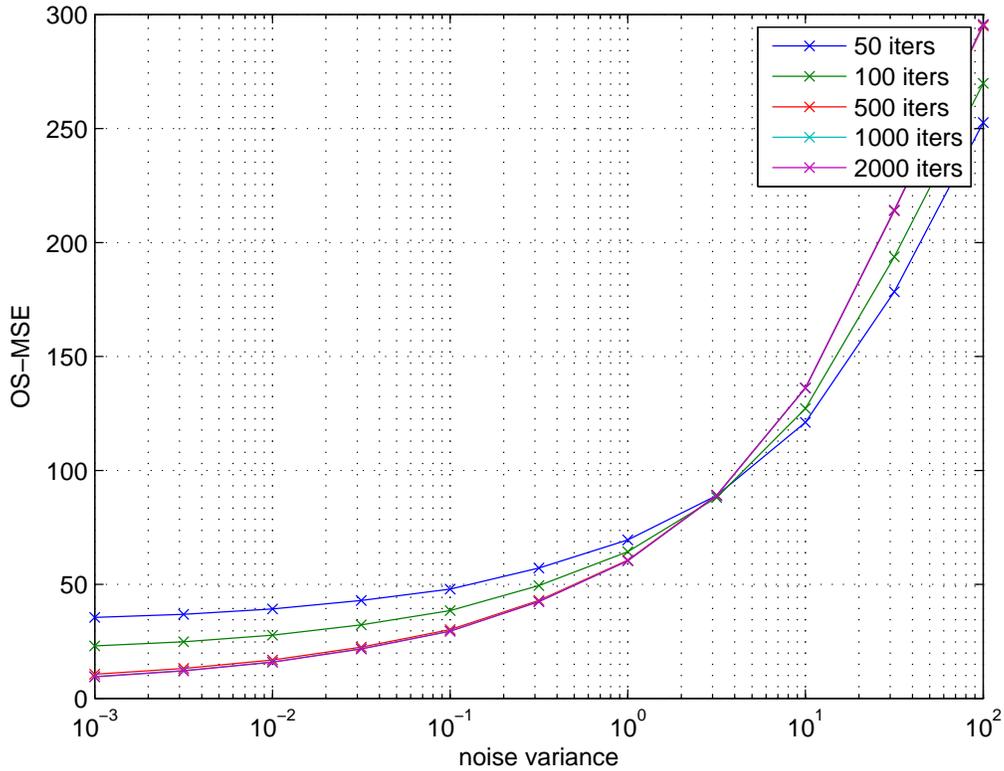


Figure 3-8: OS-MSE as a function of noise variance and number of iterations for the liftingbody image when the iteration was initialized with the degraded image magnitude. The internal DFT size was twice the image size.

iteration is run to convergence, the starting point doesn't matter. However, if a sufficiently accurate initial guess of the magnitude is paired with a sufficiently noisy phase, there are some gains to be had by stopping before convergence, though what is "sufficiently accurate" and "sufficiently noisy" will have to be determined in an ad-hoc manner and may require some trial and error. Additionally, if, because of computing constraints, the iteration cannot be run to convergence, starting with the degraded image will likely give a better result and is the recommended procedure.

In light of this, the remainder of this chapter will continue to use results obtained from a constant initial magnitude. Though in certain situations some ad-hoc improvements may be possible, these will not be explored. The focus of the discussion is on what can be achieved based on phase information, and starting with a constant both eliminates the need for magnitude information and provides a consistent starting

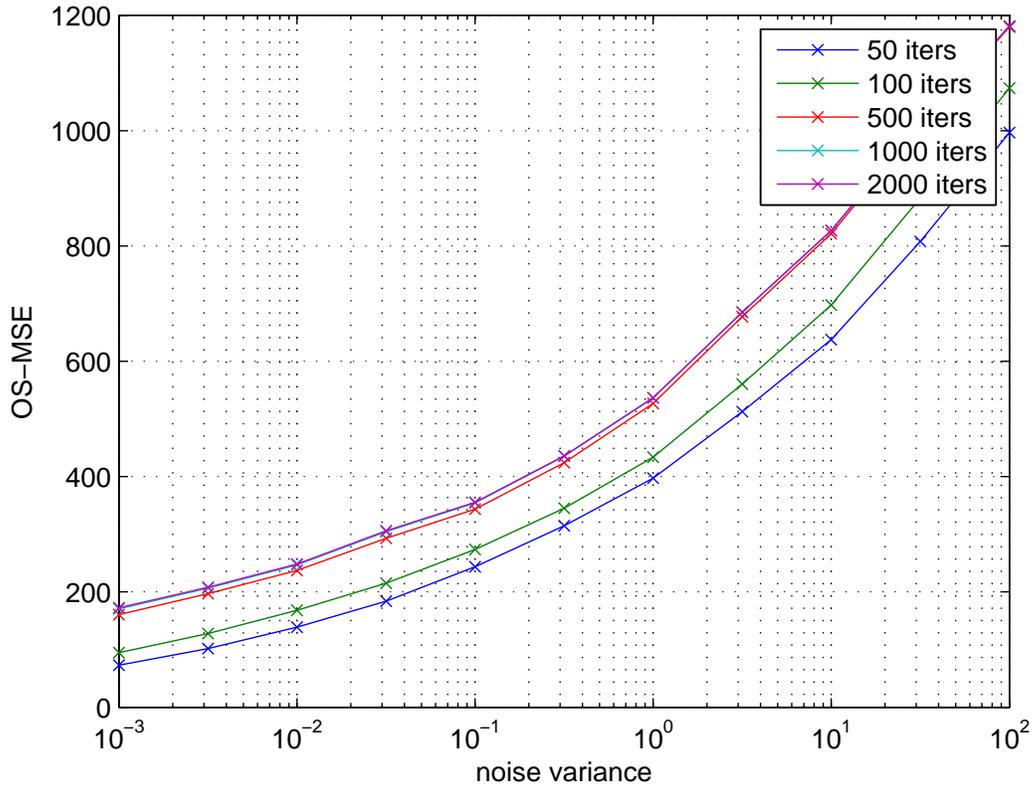


Figure 3-9: OS-MSE as a function of noise variance and number of iterations for the bag image when the iteration was initialized with the undegraded image magnitude. The internal DFT size was twice the image size.

point for all images.

### 3.3 Comparison to Other Methods

Phase-based deconvolution is, strictly speaking, an inverse filtering technique and not an entire image restoration system. That is to say, it makes no attempt to account for or correct noise in the image. Therefore, the proper comparison is to other inverse filtering techniques. However, phase-based deconvolution has proven to be so robust in the presence of noise that it can also be compared to full image restoration algorithms that do account for noise.

The methods included in these comparisons are not an exhaustive list of the possibilities—there are far too many algorithms. The comparisons provided are only

meant to serve as a dramatic demonstration of the result that phase-based deconvolution is extremely robust to noise. They do not imply that phase-based deconvolution should be used as an image restoration system.

In all cases, the phase-based reconstruction iteration was initialized with a constant and run to convergence with an internal DFT that was five times the degraded image size.

### 3.3.1 Comparison to Inverse Filtering Techniques

There were three “inverse filtering” methods considered, namely:

*Direct Inverse Filtering*, simply the inverse DFT of the  $N_1 \times N_2$  point DFT of the degraded image divided by the  $N_1 \times N_2$  point DFT of the blurring filter, where the degraded image was  $N_1 \times N_2$  pixels.

*Thresholded Inverse Filtering*, where the inverse filter was limited to have a magnitude of no more than 1,000 at all points.

*No Processing*, where no processing was performed on the degraded image.

Phase-based deconvolution performed much better than either direct or thresholded inverse filtering in all cases, and was usually better than doing no processing, though at high noise levels the moon, pout, liftingbody, cell, and eight images were slightly better left alone. (The moon image had not reached convergence when the test ended, and so further improvement was probably possible if it had been allowed to run for more iterations.)

The results for the cameraman and eight images were typical and are shown in Fig. 3-10. Visual artifacts in the cameraman image are shown in Fig. 3-11.

### 3.3.2 Comparison to Image Restoration Techniques

The results of phase-based deconvolution will now be compared with the results given by several methods of image restoration as a further demonstration of phase-based deconvolution’s robustness in the presence of noise. The methods used are

*Wiener Filtering*, with the power spectral densities of the noise and signal assumed

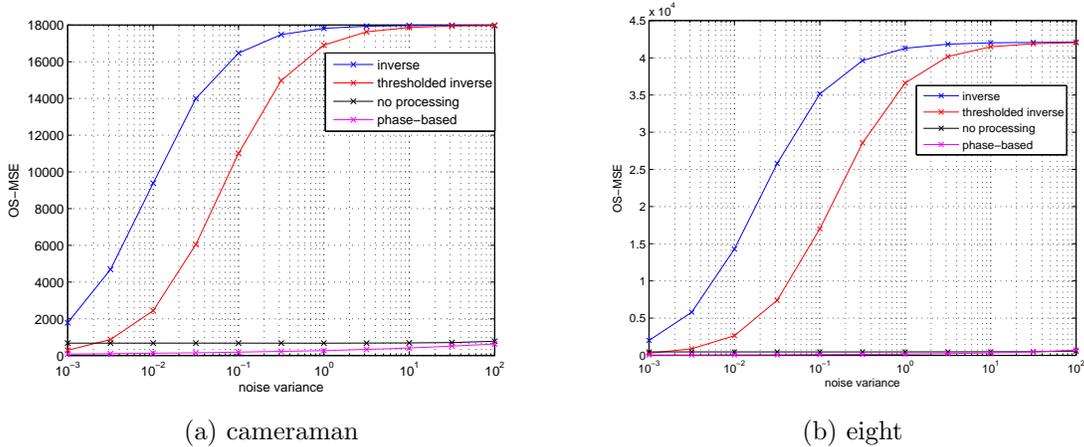


Figure 3-10: Comparison of deconvolution methods

to be constants, so that  $\frac{S_v(\omega_1, \omega_2)}{S_f(\omega_1, \omega_2)}$  was just the noise to signal ratio. Noise statistics were assumed known and the noise to signal ratio was estimated from the corrupted image.

*Regularized Filtering*, with the noise statistics assumed known and the Laplacian operator used for smoothing.

*Richardson-Lucy Iteration*, taking into account white noise, with the noise statistics assumed known. The original (undegraded) image was used to determine the first local minimum in OS-MSE as a function of the number of iterations for each situation, and that number of iterations was used for that situation.

In general, phase-based deconvolution did better than Wiener filtering and was comparable to the Richardson-Lucy iteration and regularized filtering in the OS-MSE sense at low noise values. It tended to do somewhat worse than the other methods at high noise values. The results for the liftingbody, cameraman, and bag images, shown in Fig. 3-12, were representative.

Visual quality tracked OS-MSE fairly well in this test, but different restoration methods produced different artifacts, and which artifacts were considered most unpleasant would probably depend on the application. The cameraman image was typical, and some sample images are shown in Fig. 3-13.

It is worth emphasizing again that, unlike phase-based deconvolution, all other methods require some knowledge of the noise statistics. Because other methods try to



(a) no processing



(b) direct inverse filtering



(c) thresholded inverse filtering



(d) phase-based deconvolution

Figure 3-11: Cameraman image deconvolved with several different methods. Noise variance was  $10^{-2}$ .

correct for the presence of noise and phase-based deconvolution does not, phase-based deconvolution is most competitive at high signal to noise ratios and becomes worse as the noise increases.

Additionally, it should be noted that the original image was used to determine how long the Richardson-Lucy iteration should run. In practice, the original image would not be available and the number of iterations would have to be estimated. This means that results from that iteration will not typically be quite as good as shown here.

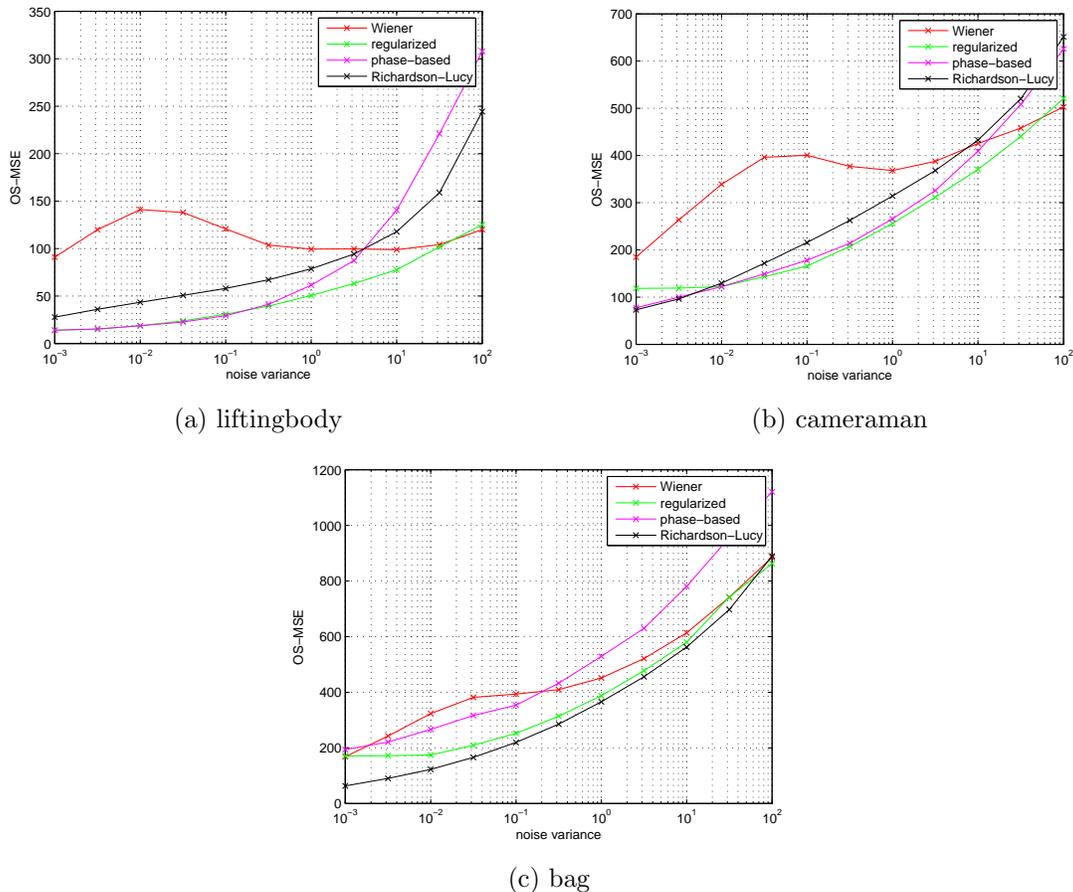


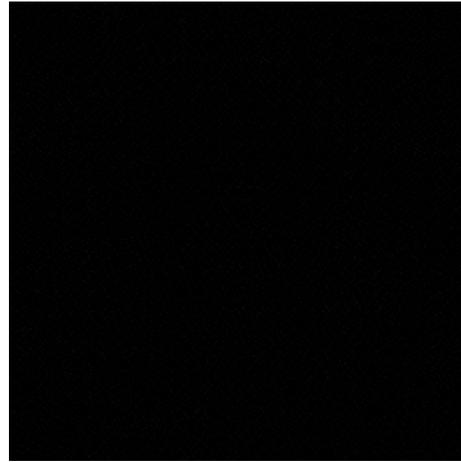
Figure 3-12: Comparison of restoration methods

### 3.4 Summary

In this chapter, we have shown that phase-based deconvolution is robust to the presence of noise. When initialized with a constant, the phase-based reconstruction iteration performs best when run to convergence, though there are minor exceptions to this rule. Larger internal DFT sizes also tend to perform better. There are some ad-hoc gains possible if a “noisy enough” phase is paired with an “accurate enough” estimate of the magnitude and the iteration is stopped short of convergence, but these may be hard to realize in practice. Phase-based deconvolution performs much better than other deconvolution methods in the presence of noise, and is competitive with restoration methods at low noise levels.



(a) degraded



(b) direct inverse filtering



(c) Wiener filtering



(d) regularized filtering



(e) Richardson-Lucy iteration



(f) phase-based deconvolution

Figure 3-13: Cameraman image restored with several different methods. Noise variance was 1.

# Chapter 4

## Performance with Imperfect Knowledge of the Blurring Function

### 4.1 Attributes of Blurring Functions

Blurring functions are, by definition, two-dimensional lowpass filters. Additionally, a large class of blurring functions are symmetric (for example, those caused by lens misfocus [9]), and so their phases are limited to 0 or  $\pi$ . Among such functions, the Fourier transforms are sinc-like. For example, the 512x512-point Fourier transform of an 11x11-point Gaussian filter with  $\sigma = 5$  (the blurring function used in the previous chapter) is shown in Fig. 4-1.

It is generally the case with such functions that the number of points in the filter determines the width of the main lobe, while the shape, or taper, of the filter merely changes the height of the side lobes. This has significant implications for the phase of the Fourier transforms of such filters. Specifically, the phases of two filters that are the same size will be very similar regardless of their respective tapers. This is illustrated dramatically in Fig. 4-2, which shows an 11x11-point Gaussian filter with  $\sigma = 5$ , an 11x11-point averaging filter, and the difference between the phases of their

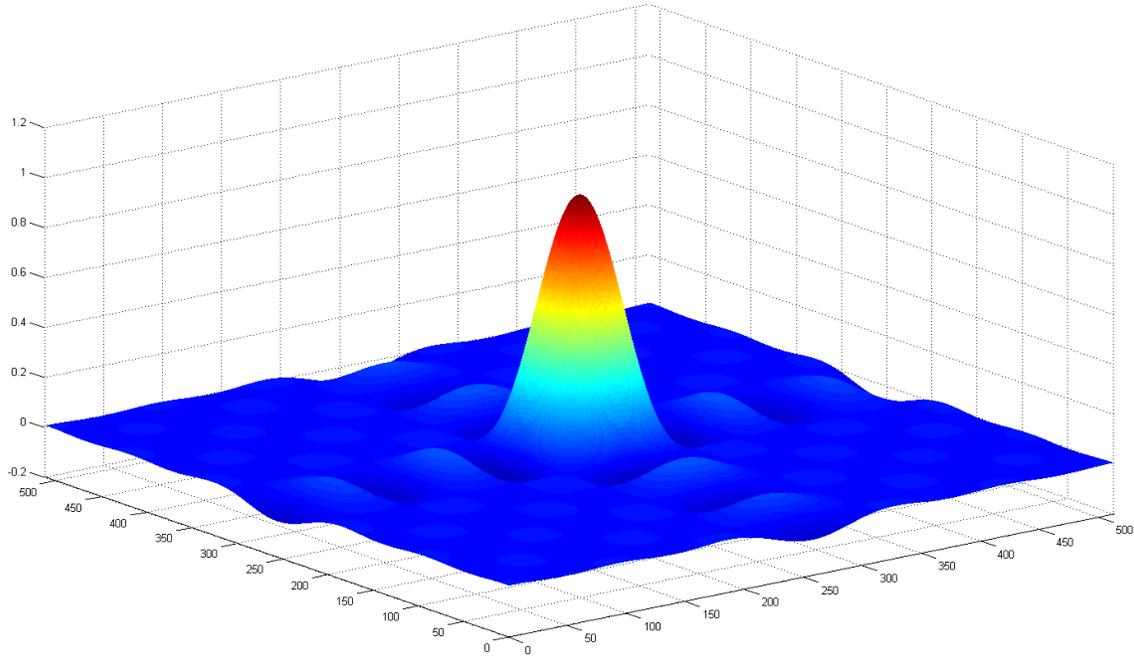


Figure 4-1: Fourier transform of an 11x11-point Gaussian filter with  $\sigma = 5$

Fourier transforms.

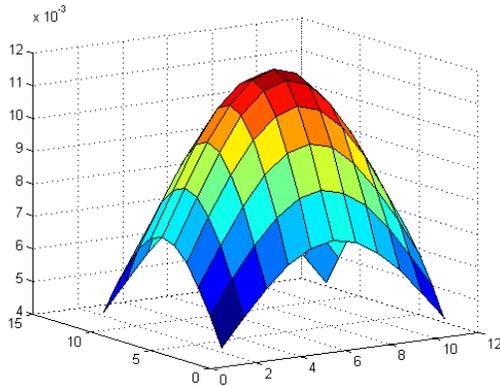
In practice, situations arise in which the blurring function is not known perfectly, but a reasonable guess can be obtained. Because of the similarity in phase across many different blurring functions of the same size, it seems likely that phase-based deconvolution will be fairly robust to imperfect knowledge of the blurring function.

In the remainder of this chapter, we will characterize the performance of phase-based deconvolution when the blurring function is known imperfectly and compare phase-based deconvolution to other deconvolution and restoration methods.

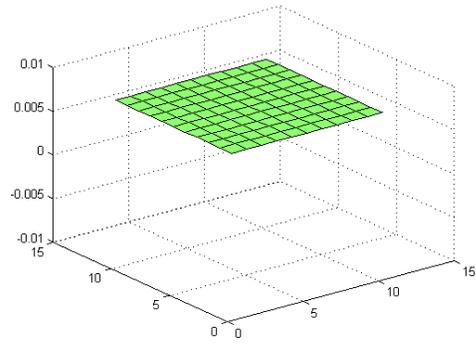
## 4.2 Characterization of Phase-Based Deconvolution

### 4.2.1 Experimental Setup

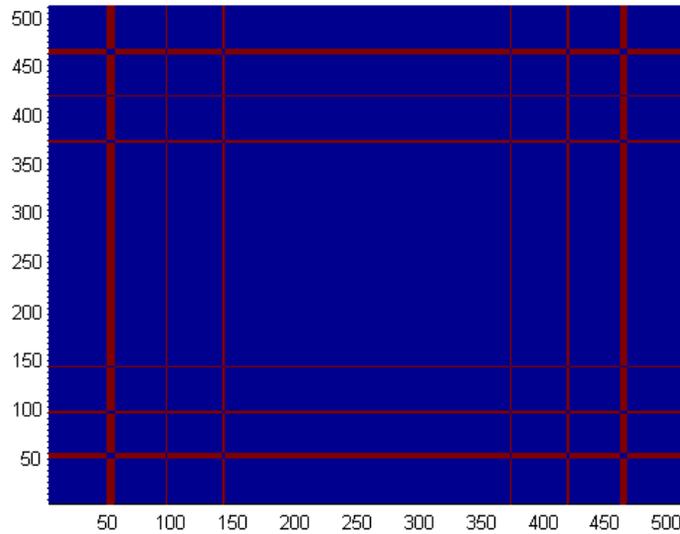
The experimental setup in this chapter will be similar to that described in Chapter 3. Initially, the additive noise will be assumed to be zero so the effects of an incorrect



(a) 11x11-point Gaussian filter with  $\sigma = 5$



(b) 11x11-point averaging filter



(c) Difference in the angles of the 512x512-point Fourier transforms of the two filters. Blue regions indicate the angles were the same, while red indicates they differed by  $\pi$ .

Figure 4-2: Comparison of the phases of two lowpass filters

guess of the blurring function can be seen clearly. Noise will be added in again at the end of the chapter to increase the realism of the model.

The reference images of Fig. 3-1 will still be used and the blurring function will still be the 11x11-point Gaussian filter with  $\sigma = 5$ . As in Chapter 3, the region of support will be assumed known and the error criterion will be the optimal scaling mean squared error defined in Eqs. 3.3 and 3.4.

There will be two deblurring functions considered: An 11x11-point averager and a 9x9-point averager. These were chosen because an averager, being a constant over its region of support, is the simplest blurring function possible, and therefore would

be a natural initial guess of an unknown blurring function. Comparing the results obtained with these two averagers will also serve to highlight the relative importance of the size of the blurring function and the relative unimportance of its taper to phase-based deconvolution.

All blurring and deblurring functions considered will be centered at the origin. Any shift in the blurring function away from the origin merely introduces a linear phase term, which, assuming it is known, can be easily accounted for after deconvolution.

As in Chapter 3, the effect of the number of iterations, size of the internal DFT, and starting point will be considered in turn. It is necessary to run these experiments again because in Chapter 3 the errors in the phase were of random size and direction, having the greatest effect on those frequencies for which the signal to noise ratio of the degraded image was low. In contrast, when the errors in phase are due to imperfect knowledge of the blurring function, they will be fewer but more dramatic (those phases which are wrong will be wrong by  $\pi$ ) and occur in a pattern determined by the blurring and deblurring functions. There is, therefore, no particular reason to expect the results of Chapter 3 to hold.

### 4.2.2 Effect of the Number of Iterations

As with noisy signals, there is no theoretical guarantee of convergence when the deblurring filter is not the same as the blurring filter. However, all the experiments in this chapter did exhibit convergent behavior. Unlike the results obtained from signals corrupted with additive white Gaussian noise, however, the convergent solution was not always the optimal solution.

When the deblurring filter was an 11x11-point averager, the preconvergent solutions were often sharper than the convergent solutions, though they were also grainier and had more pronounced ringing. Whether the effect of the increased sharpness or the effect of the increased graininess and ringing dominated depended on the image being restored. In the case of the moon, pout, liftingbody, circuit, coins, and eight images, the convergent solution was the best in the OS-MSE sense. For the other images, there was a preconvergent solution, typically around a few hundred iterations,

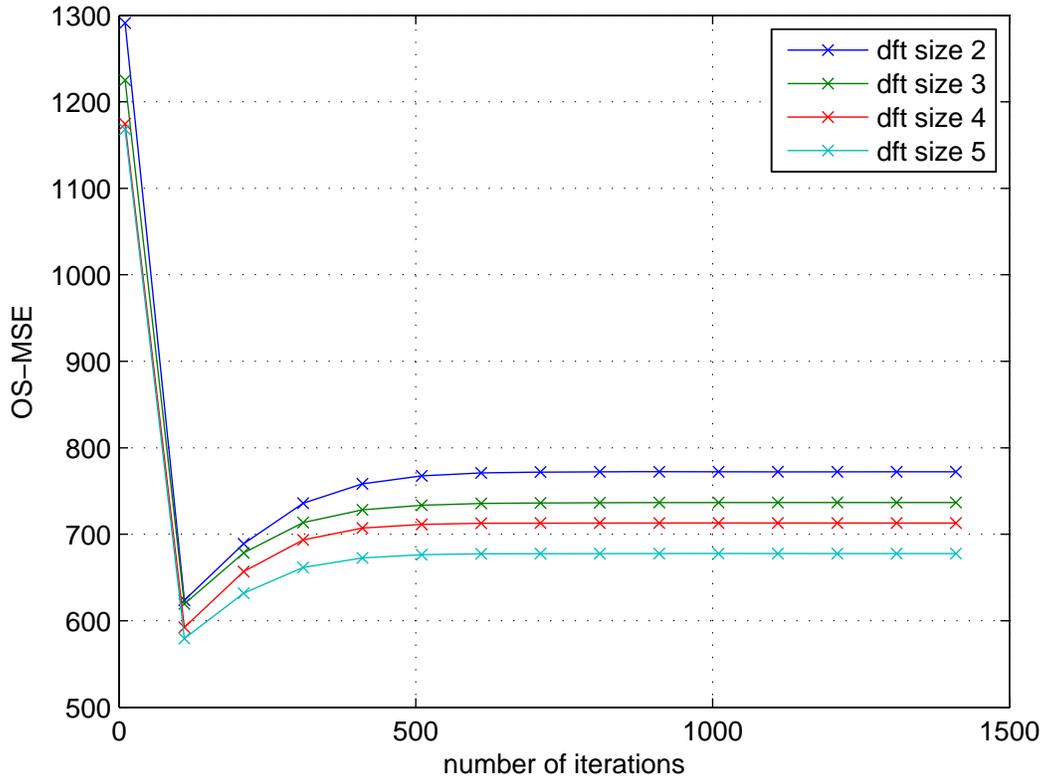


Figure 4-3: OS-MSE as a function of number of iterations and DFT size for the bag image when the deblurring function was an 11x11-point averager

that was at least slightly better than the convergent solution in the OS-MSE sense.

The bag image was the most dramatic example of a pre-convergent solution being better than a convergent solution, and results are shown in Figs. 4-3 and 4-4. (The effects of different DFT sizes will be discussed in the next subsection, and only the curve corresponding to an internal DFT size of twice the image size in Fig. 4-3 need be considered here.) The results for the cameraman image were more typical and are shown in Figs. 4-5 and 4-6. The liftingbody image was representative of the images for which the convergent solution was the best, and results are shown in Fig. 4-7.

When the deblurring filter was a 9x9-point averager, there was an optimal pre-convergent solution for the pout, cell, coins, eight, and tire images. As expected, in all cases the results were worse than what was obtained when using an averager of the correct size. The liftingbody image was representative of those images for which the convergent solution was optimal, and the results are shown in Fig. 4-8.

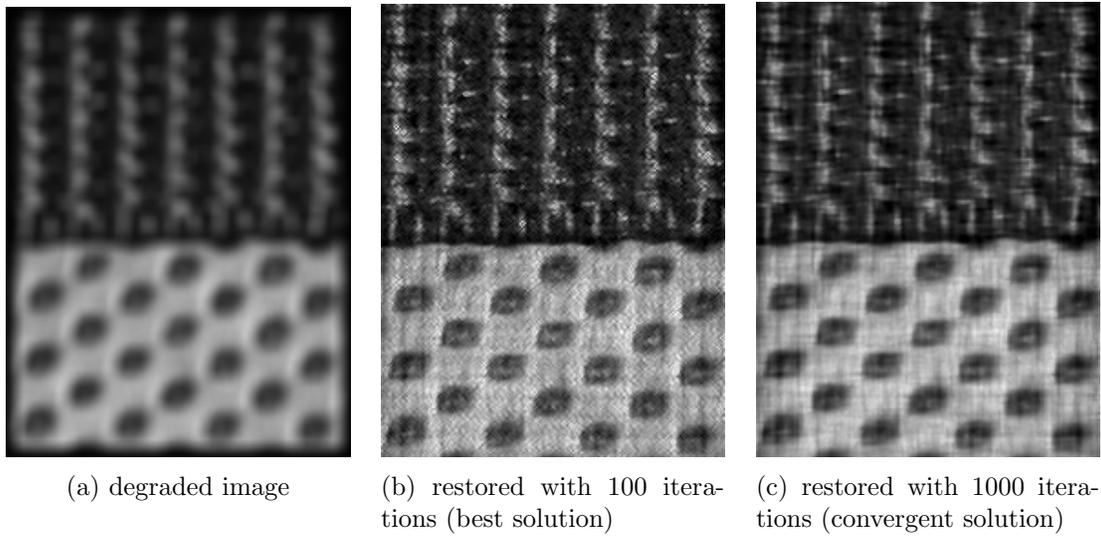


Figure 4-4: Illustration of the artifacts produced in the bag image as the phase-based reconstruction iteration converged when the blurring filter was an 11x11-point Gaussian,  $\sigma = 5$  and the deblurring filter was an 11x11-point averager. In this example, a pre-convergent solution is optimal. Internal DFT size was twice the image size.

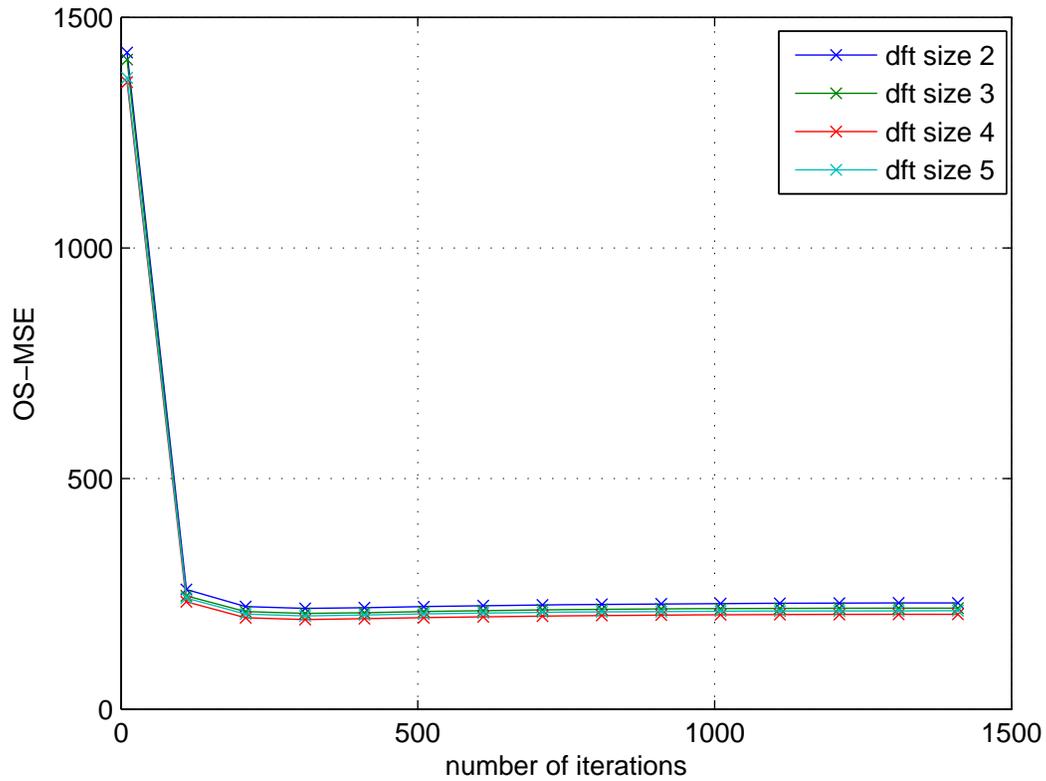


Figure 4-5: OS-MSE as a function of number of iterations and DFT size for the cameraman image when the deblurring function was an 11x11-point averager

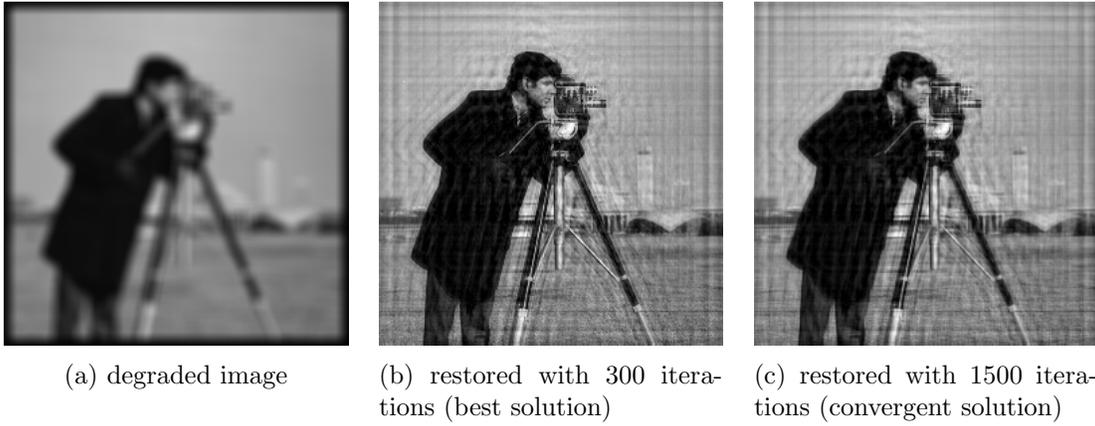


Figure 4-6: Illustration of the artifacts produced in the cameraman image as the phase-based reconstruction converged when the blurring filter was an  $11 \times 11$ -point Gaussian,  $\sigma = 5$  and the deblurring filter was an  $11 \times 11$ -point averager. In this example, a preconvergent solution is optimal. Internal DFT size was twice the image size.

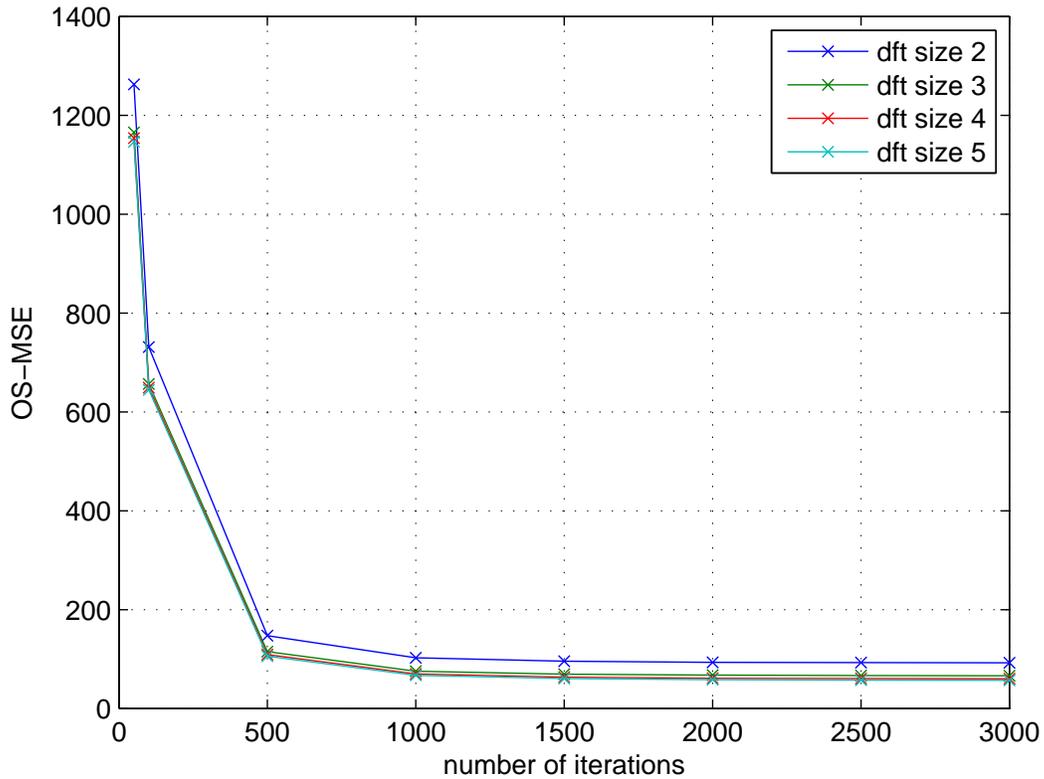


Figure 4-7: OS-MSE as a function of number of iterations and DFT size for the liftingbody image when the deblurring function was an  $11 \times 11$ -point averager

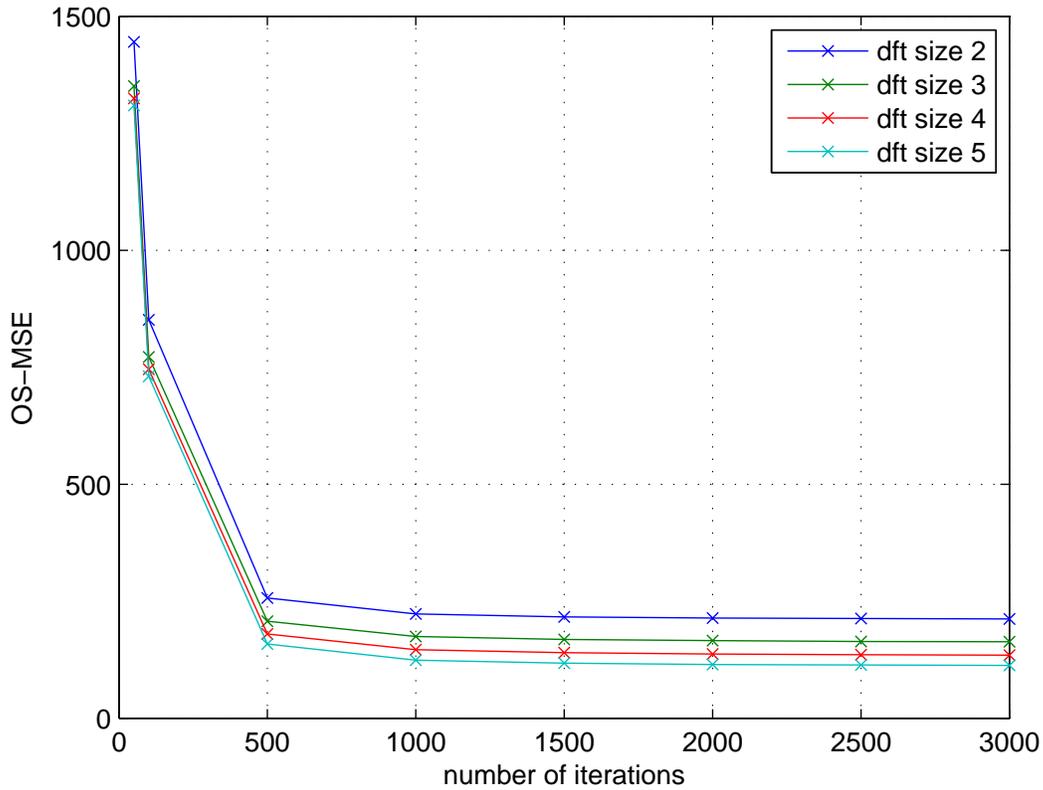


Figure 4-8: OS-MSE as a function of number of iterations and DFT size for the liftingbody image when the deblurring function was a 9x9-point averager

These results lead to several observations. First, if the texture of the original image is unknown, it will be difficult to determine to what extent the graininess in the restored image (in either the convergent or preconvergent solutions) is an artifact of restoration. Therefore, it may be difficult to know when to stop the iteration for an optimal result. Relatedly, this may be a situation in which OS-MSE is not necessarily an accurate indicator of visual quality. The “correct” balance of image sharpness and graininess or ringing is really a matter of application, and so OS-MSE is only a coarse indicator of quality.

Nevertheless, when the deblurring filter was the correct size, both the preconvergent and convergent restored images were much sharper and showed much more detail than the blurred image itself. This indicates that it would be reasonable, when dealing with uncertainty in the taper of the blurring function, to run the phase-based

deconvolution algorithm for a few different numbers of iterations (including one that produces the convergent solution) and choose the result that looks the best. While this procedure is unlikely to produce the optimal solution in the OS-MSE sense, it is very likely to give a result that is better than the degraded image.

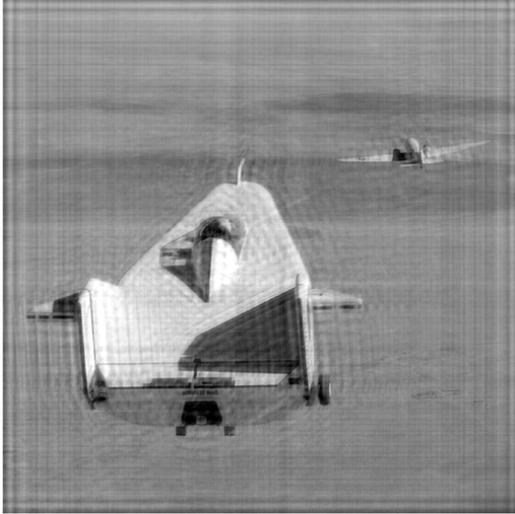
### **4.2.3 Effect of the Size of the DFT**

When the deblurring filter was either an 11x11-point or a 9x9-point averager, the size of the DFT used by the iteration did not have much of an effect on the final result. As a general trend, a larger DFT size was slightly better, and the most dramatic jump was from a DFT size of twice the degraded image size to a DFT of three or more times the image size, as illustrated in Figs. 4-7 and 4-8. The corresponding restored images when the deblurring filter was an 11x11-point averager are shown in Fig. 4-9. Careful inspection does reveal some slight variations in the ringing patterns, but, above a DFT size of twice the image size, it is difficult to say which is the best solution, though by OS-MSE, quality does improve with size in this case.

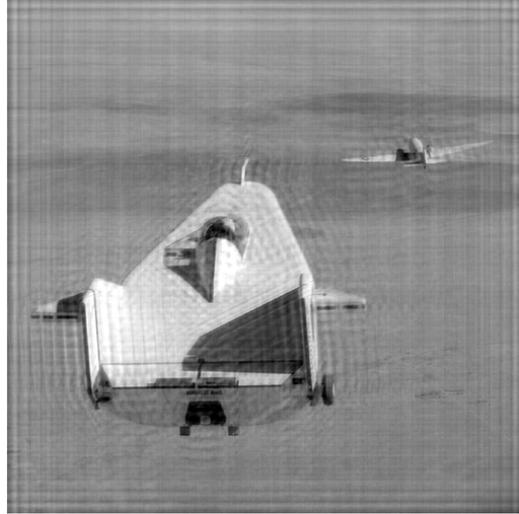
Again, phase-based deconvolution is computationally intensive regardless of the DFT size and will be most useful in situations where computational power is not an issue. This being the case, there is no reason not to use a larger DFT size to capture whatever gains are available.

### **4.2.4 Effect of the Starting Point**

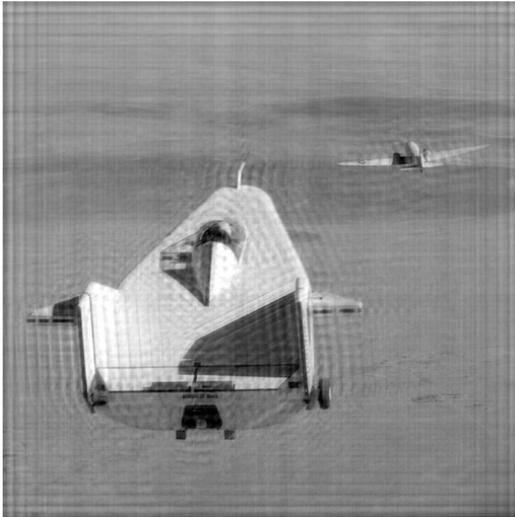
Experiments performed with the degraded image as the starting point for the phase-based deconvolution iteration yielded some interesting results when the deblurring filter was an 11x11-point averager. In every case but the cell image, more iterations led to a lower error, so the convergent solution was always the best. Additionally, for any given internal DFT size, the iteration converged to the same solution whether it was initialized with a constant or with the degraded image. These two observations together indicate the potential gains from stopping the iteration early when it is started with a constant cannot be realized when it is started with the degraded image.



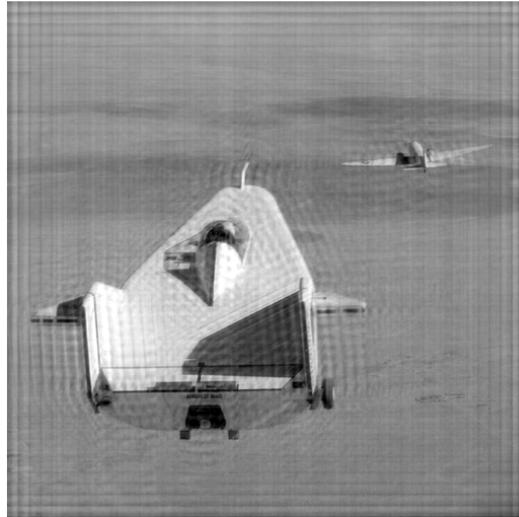
(a) DFT size: twice the image size



(b) DFT size: three times the image size



(c) DFT size: four times the image size



(d) DFT size: five times the image size

Figure 4-9: Illustration of the effect of DFT size on visual quality when the deblurring filter was an 11x11-point averager. Images correspond to the the convergent solutions.

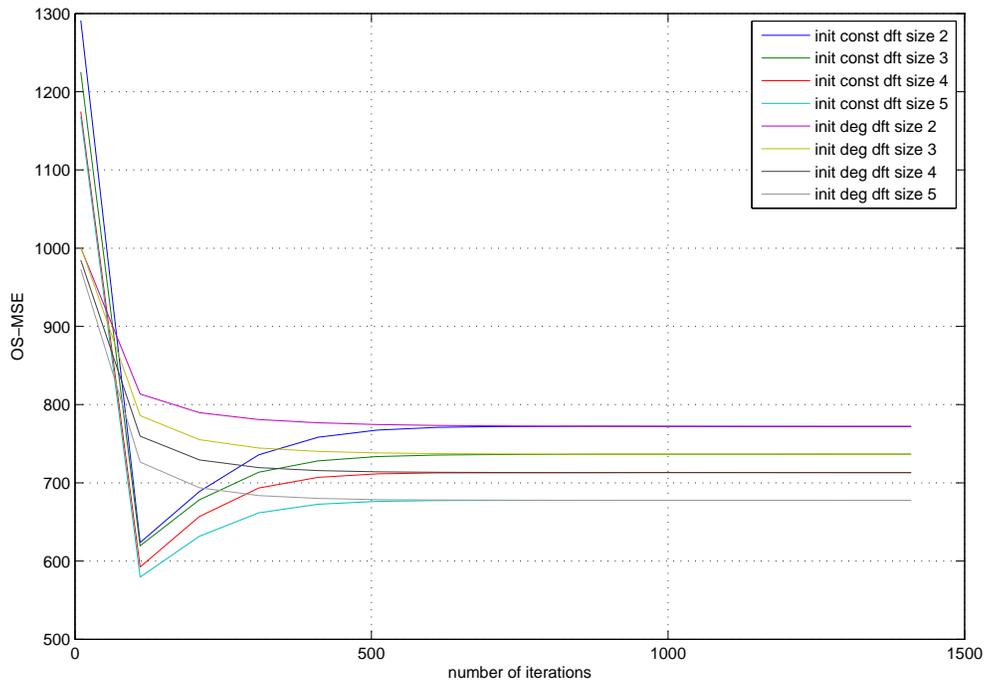
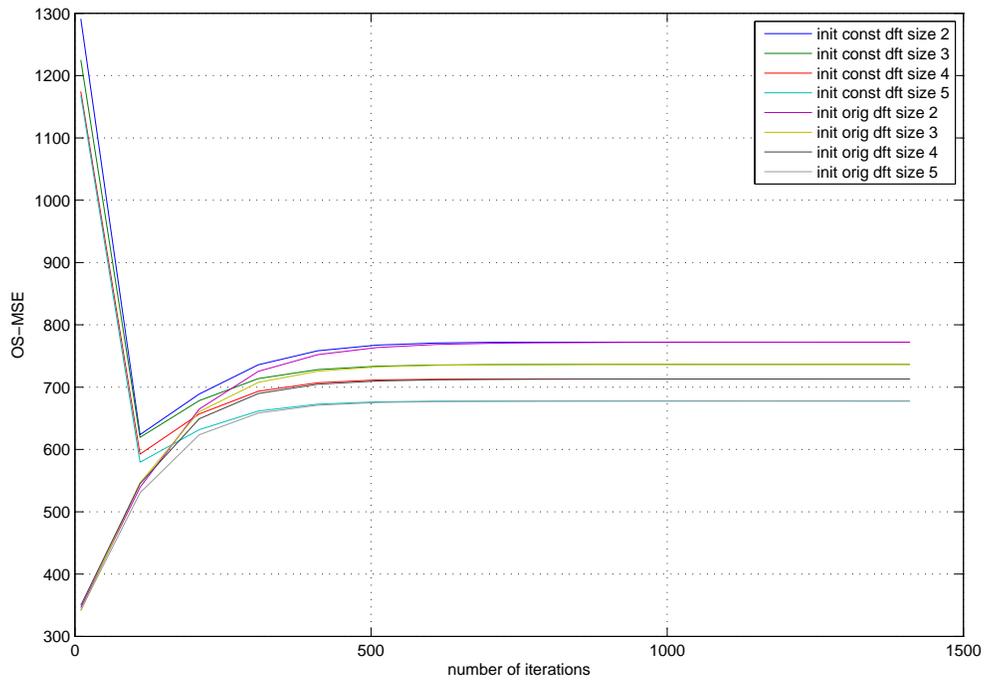


Figure 4-10: OS-MSE as a function of number of iterations, DFT size, and starting point for the bag image. Starting points used are a constant (“init const”) and the degraded image magnitude (“init deg”). The deblurring filter was an 11x11-point averager. Notice how, for any given DFT size, the solutions converge to the same answer, regardless of starting point.

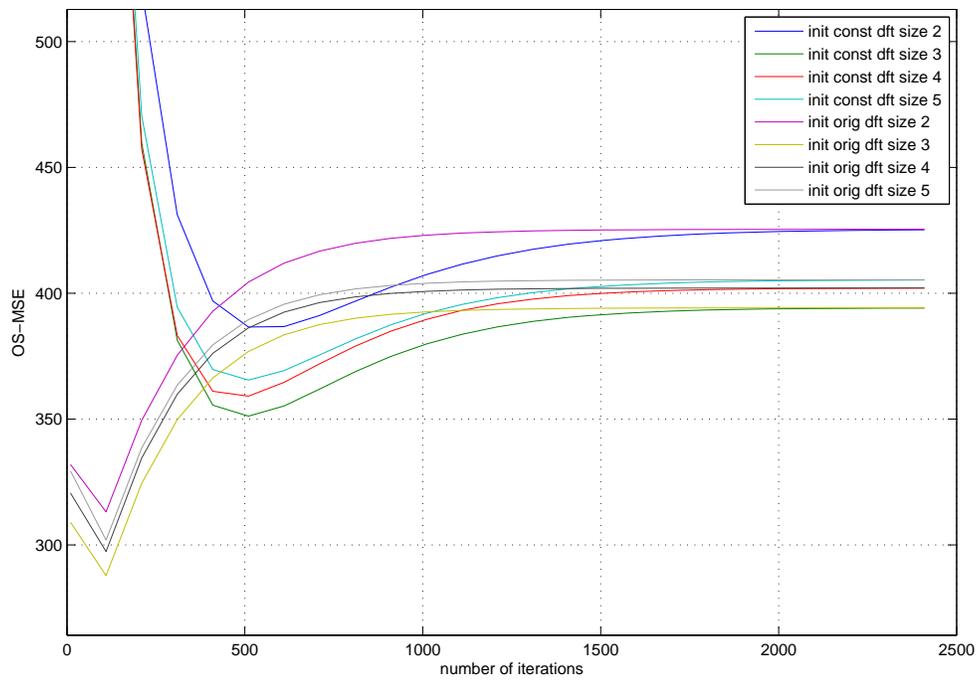
These features are illustrated in Fig. 4-10. Because the convergent solutions are the same and the pre-convergent gains are unavailable when starting with the degraded image, there seems to be no advantage to doing so.

When the correct image magnitude was used as the starting point, it was almost always the case that a low number of iterations (about 100 or less) gave the best results. This is likely the same effect seen with the deconvolution of noisy images: An “accurate enough” estimate of the magnitude is only corrupted by attempting phase-based reconstruction with a “noisy enough” phase. Interestingly, the convergent solutions were again the same as those obtained when starting with a constant, though convergence was typically reached much sooner. This is illustrated in Fig. 4-11.

Results were similar when the deblurring filter was a 9x9-point averager, but, when the degraded image magnitude was used as the starting point, fewer iterations were better for the cell and eight images, and the optimal number of iterations depended



(a) bag



(b) westconcordorthophoto

Figure 4-11: OS-MSE as a function of number of iterations, DFT size, and starting point for the bag and westconcordorthophoto images. Starting points used are a constant (“init const”) and the original image magnitude (“init orig”). The deblurring filter was an 11x11-point averager.

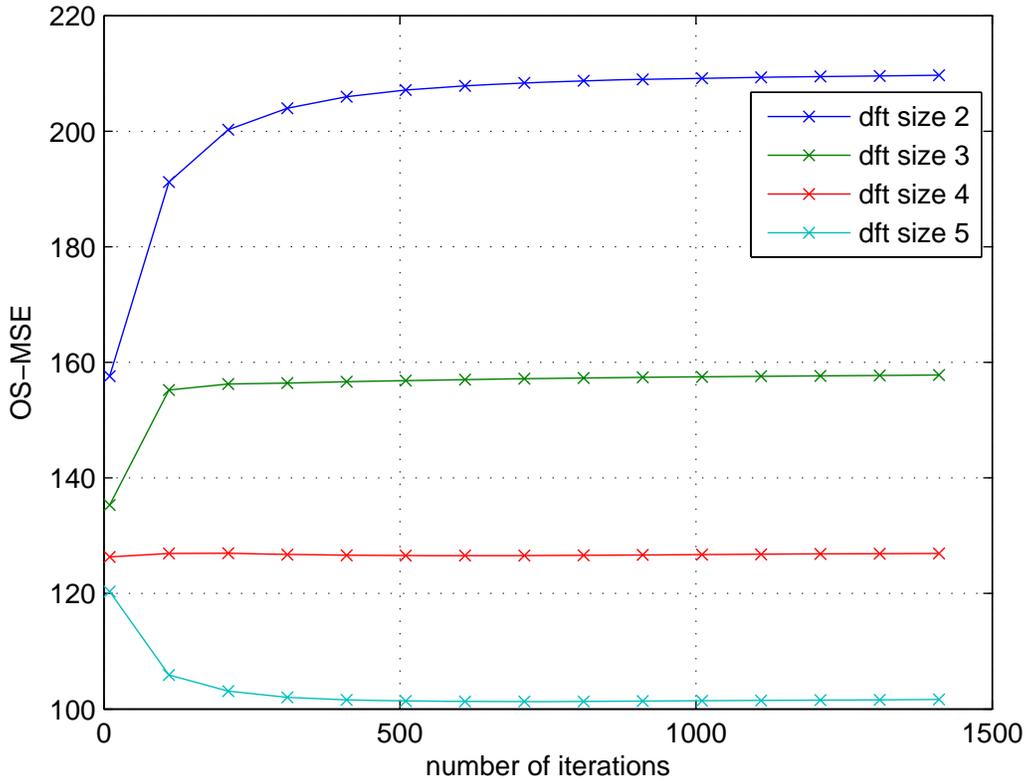


Figure 4-12: OS-MSE as a function of number of iterations and DFT size for the liftingbody image. The starting point was the degraded image magnitude and the deblurring filter was a 9x9-point averager.

on the DFT size for the pout and liftingbody images, as shown in Fig. 4-12. (This was also seen in the liftingbody image when the deblurring filter was an 11x11-point averager and the undegraded image magnitude was used as the initial magnitude.) Again, the convergent solutions always matched what was obtained when the initial magnitude was a constant.

When the deblurring filter was a 9x9-point averager and the starting point was the magnitude of the undegraded image, a low number of iterations (a few hundred or less) was almost always best. The one exception was the cameraman image, which improved with increasing iterations until convergence was reached. Again, the iteration converged to the same value as it did when the initial magnitude was a constant.

These results indicate that, if the algorithm is run to convergence, the starting point doesn't matter. However, there are gains to be had if an "accurate enough"

estimate of the magnitude is used as the starting point and the iteration is stopped short of convergence. What constitutes “accurate enough” is difficult to say, but it often must be better than the degraded image itself, and so such an estimate is not likely to be available and the iteration should simply be started with a constant initial value.

## 4.3 Comparison to Other Methods

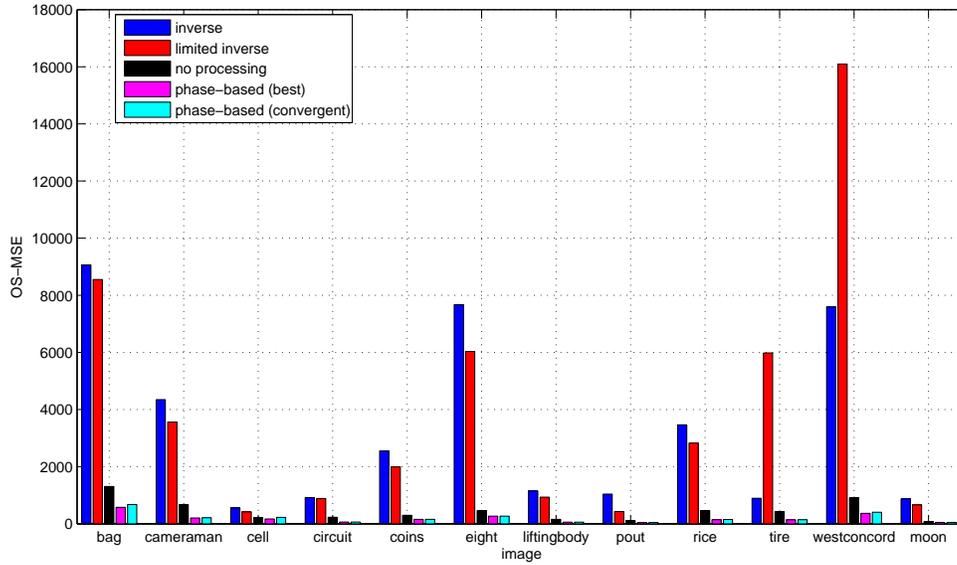
As in Chapter 3, phase-based deconvolution will now be compared first with other inverse filtering techniques and then with image restoration techniques. In all cases, the phase-based deconvolution iteration was initialized with a constant and used an internal DFT size of five times the size of the degraded image. Because there were situations in which stopping the iteration short of convergence was advantageous, both the best (to within 100 iterations) and the convergent solution will be presented in the comparisons.

### 4.3.1 Comparison to Inverse Filtering Techniques

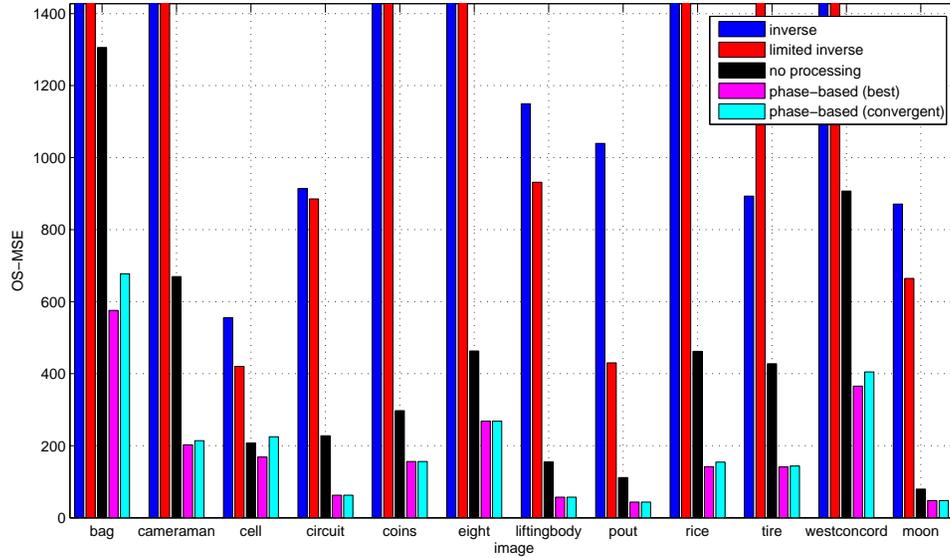
The same inverse filtering methods considered in Chapter 3 will be used here, namely inverse filtering, thresholded inverse filtering (which prevents the inverse filter from exceeding a magnitude of 1,000), and doing no processing. The results of this comparison when the deblurring filter was an 11x11-point averager are shown in Fig. 4-13. The results for a 9x9-point averager are shown in Fig. 4-14.

From these results, it can be seen that phase-based deconvolution was much more robust to errors in the deblurring function than either direct or thresholded inverse filtering, and was almost always an improvement on doing nothing, regardless of whether the iteration was run to convergence or stopped at an optimal point before.

As expected, when the deblurring filter was a 9x9-point averager, the results of phase-based deconvolution were worse than when the deblurring filter was an 11x11-point averager. However, even in this case the restored image was often better than the degraded image.

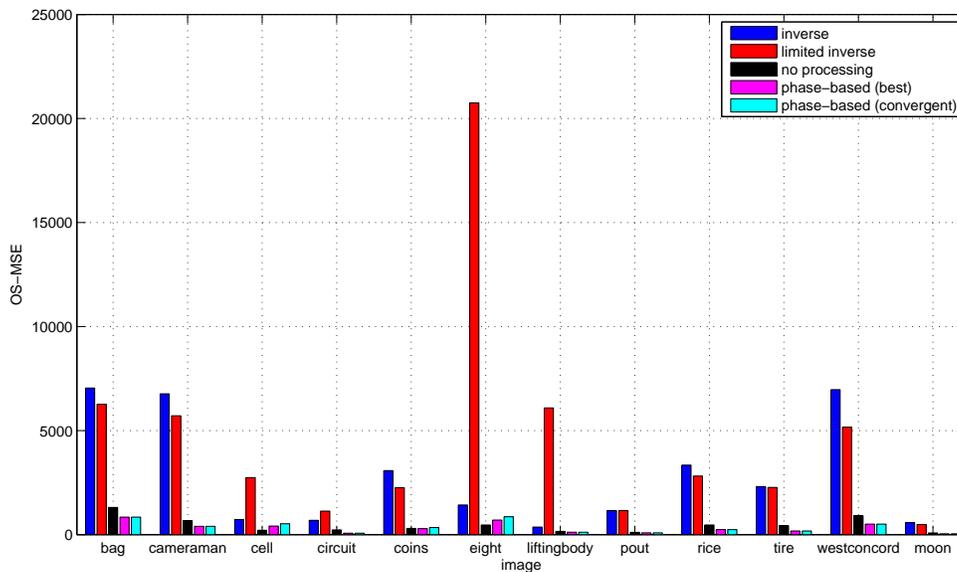


(a) Comparison of inverse filtering techniques

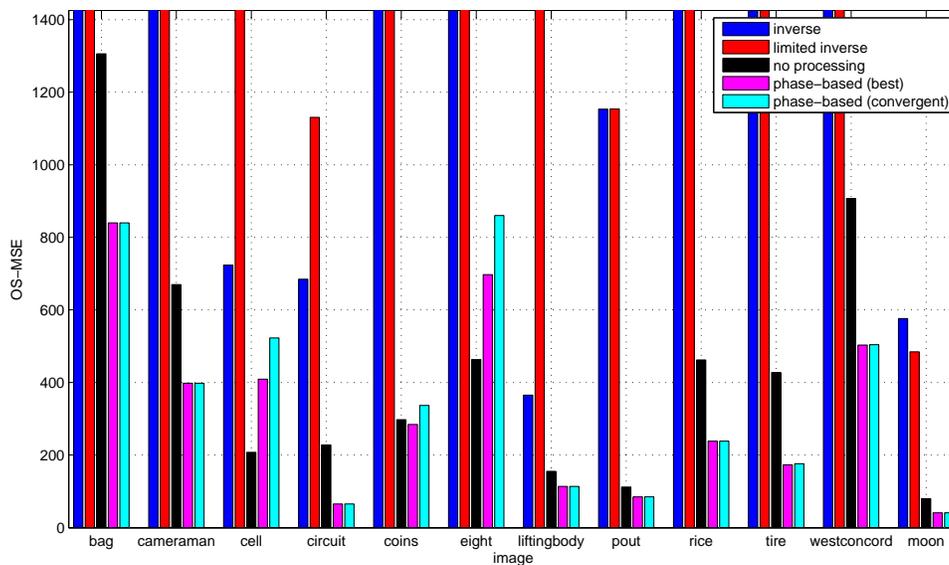


(b) Fig. 4-13a rescaled to highlight phase-based deconvolution

Figure 4-13: Comparison of inverse filtering techniques when the blurring filter was an 11x11-point Gaussian with  $\sigma = 5$  and the deblurring filter was an 11x11-point averager



(a) Comparison of inverse filtering techniques



(b) Fig. 4-14a rescaled to highlight phase-based deconvolution

Figure 4-14: Comparison of inverse filtering techniques when the blurring filter was an 11x11-point Gaussian with  $\sigma = 5$  and the deblurring filter was a 9x9-point averager

### 4.3.2 Comparison to Image Restoration Techniques

Again, the same image restoration algorithms used in Chapter 3 will be used here, namely Wiener filtering, regularized filtering, and the Richardson-Lucy iteration. In this case, there arises some ambiguity in what to use for the noise parameters input to the Wiener and regularized filters. There is no noise in the degraded image. However, there is some “noise” in the restoration in the incorrect guess of the blurring function. This noise is difficult to describe without knowledge of the correct blurring function, which obviously isn’t available. Therefore, it is incorrect to say there is no noise when using the Wiener and regularized filters, but what is correct to say instead is ambiguous.

For the purposes of these experiments, the noise used was simply Gaussian noise with a low, arbitrary variance. This is certainly not the right answer. However, it did achieve the desired effect of causing the Wiener and regularized filters to behave as though there was some low level of noise in the system. The performance of these algorithms could likely be improved with a better description of the noise, but a better description of the noise is equivalent to better knowledge of the blurring filter, which is assumed to be unavailable.

The Richardson-Lucy iteration required no such guesses because, as a probabilistic method based on Poisson processes, it does not rely directly upon the noise statistics as do Wiener and regularized filtering.

The results of this comparison when the deblurring filter was an 11x11-point averager are shown in Fig. 4-15. These results demonstrate that phase-based deconvolution does as well as or better than any of the restoration methods in almost all cases. The only exceptions were the cell image and the moon image.

The cell image was a bit of an anomaly. The artifacts present in any of the restorations of this image were not significantly different from the artifacts present in any of the restorations of the other images, but, in the case of the cell image, these artifacts seem to have affected OS-MSE differently. To a lesser extent, this also seems to have been the case for the moon image. The restored cell and cameraman images

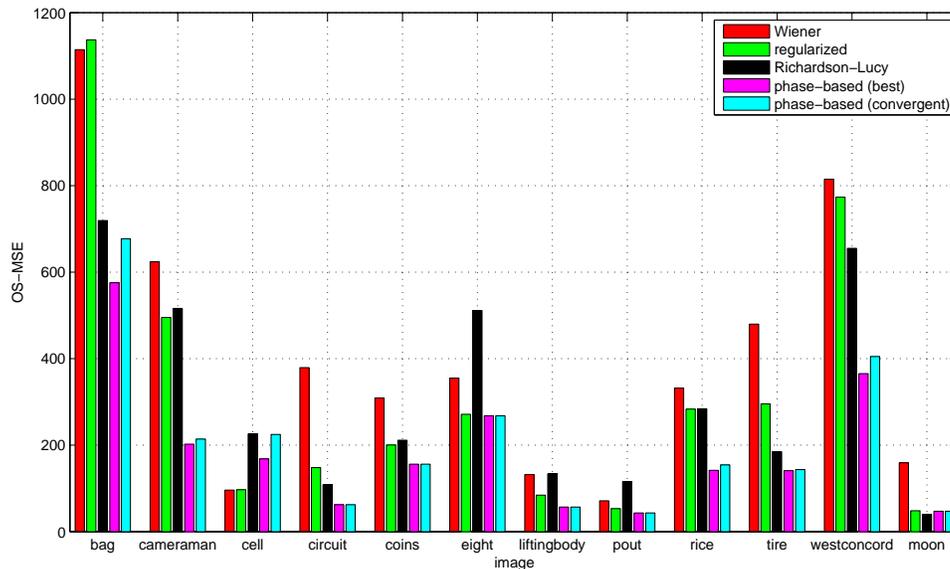


Figure 4-15: Comparison of restoration techniques when the blurring filter was an 11x11-point Gaussian with  $\sigma = 5$  and the deblurring filter was an 11x11-point averager

are shown in Figures 4-16 and 4-17, respectively, and the artifacts are representative of what was present in the other images.

In Fig. 4-16, the result of the Richardson-Lucy iteration was no different from the blurred image. This was also the case with the eight and pout images, and in no case did the Richardson-Lucy iteration, which does seem to be very sensitive to errors in the deblurring function, run for more than 25 iterations.

As can be seen from these examples, there is some variety present in the artifacts caused by different restoration methods. However, phase-based deconvolution, especially when run only to the optimal solution, does generally produce the sharpest, clearest image.

The results when the deblurring filter was a 9x9-point averager, which are shown in Fig. 4-18, were more ambiguous. In comparing Figs. 4-18 and 4-15, it is interesting to note that not only did phase-based deconvolution get worse, but the other restoration methods also often got better, implying the 9x9-point averager was in some sense a better guess of the overall blurring function than the 11x11-point averager or that the given noise was a better guess of the resulting error.

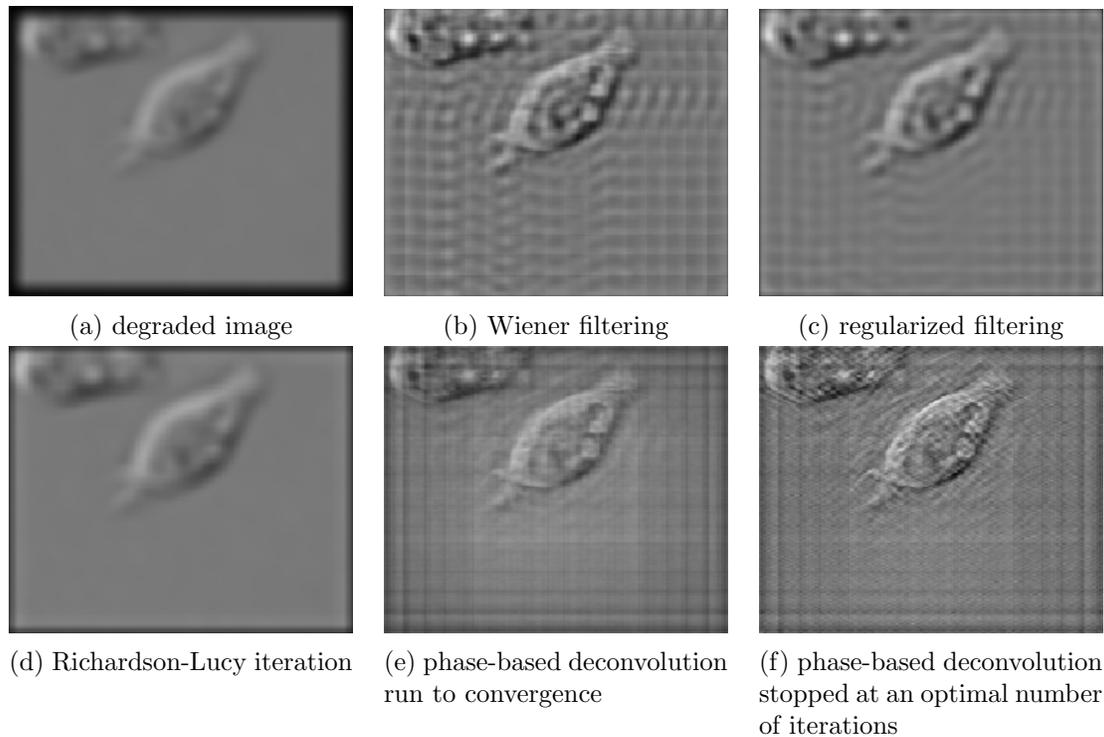


Figure 4-16: Cell image restored with several different methods when the blurring filter was an 11x11-point Gaussian with  $\sigma = 5$  and the deblurring filter was an 11x11-point averager

The cameraman image was representative of the artifacts caused by the different restoration methods and is shown in Fig. 4-19. Although the OS-MSEs of the images restored with regularized filtering, the Richardson-Lucy iteration, and phase-based deconvolution are very similar, the artifacts are quite different. Which is really the “best” solution is a matter of application. The image obtained with phase-based deconvolution was typically sharper than the others, but at the cost of the ringing artifacts shown in Fig. 4-19d.

These results indicate that phase-based deconvolution should certainly be used when the size of the blurring function is known accurately but the taper is not. If the size of the blurring function is not known, phase-based deconvolution may still have some advantages depending on the application and the preferences of the viewer.

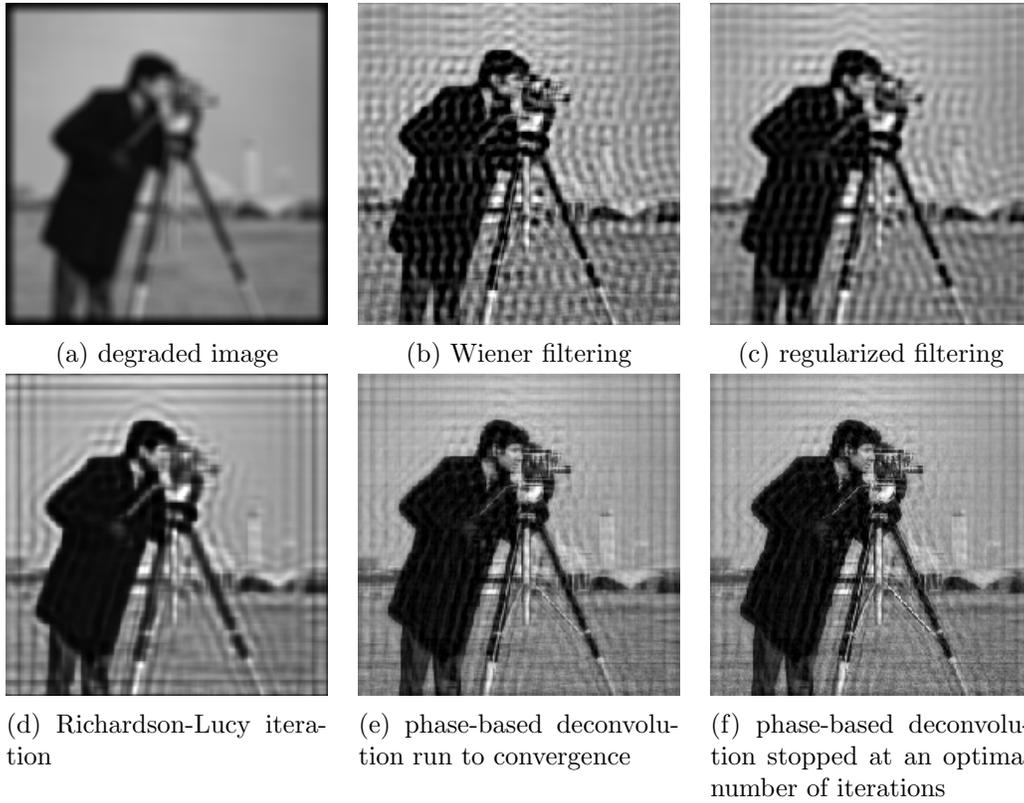


Figure 4-17: Cameraman image restored with several different methods when the blurring filter was an 11x11-point Gaussian with  $\sigma = 5$  and the deblurring filter was an 11x11-point averager

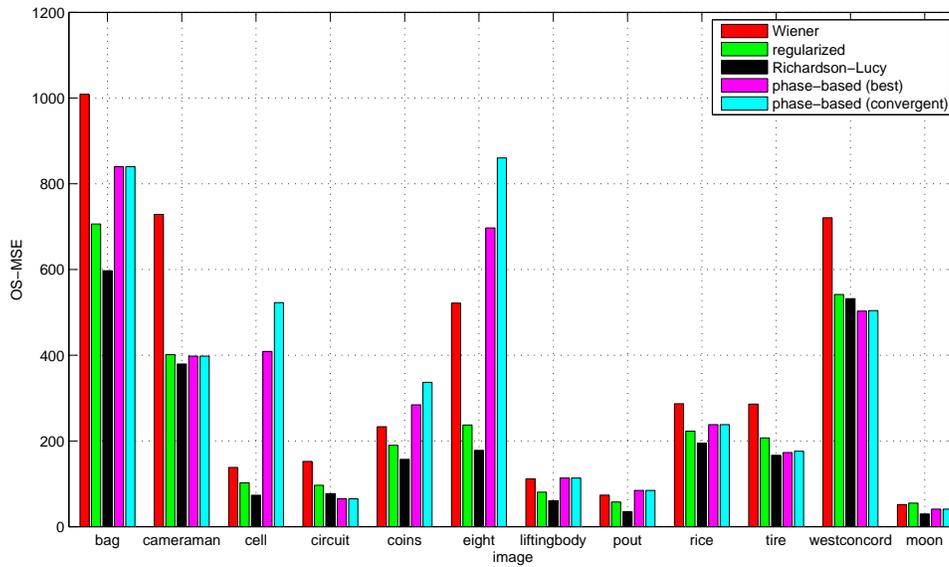


Figure 4-18: Comparison of restoration techniques when the blurring filter was an 11x11-point Gaussian with  $\sigma = 5$  and the deblurring filter was a 9x9-point averager



Figure 4-19: Cameraman image restored with several different methods when the blurring filter was an 11x11-point Gaussian with  $\sigma = 5$  and the deblurring filter was a 9x9-point averager

## 4.4 Noise

### 4.4.1 Characterization of Phase-Based Deconvolution

When white Gaussian noise was added to the blurred image before it was deconvolved with the incorrect deblurring function, the results depended on the amount of noise added. For low noise values (what constituted “low” noise varied from one image to the next), results were similar to the noiseless case. That is, for those images where stopping short of convergence was advantageous in the noiseless case, it was also generally advantageous in the low noise case, and for those where it wasn’t

advantageous in the noiseless case, it wasn't ever advantageous in the low noise case. However, the gains tended to be smaller and to disappear completely as the noise increased. In all cases but the cell image, there was some noise level above which the convergent solution was always the best. Additionally, larger DFT sizes tended to do slightly better than smaller ones.

These results indicate that, in general, the phase-based reconstruction iteration should be run to convergence when the blurring function is known imprecisely and the image is noisy. This solution may not be optimal in the OS-MSE sense, but it will likely be close. This eliminates the difficult problem of determining the correct stopping point.

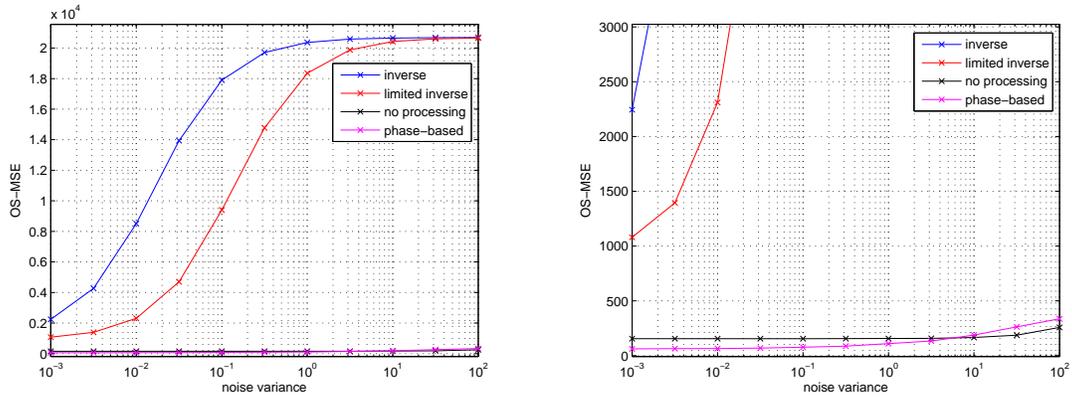
#### 4.4.2 Comparison to Other Methods

In this comparison, the phase-based deconvolution iteration was run to convergence with an internal DFT size of five times the image size. The regularized and Wiener filters were constructed using only the known additive white Gaussian noise statistics, and the Richardson-Lucy iteration was run for a locally optimal number of iterations, as in previous comparisons. The moon image was not included because of the very large number of iterations required to achieve convergence for phase-based deconvolution of that image.

In comparison to other methods, phase-based deconvolution generally performed quite well when the deblurring filter was an 11x11-point averager and the noise was low. Again, what "low" meant varied from one image to the next, but a noise variance of about 1 was a typical cutoff point. At higher noise values, other restoration methods, and sometimes even doing no processing, became better. The results for the liftingbody image were typical, and are shown in Fig. 4-20.

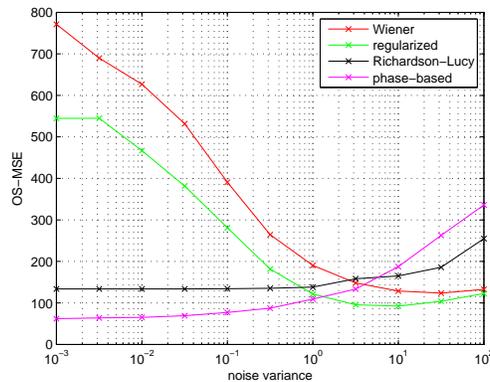
It is interesting that regularized and Wiener filtering become more accurate as the noise increases. This is likely because regularized and Wiener filters require the noise statistics. The parameters associated with higher noise levels probably provide a more accurate description of the "noise" due to the incorrect deblurring function.

As in the noiseless case, different restoration methods produce different artifacts,



(a) comparison of deconvolution methods

(b) comparison of deconvolution methods, zoomed in for detail



(c) comparison of restoration methods

Figure 4-20: Illustration of how phase-based deconvolution of the liftingbody image compares to some existing methods when the deblurring filter was an 11x11-point averager and there was noise

and so OS-MSE is only a rough indicator of visual quality, and what image is really the “best” may vary based on application. The visual results for the liftingbody image with a noise variance of 10 are shown in Figure 4-21.

Results were similar when the deblurring filter was a 9x9-point averager, though the noise level at which phase-based deconvolution became worse than the other methods was generally lower, and sometimes there was no noise level at which phase-based deconvolution was better than the Richardson-Lucy algorithm. Results for the liftingbody image are shown in Fig. 4-22.

As before, the artifacts produced by different methods are quite different, and so there will be an element of subjectivity in choosing which is really the “best” restoration. Examples of these different artifacts for the liftingbody image are shown

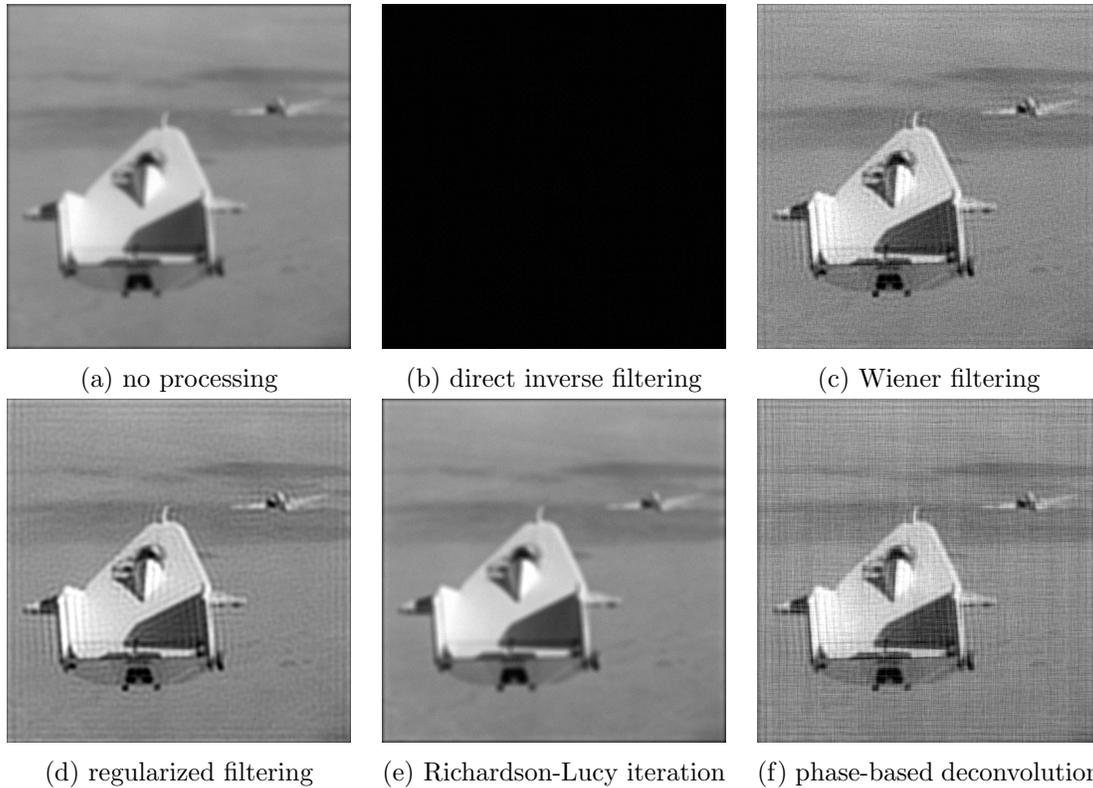


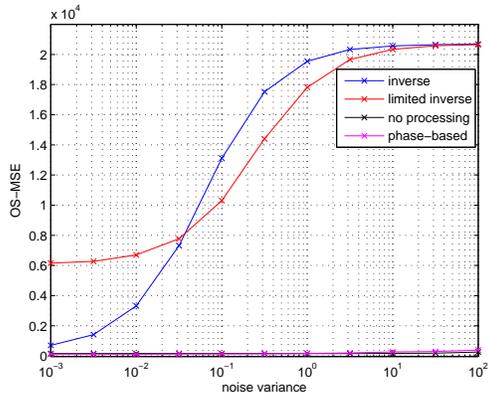
Figure 4-21: Liftingbody image restored with several different methods when the blurring filter was an 11x11-point Gaussian with  $\sigma = 5$ , the deblurring filter was an 11x11-point averager, and there was additive white Gaussian noise with variance 10

in Fig. 4-23.

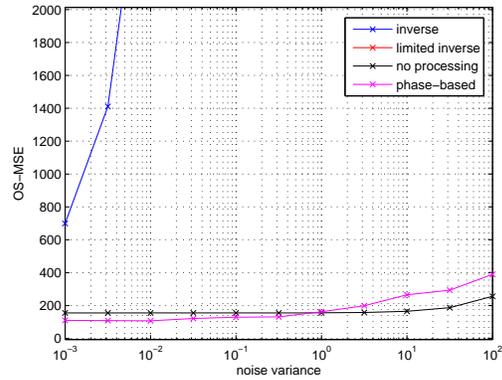
These results confirm what was found in the noiseless case and when the deblurring filter was correct: Phase-based deconvolution performs best relative to other methods when the signal to noise ratio is high and when the size of the blurring function is known.

## 4.5 Summary

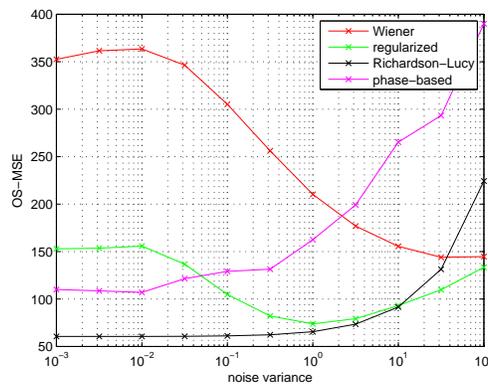
When the size of the blurring function is known but its taper is not, phase-based deconvolution does very well. Although the optimal solution may require the iteration to be stopped before convergence, the convergent solution is also generally acceptable. A larger internal DFT size also tends to do somewhat better than a smaller one, and the algorithm converges to the same solution regardless of the starting point. At low



(a) comparison of deconvolution methods



(b) comparison of deconvolution methods, zoomed in for detail



(c) comparison of restoration methods

Figure 4-22: Illustration of how phase-based deconvolution of the liftingbody image compares to some existing methods when the deblurring filter was a 9x9-point averager and there was noise

noise levels, phase-based deconvolution, run to either the optimal or the convergent solution, produced results that were almost always better than those obtained by the other deconvolution and restoration methods. At high noise levels, phase-based deconvolution performed much better than direct or thresholded inverse filtering, but became less competitive with the restoration methods, consistent with what was observed in the previous chapter. When the size of the blurring function was not known accurately, the results were generally similar, except that phase-based deconvolution did not perform as well, and so the noise threshold at which it became worse than the restoration methods was lower.

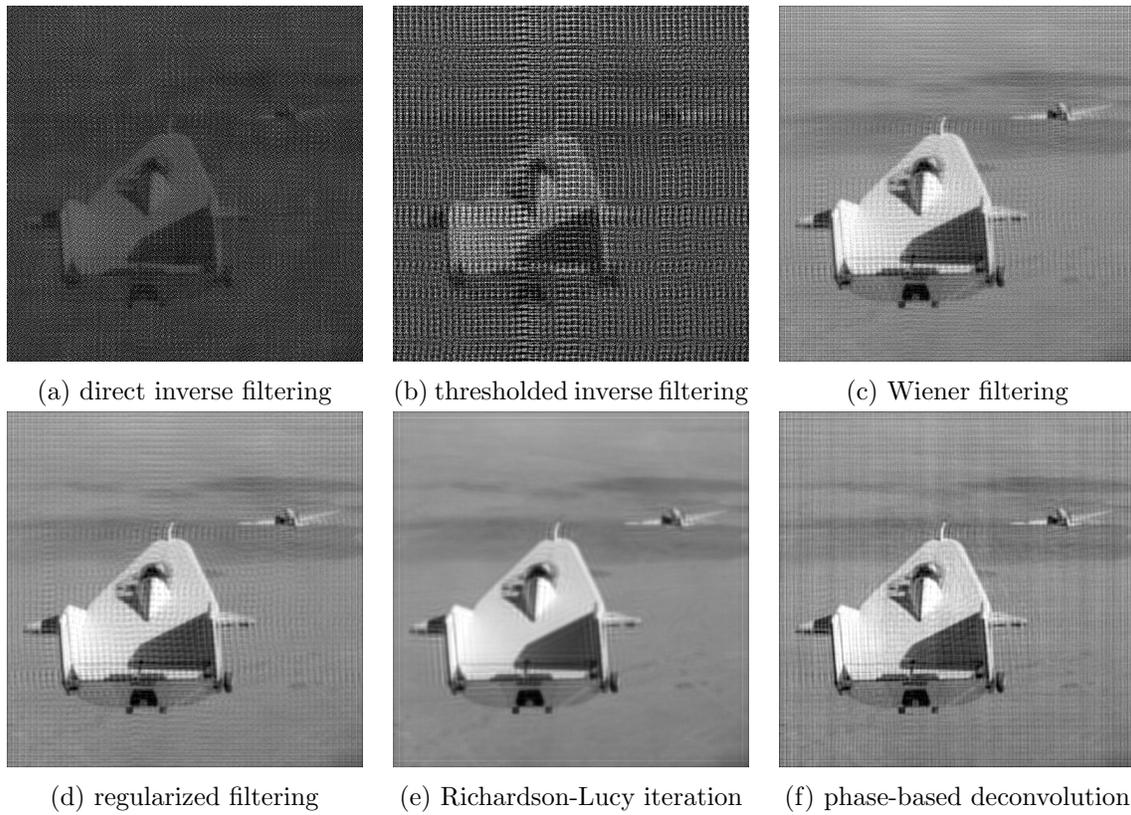


Figure 4-23: Liftingbody image restored with several different methods when the blurring filter was an 11x11-point Gaussian with  $\sigma = 5$ , the deblurring filter was a 9x9-point averager, and there was additive white Gaussian noise with variance .1

# Chapter 5

## Obtaining the Region of Support

Experiments performed so far have assumed that the region of support (ROS) of the image was known. This may not always be a good assumption. For example, a typical astronomical image could consist of a star or other bright object on a black field. In such a situation, the region of support would have to be determined in some way.

This chapter will look at three images: A white circle on a black field, a white circle with a gray halo on a black field, and a variation on the moon image in which the moon itself has been kept, but the background has been made black. (Though the background of the moon image shown in Fig. 3-1a appears to be black, it is really several shades of very dark gray—the background pixels have low values, but are not exactly zero.) These images are shown in Fig. 5-1. Fig. 5-1c is shown at 75% of the original resolution for copyright reasons.

This chapter will proceed as follows. First, several methods of obtaining the ROS will be outlined. Then, the performance of these methods will be compared in the cases of perfect knowledge of the blurring function and no noise, perfect knowledge of the blurring function and additive white Gaussian noise, and imperfect knowledge of the blurring function and no noise. Finally, the results will be summarized.

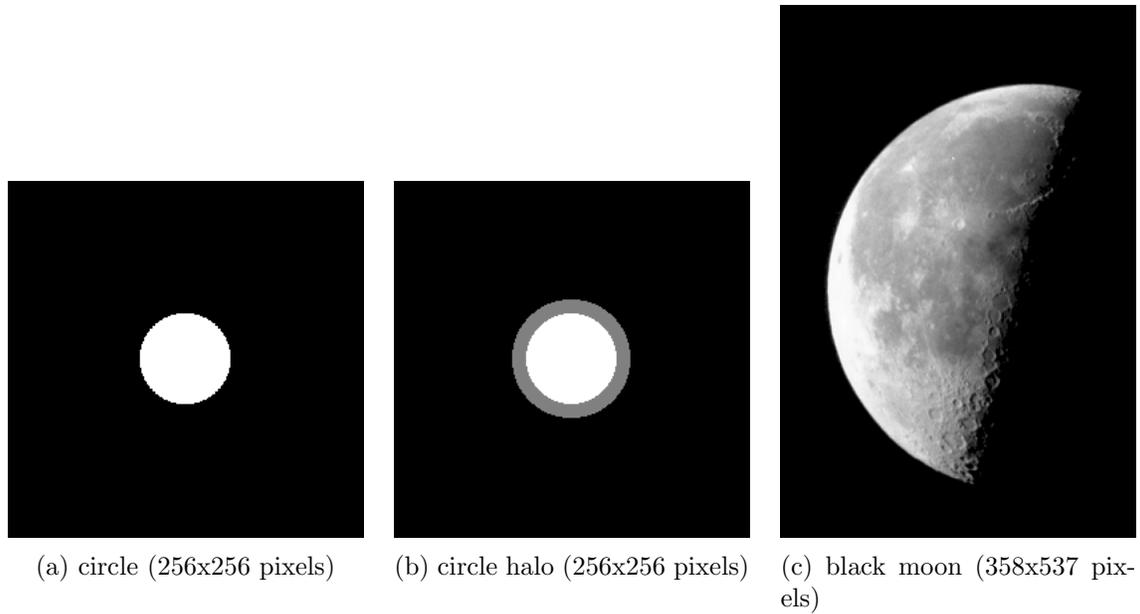


Figure 5-1: Original images used in the region of support experiments

## 5.1 Description of Methods

Three possible methods of obtaining the ROS will be investigated here. First, the ROS may simply be assumed to be the ROS of the degraded image. Second, morphological processing may be performed on the degraded image to obtain the ROS, and, finally, markers can be put in the degraded image to set the ROS to be known. Each of these methods will now be discussed in detail.

### 5.1.1 ROS from the Degraded Image

In this case, the first and last nonzero values in the degraded image will be assumed to indicate the ROS, which will then be adjusted to account for the increased region of support caused by the convolution with the blurring function. That is, if an  $11 \times 11$ -point blurring function that is symmetric around zero is used, and the first nonzero pixel in the degraded image occurs at, for example, index 3, the region of support will be assumed to start at index 8. In the presence of noise, this is very likely to give regions of support that are much larger than the correct ones, but it is not likely to give regions of support that are much smaller than the correct ones. It will be

explained in the next subsection why this is desirable.

### 5.1.2 ROS from Morphological Processing

When determining the ROS, it is important to note that the image is blurred before it is corrupted by additive noise. That is, the noise will not be blurred. This can be exploited by performing the morphological opening operation on the degraded image, where the function used to perform the opening operation is simply the same size as the blurring function. This will eliminate most of the noise outside the ROS, which will not be contiguous groups of pixels as the features degraded by the blurring function will be. Then, the ROS can be determined from the morphologically processed picture. Furthermore, the area outside the determined ROS can be set to zero, thus eliminating much of the noise in the image.

Binary morphological processing requires a threshold to be set above which all values are considered to be one and below which all values are considered to be zero, and so the question of what this threshold ought to be must be addressed. A higher threshold is more likely to eliminate noise, but may also eliminate low signal values, and thus give a ROS that is too small, and a lower threshold is more likely to include noise, but is less likely to eliminate signal, and thus give a ROS that is too large.

Experiments to determine the effect of a wrong guess of the ROS indicate that it is much more harmful to guess too small than too large. The black moon results were representative and are shown in Fig. 5-2. These results are for the case of no noise and perfect knowledge of the blurring function. The only “unknown” was the ROS.

As suggested by these results, the threshold was set very low in all following experiments. Specifically, it was set to 1% of the standard deviation of the noise. This value was determined experimentally and is somewhat arbitrary. In practice, when quantization has been performed (it was not in these experiments to avoid confounding the results), a threshold value of one or two would probably be acceptable.

As with measuring the ROS from the degraded image, the size of the blurring function is used to adjust the results obtained from morphological processing.

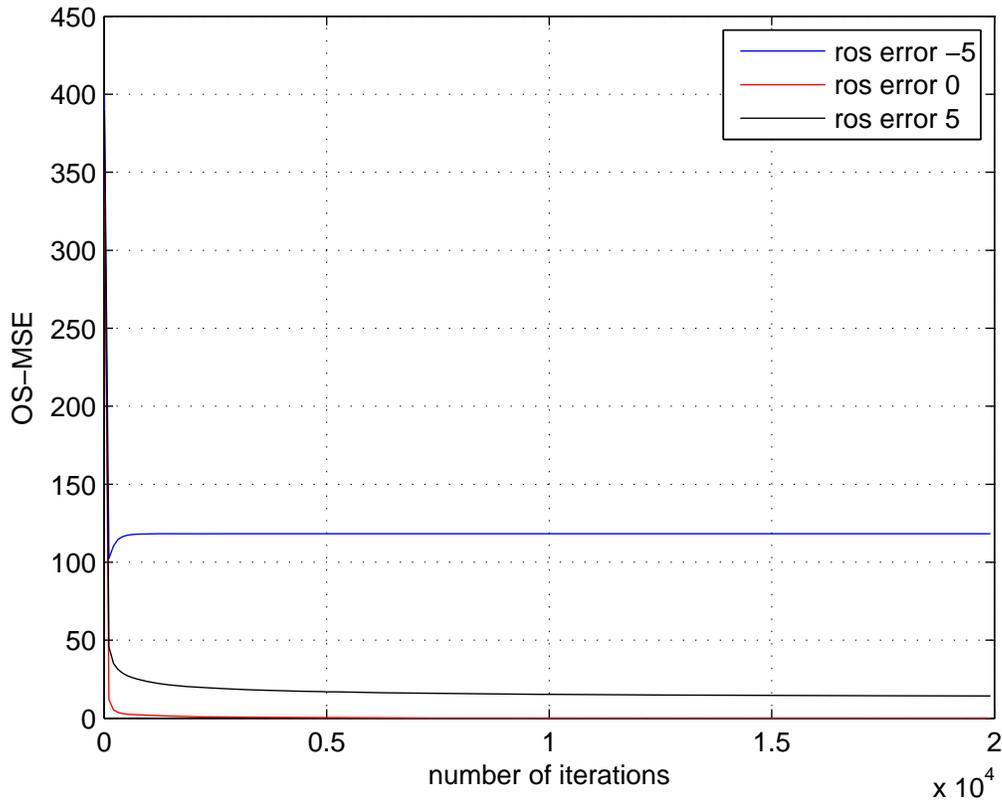


Figure 5-2: OS-MSE as a function of number of iterations and ROS error for the black moon image. ROS error is measured in pixels too large on every side.

### 5.1.3 ROS with Markers

A third method of determining the ROS is to add markers at the edges of the image. The procedure is as follows: First, select a shape to use as a marker. In these experiments, a 10x10 white square was selected, but this is not the only option. Second, blur the marker with the assumed blurring function. Third, add the blurred marker at the corners of the degraded image. The ROS is now known because it has been artificially made to be the entire degraded image, after correcting for the size of the blurring function as before. This is illustrated in Fig. 5-3.

This procedure is valid because convolution is linear. That is, the degraded image

$$g_1(n_1, n_2) = f(n_1, n_2) * b(n_1, n_2) \quad (5.1)$$

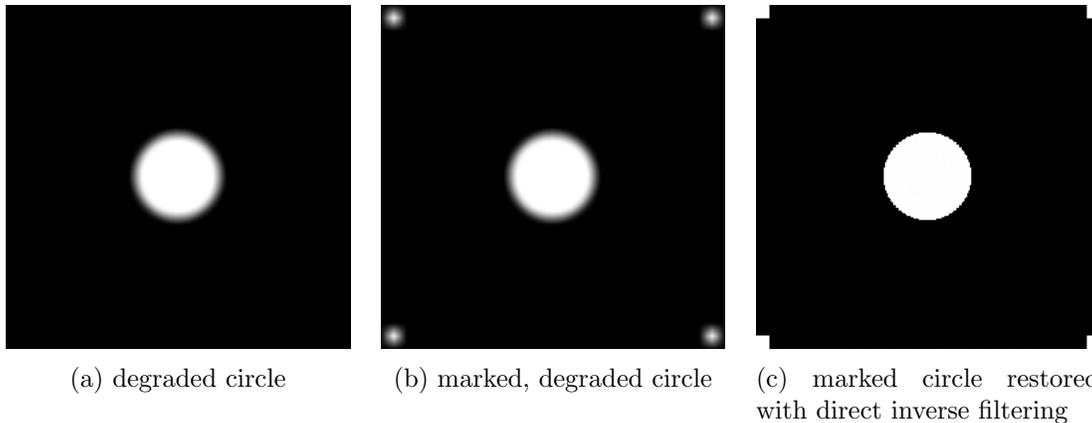


Figure 5-3: Illustration of the procedure used in marking a blurry image with an unknown region of support

is given. The degraded markers can be computed as

$$g_2(n_1, n_2) = m(n_1, n_2) * b(n_1, n_2) \quad (5.2)$$

where  $m(n_1, n_2)$  represents the undegraded markers (in this case,  $m(n_1, n_2)$  is simply a black image of the same size as the original image with a 10x10 white square in each corner), and  $g_2(n_1, n_2)$  is the resulting degraded markers. The marked, degraded image is then

$$g(n_1, n_2) = g_1(n_1, n_2) + g_2(n_1, n_2) \quad (5.3)$$

$$= f(n_1, n_2) * b(n_1, n_2) + m(n_1, n_2) * b(n_1, n_2) \quad (5.4)$$

$$= (f(n_1, n_2) + m(n_1, n_2)) * b(n_1, n_2) \quad (5.5)$$

And so the deconvolution of  $g(n_1, n_2)$  with  $b(n_1, n_2)$  will simply yield the sum of the original image and the markers. The markers are known, and so the original image can easily be obtained.

## 5.2 Comparison of Methods

The performance of the three methods described will now be compared in three situations: The absence of noise and perfect knowledge of the blurring function, the

presence of noise and perfect knowledge of the blurring function, and the absence of noise and imperfect knowledge of the blurring function.

For the purposes of these experiments, “convergence” of the phase-based reconstruction algorithm will be defined as the point at which the OS-MSE changes by less than .1 between a hundred iterations of the algorithm.

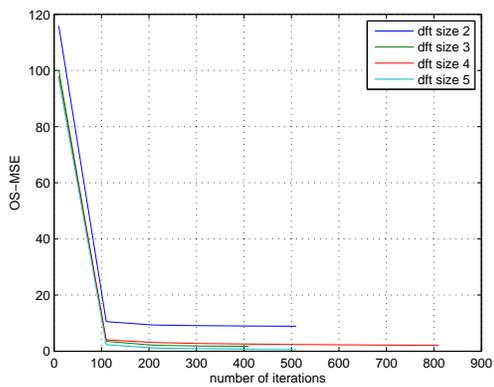
The algorithm will be stopped after the application of the frequency domain constraints. The issue hasn’t arisen before because, in previous chapters, the entire image was included in the region of support. However, here it becomes important to note where the iteration is stopped. The justification for stopping after the application of the frequency domain constraints is twofold. First, in the ideal case, the ROS would be the only unknown. Therefore, when convergence is reached and the two constraints are being violated circularly, it is reasonable to end with the correct constraint met and the incorrect constraint violated. Second, letting the spatial constraints be violated will leave artifacts visible that would otherwise be set to zero, and this will give some intuition about how the algorithm is trying to resolve the conflicting constraints.

### **5.2.1 No Noise, Perfect Knowledge of the Blurring Function**

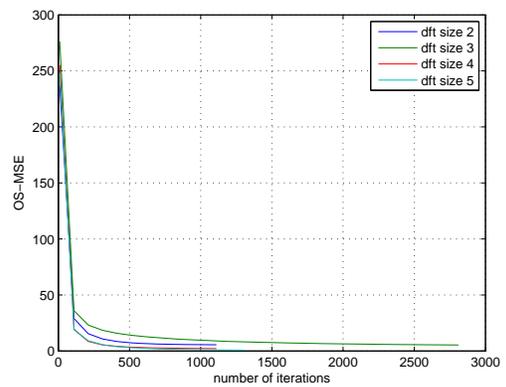
In this situation, all methods should, in theory, give perfect results. The only question is how many iterations are required to achieve those results. However, in these experiments, there were sometimes artifacts visible in the images even after “convergence” had been reached. This is because the “convergence” defined here is not actually the convergent solution of the algorithm. It merely indicates the point at which gains are coming too slowly for continued running to be practical.

With morphological processing (which, in this situation, is the same as using the ROS of the degraded image), the number of iterations required for convergence depended on the image and, in the case of the circle and circle halo images, on the DFT size. Also for those images, the OS-MSE of and artifacts in the “convergent” solution depended on the DFT size. These results are illustrated in Figs. 5-4 and 5-5.

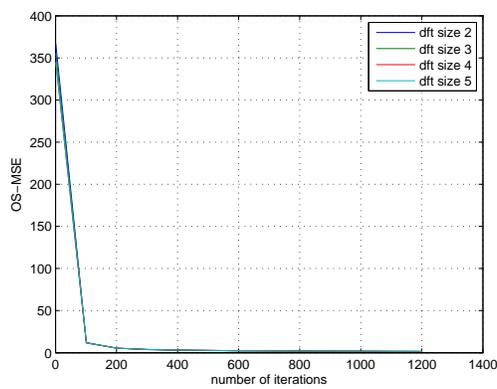
With markers, it took much longer to reach convergence. In the case of the circle and circle halo images, the iteration had not converged after even 100,000 iterations,



(a) circle

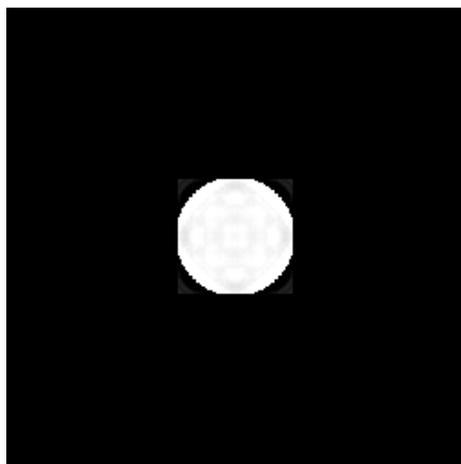


(b) circle halo

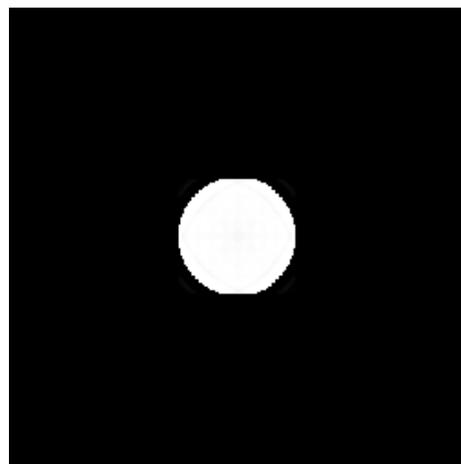


(c) black moon

Figure 5-4: OS-MSE as a function of number of iterations and DFT size when the ROS was obtained by morphological processing



(a) DFT size: twice the image size



(b) DFT size: five times the image size

Figure 5-5: Artifacts in “convergent” circle images. ROS determined (correctly) by morphological processing.

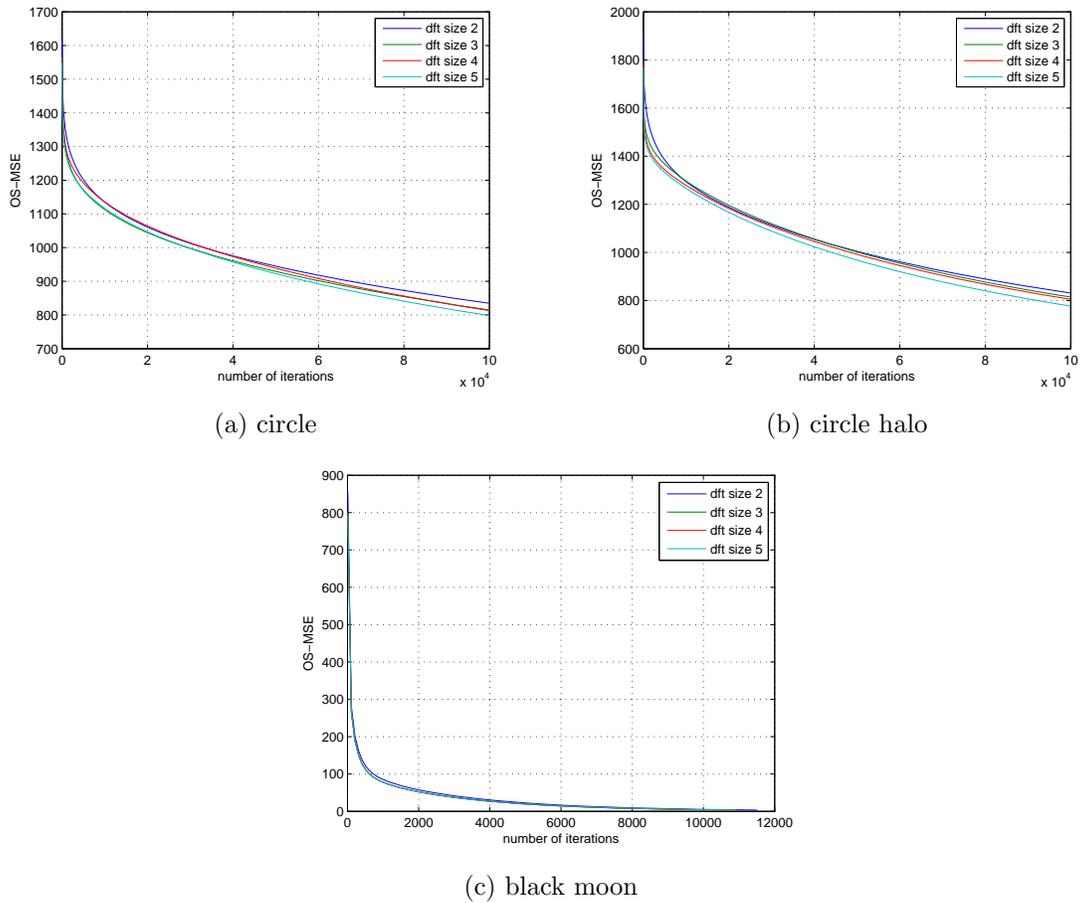


Figure 5-6: OS-MSE as a function of number of iterations and DFT size when the ROS was obtained by the use of markers

when it was stopped due to time constraints. For the black moon image, it did converge, but only after over 10,000 iterations, an order of magnitude more than required with morphological processing. These results are shown in Figs. 5-6 and 5-7. It should be noted that, while the markers have been left in the restored images in Fig. 5-7, they were excluded in the calculation of the OS-MSE. The strong dependence on DFT size seen when the ROS was obtained with morphological processing was not visible here, though it may have become an issue in the circle and circle halo images if the iteration had been allowed to continue running.

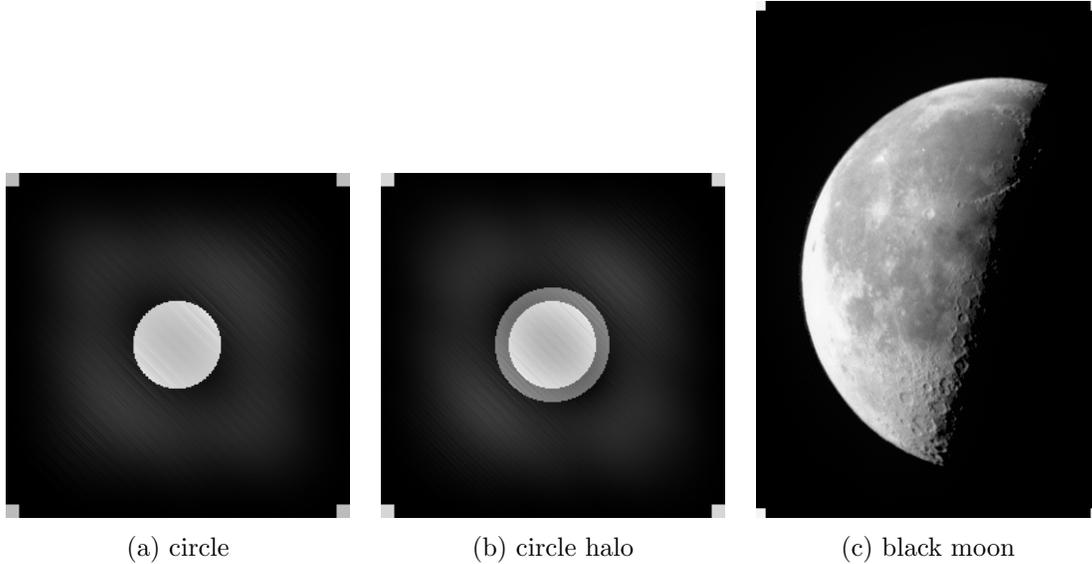


Figure 5-7: Artifacts in “convergent” images when markers were used to indicate the ROS. All images were restored with a DFT size of five times the image size.

### 5.2.2 Noise, Perfect Knowledge of the Blurring Function

In the experiments that follow, a subtlety must be dealt with that has previously been ignored. When adding white Gaussian noise to an image, it is possible to get negative magnitudes at some pixels. Obviously, this is an unrealistic model. In previous chapters, images have had sufficient non-zero content to render the issue negligible, but it becomes relevant here. In this chapter, the noise clipping inherent in any physical imaging system will be included in the model, and the degraded image will be computed as

$$g(n_1, n_2) = \max(f(n_1, n_2) * b(n_1, n_2) + v(n_1, n_2), 0) \quad (5.6)$$

where  $g(n_1, n_2)$  is the degraded image,  $f(n_1, n_2)$  is the original image,  $b(n_1, n_2)$  is the blurring function, and  $v(n_1, n_2)$  is Gaussian noise.

When the ROS of the degraded image was used as the ROS, the results were very poor. Generally, more iterations led to a lower OS-MSE, and the exceptions were not dramatic. However, convergence was not always reached after even 100,000 iterations. Larger DFT sizes, as seen previously, were also generally better, though which was the best depended on noise level and image. Even in the best case, the restoration

results were very poor.

When morphological processing was used to determine the ROS, the results were much better. Morphological processing was always able to determine the ROS to within a few pixels, which allowed much of the noise to be eliminated. The convergent solution was the best in the case of the black moon image, but in the circle and circle halo images, a few hundred iterations did somewhat better. The black moon image is generally more representative of real images and is therefore the more interesting result. As usual, the DFT size did not have a very significant impact, but larger was generally a little better.

When markers were used to set the ROS, the results were not as good as those obtained by morphological processing. The solution achieved after a few hundred iterations tended to be better than the convergent solution, and, in the case of the circle image, it was not clear whether the iteration actually would converge or just get indefinitely worse. The solution changed only slightly with the DFT size, and none of these solutions were at all competitive with morphological processing.

These results are summarized in Fig. 5-8, which shows the OS-MSE of the restored black moon images using a DFT size of twice the degraded image size and the convergent solution. A comparison of the artifacts is shown in Fig. 5-9. When the degraded image's ROS was used as the ROS, the types of artifacts present in the restored image changed as the noise increased. This change, and not any real improvement in quality, is responsible for the decrease in OS-MSE as noise increases. The subjective visual quality of these images was never acceptable. The results for the moon image are representative for morphological processing. For markers and using the degraded image, the results for the moon image were much better than they were for either circle image.

Morphological processing is clearly the best option when there is noise present. This is likely for two reasons. First, morphological processing puts the tightest bound on the ROS. Even though this bound may be too tight when there is noise, a bound slightly too tight does better than a bound much too loose and the effect of noise dominates the effect of an incorrect ROS. Second, morphological processing allows

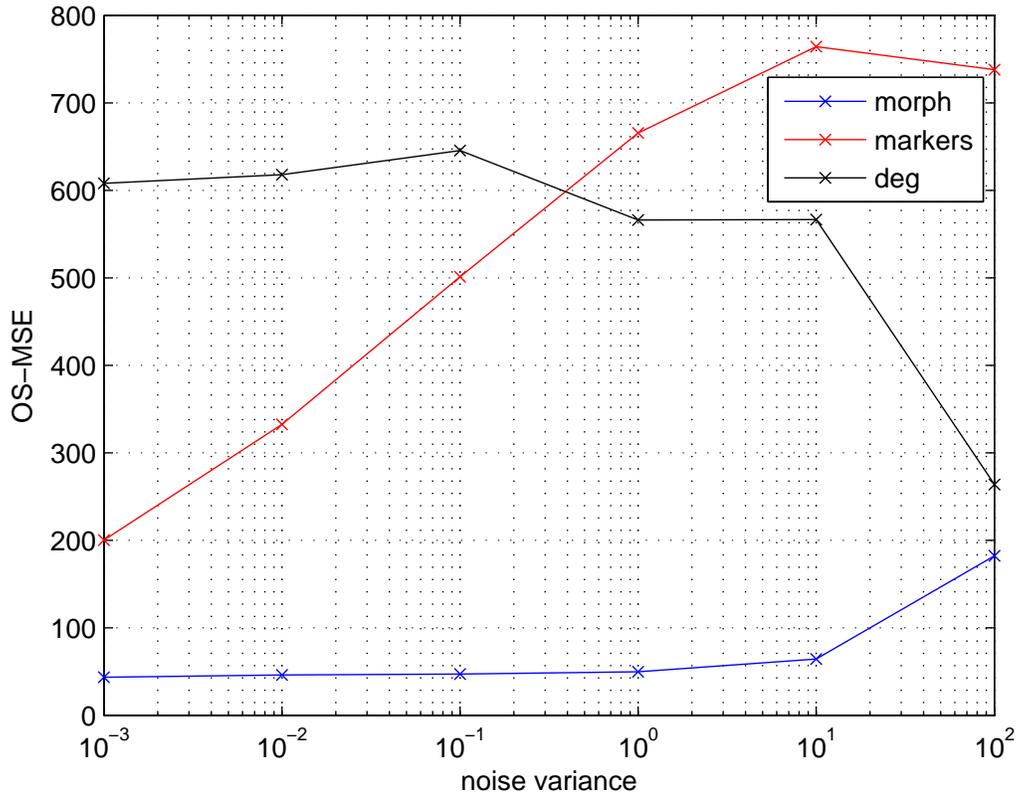
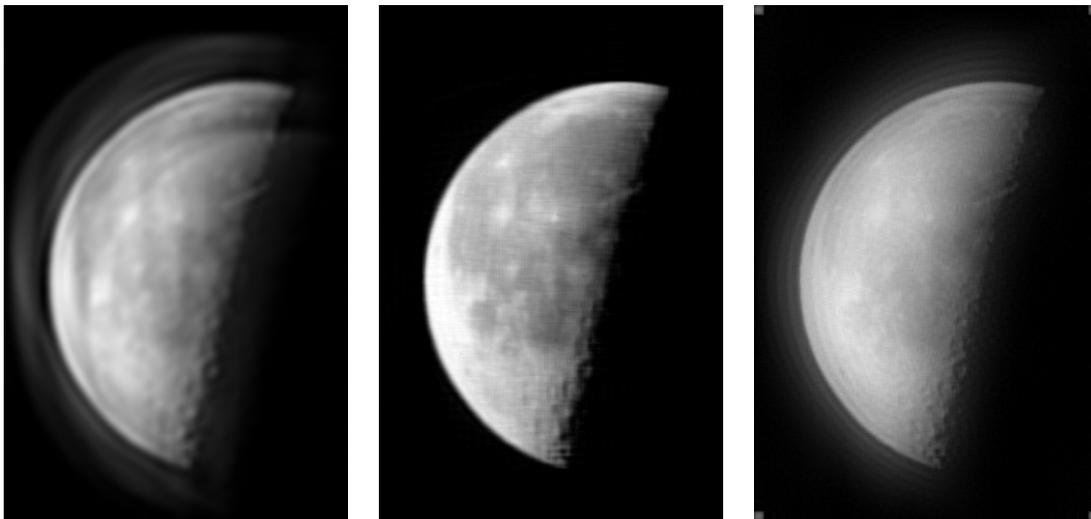


Figure 5-8: Comparison of methods of determining the ROS of the black moon image in the presence of noise



(a) ROS from degraded image (b) ROS from morphological processing (c) ROS from markers

Figure 5-9: Comparison of artifacts produced by different methods of obtaining the ROS. Noise variance was 1.

everything outside the estimated ROS to be set to zero, which eliminates much of the noise. This noise reduction is not possible with the other methods.

### 5.2.3 No Noise, Imperfect Knowledge of the Blurring Function

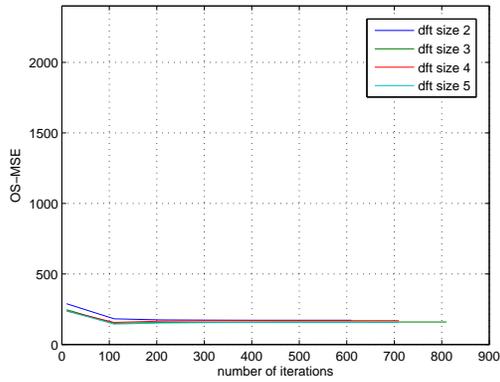
There are two situations to be considered when knowledge of the blurring function is imperfect. First, the size of the blurring function might be known. If this is the case, the fact that the specific values of the blurring function are unknown will have minimal impact on the estimated ROS, and results should be similar to those found in Chapter 4. Second, the size of the blurring function may be unknown, in which case the estimated ROS will be affected.

Experiments were performed to determine the performance of the three methods of finding the ROS in this second case. Specifically, as in Chapter 4, the blurring function was an 11x11-point Gaussian filter,  $\sigma = 5$ , and the deblurring filter was a 9x9-point averager.

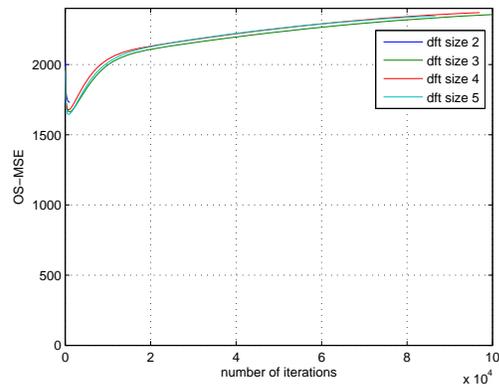
As in the ideal case, using the ROS of the degraded image and obtaining the ROS with morphological processing gave the same result, and the ROS in this case was estimated to be one pixel larger than the actual ROS. This is as expected—the edges of the blurred image were detected properly, but the size increase due to blurring was incorrectly compensated for.

In all three images, morphological processing gave much better results than the use of markers. In the circle and circle halo images, there were some gains to be had by stopping the algorithm after a few hundred iterations, but the convergent solution was also reasonable. How the DFT size affected the performance was different between the two images, but a DFT size of five times the degraded image size was the best in both cases.

When markers were used, the results were much worse. A few hundred iterations gave the best solution (which was still worse than the convergent solution obtained with morphological processing) and it was unclear whether the algorithm was going

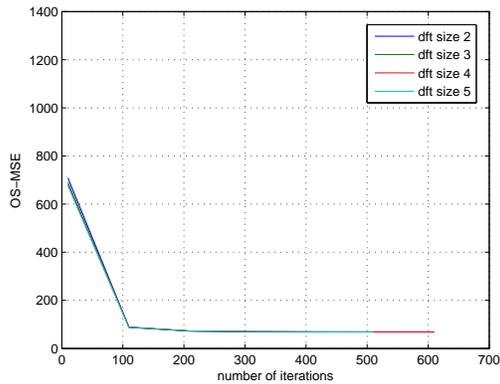


(a) ROS from morphological processing

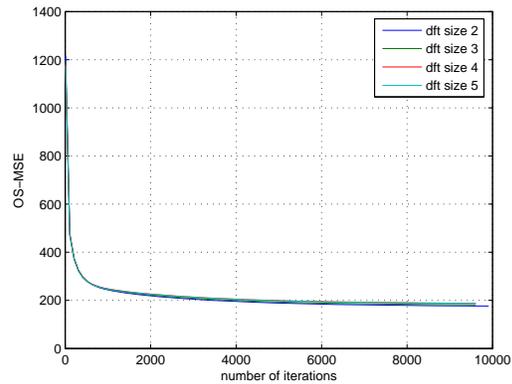


(b) ROS from markers

Figure 5-10: OS-MSE as a function of number of iterations and DFT size for the circle image when the ROS and the size of the blurring function were unknown



(a) ROS from morphological processing



(b) ROS from markers

Figure 5-11: OS-MSE as a function of number of iterations and DFT size for the black moon image when the ROS and the size of the blurring function were unknown

to converge or simply degrade the solution indefinitely. Different DFT sizes did yield different results, but none were acceptable.

These results are summarized in Fig. 5-10, which shows the results using the ROS obtained with morphological processing and the results using markers for the circle image.

Using markers gave much better results with the black moon image, as shown in Fig. 5-11. However, the OS-MSE of the result obtained by using markers was about twice as high as that of the result from using morphological processing. The result obtained using markers also took much longer to converge. The improvement from

the circle image may be because the black field is not as large in the moon image, and therefore using markers gives a tighter bound on the ROS in this case than in the case of the circle image.

### **5.3 Summary**

Three methods of obtaining the ROS were proposed in this chapter. First, the ROS of the degraded image could be used. Second, the ROS could be obtained by morphological processing. Third, the ROS could be artificially defined by the use of markers. Using a ROS obtained by morphological processing puts the tightest bound on the ROS and also allows any noise outside the estimated ROS to be eliminated. The results from this method converge much faster than the results from the other two methods, and they also have the lowest OS-MSE. Therefore, the use of morphological processing to find the ROS is recommended in all cases.

# Chapter 6

## Conclusions and Future Work

An image degraded by blurring can be restored by correcting the phase of its Fourier transform and using its region of support to iteratively reconstruct its magnitude. This method was shown to be much more robust than existing inverse filtering methods in the presence of additive noise. In some situations, this method was shown to be competitive even with image restoration methods, which rely on knowledge of the noise statistics. It was also shown that this method is robust to imperfect knowledge of the blurring function, particularly if the blurring function is zero phase and has a known size.

If the region of support of the image is unknown, it can be found with tolerable accuracy by using morphological processing on the degraded image. Again, this was shown to be robust to noise and imperfect knowledge of the blurring function.

One significant advantage of phase-based deconvolution is that it does not attempt to correct for noise, and therefore it may be particularly useful in situations where the noise statistics are unknown, irregular, or spatially varying. Additional testing is needed to see if this is, indeed, the case. Also, phase-based deconvolution has yet to be characterized in the presence of noise that is not additive. Furthermore, it may be possible to combine phase-based deconvolution with a noise reduction system to achieve better results.

Experiments have shown that the performance of phase-based deconvolution could be improved by incorporating accurate magnitude information. How to obtain such

information and how to incorporate it to realize these improvements is also an open question.

Finally, the experiments in this thesis were somewhat artificial because they took discrete-space images and blurred them with a discrete-space filter. In reality, the blurring will happen in continuous-space and then the continuous blurred image will be sampled by a camera. When there is no aliasing, the discrete-space model used here is valid. However, the effect of aliasing on phase-based deconvolution should be explored to gain a more full understanding of how this method could be applied to real images. In addition, images that are not on black fields will be windowed when captured with a digital camera. The effects of windowing on phase-based deconvolution also have yet to be characterized.

# Appendix A

## Photo Credits

1. bag, pout, tire, rice, coins, and eight: ©Mathworks, imdemo collection
2. cameraman: ©Massachusetts Institute of Technology
3. cell: Alan W. Partin, M.D., Ph.D., Johns Hopkins University School of Medicine
4. circuit: Steve Decker and Shujaat Nadeem, MIT, 1993
5. liftingbody: Public domain image, NASA, 1964-01-01, Dryden Flight Research Center #E-10962, GRIN database #GPN-2000-000097
6. moon: ©Michael Myers
7. westconcordorthophoto: Massachusetts Executive Office of Environmental Affairs, MassGIS



# Bibliography

- [1] D. S. C. Biggs and Mark Andrews. Acceleration of iterative image restoration algorithms. *Applied Optics*, 36(8):1766–1775, March 1997.
- [2] R. C. Gonzalez and R. E. Woods. *Digital Image Processing*. Addison-Wesley, 1993.
- [3] R. J. Hanisch, R. L. White, and R. L. Gilliland. Deconvolution of hubble space telescope images and spectra. In P. A. Jansson, editor, *Deconvolution of Images and Spectra*, volume 2. Academic Press, 1997.
- [4] M. H. Hayes. *Signal Reconstruction From Phase or Magnitude*. PhD thesis, Massachusetts Institute of Technology, June 1981.
- [5] M. H. Hayes. The reconstruction of a multidimensional sequence from the phase or magnitude of its Fourier transform. *IEEE Trans. Acoust., Speech, and Signal Proc.*, 30(2):140–154, April 1982.
- [6] M. H. Hayes, J. S. Lim, and A. V. Oppenheim. Signal reconstruction from phase or magnitude. *IEEE Trans. Acoust., Speech, and Signal Proc.*, 28(6):672–680, December 1980.
- [7] A. K. Jain. *Fundamentals of Digital Image Processing*. Prentice Hall, 1989.
- [8] J. Kalifa, S. Mallat, and B. Rouge. Deconvolution by thresholding in mirror wavelet bases. *IEEE Trans. Image Proc.*, 12(4), April 2003.
- [9] J. S. Lim. *Two-Dimensional Signal and Image Processing*. Prentice Hall, 1990.
- [10] L. B. Lucy. An iterative technique for the rectification of observed distributions. *The Astronomical Journal*, 79(6):745–754, June 1974.
- [11] R. M. Mersereau and R. W. Schafer. Comparative study of iterative deconvolution algorithms. In *Proc. ICASSP '78*, pages 192–195, Tulsa, OK, April 1978.
- [12] R. D. Nowak and M. A. T. Figueiredo. Fast wavelet-based image deconvolution using the EM algorithm. In *Proc. of the 35th Conference on Signals, Systems and Computers*, pages 371–375, Pacific Grove, CA, November 2001.
- [13] W. H. Richardson. Bayesian-based iterative method of image restoration. *Journal of the Optical Society of America*, 62(1):55–59, January 1972.

- [14] V. T. Tom, T. F. Quatieri, M. H. Hayes, and J. H. McClellan. Convergence of iterative nonexpansive signal reconstruction algorithms. *IEEE Trans. Acoust., Speech, and Signal Proc.*, 29(5), October 1981.