Radial Deblurring with FFTs

Except where reference is made to the work of others, the work described in this thesis is my own or was done in collaboration with my advisory committee. This thesis does not include proprietary or classified information.

_____
Christopher B. Webster

Certificate of Approval:

_____
A. Scottedward Hodel
Associate Professor
Electrical and Computer Engineering

_____
Stan Reeves, Chair
Professor
Electrical and Computer Engineering

_____
Thomas Denney
Professor
Electrical and Computer Engineering

_____
George T. Flowers
Interim Dean
Graduate School

Radial Deblurring with FFTs

Christopher B. Webster

A Thesis

Submitted to

the Graduate Faculty of

Auburn University

in Partial Fulfillment of the

Requirements for the

Degree of

Master of Science

Auburn, Alabama
August 4th, 2007

Radial Deblurring with FFTs

Christopher B. Webster

Permission is granted to Auburn University to make copies of this thesis at its
discretion, upon the request of individuals or institutions and at
their expense. The author reserves all publication rights.

_____
Signature of Author

_____
Date of Graduation

VITA

Christopher Blake Webster was born in Flowood, Mississippi on August 14th, 1982. He graduated high school in 2000 from Jackson Preparatory School in Jackson, Mississippi. He began his studies in Electrical Engineering at Mississippi State University in the fall of 2000 and later received his Bachelor's Degree from the Bagley College of Engineering in Spring of 2005. In the fall of 2005, he began his graduate studies in Electrical Engineering (specifically Image Processing) at Auburn University.

Thesis Abstract

Radial Deblurring with FFTs

Christopher B. Webster

Master of Science, August 4th, 2007
(B.S., Mississippi State University, 2005)

62 Typed Pages

Directed by Stan Reeves

Radial blurring (sometimes called zoom blurring) of an image occurs when a camera acquires an image while traveling at a high rate of speed towards an object. Removing this blur is a challenging problem because it is a shift-variant blur. As one travels outward from the center of an image, the blur length increases linearly. This shift-variant characteristic precludes the use of traditional FFT-based deblurring techniques. We propose a way around this problem by transforming the coordinate system of the blurred image into a coordinate system in which the blur is linear shift-invariant (LSI). Equivalently, the blurred image is sampled in a particular nonuniform fashion so that the blur becomes LSI in the new discrete-space coordinates. As a result, the blur can be modeled by convolution so that FFT-based deblurring can be used. In a similar fashion, rotational blur about the center point of an image can also be modeled in this way. We define the overall method as exponentially-sampled radial-space deblurring (ESRSD).

Specific focus is given to identifying the blur percentage of a given image because this value is required to perform the proposed ESRSD method. The blur identification is conducted on the image after the coordinate system has been transformed and nonuniform

sampling has occurred. Using the characteristics of a motion blur in the frequency do-main, a minimization problem is then set up to acquire the blur percentage based on the given data. Results demonstrate that the method yields high-quality restored images in a computationally efficient manner.

Style manual or journal used <u>Journal of Approximation Theory (together with the style</u>
<u>known as "aums"). Bibliograpy follows van Leunen's *A Handbook for Scholars.*</u>

Computer software used <u>The document preparation package TeX (specifically LaTeX)</u>
<u>together with the departmental style-file `aums.sty`.</u>

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

Sometimes referred to as a zoom blur, a radial blur arises when a picture (or video) is taken as an imaging device is rapidly approaching an object of interest. While we have found no literature explicitly addressing radial blurring or deblurring, this blur can occur under certain conditions. Such situations can occur with aerial photography as well as video-based missile systems. Figure 1.1 is a diagram of the origin (and mathematical model) of the blur. The variable x represents the horizontal distance from the camera location to the object
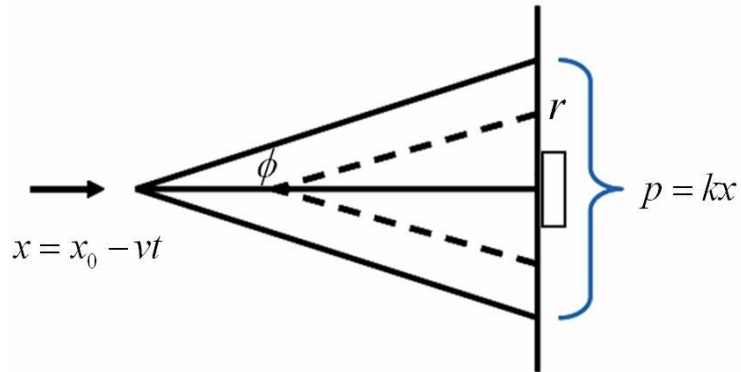
Figure 1.1: Blur Origin

in question, and v is the camera speed. The hollow rectangle is an object of interest, and p can be viewed as the vertical distance in the object plane seen by the camera. As time t increases and the camera approaches the object, p becomes smaller while the size of the object remains the same. This process creates a time-dependent scaling factor of the object

1

in the camera image plane denoted by k. If the exposure time is significant relative to the speed and distance, the image will reflect a blurring of the object as it is progressively scaled and recorded by the camera. The result of this process is the integration (summation) of a sequence of continually scaled images as shown in Figure 1.2.
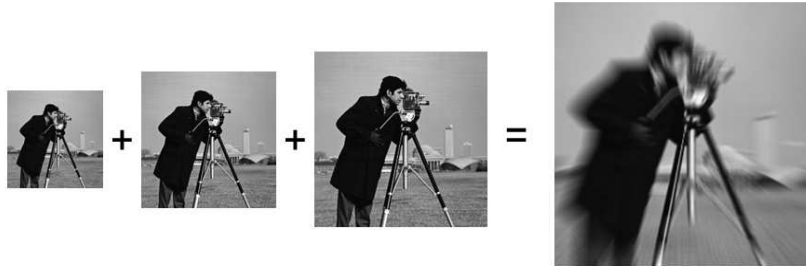
Figure 1.2: Radial blurring

The major challenge with a radial blur is that its effects are shift-variant. Whereas typical motion blurring (in the plane of the camera) occurs when an object moves at a uniform rate during exposure, in radial blurring the apparent motion is different at each point in the object. More precisely, the blur increases linearly as one travels outward from the center of the image. This fact renders traditional convolution-based deblurring methods unusable [1]. In traditional FFT-based restoration algorithms, the system (1.1) must be shift-invariant. Deblurring methods such as inverse filters, regularization filters, and Wiener filters all require a blur to be shift-invariant. We focus specifically on reguarlization filters.

The majority of image degradation problems can be described as a linear blur in the presence of additive noise [2]. Often this degradation is represented as

$$g = Df + n, \tag{1.1}$$

where f and g are the lexicographically ordered original and degraded images, D describes the linear degradation process, and n is random noise (correlated with the image f). Given only g and D, the purpose is then to obtain an estimate $\hat{f}$ of the original image f. An estimate $\hat{f}$ can be chosen that minimizes

$$\left\| g - D\hat{f} \right\|_R^2 = \left(g - D\hat{f} \right)^T R \left(g - D\hat{f} \right), \tag{1.2}$$

where the diagonal matrix R locally weights the restoration process. Problems may arise if either the noise term in 1.1 is significant or D is singular [2]. The minimization $(\hat{f})$ will then be unstable. These ill-posed problems are dealt with through the well-established technique of regularization [3]. Regularization assumes that the true image is smoother than $\hat{f}$, so a "roughness" penalty is imposed on the result. This is accomplished by adding a stabilizing functional on the previous minimization expression. The following expression is then minimized with respect to $\hat{f}$:

$$\psi(\hat{f}) = \left\| g - D\hat{f} \right\|_R^2 + \alpha \left\| L\hat{f} \right\|_S^2, \tag{1.3}$$

where α (a scalar) controls the degree of regularization on the solution, and L is the regularization operator. Note that

$$\left\| L\hat{f} \right\|_S^2 = \left(L\hat{f} \right)^T S \left(L\hat{f} \right), \tag{1.4}$$

where S (a matrix) controls the local degree of smoothing. R and S are both set to identity, and D represents a shift-invariant point-spread function. The frequency domain deblurring is then represented as

$$\hat{F} = \frac{\tilde{D}^*}{\tilde{D} + \alpha \tilde{L}} G \tag{1.5}$$

3

where $\tilde{D}$ is the FFT of the point-spread function, $\tilde{L}$ is the FFT of the regularization kernel, G is the FFT of the blurry image, α is the regularization parameter, and $\hat{F}$ is the FFT of the deblurred image. Areas of low spatial activity require less smoothing than areas with high spatial activity. However, it has been shown that edge structures (areas of high spatial activity) can produce significant ringing artifacts when shift-invariant regularization is used [4]. Spatially varying the regularization can reduce the ringing in the restored image [5].

Iterative algorithms are generally necessary when the blur is not linear shift-invariant (LSI) [6], and these algorithms typically require a great deal more computation than closed-form algorithms based on the FFT. The inability to use FFTs in the deblurring process may be seriously problematic in the case of video-based guidance due to the need for real-time restored images of the object being tracked. If the image can be resampled into a space in which the blur is a constant linear motion blur, then convolution—and hence FFTs—can be used to accomplish deblurring.

A related, well known idea—homomorphic filtering—has been studied by Oppenheim et al. in which they use a pointwise transformation on the dependent axis (intensity of the image) before and after convolution [7]. In exponentially sampled radial-space deblurring (ESRSD), the independent axis (spatial variable) undergoes the transformation instead. That is, the spatial domain is warped so that an operation that was otherwise another type of integral becomes a convolution sandwiched between two geometric transformations.

We propose to transform the coordinate system of the blurred image into one in which the blur is LSI. Equivalently, the blurred image will be sampled in a particular nonuniform fashion so that the blur becomes LSI in the new discrete-space coordinates. As a result, the transformed radial blur can be modeled by convolution so that we can perform closed-form FFT-based deblurring.

Mathematical Development

In this chapter we present mathematical background to be used in formal development of the ESRSD outlined in Chapter 1. The details of ESRSD will be presented in Chapter 3. A system with a shift-variant blur will be geometrically transformed into a new coordinate space in which the blur is represented by a convolution, and hence shift-invariant. The representative equation for the original blurred system is

$$y_\theta(\phi) = \int_0^{t_1} g_\theta(r(\phi, t)) dt, \qquad (2.1)$$

where r is the object-plane radial coordinate, θ serves as the angular coordinate in both the object plane and sensor plane, g_θ is the unblurred image, y_θ is the blurry image, ϕ is the viewing angle associated with a particular pixel in the sensor, and t represents time. Because the blur is one-dimensional for any given θ, the observation equation is expressed with a subscript θ for each y (and subsequently each g). ϕ is taken now to be the view angle of an arbitrary pixel whose linear distance from the center of the object is p. ϕ serves a similar role in the sensor plane to that served by r in the object plane. From the diagram in Figure 1.1,

$$r = x \tan(\phi), \ \frac{dx}{dt} = -v \qquad (2.2)$$

where x is the horizontal spatial coordinate. Through a change of variables, we obtain

$$y_\theta(\phi) = \frac{1}{v} \int_{x_0 - vt_1}^{x_0} g_\theta(x \tan(\phi)) dx \qquad (2.3)$$

To transform the blur to a convolution model, the spatial dimension must be changed so that scaling becomes a shift. In the following equations, a change of variables will be performed so that the limits of integration in 2.3 will become logarithmic terms and the integrand will contain an exponential term. Let

$$z = \log(x) + \log(\tan(\phi)), \ \ e^z = x \ \tan(\phi) \tag{2.4}$$

Then we differentiate 2.4 to obtain

$$dz = \frac{1}{x}dx = \frac{\tan(\phi)}{e^z}dx \tag{2.5}$$

Solving for dx, we now have

$$dx = \frac{1}{\tan(\phi)}e^z dz \tag{2.6}$$

Using equations (2.4), (2.5), and (2.6), we do a change of variables (on the limits of integration and the integrand) on 2.3 to obtain

$$y_\theta(\phi) = \frac{1}{\tan(\phi)} \int_{\log(d_1)+\log(\tan(\phi))}^{\log(d_2)+\log(\tan(\phi))} e^z g(e^z) dz \tag{2.7}$$

where

$$d_1 = x_0 - vt_1, \ \ d_2 = x_0$$

Equation 2.7 may be expressed as a convolution as follows. Define

$$h_\theta(z) = e^z g_\theta(e^z), \ \ \rho_i = \log(d_i), \ \ \alpha = \log(\tan(\phi)) \tag{2.8}$$

Using the definition of α in 2.8, we define $\tilde{y}_\theta(\alpha)$ such that

$$\tilde{y}_\theta(\alpha) = \tilde{y}_\theta(\log(\tan(\phi))) = y_\theta(\phi) \tag{2.9}$$

Substituting the variables h_θ, p_i, and $\tilde{y}_\theta$ into 2.7 results in

$$\tilde{y}_\theta(\alpha) = \int_{\rho_1+\alpha}^{\rho_2+\alpha} h_\theta(z)dz \tag{2.10}$$

For convenience, the limits on integration are expressed as multiplying the integrand by a pulse function in (2.11):

$$\tilde{y}_\theta(\alpha) = \int_{-\infty}^{\infty} [u(z - \alpha - \rho_1) - u(z - \alpha - \rho_2)]h_\theta(z)dz \tag{2.11}$$

We observe that this is equivalent to

$$\tilde{y}_\theta(\alpha) = [u(\alpha - \rho_1) - u(\alpha - \rho_2)] * h_\theta(\alpha) \tag{2.12}$$

Accordingly, (2.12) demonstrates that the forward model has been transformed into convolution with a uniform motion blur in a spatially transformed coordinate system. Since this motion blur is a shift-invariant blur, traditional FFT-based deblurring procedures can now be used to deblur and restore the image. The mathematical analysis now supports the theory. The next chapter discusses the details regarding the sampling and warping of the spatial domain.

CHAPTER 3

EXPONENTIALLY SPACED SAMPLING

3.1 Sampling Parameters

Chapter 2 demonstrates how the mathematical development supports the theory behind
ESRSD; however, the processes of geometric transformation and spatial warping need to
be analyzed in greater detail. To express the blur operation in a discrete-space setting, the
image must be radially sampled with an exponentially-behaved spacing to achieve a shift-
invariant result. By radial sampling, we mean along a line of pixels from one boundary of
the image to the opposite boundary while passing through the center. These lines of pixels
are taken at various angles, and then the lines of pixels are placed in columns to form a
new rectangular image. Because these sample locations do not lie on a regular grid, an
interpolation scheme must be used [8, 9]. A visualization of this sampling pattern is shown
in Figure 3.1. The center of the image is located at (0,0), and the square represents the
boundary of the image. Notice that the points shown in Figure 3.1 are much less dense
than the actual sampling pattern, both in terms of radial points and angles. The reduced
sampling density of Figure 3.1 is used for the clarity of the reader. The procedure of placing
the radial lines of points into columns is shown in Figure 3.2. Forming the columns in this
way produces an image in which the blur is located completely in the vertical dimension.
This 1-dimensional behavior has other added benefits that will be discussed in Section 4.2.

Because the sampling is done radially, two parameters to consider are the quantity
of sampling angles and the relative sample spacing. As the number of angles increases,
the detail retained from the original image data increases. However, this also increases the

8
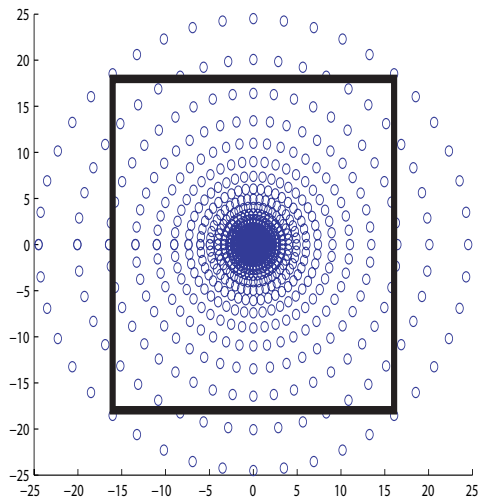
Figure 3.1: Sampling Layout
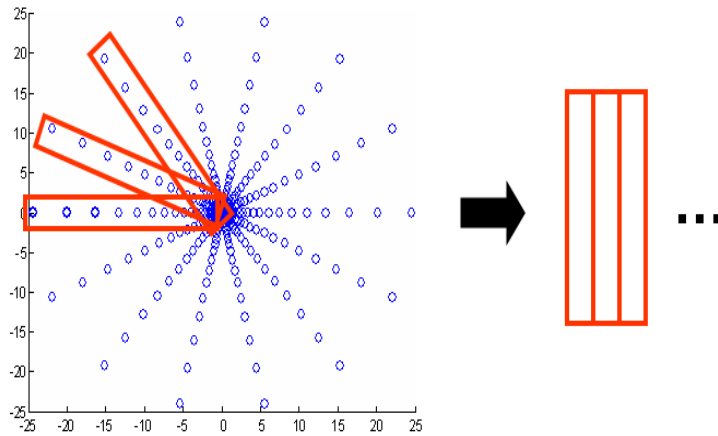
Figure 3.2: Formation of the radial-space image

computational time proportionately. The sampling must be dense enough to fully sample the image at the outer boundaries. The exponentially-behaved spacing of the sampling pattern varies based on the size of the input image. Figure 3.3 is the original image compared to a image after radial sampling and 2-dimensional interpolation. Sufficient sampling densities were used in both angular and radial dimensions to ensure that aliasing was avoided. While there is a slight loss in clarity, Figure 3.3 shows are no significant artifacts introduced from sampling and interpolation procedures.

Figure 3.3: Original and radially-sampled/restored images

The radial spacing at the outer boundaries must be equivalent to the sampling of the blurred image to avoid aliasing. This introduces a tradeoff in that the center of the image does not need as dense a sampling pattern as the outer region of the image. Sampling along the proper exponential curve to avoid aliasing has the effect that the center region of the image is actually oversampled. This excess of data points slows down computation, so a method to address this problem was devised.

3.2 Annular Region Method

In order to properly account for the tradeoff between oversampling and aliasing, the Annular Region (AR) method was developed. In this method the image is segmented into annular regions, and the deblurring procedures are carried out on each region individually. Slight overlap between the regions reduces the effects of artifacts introduced from this new segmentation. By segmenting the image into these regions (Figure 3.5), the sampling density of the inner regions of the image can be less than the sampling density of the outer regions of the image while still following the proper exponential behavior. Figure 3.4 shows a visualization of variable sampling density along one exponential curve. As the regions
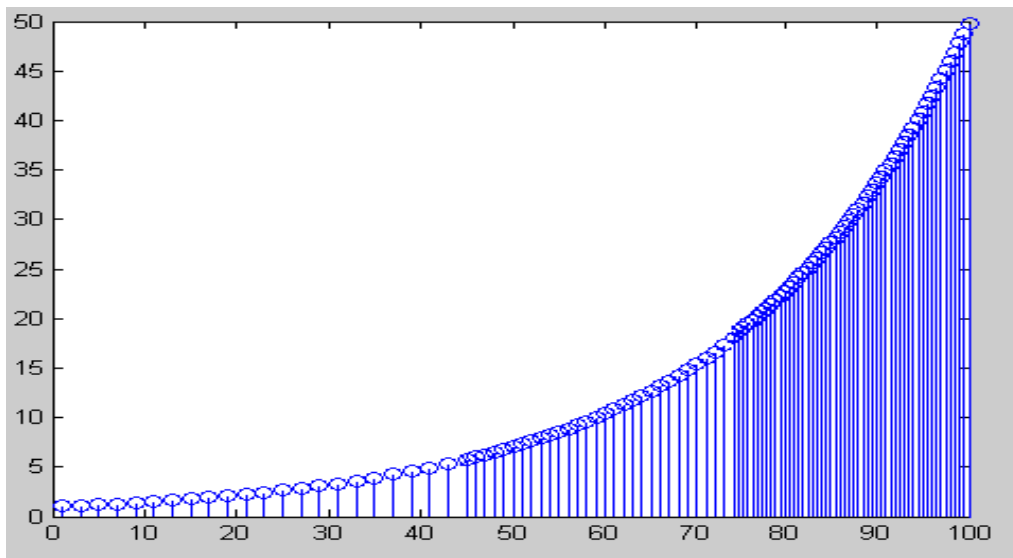
Figure 3.4: Variable Sampling Density

get further from zero (the center of the image), their radial sampling density is increased. The minimum radial sampling density is chosen for each region to avoid aliasing in that region only. In a like manner, angle density can also be varied among the regions. Figure

3.5 shows how the regions are segmented. In this figure, there are three separate regions. For the majority of images, three regions is sufficient to provide significant computational savings with high quality results having well-defined edges and only slight artifacts.
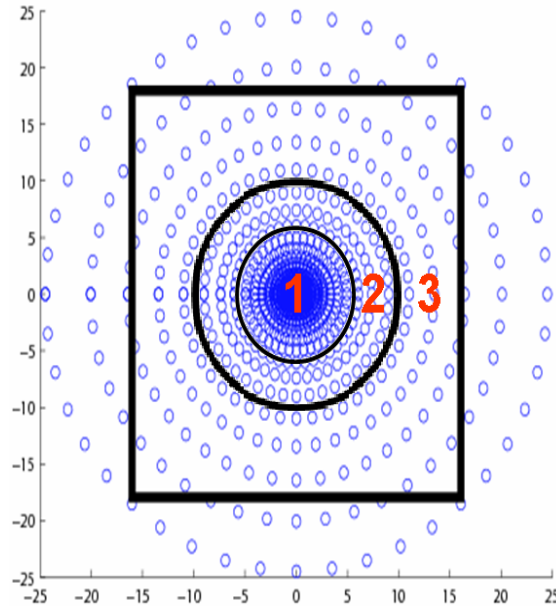
Figure 3.5: Annular Region Segmentation

For large images (500 x 500 or greater), more annular regions may be needed. Choosing the number of annular regions for a given image depends on the type of detail in the image, but thin regions (with respect to the amount of radial points) tend to produce boundary artifacts. Radially-large annular regions tend to slow down computation (due to the interpolation). When using different sampling parameters (radial or angular sampling density) for each region, the radial-space images do not have the same linear dimensions. This means that each region must undergo separately the 2-dimensional interpolation into the original space. Once each region is transformed back into the original space, the regions are then added together to produce the complete deblurred image. This addition is all

12

that is necessary because the values outside the boundary of each region are zero. Special care must be taken to ensure that the borders of the regions line up correctly. This is accomplished by recording the starting and ending borders of each region with reference to the original space. The starting border for each region is the border closest to the center of the image, and the ending border is the border closest the outer boundary of the image. The radial distance to each region border is recorded when the image is first segmented. This is important because during the deblurring procedure, each region has a small overlap into the other regions. A small overlap aids in the elimination of boundary artifacts during reconstruction. After the deblurring is complete, the previously recorded border values are used to piece together the annular regions into a complete image.

One side benefit from using the AR method is that different regularization kernels and parameters can be used for each region. This turns out to be beneficial because of the effect of the regularization parameter α. Let

$$y_r = \text{ifft} \left(\frac{H^*(k)}{|H(k)|^2 + \alpha\,|L(k)|^2} X(k) \right) \tag{3.1}$$

be the regularization function in the frequency domain, where $H(k)$ is the DFT sequence (of index k) of the point-spread function $h[n]$, $L(k)$ is the regularization kernel, $X(k)$ is the FFT of a radial line of points from the blurred image $x[n]$, and y_r is the deblurred radial line of points. As the regularization parameter α decreases in magnitude, the sharpness of the reconstructed image increases. However, the ringing in the image also becomes more pronounced. Therefore, in cases where characteristics are known about the blurred image, the regularization parameter α can be varied to accommodate these characteristics. For example, suppose the center of the image is known to have a high amount of detail. We can choose a small value of the regularization parameter α for the inner region to reconstruct

the inner region as sharply as possible while enforcing a smoother reconstruction of the outer regions.

3.3 Rotational Blur

While the focus of ESRSD is not rotationally blurred images, the mathematical development and the sampling parameter stipulations still hold true for this slightly different type of blur. Figure 3.6 shows a radial-space image and the comparison between radial blur and rotational blur. Once in the radial-space domain, a radial blur corresponds to a vertical motion blur. Similarly, a rotational blur corresponds to a horizontal motion blur in that same domain.
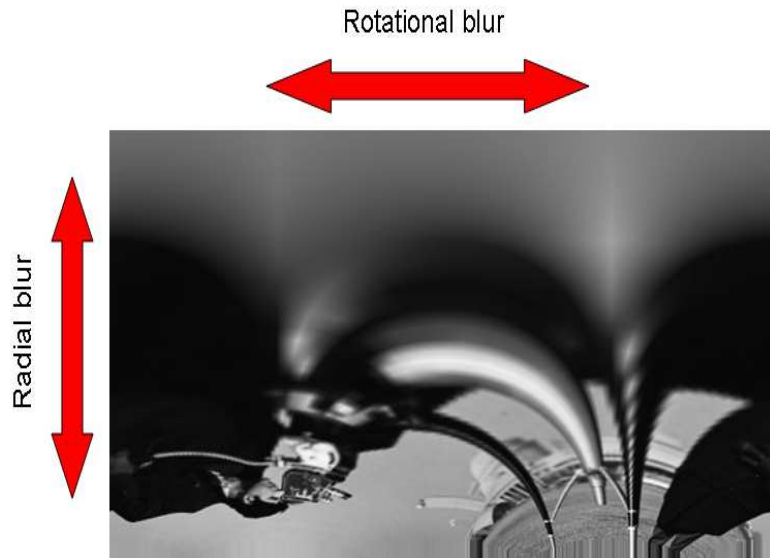
Figure 3.6: Radial and rotational blur comparison

Although mathematically similar to the radial blurred cases, there are a few differences to note regarding rotational blur. When using FFT-based deblurring techniques, the deblurring procedure assumes that the image is periodic. This is not true for most images, so

14

artifact-reducing boundary techniques are usually used. However, in the horizontal dimension, the radial-space image is periodic. Because of this, FFT-based deblurring methods work very well without any pre-processing techniques (such as symmetric extension). The AR method can be applied to rotationally blurred images as well. The segmentation of a rotationally blurred image is easier because no overlap is needed from region to region. In a radial-space image, the blur is orthogonal to the direction of segmentation. Therefore, the segmentation does not produce boundary artifacts during the deblurring procedure. A somewhat disadvantageous difference between the rotational and radial blurring is that the rotational blur has a higher likelihood to extend beyond the original image boundaries. The deblurred image loses significant detail in the regions where this occurs.

Various radial and angular sampling parameters have been analyzed. The AR method of ESRSD demonstrates a way to avoid aliasing while also avoiding oversampling on a region by region basis. On a limited basis, rotational blurring has also been analyzed. Chapter 4 will discuss the implementation of ESRSD with respect to both radial and rotational blurring.

CHAPTER 4

DEBLURRING METHOD

4.1 Procedure

ESRSD begins with the radial sampling of the blurry (and noisy) image (Figure 4.1), as detailed in Chapter 3.

Figure 4.1: Radially blurred image plus 30dB of noise

For comparison purposes, the original unblurred image is shown in Figure 4.2. To obtain the entire image in the sampling process, the maximum radius of sampling must be equal to $\frac{\sqrt{2}}{2}N$, where N is the length of a side of the image. After stacking these radial samples into columns, the resulting image resembles Figure 4.3.

The black arches at the bottom of the image result from sampling at a large enough distance to obtain the corners of the original image. Because the sampling pattern is a circle, some of the sample points will fall outside the image frame. These undefined samples

Figure 4.2: Original image
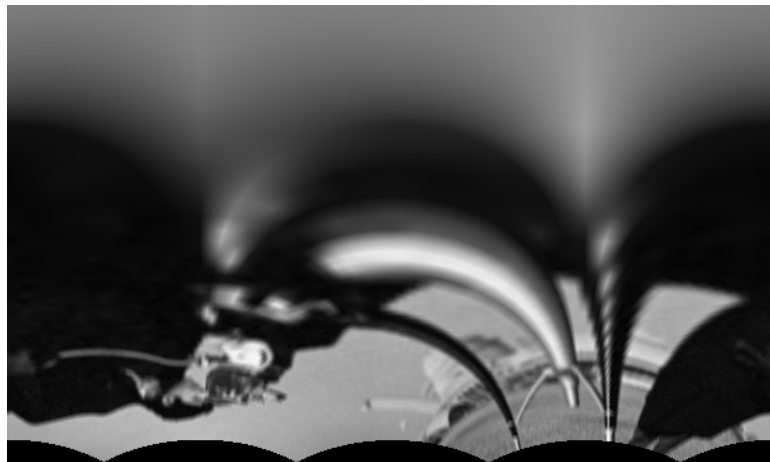
Figure 4.3: Radial-space image

are supplied by replicating the last known good pixel value inside the image in that direction (Figure 4.4). The replication performs well because it produces a smooth transition to the boundary of the image.

Figure 4.4: Radial-space image after border extension

The blur is now a one-dimensional motion blur at an angle of 90 degrees. The radial-space image is then deblurred (columnwise) via a simple FFT-based regularized restoration algorithm. For simplicity, the regularization operator L was set to identity, and the regularization parameter was chosen based on experimentation. The experimentation involved running the same blurry image through the deblurring procedure with varying values of the regularization parameter α. The order of magnitude of the regularization parameter determined the characteristics (smoothness or sharpness) of the result. Symmetric extension was used to extend the boundaries to reduce edge artifacts, although more complex but effective methods are available [10].

When using the AR method, sharp edges in the outer region of the image can produce sharp ringing artifacts. In order to lessen the effect of the ringing on the interpolation to follow, a shift-variant regularization is applied [11]. This regularization takes as its input a

modified edgemap of the radial-space blurry image. The edgemap is modified by extending the edges by 1 pixel in either direction. Figures 4.5 and 4.6 show the difference between the modified and unmodified edgemaps.

Figure 4.5: Radial-space outer region edgemap

Figure 4.6: Radial-space outer region edgemap after edge extension

Recall that the edges are all 1-dimensional (vertical) due to the sampling process. Since the edgemap is formed from the blurry image, the edges are not perfectly sharp. Extending the edges in the edgemap accounts for the fact that a slightly blurred edge may be larger
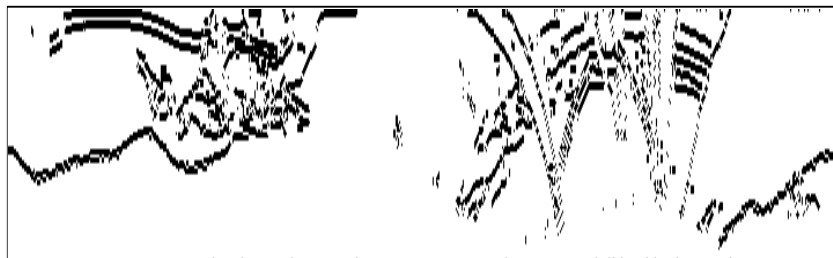
19

than 1 pixel long. Regularization is turned off around the edges so that the smooth regions stay smooth and the edges stay sharp. This also helps lessen the effect of noise during the interpolation procedures. This shift-variant regularization requires a matrix inversion where the size of the matrix is based on the number of edge points in the image [11]. While matrix inversions are typically very computationally costly, all of the operations in this regularization are in one dimension. In a 1-dimensional case, the matrix inversion is reduced to a division by a constant. Therefore, this regularization procedure is very fast when compared to the same procedure on 2-dimensional images (see Section 4.2).

Once the deblurring is complete, the radial sampling is reversed—a 2-dimensional interpolation is performed to fill in the unknown pixels using a nonuniform interpolation algorithm based on Delaunay triangulation [12]. This process (the 2-dimensional interpolation) is the main bottleneck in the procedure, requiring more than 90% of the total computation time. Fortunately, most of the computation can be done in advance to eliminate this bottleneck.

4.2   Computational Analysis

The elements of computational complexity in ESRSD include linear 2-dimensional interpolations, cubic 2-dimensional interpolations, and restoration via FFTs. The AR method has multiple occurences of the interpolation procedures but is otherwise the same. The edge-preserving regularization (EPR) adds some computational complexity of the form of edge detection and regularization. The linear interpolation requires $4mn$ floating point operations (flops) for an $m \times n$ image. All of the FFT-based operations that would normally be conducted in two dimensions for an image can instead be conducted in one dimension. After the interpolation into a radial space (see Figure 4.4), the operations are conducted in one dimension for every column in the image. This reduces the computational cost (for a

20

$p \times q$ image in radial-space) from a 2-dimensional FFT of $.5pq \log_2(pq)$ to a one-dimensional FFT calculated q times, each of $p \log_2(p)$. The relationship between m, n and p, q is

$$p = \frac{\sqrt{2}}{2} n s_r, q = 360 s_a$$

where s_r is the radial sampling density (defined in samples per pixel), and s_a is the angular sampling density (defined in samples per angle). Because the sampling pattern is a circle, 360 is simply the number of degrees in the circle. The AR method produces regions in which $q >> p$, hence the reduction in computational time is very evident. The regularization requires $4pq$ for the matrix multiplications, additions, and divisions (see 3.1). The bulk of the computational cost comes from the second 2-dimensional interpolation. Because the space is warped from the deblurring procedure, a Delaunay triangulation must be used in coordination with another interpolation. These two procedures together require $4(pq)^2$ flops. The triangulation can be computed in a pre-processing stage because the triangulation will not change for images of the same size and AR segmentation. The flop count is reduced to $4pq$ for the remaining interpolation. The iterative EPR algorithm consists of mainly FFTs and matrix multiplies; however, it is run on each column separately as opposed to the entire image at once. The resulting flop count is $pq(11 \log_2(p) + 14)$ per iteration. Some of the calculations are redundant because the EPR algorithm was not modified explicitly for the AR method. The overall flop count of the entire procedure then becomes

$$A \left(4n^2 + 1080\sqrt{2} n s_a s_r \left[2 \log_2(\frac{\sqrt{2}}{2} n s_r) + 180\sqrt{2} n s_a s_r + 3 \right] \right)$$

where the original image is square (i.e., $m = n$), and A is the number of annular regions. Note that this overall flop count includes an iteration of the EPR algorithm which may not be necessary for inner regions in the image. The parameters s_a and s_r can vary from

region to region, and in fact it is beneficial to do so. The inner regions require much less sampling densities than the outer regions, so p and q are much smaller and hence much less computation is required. Without using the AR method, the sampling densities required on the outer region (to avoid aliasing) must be used on the entire image. This introduces an excess of data points for the inner regions, and this excess of data points slows down computation.

The reduced computation leads to a very noticeable decrease in running time. A Delaunay triangulation was run on an entire image (originally 256×256) using the minimum values of s_a and s_r to avoid aliasing. The triangulation took about 53 seconds. The image was then segmented into 3 annular regions, each with minimum values of s_a and s_r with respect to their region. A triangulation was conducted on each of these regions, and the running time of all 3 added together was only 14 seconds.

4.3   Blur Modeling

In order to test the functionality of the process, test images can be constructed via two different methods. The first method involves implementing the discretized blurring model on an unblurred image. The image is sampled radially, then blurred columnwise using FFTs. The image is then reconstructed in its original space using 2-D interpolation. This constitutes a radially blurred image that can then be used to test radial deblurring algorithms. Sufficient radial and angle sampling densities are used to ensure that the sampling and 2-dimensional interpolation do not introduce large artifacts (see Figure 3.3).

The second method forms the blurry image by using resizing and summation. The original image is taken and resized by uniform increments of the scale factor to be larger and larger, and all the resulting images are then added together. The process can be seen in Figure 1.2. This method is useful in that artifacts due to the radial sampling process

can be avoided, and one can test the algorithm when the blurring and deblurring process are not structurally the same. Furthermore, different boundary conditions can be used as compared to the FFT-based radial blurring method.

The rotationally blurred images are formed using the discretized blurring model on an unblurred image. After radial sampling, the image is blurred using FFTs by row instead of by column. Implementing this blur by row instead of by column is the main difference between the rotaionally and radially blurred images (see Figure 3.6).

4.4   Radial Test Results

The procedure by which ESRSD is carried out has been explained, and the following are test results for various images. The different results sections pertain to different ways in which the blurry image was formed. Recall that there are two ways in which the radially blurred images can be formed, and the results for each of these ways are analyzed. Some preliminary results for rotationally blurred images are examined as well.

4.4.1   Results for FFT-blurred Images

The most common degradations in the reconstructed images were ringing around sharp edges and slight loss of clarity in highly detailed areas. These effects intensified as the size of the blur increased. A peak signal-to-noise (PSNR) measurement is used to analyze the quality of reconstruction. Define PSNR as

$$PSNR = 10log_{10}\left(\frac{max_I^2}{MSE}\right) \tag{4.1}$$

where max_I is the maximum intensity value of the image, and MSE is defined as

$$MSE = \frac{1}{mn} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} \|I(i,j) - K(i,j)\|_2^2 \qquad (4.2)$$

for two images I and K of size $m \times n$.

Figure 4.7 shows an image blurred by 10% (plus 30dB noise), and Figure 4.8 shows the deblurred image without using the AR method. The PSNR of Figure 4.7 is 16.23 dB, and the PSNR of Figure 4.8 is 20.17 dB. Note that all of the PSNR measurements listed are referenced to the original unblurred and noise-free image. Although the details and edges

Figure 4.7: Radially blurred image plus 30 dB of noise

of the cameraman himself seem to be preserved, this is overshadowed by the presence of artifacts present over the entire image. Figure 4.9 shows an image deblurred with the AR method but without any EPR (edge-preserving regularization). The PSNR of this result is 20.33 dB.

Figure 4.10 is the result when the edge restoration algorithm is used to clean up the ringing and restore the sharp edges. Slight artifacts are noticeable, but the visual improvement is very apparent. The PSNR of this result is 22.78 dB, which is also an improvement over the previous images.

Figure 4.8: Deblurred image without AR method

Figure 4.9: Deblurred image with AR method but no EPR

Figure 4.10: Deblurred image with AR method and EPR for 10% blur

Figure 4.11 shows a slightly smaller percentage blur and the reconstruction with the AR method and EPR. The artifacts are almost completely gone from this image, and the PSNR is up to 23.51 dB. As the blur percentage increases past 10%, more artifacts are introduced, and noise has a much greater effect on the reconstruction.

Figure 4.11: FFT-blurred and restored images for 6% blur

The artifacts in the reconstructed image in Figure 4.12 are much more pronounced than in Figures 4.10 and 4.11. Any reconstruction on a blur greater than 14% produces artifacts that dominate the entire image. Excessive smoothing parameters can be used in an attempt to counter this, but the cost to visual sharpness is very high. Even though the PSNR is still 22.56 dB, the visual clarity is much worse than Figures 4.10 and 4.11.

Figure 4.12: FFT-blurred and restored images for 14% blur

The method was tested on various other images as well (all with 30 dB noise). Figures 4.13 and 4.14 are two images in which real world radial blur may occur. Figure 4.13 demonstrates that the method can perform well on an image such as a jet in flight. The method was also tested against a natural background, as shown in Figure 4.14. The PSNR of Figure 4.13 is 24.84 dB, while the PSNR of Figure 4.14 is 25.14 dB. These values are slightly higher than the cameraman PSNR values because there is a much lower level of detail in Figures 4.13 and 4.14.

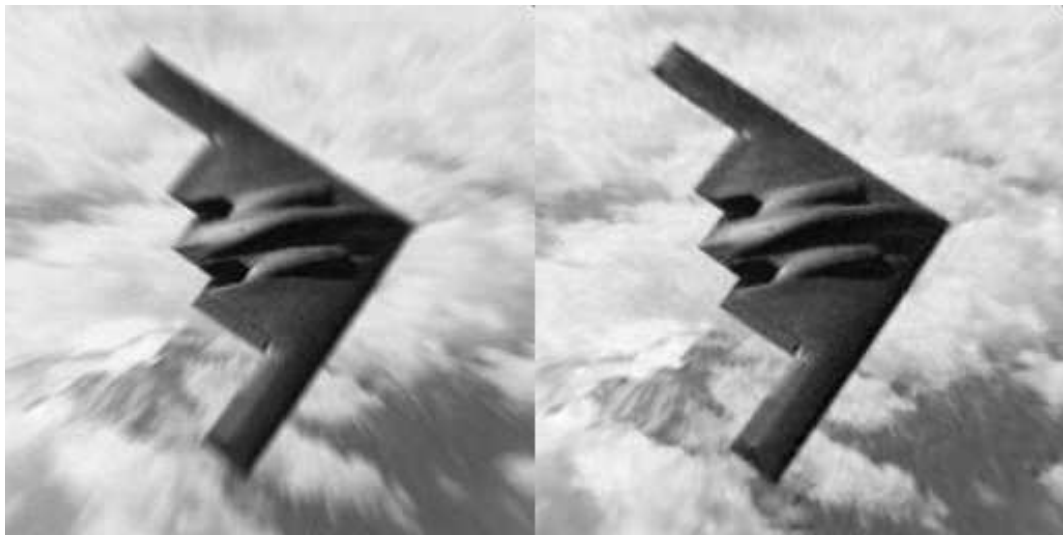Figure 4.13: FFT-blurred and restored images for 10% blur

Figure 4.14: FFT-blurred and restored images for 10% blur

4.4.2 Results for Summation-formed Images

To test the diversity of the AR method of ESRSD (with EPR), the summation-formed images were also deblurred. By diversity, we mean the ability to handle different models of radial blurring. Because the interpolation method itself produces errors, it is important to establish that the method will work when the method used to form the image is not exactly the same as the interpolation used in the deblurring algorithm. Figure 4.15 shows a summation-formed image (plus 30 dB noise) and the subsequent deblurred image via the AR method (with EPR). The PSNR of this result is 23.01 dB.

Figure 4.15: Summation-blurred and restored images for 6% blur

Figures 4.16 and 4.17 show blurry and deblurred images for higher blur percentages. The PSNR for Figures 4.16 and 4.17 are 21.99 dB and 21.47 dB, respectively. Note that these values are consistently lower than the results for the FFT-formed images.

Other blurry images were formed to test the robustness of the method as well as the modeling ability of the summation process. Figures 4.18 and 4.19 show real-world images that could experience radial blurring. Figure 4.18 tests the procedure on a summation-formed image of a jet; however there is an artificial background. The background in Figure

Figure 4.16: Summation-blurred and restored images for 10% blur

Figure 4.17: Summation-blurred and restored images for 14% blur

4.19 more closely resembles a real-world background of an image taken in flight. The PSNR of Figure 4.18 is 24.01 dB, while the PSNR of Figure 4.19 is 24.73 dB.

Figure 4.18: Summation-blurred and restored images for 14% blur
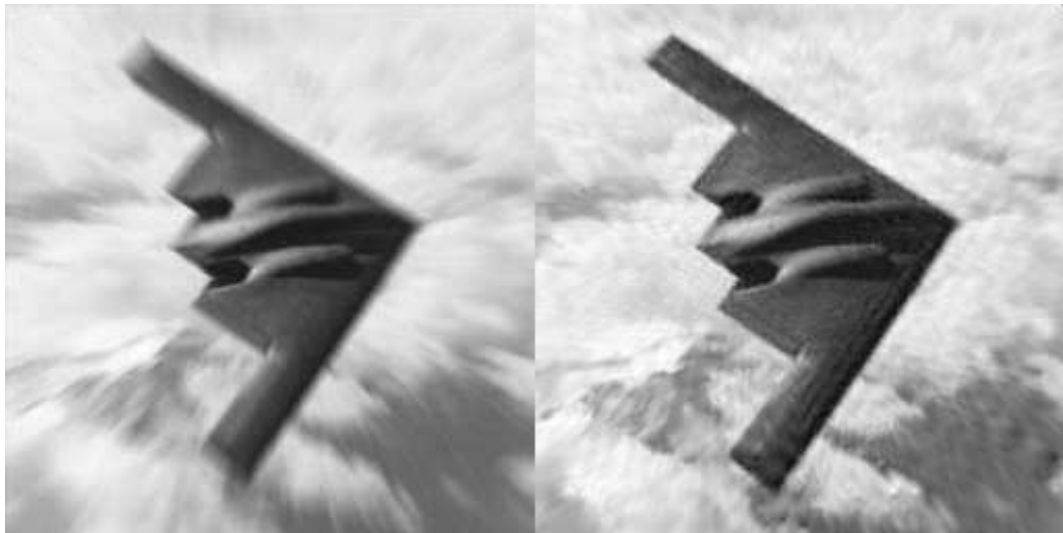

Figure 4.19: Summation-blurred and restored images for 14% blur

4.5 Preliminary Rotational Test Results

The AR method of ESRSD was tested on various rotationally blurred images as well. The procedure was not specifically modified to handle these types of blurred images, so the results are not quite as high-quality as the radially deblurred results. Figure 4.20 shows a rotationally blurred and restored image for a relatively small blur. The PSNR of this result is 27.02 dB. Figures 4.21 and 4.22 show steadily increasing blur sizes and the

Figure 4.20: Blurred and restored images for a 2.5° blur

resulting deblurred images. The PSNR of Figures 4.21 and 4.22 are 26.43 dB and 25.68 dB, respectively. As with radial blurring, the artifacts resulting from deblurring steadily increase with respect to the severity of the blur. As a worst-case scenario, Figure 4.23 shows a very severe blur and reconstruction. Due to the extreme amount of blur, there are major artifacts present in the resulting image. The PSNR of this image drops down to 24.95 dB. The PSNR values of these deblurred images are actually consistently higher than their radially deblurred counterparts. This is because the original image does not need to be altered in any way when making the error measurements. This is not the case with the radially blurred images, as detailed in Chapter 6.

Figure 4.21: Blurred and restored images for a 4° blur

Figure 4.22: Blurred and restored images for a 5.5° blur

Figure 4.23: Blurred and restored images for a 7.5° blur

5.1 Power Spectrum Method

ESRSD requires that the blur percentage be known. In a real-world scenario, the blur percentage will depend on the distance to the object as well as the relative velocity between the two objects, either of which may be unknown. Therefore, a blur identification technique was developed. This technique is based on the fact that after the intial warping of the image into a radial-space domain, the blur can be modeled as a motion blur in one dimension. In the time domain, a motion blur is represented as a pulse. The Fourier transform of a pulse in the time domain is a sinc function in the frequency domain. Using a power spectrum method approach, the existing data is compared to a blur model for various blur amounts [13]. Let D be the average power spectrum of the columns:

$$D(k) = \sum_{i=0}^{N-1} |X_i(k)|^2, \tag{5.1}$$

where $X_i(k)$ is the FFT of the ith column of the radial-space image of dimension $M \times N$. We seek to match a model $\tilde{M}(k)$,

$$\tilde{M}(k) = \left(\frac{\sin(L\pi k/N)}{L\pi \left|1 - \alpha e^{-j2\pi k/N}\right|^2 k/N} \right)^2 \tag{5.2}$$

to the power spectrum term D using normalized error criterion, where $k = 0 : M - 1$, L is the blur length, and α is a scalar that controls how quickly the overall frequency content

rolls off. For a given D and $\tilde{M}$, we define a scaling term A to be

$$A = \frac{\sum_k \tilde{M}(k)D(k)}{\sum_k \tilde{M}(k)^2} \tag{5.3}$$

The modeling error is then represented by Q, where

$$Q = \frac{\sum_k \left| D(k) - A\tilde{M}(k) \right|^2}{\sum_k \left| A\tilde{M}(k) \right|^2} \tag{5.4}$$

Now, 5.4 can be minimized with respect to A to determine the proper scaling for the model $\tilde{M}$.

Through experimentation, it was discovered that both the power spectrum and generalized cross-validation (GCV) methods work best when the outer annular region of the image is used because the resolution of the blur is greater in this region. Also, it was found that symmetrically extending X_i in the vertical dimension made the identification more accurate. Symmetric extension provides a smooth boundary condition to prevent spurious high frequencies. Figure 5.1 shows semilog plots of the data D and the model (times the scale factor) $A\tilde{M}$, respectively. This comparison is for an exact match (between model and data) on the blur percentages. Note that "blur percentage" and "blur length" are merely different representations of the same effect. The blur percentage takes into account the radial sampling density used.

The differences between the model and the actual data arise from noise and interpolation, with noise being the main contributer. As the noise level increases, the identification becomes less and less reliable. However, this method does have a small tolerance which will be discussed in the section 5.3 of this chapter.

Figure 5.2 shows a 8% blur length difference between the model and the data. In Figure 5.1, the zeros of the data and the model line up exactly. In Figure 5.2, there is a slight
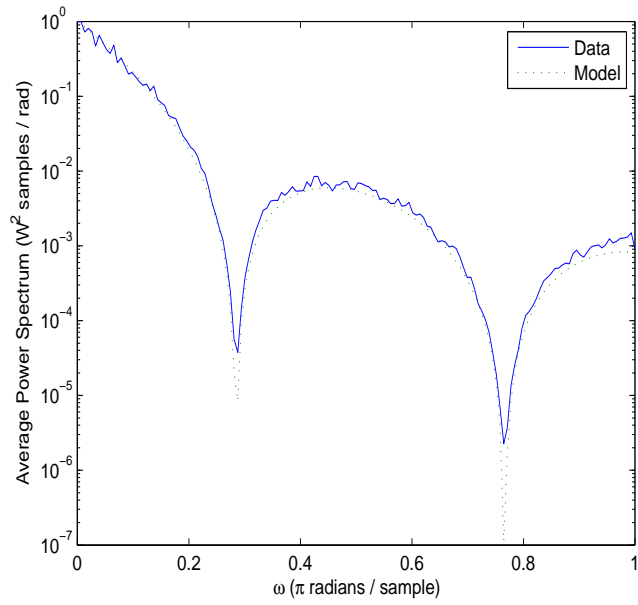
Figure 5.1: Model and data comparison for matching blur lengths

misalignment in the zeros of the data and the model. This misalignment produces a larger error between the data and the model as compared to Figure 5.1. Comparing these error values over a range of models produces a minimum at the correct blur value. The minimum value obtained from this method is actually a blur length (corresponding to a motion blur). Recall that this length is related to a percentage blur based on the sampling density used in the interpolation.

Figure 5.2: Model and data comparison for 8% difference in blur length

5.2 Generalized Cross-Validation

A slightly more complex blur identification method is that of generalized cross-validation (GCV). This parametric identification technique handles more general blu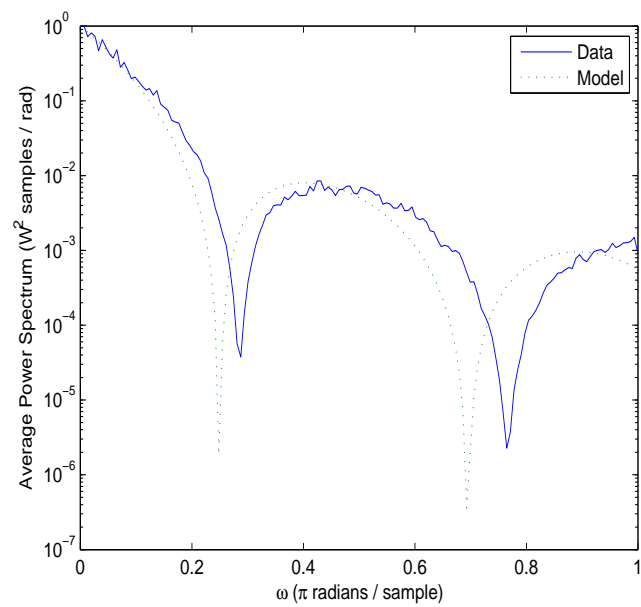rs than that of the power spectrum method. The image is modeled as an autoregressive (AR) process. To distinguish the effects of the blur from the effects of the image, the blur is then modeled as a moving-average (MA) process [14]. An autoregressive moving-average (ARMA) model is used to represent the blurred image, and the point-spread function (PSF) of the blurred image is represented by the MA process. The blur identification problem becomes a 2-D ARMA model parameter identification [15]. In this way, determining the parameters of the ARMA model produces a blur identification [14]. Traditionally, GCV functions as a criterion to estimate the optimal regularization parameter in smoothing problems [16]. GCV does this by determining the parameter or parameters that minimize a weighted sum of prediction errors. Certain properties of this criterion are superior to other parametric methods (such as Maximum Likelihood) with regards to regularization parameter estimation [17]. GCV can then be expanded from image restoration (optimal smoothing) to blur identification problems [18].

Similar to the power spectrum method, GCV performs best when we symmetrically extend the outer region of the image before passing it through the blur identification process. Define

$$T = \sum_{k,l} \frac{\alpha U_{k,l}^2}{|H_{k,l}|^2 + \alpha U_{k,l}^2} |X_{k,l}|^2 \tag{5.5}$$

and

$$B = \left(\sum_{k,l} \frac{\alpha U_{k,l}^2}{|H_{k,l}|^2 + \alpha U_{k,l}^2} \right)^2 \tag{5.6}$$

where X is the FFT of the blurry image (of dimension $k \times l$), U is the FFT of the regularization kernel, H is the FFT of the point spread function (based on blur length input L), and α is the regularization parameter. A minimization is performed over V with respect to L and α, where

$$V = \frac{T}{B} \tag{5.7}$$

Division by zero is impossible because the regularization parameter α is not allowed to be zero, and the regularization kernel U will not be chosen such that $U_{k,l} = 0$ for all k, l.

5.3   Results

Both methods of blur identification produce desirable results. We examine and analyze an identification example using each method. In each method, there are two parameters (α and L) over which the function must be minimized. The power spectrum method was tested on the previously mentioned noisy and 10% blurred image (see Figure 4.7). Figure 5.3 is an image object where the pixel colors represent the magnitude of the power spectrum function for a given α and L.

The minimum value over this range of α and L occurs at a value of $\alpha = .5$ and $L = 5$. Recall that these methods deal with blur lengths and not blur percentages. Because of the sampling density used for this image, a blur length of 5 corresponds to a 10% blur. Therefore, the identification is correct. One thing to note is that a value very close to the minimum occurs at $\alpha = .6$ and $L = 4.75$. In some cases, the minimum value does not occur at exactly the correct blur length. Deblurring with an improper blur length will be examined later in the section.

GCV was tested on the previously mentioned noisy and 14% blurred image (see Figure 4.12). For this image and sampling density, a blur length of 7 corresponds to a 14% blur.
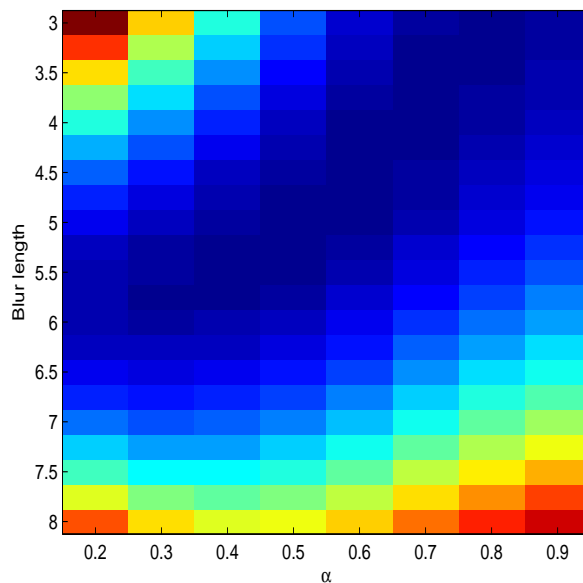
Figure 5.3: Image object of power spectrum function output

Figure 5.4 is an image object representing the magnitude of the GCV function for a given α and L.

The minimum value over this range of α and L occurs at a value of $\alpha = .6$ and $L = 7$. Again, the identification is correct. Although both methods work equally well with regard to accuracy, the power spectrum method runs slightly faster. If the number of floating point operations is considered, it is obvious to see why this is the case. The number of calculations required to compute D and $A\tilde{M}$ is $O(m(nlog(m) + 8))$ for an image of dimension $m \times n$. Comparatively, the number of calculations required to compute T and B for the GCV method is $O(m(3nlog(m) + 15n))$ for an image of dimension $m \times n$.

To test the effect of mismatching blur and deblur lengths, Figure 4.7 is deblurred with a slightly miscalculated blur length. Suppose the minimum that was reached through the identification was $\alpha = .6$ and $L = 4.75$. Figure 5.5 shows a comparison between two deblurred images for values of $L = 5$ and $L = 4.75$, respectively.
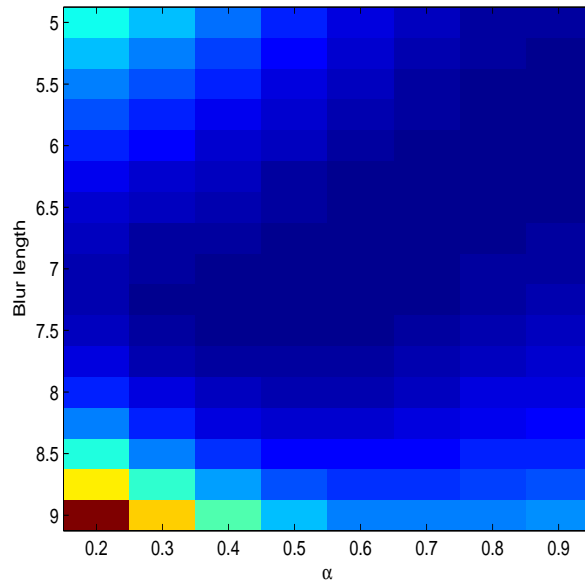
Figure 5.4: Image object of GCV function output

Figure 5.5: Image comparison for deblur lengths of 5 and 4.75

There are very few noticeable differences between the two images. The PSNR values of the two images are 22.80 dB and 22.77 dB, respectively. Even when the blur is not exactly known, the deblurring procedure produces a high quality result. The tolerance for a given blur length is around $\pm 10\%$ (determined experimentally). Once the identified minimum strays too far from the actual blur length, the results begin to degrade. Figure 5.6 shows a comparison between two deblurred images for values of $L = 5$ and $L = 4.25$, respectively.

Figure 5.6: Image comparison for deblur lengths of 5 and 4.25

The loss of detail between the two images is very apparent. The PSNR value of the image deblurred with $L = 4.25$ is 22.61 dB. Although large shapes are still recognizable, the loss of sharpness and introduction of ringing are becoming more and more apparent. This trend continues as the deblur length mismatched to a larger degree. As another example, Figure 5.7 shows a comparison between two deblurred images for values of $L = 5$ and $L = 5.25$, respectively.

This comparison demonstrates that small differences (such as ringing) may become recognizable when the exact blur length is not used to deblur. The results are still high-quality, but not as high-quality as when the exact blur length is used.

Figure 5.7: Image comparison for deblur lengths of 5 and 5.25

The further compare the two identification methods, a series of 100 identification trials was performed in order to analyze the consistency of the each identification method. The image contained 30 dB noise, however a different randomization of noise was used for each trial. The correct blur length for these trials was 7. The mean minimization over 100 trials for the power spectrum method was 7.097, and the standard deviation was .0103. For GCV, the mean minimization was 7.1167, and the standard deviation was .0114. Both identification methods actually overshoot the actual blur length, but each one produces a blur length well inside the tolerance allowed by the deblurring procedure. Each method was also tested on an image with a correct blur length of 5 and 30 dB noise. The mean of the power spectrum method was 4.9361, and the standard deviation was .0092. The GCV method produced slightly better results, with a mean of 4.947 and a standard deviation of .0088.

The standard deviation and mean both confirm the consistency of the blur identification methods. The standard deviation is less than 1%, and the mean minimization is within the acceptable tolerance of ESRSD. Even though the results are not exact, they are close enough

44

for ESRSD to achieve a high-quality result. The results also suggest that GCV produces more accurate results at lower blur lengths, whereas the power spectrum method works better at higher lengths.

CHAPTER 6

CONCLUSION

In this work, we have discussed the challenging problem of radial deblurring. Handling shift-variant blurs is difficult because of the inability to use FFTs to deblur. Warping the spatial domain to transform a shift-variant blur to a shift-invariant blur is an effective solution to this problem. The early implementations of ESRSD used a warping was over the entire image at once. Later implementations (AR method) demonstrated that dividing the image up into annular regions produced better results than just one complete warping. The AR method also allowed us to use variable sampling rates and regularization parameters for each region in the image so that computational time was saved and excessive sampling was avoided. Another benefit of this method is that 3 separate 2-dimensional interpolations can be performed. These 3 separate interpolations reduce the computation significantly because the number of data points needed is reduced to a minimum for each region. The Delaunay triangulation portion of the interpolation can be computed once in a pre-processing stage, and this process takes more than 90% of the total computational time. Even with this pre-processing, the total running time for 256 x 256 image is 1.2 seconds without any edge restoration. Edge restoration takes another 2-4 seconds per iteration (on a P4, 3.2 GHz machine). While the method works well in producing high-quality deblurred images, the real-time capability of the algorithm is not yet realized.

The PSNR values for the deblurred images were not quite as high as we expected. Obtaining these measurements introduced some errors because the two images which were being compared were not the same size. Dimensionally they had the same number of pixels

(256×256), but the object inside the image was not the same size due to the deblurring procedure. The original image had to be resized to match the deblurred image. The resized image was then cropped to dimensionally match the deblurred image.

The blur identification procedure produces accurate results within $\pm 5\%$ of the actual blur value on average. This is ideal because the deblurring method has an error tolerance greater than this in almost all cases. The identification function only takes .1 seconds to run, so the time required to obtain a minimum depends on the predetermined range of two parameters over which the function is to be minimized. This range can be made very small after acquiring certain characteristics about how the image is obtained.

The rotationally deblurred images are not as high quality as the radially deblurred images; however, this is to be expected. The method in question was not specifically tuned for these types of images; however, minor test cases were conducted as a preliminary proof of concept. The mathematical development holds true for these cases, and the results back up the mathematical development.

CHAPTER 7

PROPOSED FUTURE WORK

The proposed method has been tested on a wide variety of images and blur percentages. These images represent situations in which a real-world radial blur could occur. The method performs well in deblurring these images; however, the procedure is still not computationally fast enough to be conducted in real time. We hope to streamline the procedure enough so that it can be implemented in a real-time scenario. In cases where blur identification is not needed, the streamlining would consist of reducing the computational time of the 2-dimensional interpolations and the FFTs. The blur identification computational time can be decreased by narrowing the range of of the blur length over which the function is to be minimized. α may be able to be treated as a fixed value in some cases, but more research is needed to determine this. The EPR algorithm has also not been optimized for speed. The algorithm was chosen for its functionality, therefore a faster or more efficient procedure may be available.

This procedure has thus far only been tested on images that are modeled to have a radial blur. In the future, we hope to obtain some images (or video) that has actually been radially blurred during acquisition. Examining the performance of the method on these images will help further perfect the radial blur model and make it possible to more accurately identify blur percentages.

Although the main focus of this method is radial deblurring, it has been shown that the method will work for rotational deblurring as well. With minor tweaking, the method performs reasonably well on rotationally blurred images as it is. Certain characteristics

about rotationally blurred images can be analyzed to further modify the method. Obtaining real-world rotationally blurred images would also allow for a more in-depth analysis of the rotational blur model as well.

BIBLIOGRAPHY

[1] H. C. Andrews and B. R. Hunt, *Digital Image Restoration*, Prentice-Hall, New Jersey, 1977.

[2] S. J. Reeves and R. M. Mersereau, "Regularization parameter estimation for iterative image restoration in a weighted Hilbert space," *Proceedings of IEEE on ASSP*, pp. 1885-1888, April 1990.

[3] A. N. Tikhonov and V. Y. Arsenin, *Solutions of Ill-Posed Problems*, Winston, 1977.

[4] S. J. Reeves, "Optimal space-varying regularization in iterative image restoration," *IEEE Transactions on Image Processing*, vol. 3, pp. 319-324, May 1994.

[5] R. L. Lagendijk, J. Biemond, and D. E. Boekee, "Regularized iterative image restoration with ringing reduction," *IEEE Transactions on ASSP*, 36(12): 1874-1888, Dec. 1988.

[6] Al Bovik, Ed., *Handbook of Video and Image Processing*, chapter Iterative Image Restoration, pp. 235–252, Academic Press, Maine, second edition, 2005.

[7] A. V. Oppenheim, R. W. Schafer, and T. G. Stockham, Jr., "Nonlinear filtering of multiplied and convolved signals," *Proceedings of the IEEE*, vol. 56, no. 8, pp. 1264–1291, August 1968.

[8] H. H. Hou and H. C. Andrews, "Cubic splines for image interpolation and digital filtering," *ASSP*, vol. ASSP-26, pp. 508–517, 1978.

[9] M. Unser, A. Aldroubi, and M. Eden, "Fast b-spline transforms for continuous image representation and interpolation," *PAMI*, vol. PAMI-13, pp. 277–285, 1991.

[10] S. J. Reeves, "Fast image restoration without boundary artifacts," *IEEE Transactions on Image Processing*, vol. 14-10, pp. 1448–1453, 2005.

[11] R. Pan and S. J. Reeves, "Efficient Huber-Markov edge-preserving image restoration," *IEEE Transactions on Image Processing*, vol. 15-12, pp. 3728–3735, December 2006.

[12] C. B. Barber, D. P. Dobkin, and H. T. Huhdanpaa, "The Quickhull algorithm for convex hulls," *ACM Transactions on Mathematical Software*, vol. 22, no. 4, pp. 469-483, Dec. 1996.

[13] T. G. Stockham, Jr., T. M. Cannon, and R. B. Ingebretsen "Blind deconvolution through digital signal processing," *Proceedings of the IEEE*, vol. 63, no. 4, pp. 678–692, April 1975.

[14] S. J. Reeves and R. M. Mersereau, "Blur Identification by the method of generalized cross-validation," *IEEE Transactions on Image Processing*, vol. 1-3, pp. 301–311, 1992.

[15] A. M. Tekalp, H. Kaufman, and J. W. Woods, "Identification of image and blur parameters for the restoration of noncasual blurs," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. ASSP-34, pp. 963–972, August 1986.

[16] P. Craven and G. Wahba, "Smoothing noisy data with spline functions — estimating the correct degree of smoothing by the method of generalized cross-validation," *Numerische Mathematik*, no. 31, pp. 377–403, 1979.

[17] G. Wahba, "A comparison of GCV and GML for choosing the smoothing parameter in the generalized spline smoothing problem," *Annals of Statistics*, vol. 13, no. 4, pp. 1378–1402, 1985.

[18] S. J. Reeves and R. M. Mersereau, "Optimal estimation of the regularization parameter and stabilizing functional for regularized image restoration," *Optical Engineering*, vol. 29, pp. 446–454, May 1990.