Investigation of a Localized Approach to Shift-Variant Image Restoration and Robust Autofocusing

A Dissertation Presented

by

Younsik Kang

to

The Graduate School

in Partial Fulfillment of the

Requirements

for the Degree of

Doctor of Philosophy

in

Electrical Engineering

Stony Brook University

State University of New York

at Stony Brook

2011

Copyright by
Younsik Kang
2011

Stony Brook University

The Graduate School

Younsik Kang

We, the dissertation committee for the above candidate for the
Doctor of Philosophy degree,
hereby recommend acceptance of this report.

Muralidhara Subbarao, Preliminary Examination Advisor
Professor, Department of Electrical & Computer Engineering

Sangjin Hong
Associate Professor, Department of Electrical & Computer Engineering

---

Monica Fernandez-Bugallo
Assistant Professor, Department of Electrical & Computer Engineering

---

Jeffrey Ge
Professor, Department of Mechanical Engineering

This dissertation is accepted by the Graduate School

Dean of the Graduate School

Stony Brook University

The Graduate School

Younsik Kang

We, the Dissertation committee for the above candidate for the
Doctor of Philosophy degree,
hereby recommend acceptance of this report.

Muralidhara Subbarao, Dissertation Advisor
Professor, Department of Electrical & Computer Engineering

Sangjin Hong
Associate Professor, Department of Electrical & Computer Engineering

Monica Fernandez-Bugallo
Assistant Professor, Department of Electrical & Computer Engineering

Jeffrey Ge
Professor, Department of Mechanical Engineering

This dissertation is accepted by the Graduate School

Lawrence Martin
Dean of the Graduate School

Abstract of the Dissertation

Investigation of a Localized Approach to Shift-Variant Image Restoration and Robust Autofocusing

by

Younsik Kang

Doctor of Philosophy

in

Electrical Engineering

Stony Brook University

2011

Images of three-dimensional (3D) scenes or dynamic scenes captured by a digital camera are in general blurred by different degrees at different points in the image. The blur level at a pixel depends on the point spread function (PSF) of the camera near that pixel. The blur may be due to both defocus and motion blur effects. The PSF itself depends on the shape, distance, and motion of objects in the scene as well as imaging parameters of the camera such as focal length, position of image sensor, and camera aperture. Images where the blur PSF changes with spatial position of pixels are said to be shift-variant blurred images. In some special cases such as a planar object perpendicular to the direction of view, the PSF does not change but remains the same at all pixels. In this case the image is said to be a shift-invariant

or convolution blurred image. Image restoration or deblurring is the problem of recovering the focused image from blurred images in the presence of noise. This thesis addresses two closely related problems. One is determining the distance of objects through image defocus analysis for rapid autofocusing of digital cameras. Another is image restoration or recovering the focused image from shift-variant blurred images.

Shift-invariant deblurring or deconvolution is a special case of the shift-variant deblurring. Application of this special case to autofocusing of digital cameras is presented in this thesis. In particular, we present a new Robust Depth-from-Defocus (RDFD) method for autofocusing that combines the advantages of both Depth-from-Focus (DFF) and Depth-from-Defocus (DFD). RDFD does not require pixel correspondence between samples and determines object depth by analyzing local energy of image. RDFD is shown to be computationally faster and robust in the presence of small image shifts due to camera motion and hand-shake during image capture. A new polynomial model for Focus Measure (FM) is proposed. This new model approximates actual focus measure as a polynomial of a blur parameter σ. This model can be used to predict the variation of focus measure with camera parameters near the focal position where the blur level is small.

Many methods have been proposed to deblur or restore shift-variant blurred images. These methods typically suffer from exorbitant computational cost and memory requirement. Matrix approach is a widely used method for shift-variant image restoration. It formulates shift-variant system as a matrix equation and solves it by inverting a large matrix. Both computational cost and noise sensitivity are serious problems. An $N \times N$ image requires inverting an $N^2 \times N^2$ matrix, which is an $O(N^6)$

operation. Therefore computational cost becomes very large even for small images. This method is highly noise sensitive and mere quantiztion noise alone can result in large errors in image restoration. Regularization is needed to reduce noise sensitivity which requires additional computation and complication in the form of selecting optimal regularization technique and parameter.

In order to reduce computational cost, some methods use local piecewise shift-invariance approximation in small image blocks to model shift-variant blurring. Each image block is restored separately through deconvolution in the Fourier domain using the Fast Fourier Transform. The resulting restored image blocks are synthesized to reconstruct the whole image. Due to the inherent approximation used, the error in restoration could be large. In addition, this method could result in highly visible blocking artifacts or discontinuities near the boundary between image blocks.

This thesis investigates a new approach to shift-variant image restoration based on a complete localization of the blurring process using Rao Transform (RT). This approach is fully localized, non-iterative, and effective. It assumes a local polynomial model for the focused image, e.g. a bicubic model in a small image region such as 9×9. The blurred image intensity at each pixel is expressed in terms of the derivatives of the focused image at the same pixel and moments of the derivatives of the shift-variant PSF at the same pixel. This differential equation is inverted locally by considering the spatial derivatives of the blurred image. The resulting solution provides a computationally efficient, parallel, and closed-form solution to the shift-variant image restoration problem. This method is extended by incorporating regularization techniques to derive robust image restoration techniques. In comparison with com-

peting techniques, this new method is fully localized and it involves non-iterative computation in the spatial domain. It enables deblurring to be carried out in the image space and to restrict computations to use local image data. Therefore, this new method permits very fine-grain parallel implementation and is suitable for real-time applications.

In this thesis, the RT theory underlying the image restoration method is reviewed, tested, and evaluated in the case of shift-variant blur. The performance of RT for various shift-variant systems including one dimensional blur, two dimensional blur, and multiple blur are investigated. Analysis of noise sensitivity of spatial differentiation filters is presented. A unified expression is derived for multiple blur that includes motion blur and shift-variant defocus. Results of both simulation and real experiments are presented for digital still cameras and cell-phone cameras.

Performance of the new localized approach is compared with matrix approaches such as Singular Value Decomposition (SVD) and Conjugate Gradient method. The results indicate that the new approach offers significant computational speed up and improved accuracy in comparison with general Singular Value Decomposition (SVD) and Conjugate Gradient method. In particular, when combined with regularization, the new approach provides computationally efficient and more accurate results.

Contents

List of Figures **xii**

List of Tables **xxi**

1 Introduction **1**

1.1 Literature Review 3

1.1.1 Autofocusing Techniques 3

1.1.2 Inverse Filtering of Shift-Variant Blur Kernels 8

1.2 Dissertation Organization 12

2 Robust Depth-from-Defocus **15**

2.1 Introduction 16

2.2 Camera Model 18

2.2.1 Image formation 19

2.3 S-Transform 22

2.4 Robust Depth from Defocus (RDFD) 24

2.4.1 RDFD based on Green's Identity 28

2.4.2 Experiments 30

2.5 Two Variations of DFD: FET and BET 34

2.6 Relation between Depth from Focus (DFF) and the Blur Parameter σ of Depth from Defocus (DFD) 35

2.7 Summary 38

3 Restoration of One-Dimensional Shift-Variant Blur 41

3.1 Problem of Modeling of Shift Variant System 42

3.1.1 Matrix Form of RT for One-Dimension 46

3.1.2 Example 48

3.2 Advantage of RT 50

3.3 Noise Sensitivity Analysis of Inverse RT Filter 51

3.4 Experiment 57

3.4.1 Restoration of Sinusoidal Pattern 58

3.4.2 Experiment on Barcode Image with Gaussian SV-PSF 61

3.5 Summary 64

4 Shift-Variant Image Deblurring: Two-dimensional Case 65

4.1 Brief Review of Prior Art 66

4.2 Theory of RT for Two-dimension 68

4.2.1 Example 74

4.2.2 Computational Complexity 75

4.3 Experiments 76

4.3.1 Simulation Results 76

4.3.2 Results on Real Images 80

4.3.3 Experiment on a Cylindrical Object 84

4.3.4 Experiment on Spatial Differential windows derived by Polynomial Fitting 89

4.4 Summary 94

5 Motion and Defocus Blur **95**

5.0.1 Shift-Invariant Motion Blur 95

5.0.2 Experiments on Shift-invariant Motion Deblur 98

5.0.3 Multiple Blur Model 103

5.0.4 Experiments on Multiple Blur Model 106

6 Comparison of Matrix Approaches and RT for Image Deblurring 110

6.1 Introduction 110

6.2 Matrix approach to shift-variant imaging system 111

6.2.1 Matrix formation 111

6.3 Kernel Matrix Inversion 113

6.3.1 Singular Value Decomposition 113

6.3.2 Conjugate Gradient Method 116

6.3.3 Complexity of SVD 117

6.4 Image Restoration based on RT and SVD 118

6.5 Image Restoration based on Conjugate Gradient Method 124

6.6 Regularization techniques 125

6.6.1 Regularization of SVD 126

6.6.2 Regularization of RT 127

6.6.3 Experiments on Regularization of SVD 129

6.6.4 Experiments on Regularization of RT 138

6.7 Summary of experimental results 142

7 Conclusion **143**

A Derivation of Differentiation Filters by Polynomial Fitting **i**

List of Figures

1-1 Autofocusing using Phase Detection 5

2-1 Camera Model with Single Convex Lens 19

2-2 Camera Model with double Convex Lens 20

2-3 Test objects. Left: “Checkerboard”. Right: “BigLetter”. 31

2-4 Results of DFD and RDFD with “Checkerboard”. (a) Top left: DFD results with image shift. (b) Top right: RDFD results with image shift. (c) Bottom left: DFD result with scaling. (d) Bottom right: RDFD results with scaling. 33

2-5 Results of DFD and RDFD with “BigLetter”. (a) Top left: DFD results with image shift. (b) Top right: RDFD results with image shift. (c) Bottom left: DFD result with scaling. (d) Bottom right: RDFD results with scaling. 34

2-6 Polynomial model of Focus Measure 37

2-7 Polynomial Model of focus Measure. (a) Top: Checkerboard image, (b) Middle: Iogray image, (c) Bottom: Pumpkin image 39

2-8 Focus Measure with Real Image. (a) Top: BigLetter image, (b) Bottom: FM of image in (a) 40

3-1 Shift-variant Blurred, Restored. Cross section plots are shown below. Input function = $sin(x)$, $N = 8$, $M = 2$, $x_{max} = 10$, $x_{min} = -10$, Sampling Frequency=200, Gaussian PSF: $\sigma(x) = 0.5 + 0.1x$. 58

3-2 Cross section plots of Fig.3-1: The left-side plot shows blurred sine wave, and the right-side plot shows restoration result. 58

3-3 Simulation of 1-D blur with two sinusoidal waves: Top: $N = 2$ and $M = 2$, Middle: $N = 4$ and $M = 2$, Bottom: Input signal frequency vs. Taylor series order 60

3-4 Simulation of 1-D blur with barcode image. Left: Input focused Image. Right: plot of parameter sigma of the shift-variant Gaussian PSF. . . 61

3-5 Shift-variant blurred and restored images. The parameter σ of Gaussian PSF is varied linearly along horizontal axis. σ has a maximum value at the image center. 62

3-6 Plots of Mean Square Error: Gaussian Blur 63

4-1 Restoration by RT with shift-variant PSF. Focused position is at 500 step number. First column is blurred image and second column is restored image. First row: Blurred by $\sigma_{max} = 1.0$ pixel. Second row: Blurred by $\sigma_{max} = 2.0$ pixel. Third row: Blurred by $\sigma_{max} = 3.0$ pixel. 77

4-2 RMS Error of RT Restoration. $0.6 \leq \sigma \leq 2.0$ 78

4-3 Sigma Plot with $\sigma_{max} = 3.0$ pixel 78

4-4 Restoration of blurred image with shift-variant Gaussian PSF. (a) Top: Input focused image (source: NASA website), (b) Middle : Computed shift-variant Gaussian blurred image, (c) Bottom: Result of shift-variant image restoration by RT. 79

4-5 Restoration using RT on E-1 camera image. (a) Top: Real image of a horizontally slanted planar object with shift-variant blur, (b) Bottom: Result of shift-variant image restoration by RT. 81

4-6 Restoration by RT with inclined Cell-phone Image. (a) Left top: Input Cell-phone Image, (b) Right top: Restored Image, (c) Bottom: Estimated σ plot. 82

4-7 Restoration by RT with inclined Cell-phone Image. (a) Left top: Input Cell-phone Image, (b) Right top: Restored Image, (c) Bottom: Estimated σ plot. 83

4-8 Pinhole camera model and cylindrical Object 84

4-9 Cylinder image for pinhole camera model. (a) Left: Planar image pattern, (b) Right: Cylinder image 85

4-10 Depth of Fig. 4-9 (b) 86

4-11 Image restoration result of Cylinder Image. (a) Left top: Blurred image with S = 1.98, (b) Right top: Restored image of (a), (c) Left middle: Blurred image with S = 2.00 (d) Right middle: Restored image of (c), (e) Left bottom: Blurred image with S = 2.02 (f) Right bottom: Restored image of (e) 87

4-12 Estimated blur circle radius of Fig. 4-11 (a) Top: Radius of blur circle with S = 1.98, (b) Middle: Radius of blur circle with S = 2.00, (c) Bottom: Radius of blur circle with S = 2.02 88

4-13 Sample Images used for Simulation of spatial window. (a) Left: Iogray image, (b) Right: Moon image 90

4-14 Result of RT with Moon image. (a) Left top: RT using 3×3 window RT with $\sigma_{max} = 2.0$, (b) Right top: RT using 5×5 window $\sigma_{max} = 2.0$, (c) Left bottom: RT using 3×3 window RT with $\sigma_{max} = 3.0$, (d) Right bottom: RT using 5×5 window $\sigma_{max} = 3.0$ 91

4-15 Result of RT with Iogray image. (a) Left top: RT using 3×3 window RT with $\sigma_{max} = 2.0$, (b) Right top: RT using 7×7 window $\sigma_{max} = 2.0$, (c) Left bottom: RT using 3×3 window RT with $\sigma_{max} = 3.0$, (d) Right bottom: RT using 7×7 window $\sigma_{max} = 3.0$ 92

4-16 The restoration result of inclined planar image. (a) Top: the restored Image using 3×3 spatial filter, (b) Middle: the restored image using 5×5 window (c) Bottom: the restored image using 7×7 window. . . 93

5-1 Model of Motion PSF 96

5-2 Clear Images used for motion deblur. Left: Checkerboard, Middle: Moon, Right: Iogray 99

5-3 Image Restoration (Motion Length=2 and Motion Angle=0°). Left: Blurred Image, Right: Restored Image 99

5-4 Image Restoration (Motion Length=6 and Motion Angle=0°). Left: Blurred Image, Right: Restored Image 100

5-5 Image Restoration (Motion Length=6 and Motion Angle=0°). Left: Blurred Image, Right: Restored Image 100

5-6 Image Restoration (Motion Length=6 and Motion Angle=0°). Left: Blurred Image, Right: Restored Image 101

5-7 Image Restoration (Motion Length=6 and Motion Angle=45°). Left: Blurred Image, Right: Restored Image 101

5-8 Image Restoration (Motion Length=6 and Motion Angle=45°). Left: Blurred Image, Right: Restored Image 102

5-9 Image Restoration (Motion Length=6 and Motion Angle=45°). Left: Blurred Image, Right: Restored Image 102

5-10 Image restoration result of multiple blur. (a) Left top: Blurred Checker image with $\sigma_{max} = 2$, motion length = 5, and angle = 0° (b) Right top: Restored image of (a), (c) Left middle: Blurred Iogray image with $\sigma_{max} = 2$, motion length = 3, and angle = 45° (d) Right middle: Restored image of (c), (e) Left bottom: Blurred Moon image with $\sigma_{max} = 2$, motion length = 7, and angle = 90° (f) Right bottom: Restored image of (e) 108

5-11 Image restoration result of multiple blur. (a) Left top: Blurred Checker image with $\sigma_{max} = 3$, motion length = 7, and angle = 45° (b) Right top: Restored image of (a), (c) Left middle: Blurred Iogray image with $\sigma_{max} = 3$, motion length = 5, and angle = 90° (d) Right middle: Restored image of (c), (e) Left bottom: Blurred Moon image with $\sigma_{max} = 3$, motion length = 3, and angle = 0° (f) Right bottom: Restored image of (e) 109

6-1 Sample image for simulation. (a) Left: "Checkerboard", (b) Right: "Moon". 119

6-2 Plot of $\sigma_{max} = 1$. (a) Left: σ for RT, (b) Right: σ for SVD. 120

6-3 Simulation results of "Checkerboard" with $\sigma_{max} = 1$. (a) Left top: Blurred image , (b) Right top: Restored image using plain SVD without truncation or regularization, (c) Bottom: Restored image using RT. 120

6-4 Simulation results of "Checkerboard" with $\sigma_{max} = 3$. (a) Left: Blurred image , (b) Middle: Restored image using plain SVD without truncation or regularization, (c) Right: Restored image using RT. 121

6-5 Simulation results of "Moon" with $\sigma_{max} = 2$. (a) Left: Blurred image , (b) Middle: Restored image using plain SVD without truncation or regularization, (c) Right: Restored image using RT. 121

6-6 Simulation results of "Moon" with $\sigma_{max} = 3$. (a) Left: Blurred image , (b) Middle: Restored image using plain SVD without truncation or regularization, (c) Right: Restored image using RT. 122

6-7 RMS error comparison of SVD and RT 122

6-8 Plot of average Singular Value of PSF. 123

6-9 RMS error comparison of Conjugate Gradient and RT. Left: Checkerboard image. Right: Moon image. 124

6-10 Restoration result of Conjugate Gradient with $\sigma_{max} = 3$ Top: 15 iterations. Bottom: 35 iterations. 125

6-11 Sample image for simulation. (a) Left: "Checkerboard", (b) Right: "Moon". 129

6-12 Plot of blur parameter σ with $\sigma_{max} = 1$. (a) Left: Original σ, (b) Right: Approximated σ for SVD. 129

6-13 Restored image by Truncated SVD with L-curve method. (a) Left: $\sigma_{max} = 2.0$, (b) Right: $\sigma_{max} = 3.0$. 130

6-14 Blurred images. (a) Top: Blurred images by blur parameter $\sigma_{max} = 2.0$, (b) Bottom: Blurred images by blur parameter $\sigma_{max} = 3.0$. . . . 131

6-15 Regularization result of Tikhonov SVD with $\sigma_{max} = 2.0$. (a) Left: Restored image by Tikhonov SVD with parameter 0.001, (b) Right: Restored image by Tikhonov SVD with parameter 0.032 133

6-16 Regularization result of Tikhonov SVD with $\sigma_{max} = 3.0$. (a) Left: Restored image by Tikhonov SVD with parameter 0.001, (b) Right: Restored image by Tikhonov SVD with parameter 0.032 133

6-17 Regularization result of Tikhonov SVD with $\sigma_{max} = 2.0$. (a) Left: Restored image by Tikhonov SVD with parameter 0.001, (b) Right: Restored image by Tikhonov SVD with parameter 0.031 134

6-18 Regularization result of Tikhonov SVD with $\sigma_{max} = 3.0$. (a) Left: Restored image by Tikhonov SVD with parameter 0.001, (b) Right: Restored image by Tikhonov SVD with parameter 0.031 134

6-19 Regularization result of TSVD with $\sigma_{max} = 2.0$. (a) Left: Restored image by TSVD with parameter 0.001, (b) Right: Restored image by TSVD with parameter 0.027 135

6-20 Regularization result of TSVD with $\sigma_{max} = 3.0$. (a) Left: Restored image by TSVD with parameter 0.001, (b) Right: Restored image by TSVD with parameter 0.023 135

6-21 Regularization result of TSVD with $\sigma_{max} = 2.0$. (a) Left: Restored image by TSVD with parameter 0.001, (b) Right: Restored image by TSVD with parameter 0.039 136

6-22 Regularization result of TSVD with $\sigma_{max} = 3.0$. (a) Left: Restored image by TSVD with parameter 0.001, (b) Right: Restored image by TSVD with parameter 0.038 136

6-23 RMS error of Tikhonov SVD with various regularization parameter. (a) Left: Checkerboard image, (b) Right: Moon image. 137

6-24 RMS error of Truncated SVD with various regularization parameter. (a) Left: Checkerboard image, (b) Right: Moon image. 137

6-25 Regularization result of RT with Checkerboard image (a) Left top: General RT (no regularization) with $\sigma_{max} = 2.0$, (b) Right top: Regularized RT with $\sigma_{max} = 2.0$ and regularization parameter 0.05, (c) Left bottom: General RT with $\sigma_{max} = 3.0$, (d) Right bottom: Regularized RT with $\sigma_{max} = 3.0$ and regularization parameter 0.05 138

6-26 Regularization result of RT with Moon image (a) Left top: General RT with $\sigma_{max} = 2.0$, (b) Right top: Regularized RT with $\sigma_{max} = 2.0$ and regularization parameter 0.05, (c) Left bottom: General RT with $\sigma_{max} = 3.0$, (d) Right bottom: Regularized RT with $\sigma_{max} = 3.0$ and regularization parameter 0.05 139

6-27 Test image and estimated σ. (a) Left: The image of a slanted plane with shift-variant blur captured by Olympus E-1 camera and used in the experiments. (b) Right: A plot of the estimated Gaussian blur parameter σ along the horizontal axis. 140

6-28 Restored image with/without regularization based on RT. (a) Top: restored Image without regularization, (b) Middle: restored and regularized image with $\alpha = 0.15$ (c) Bottom: restored and regularized image with $\alpha = 0.35$. 141

List of Tables

Chapter 1

Introduction

Image processing plays a critical role in industrial automation, surveillance systems, medical imaging, etc. In all imaging systems, image deblurring or image restoration in the presence of noise is an important problem. Image blurring can be shift-invariant (space-invariant) or shift-variant (space-variant). In the shift-invariant case, the blurring point spread function (PSF) of the camera system is invariant or unchanging with spatial location. In the shift-variant case, the PSF changes with position on the image. For the shift-invariant image blur, Fourier Transform provides a very useful mathematical tool that transforms convolution in time/space domain to multiplication in the frequency domain. Therefore the image blurring process can be simplified in the frequency domain and the inversion of image blur can be done through deconvolution. However, shift-variant image blur is more common and in this case Fourier analysis does not provide a satisfactory solution for image deblurring.

Several approaches have been proposed in the literature for shift-variant image restoration. One approach is to divide a large image into many smaller image blocks,

and to approximate the shift-variant blur to be shift-invariant in each image block and restore it. Another approach is to formulate a large matrix equation and solve it directly. Some approaches restrict the PSFs to be a Gaussian or a rotationally symmetric function, or a separable function. These approaches are often computationally intensive, error-prone, and restrictive. In this thesis, a significant part of the research is related to the investigation of a new localized approach to shift-variant image restoration. It is based on the recently proposed Rao Transform (RT) which restates the traditional shift-variant image blurring model without loss of generality in a completely local and easily invertible form but exactly equivalent to the traditional model. The new approach investigated here has significant theoretical and computational advantages in comparison with conventional approaches including those based on Singular Value Decomposition. It provides mathematical simplicity, ease of implementation, computational locality, efficiency, and a solution for arbitrary PSFs. It uses a weak assumption that the focused image in a small image region can be modeled as a polynomial at each point in the blurred image. The performance of the new approach is evaluated and compared with competing techniques in the current literature.

Another significant part of this thesis is devoted to autofocusing of digital cameras to capture focused images. We propose a new passive autofocusing technique (RDFD) which is robust with respect to hand-shake or small camera motion during image capture. This new autofocusing technique does not require pixel correspondence between sample images but only region correspondence. Therefore it is less sensitive to camera motion. We have performed many simulation and real experiments on

RDFD. These experiments indicate that RDFD is fast, sufficiently accurate, and can be applied in practice.

1.1 Literature Review

Autofocusing technique has proceeded from attaching additional depth detector to estimating depth from image data. While depth detection from image enables camera to be more compact, it still requires improvement in noise robustness and accuracy. In image restoration, shift-invariant deblurring techniques have been extended and new approaches have been proposed for shift-variant deblurring. In this section we summarize some existing techniques for autofocusing and image restoration of shift-variant blur. We review the advantages and disadvantages of different techniques and present motivation for our research.

1.1.1 Autofocusing Techniques

There are two main approaches to autofocusing: active and passive autofocusing. Active autofocusing is mainly based on distance detection (or ranging) technique by projecting some energy instead of using ambient light. It projects a carrier signal such as ultrasonic, infrared or even structured light onto the object surface and the sensor detects the reflected signal. From the relation between the reflected signal and the projected signal the object distance is obtained. Then, the system adjusts its lens position. Laser ranging is a widely used technique for active autofocusing. The disadvantage of active autofocusing system is that additional hardware such as beam

projector or sensor is required and the object material influences ranging result. The advantage of an active autofocusing system is that it works in dark illumination.

Passive Autofocusing determines the distance/depth of object by analyzing image data that enters the optical system. It generally does not emit any energy that active autofocus methods use. Examples of this method are Phase detection, Depth-from-focus (DFF) [4, 6], and Depth-from-Defocus(DFD) [7, 9, 12, 14, 17–20]. Phase Detection (PD) is widely used in high-end Single-Lens Reflex (SLR) cameras. This technology was first patented by Honeywell in the 1970s [75, 76]. Phase detection (PD) is achieved by dividing the incoming light into pairs of images and comparing. The beam splitter directs light to two image sensors. The distance between two split images are compared to the distance of two split focused images. When focused distance is shorter than object distance, narrow spaced split image pair is observed. On the contrary, wide spacing happens when the object is inside the focused distance. PD module can detect distance in a certain range due to limit of physical module size. Multiple PD modules make camera structure complex and cause cost increase.

Depth-from-Focus (DFF) determines object distance by image focus analysis. A sequence of images of a scene is obtained by continuously varying the distance between the lens and the image detector [4, 6]. For different focus setting, a point in the scene will be blurred to different degrees in each image frame. Since defocusing is lowpass filtering, the sharpness of image can be a good criteria to decide the degree of blur. The image having maximum sharpness is selected as to be focused the most, and measuring the sharpness of image is called Focus Measure (FM). Differentiation such as Laplacian operation and Sobel operator are widely used for FM. In 1970,

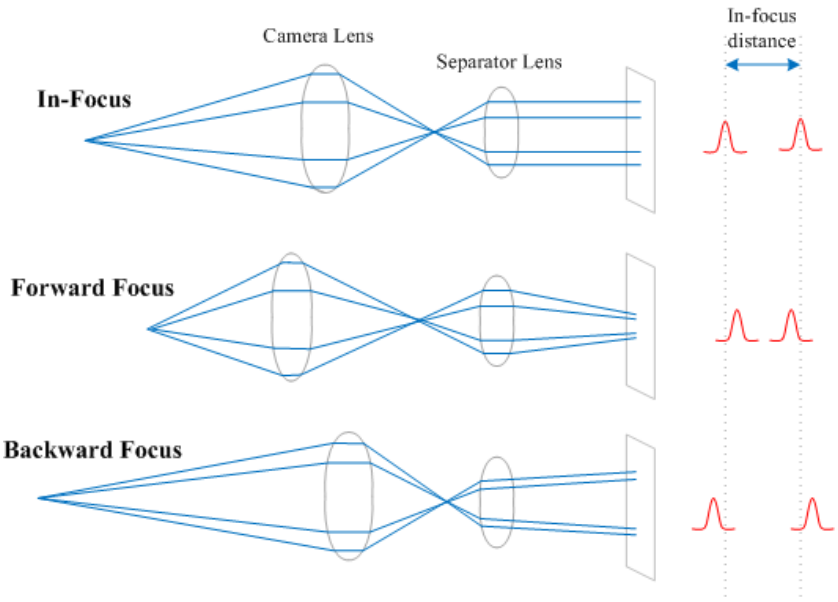


Figure 1-1: Autofocusing using Phase Detection

Tenenbaum developed a focus measure method based on the gradient magnitude from the Sobel operator [1]. Its FM is the summation of squared gradient magnitude. Unlike Tenenbaum who used two-dimensional convolution mask, Boddeke proposed one-dimensional FM algorithm in 1994 [2]. Another method for image sharpness measurement is the use of second-derivatives. Laplacian operator is one of the simplest second-derivative operator for DFF algorithm [6]. By calculating the absolute value or the squared second-derivative, the highest FM is obtained for the sharpest image. To increase robustness to noise, variant of a gradient magnitude and Laplacian gradient magnitude was proposed by Pech-Pacheco and Cristobal [3].

The disadvantage of DFF is to require many images for comparison and optimal distance determination. Lens movement consumes electrical energy, and capturing and processing a large number of images requires computational time. Due to changes in magnification, it is difficult for DFF to improve the accuracy of object distance.

DFF does not need camera calibration and pixel correspondence, and it is relatively robust against noise and small object motion.

DFD is based on defocus analysis. An image is considered as a collection of point sources of light. Light from every point source passes through a convex lens and is projected on the image sensor plane. When the lens is focused on light source, the projected light from one point light source converge to one point. If not focused, the light is distributed over a circular area on the image plane. This distribution is the Point-Spread-Function (PSF), and the radius of light spread is an important parameter to estimate the PSF. PSF is dependent on both object distance and camera parameters. Therefore, if the camera parameters and the PSF are known, one can deduce object distance. DFD is the process that estimates PSF parameters and decides object distance. To estimate object distance, DFD requires only two or three images captured with different camera parameter setting. Pentland proposed two methods of DFD in 1980's [12]. The first method used some well known image pattern like edges. By analyzing the spread of the known pattern, the degree of blur was measured. This method had a limiting factor since it was based on the assumption that the image pattern is known. The second method was to capture two different images with different lens aperture. One lens aperture was set small like pin-hole, and the other aperture was set normal. Because differing aperture size causes differing blur of same point, the same point will be focused differently in the two images. Analyzing the relation between the distance, lens aperture, and blur circle radius, object distance was estimated. A very small aperture was difficult to implement in practical application. Since less light passes through a small aperture, long exposure

time is required. Diffraction degrades image quality. Subbarao proposed a new approach with arbitrary camera setting [15]. Using power spectral density of Gaussian PSF, blur parameter was estimated from two images with different camera setting. Pentland claimed that spatial filtering can be an alternative method when spectral power analysis was carried out in limited frequency range [13]. By Parseval's theorem, spectral energy in limited frequency range can be translated into the energy of image that is filtered in spatial domain. This alternative filtering leads to the simple implementation and fast calculation. Y. Xiong and S. A. Shafer proposed an iterative method, named Maximal Resemblance Estimation, for reducing noise influence of windowing in DFD [35]. At every iteration step, the convolution of image, Gaussian spatial filtering, and Fourier Transform were performed. The variance of gaussian filter was updated until the estimated value converges to true value. Subbarao and Surya [8] introduced spatial domain approach based on S-transform. This method approximated original image by a local cubic polynomial, and depth was estimated using a novel deconvolution formula where the original image is expressed in terms of image derivatives and the moments of PSF. Since the blur parameter is obtained from the deconvolution formula, the depth estimation process requires only two image derivatives and a moment of PSF. This method leads to simple implementation and localized computation. Watanabe and Nayar proposed a small set of filters and operators for passive DFD [33]. These operators, when used in conjunction, yield invariance to texture frequency while computing depth. Using cubic polynomial, they approxiamte the ratio of subtraction and summation of two spectral images. Its ratio has monotonic characteristics for any radial frequency, and the coefficient of the cu-

bic polynomial is a function of frequency. Inverse Fourier transform of the coefficient function is the proposed broadband rational operator.

Depth From Defocus (DFD) needs acquiring only two or three images, and recovers the depth information in a scene by computing the degree of blur. This makes it suitable for real-time autofocusing and distance measurement. However, There are some error factors in DFD. Most DFD methods begin with the assumption that pixel correspondence is given while different camera parameters produce different magnifications. For accurate depth estimation, DFD has to compensate for the discrepancy of image magnification. The second error factor is the non-uniform illumination. If the illumination is not uniform over the image region, the relation between blurs deviates from our expectation, and depth estimation fails. Also, motion of either object/camera influences blur.

1.1.2 Inverse Filtering of Shift-Variant Blur Kernels

Image formation for shift-invariant PSF is typically described as a linear process. Such systems are formulated by the convolution of image function and PSF:

$$g(x) = f(x) * h(x) = \int f(u)h(x-u)du \tag{1.1}$$

where $g(x)$, $f(x)$, $h(x)$, and $*$ denote the observed image function, the original image function, PSF, and the convolution operator.

The classical definition of shift-variant system is expressed by the Fredholm inte-

gral equation of the first kind:

$$g(x,y) = \int_a^b f(u)h(x,u)du \tag{1.2}$$

The shift-invariant system Eq.(1.1) can be easily inverted in the Fourier domain. The Fourier transform simplifies the above convolution into a simple scalar multiplication. However, for shift-variant system Eq.(1.2), it is necessary to look for other representations that play a similar role to that of Fourier transform in the case of shift-invariant filtering. Finding new mathematical model for shift-variant system is an important challenge. Instead of finding new mathematical model, if we can consider a shift-variant system as a shift-invariant system, we can use deconvolution technique based on Fourier transform for obtaining input signal. Examples of this include sectioning method, image warping technique, and coordinate transform and resampling. Matrix approach is another useful method to invert system equation. Formulating a matrix equation is very straightforward, and solution of the matrix equation can be obtained by inverting the kernel matrix. But, inverting a kernel matrix is a very computationally intensive task for adoption in a practical application.

One of the first approaches to the restoration of spatially variant blurs was the development of the inverse filters by Sawchuk [54, 57] and Robbins and Huang [58]. Robbins and Huang proposed inverse filtering method for coma-like system. They assumed that the object point u is near the optical axis so that the irradiance at the exit pupil is a constant independent of position ϵ. Sawchuk proposed the coordinate transformation of object, image and shift-variant PSF to shift-invariant system. After

the transformation, shift-invariant processing is used for restoration. Although these two methods yield good results, it is difficult to apply these methods to general shift-variant blur systems. Some special type of Shift-variant PSFs, like coma and motion blur, can be transformed into shift-invariant PSF. McNown and Hunt [51] warped the signal to make the system stationary by moving sample location like Robbins and Sawchuk. The warped signal appear stretched in some places and compressed in others while the system becomes shift-invariant. The warping technique has similar restrictions which the above methods have.

Some approaches attempt to simplify the inverse filtering of shift-variant kernel by dividing image into small subsections. These approaches treat each subsection with space-invariant assumption. Trussel and Hunt [45] proposed to extend sectional method in image processing to the shift-variant degradation process. The size of a section has to be carefully selected because small sections are less efficient than larger sections but give better quality restorations due to their adaptability to local variations in the image. Costello and Mikhael [53] have attempted to increase computational efficiency by adjusting section size. Using the characteristics that image distortion in outer area is larger than that in the central area, they assigned larger section to central area.

The convolution process can be described by matrix multiplication of the vector-form of original image and kernel matrix. The virtue of matrix form is that deconvolution is carried out by inverting the kernel matrix. For shift variant system, one can inverse filter if kernel matrix represents a shift-invariant kernel. For the inversion of matrix, Singular-Value-Decomposition (SVD) is widely used due to the simplic-

ity of calculation [21–23]. However, matrix inversion for shift-variant kernel has two major problems: the enormous size of matrix, and ill-posedness of the problem. The processing time of matrix inversion increases rapidly as the cube of the size of kernel matrix. One solution is the sectioning method mentioned above. Assuming locally shift-invariant PSF in a subsection, Hansen, Nagy and O'Leary [21] claimed that the inversion of matrix can be performed by Fast-Fourier-Transform (FFT) if the matrix is a Toeplitz matrix. Reshaping matrix kernel with Kronecker product was proposed since the inversion of Kronecker product of two matrices yields Kronecker product of two inverted matrices [22, 23].

Noise in imaging systems result in an ill-posed problem and therefore the image restoration may fail. Regularization techniques and iterative methods are used for solving ill-posed problems. Richardson-Lucy algorithm [77, 78] is an iterative maximum likelihood method without the assumption of specific statistical noise model. Angel and Jain have used a conjugate gradient method for least square restoration. Nagy and O'Leary [24] used the preconditioned conjugate gradient method. The main disadvantage of iterative methods is processing time.

Recently the spatial-domain convolution/deconvolution transform (S-Transform) was extended to solve shift-variant image deblurring based on a new transform named Rao Transform (RT) [27, 28]. RT formulates the shift-variant blurred image in terms of the derivatives of an analytic approximation to the focused image. It is capable of localized computation and it reduces computation significantly. In this sense, RT is relevant to practical application such as image restoration.

1.2 Dissertation Organization

There are two goals in this dissertation; one is to develop an autofocusing technique that is robust to camera vibration, and the other is image restoration with shift-variant blur.

In Chapter 2, a new autofocusing technique is introduced to estimate depth from two given images which are taken by moving the camera with different camera parameters. This novel technique is based on calculating the local energy of images like DFF. The relation between two blur parameters is obtained by letting energy of restored images to be identical. Focused images are obtained using the inverse S-Transform. Since the energy in a small region has less variation even though the region is shifted by a few pixels, RDFD produces consistent results. Thus, RDFD is robust against object motion and camera shaking. We also propose a novel polynomial model to approximate the Focus Measure as a function of blur parameter. The proposed polynomial model is a good approximation for small blur.

In Chapter 3 and 4, the new localized (RT) approach for one-dimensional and two-dimensional shift-variant deblurring is investigated. Brief theory and advantage of RT are discussed, and we derive expressions for noise sensitivity of image restoration based on RT. The computational complexity of RT approach for two-dimensional image is presented. The new method provides an analytic solution suitable for both theoretical analysis and numerical implementation. Expressions for local noise variance (or noise power) at a pixel in the restored image is obtained in terms of local noise variance in the input image. In the analysis of computational complexity, RT

approach has significant speed-up in comparison with conventional matrix approach (e.g. SVD). Experimental results with simulated images and real images are presented. Simulation experiments includes linearly varying and non-linearly varying SV-PSF. For simulation, sinusoidal image, barcode image, cylindrical image by pinhole camera model, and planar object images are blurred, restored and verified. Real blurred images captured by normal digital camera and cellular phone are restored. The results were found to be very satisfactory and the images were well restored.

In chapter 5, we extend RT approach to multiple blurred image. The new expression of motion blurred system is derived, and then this expression is combined with shift-variant defocus. We simulate and restore such multiple blurred images with motion angle 0°, 45°, and 90° with horizontal axis. We consider motion length up to 7 pixels. The σ_{max} of shift-variant PSF varies 0 to 3. Experimental results indicate improvement by 13-17 percent in RMS error.

In Chapter 6, some standard matrix approaches to shift-variant deblurring are summarized and their performance is compared with RT based method. The considered matrix approaches include Singular value decomposition (SVD) and conjugate gradient method. In section 6.2, kernel matrix formations with zero boundary condition, periodic boundary condition, and reflexive boundary condition are addressed. By sectioning the image, matrix approach reduces the size of kernel matrix for computational simplicity. The theories of image restoration based on SVD and conjugate gradient method are discussed, and their computational complexity is considered and compared with RT's. Experimental results indicate that the new approach offers significant computational speed up and improved accuracy in comparison with the

SVD and Conjugate Gradient method. Image restoration incorporating regularization techniques is discussed in Section 6.6.

Chapter 2

Robust Depth-from-Defocus

In this chapter, a new passive ranging technique named Robust Depth-from-Defocus (RDFD) is presented for autofocusing in digital cameras. It is adapted to work in the presence of image shift and scale change caused by camera/hand/object motion. RDFD is similar to spatial-domain Depth-from-Defocus (DFD) techniques in terms of computational efficiency, but it does not require pixel correspondence between two images captured with different defocus levels. It requires approximate correspondence between image regions in different image frames as in the case of Depth-from-Focus (DFF) techniques. Theory and computational algorithm are presented for two different variations of RDFD. Experimental results are presented to show that RDFD is robust against image shifts and useful in practical applications. RDFD also provides insight into the close relation between DFF and DFD techniques.

2.1 Introduction

Passive ranging and autofocusing techniques based on image focus and defocus analysis have been investigated by many researchers [1-11]. Depth-from-Focus (DFF) [4,6] is a technique based on image focus analysis. It requires capturing and computing focus measures for many (about 10 in a typical application) images. It involves searching for a camera parameter setting that maximizes the focus measure in a small image block. It is slow and provides a low resolution depth (one estimate in each image block), but it does not need camera calibration or pixel correspondence. It needs only image region correspondence. DFF is relatively robust and accurate. Depth-from-Defocus (DFD) [4-11] is a technique based on image defocus analysis. It uses only two or three images captured with different camera parameter settings. It compares the change in blur to estimate depth. It is fast and efficient but requires camera calibration. Spatial domain DFD techniques provide a denser depth-map (upto one depth estimate at each pixel) than DFF techniques, but they need pixel correspondence between different image frames. DFD is also less accurate and robust than DFF unless more images (about 5) are used. Fourier domain DFD [19]techniques provide lower resolution depth-maps and require more computation than spatial domain DFD techniques [9, 20].

A new passive ranging technique named Robust Depth-from-Defocus (RDFD) is presented for autofocusing in digital still/video cameras. RDFD is similar to spatial-domain DFD techniques [9,20] in terms of computational efficiency and the use of only two images, but it does not require pixel correspondence between two images cap-

tured with different defocus levels. It requires approximate correspondence between small image regions in different image frames as in the case of DFF techniques [4,6]. This is an important advantage because, in autofocusing applications, small image shifts between different captured image frames are caused by many unavoidable factors such as hand shake, camera vibration, magnification change (due to lens motion), and object motion. Both RDFD and spatial-domain DFD use a local deconvolution formula involving image derivatives for estimating an image defocus parameter. However, unlike DFD techniques, RDFD relies on computations at corresponding small image regions instead of corresponding pixels. RDFD is suitable for real-time autofocusing in digital still and video cameras where the number of image frames captured and computational resources are limited, and accurate pixel correspondence is not available.

It should be noted that Fourier domain DFD techniques [19] do not need pixel correspondence but only region correspondence (as with DFF) and therefore less sensitive to image shift caused by camera vibration, etc. However, Fourier domain techniques have other problems. First, they require more computation as Fourier coefficients will have to be computed. Next, unlike local spatial domain DFD techniques such as STM [9,20], Fourier domain techniques are global and less accurate as they ignore the image overlap problem. Fourier domain techniques use windowing to extract image blocks and compute Fourier coefficients of the image blocks as an estimate of local Fourier coefficients. As blurring just outside the border of the image blocks changes the images inside the image blocks, errors will be introduced in these techniques. RDFD uses local spatial domain computations similar to DFD-STM [9, 20] at each

pixel, and is therefore affected less by the image overlap problem.

We present the theory and a computational algorithm for two different variations of RDFD. One of the variation uses Greene's identity to avoid the use of second-order image derivatives which are highly noise sensitive. Instead, it relies more on first-order image derivatives which are a lot less sensitive to noise. Greene's identity provides a relation between the area integral involving second-order image derivatives to a boundary integral involving only first-order image derivatives. Experimental results are presented to compare the performance of RDFD with DFD in [9,20] (STM). These results show that RDFD is more resistant to small image shifts than DFD-STM, and it is useful in practical applications.

2.2 Camera Model

Many passive autofocusing techniques estimate object distance using certain metrics computed from defocused image set such as power spectral density, spectral energy in limited frequency, and relation between estimated blur parameters. These metrics depend on how lens position is related to blur and how image is formed in a camera system. In this section we review the image formation process in an optical system. This helps to understand how blur is related to object distance and lens position. We first consider a very simple model of a camera consisting of a single lens. In practice, however, camera systems consist of multiple lenses with complex structure.

2.2.1 Image formation

Consider a camera system consisting of a single convex lens as shown in Fig.2-1.

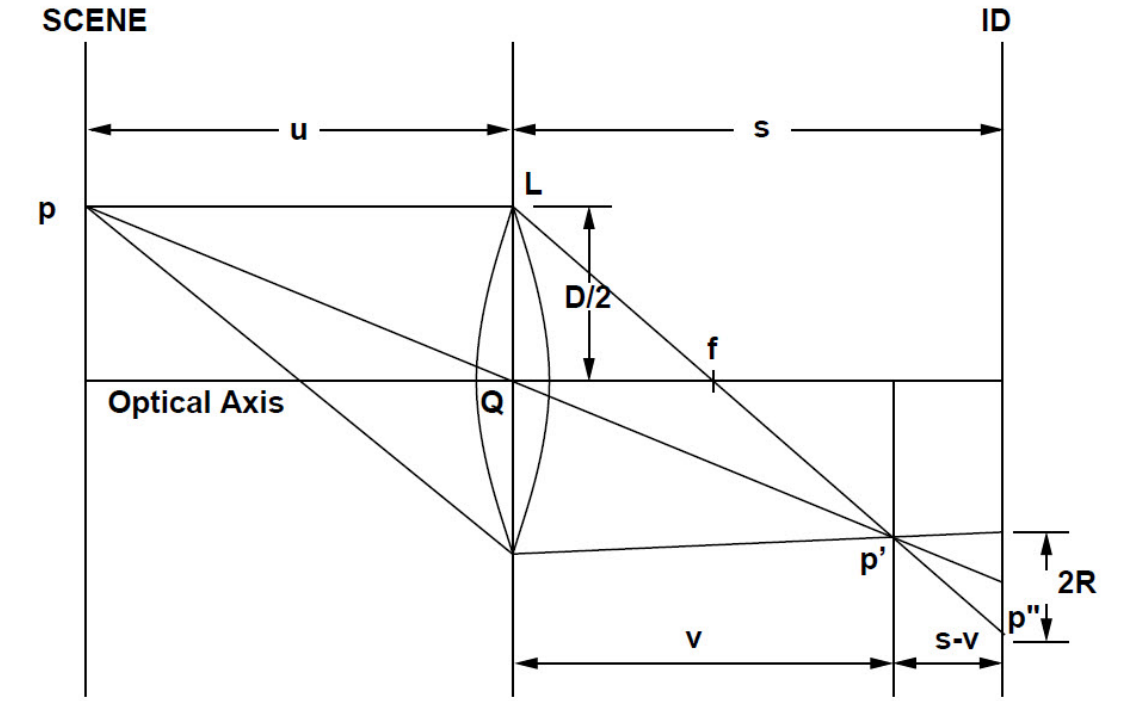


Figure 2-1: Camera Model with Single Convex Lens

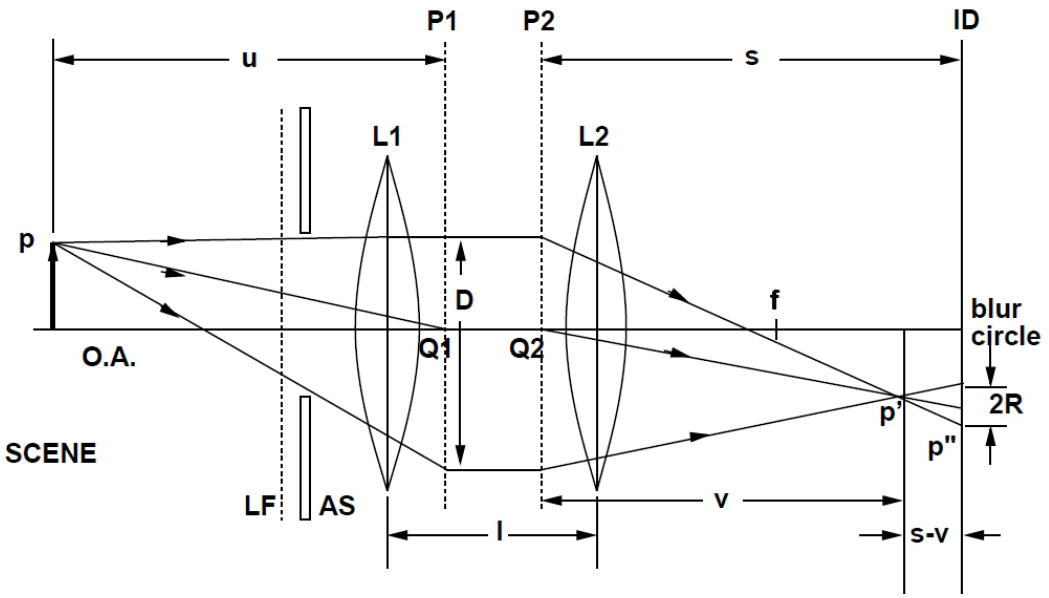


Figure 2-2: Camera Model with double Convex Lens

If there is a point light source p at a distance u from the lens, then focused image p' is formed on the other side of the lens at a distance v. The distances u and v are related by the well-known lens formula,

$$\frac{1}{u} + \frac{1}{v} = \frac{1}{f} \tag{2.1}$$

where f is the focal length of the lens.

If image sensor such as CCD array or photographic film is placed at distance v, the point light source is displayed as a point in captured image. However, if the image sensor is located at other place, s, $v \neq s$, then the point is spread into a circle with a certain radius in the captured image. In Fig.2-1, let R be the radius of the blur circle, D be the diameter of lens aperture, and s be the distance from the lens to the image sensor. Also, let q be the scaling factor defined by $q = 2R/D$. By triangle similarity, we have

$$q = \frac{2R}{D} = \frac{s-v}{v} = s\Big[\frac{1}{v} - \frac{1}{s}\Big] \tag{2.2}$$

substituting for $1/v$ from Eq.(2.1), we obtain,

$$q = s\Big[\frac{1}{f} - \frac{1}{u} - \frac{1}{s}\Big] \tag{2.3}$$

Therefore,

$$R = q\frac{D}{2} = s\frac{D}{2}\Big[\frac{1}{f} - \frac{1}{u} - \frac{1}{s}\Big] \tag{2.4}$$

R in Eq.(2.4) represents the radius of blur circle where the light energy spreads, and the blur circle corresponds to the PSF. We often model PSF with Gaussian distribution function. If we assume the PSF to be a blur circle represented by a circularly symmetric cylindrical function, then the PSF is described by

$$h(x,y) = \begin{cases} \frac{1}{\pi R^2} & \text{for } x^2 + y^2 \leq R^2 \\ 0 & \text{otherwise} \end{cases} \tag{2.5}$$

where $h(x,y)$ denotes the PSF.

In Gaussian PSF model, Gaussian PSF is defined by the relation between the Gaussian parameter σ and blur circle radius R:

$$\sigma = \frac{R}{\sqrt{2}}$$

Therefore,

$$h(x,y) = \frac{1}{\pi R^2}\, e^{-\frac{x^2+y^2}{R^2}} \tag{2.6}$$

2.3 S-Transform

A brief summary of the Spatial-domain Convolution/Deconvolution Transform or S-Transform is presented here. S-Transform [20] has been developed for images and n-dimensional signals for the case of arbitrary order of polynomials. It provides a completely local expression for the convolution and deconvolution of an image with respect to a filter or Point Spread Function (PSF) in the spatial domain. Using a local polynomial approximation of an image, spatial domain formulas are derived for convolution (corresponds to blurring or defocusing) and deconvolution (corresponds to deblurring or image restoration) with respect to a PSF.

Let $h(x,y)$ be a rotationally symmetric PSF. This assumption of rotational symmetry is a useful one, but it can be removed easily for theoretical analysis if needed. If we assume the camera to be a lossless system (i.e., no light energy is absorbed by the camera system) then

$$\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} h(x,y)dxdy = 1 \tag{2.7}$$

The moments of the PSF are defined by

$$h_{m,n} = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} x^m y^n h(x,y)dxdy \tag{2.8}$$

Since the PSF is rotationally symmetric, many odd moments are zero and it can

be shown that

$$h_{0,1} = h_{1,0} = h_{1,1} = h_{0,3} = h_{3,0} = h_{1,2} = h_{2,1} = 0, \quad h_{2,0} = h_{0,2}, \quad \text{and} \quad h_{0,0} = 1 \tag{2.9}$$

A blurred image $g(x, y)$ captured by a camera system can be modeled by the convolution of the focused image $f(x, y)$ with the corresponding PSF $h(x, y)$ as:

$$g(x, y) = f(x, y) * h(x, y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x - \zeta, y - \eta) h(\zeta, \eta) d\zeta d\eta \tag{2.10}$$

where $*$ denotes the convolution. If we approximate the focused image to be a cubic polynomial locally in a small image region corresponding to the size of the blur circle region at a pixel, then the following relation can be derived for image deblurring.

$$\begin{aligned} f(x, y) &= g(x, y) - \frac{h_{2,0}}{2}(g^{2,0}(x, y) + g^{0,2}(x, y)) + H.O.T.s \\ &= g(x, y) - \frac{h_{2,0}}{2}\nabla^2 g(x, y) + H.O.T.s \end{aligned} \tag{2.11}$$

where ∇^2 is the Laplacian operator. If we define blur parameter as

$$\sigma^2 = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} (x^2 + y^2) h(x, y) dx dy \tag{2.12}$$

then $h_{2,0} = h_{0,2} = \sigma^2/2$. Therefore Eq.(2.11) can be written as

$$f(x, y) = g(x, y) - \frac{\sigma^2}{4}\nabla^2 g(x, y) + H.O.T.s \tag{2.13}$$

We can also write an expression for the blurred image $g(x, y)$ in terms of the focused image $f(x, y)$ using Eq.(2.13) as

$$g(x, y) = f(x, y) + \frac{\sigma^2}{4}\nabla^2 f(x, y) + H.O.T.s \tag{2.14}$$

2.4 Robust Depth from Defocus (RDFD)

RDFD theory is an extension of the DFD theory based on the S Transform Method (STM) [9, 20]. Let $g_1(x, y)$ and $g_2(x, y)$ be two defocused images captured with two different camera parameter settings such as two different lens positions. Let $f(x, y)$ be the focused image and $h_1(x, y)$ and $h_2(x, y)$ be the two point spread functions (PSFs) so that

$$g_1(x, y) = h_1(x, y) * f(x, y) \tag{2.15}$$

and

$$g_2(x, y) = h_2(x, y) * f(x, y) \tag{2.16}$$

Let σ_1 and σ_2 be the blur parameters given by the square root of the second-central moment of the PSFs $h_1(x, y)$, $h_2(x, y)$, respectively (see Eq.(2.12)). The spatial domain DFD technique STM [9, 20] uses the following expression to compute a change in the blur parameters (assuming a local cubic polynomial model of the images):

$$G = \sigma_1^2 - \sigma_2^2 = \frac{4[g_1(x, y) - g_2(x, y)]}{(\nabla^2 g_1(x, y) + \nabla^2 g_2(x, y))/2} \tag{2.17}$$

where ∇^2 is the Laplacian operator.

Spatial domain DFD methods such as STM [9, 20] require pixel correspondence between two images captured with different camera settings. Pixel correspondence is needed because the expression used in solving for depth includes mixed-image terms such as $g_1(x,y) - g_2(x,y)$ in the above equation. These mixed terms need the value of two different images $g_1(x,y)$ and $g_2(x,y)$ at the same corresponding pixel position specified by (x,y). In RDFD, such terms involving both images at the same corresponding pixel are avoided. Instead focus measure terms computed over corresponding image regions are used as in the case of DFF methods. The expressions for the focus measures are derived from the local spatial-domain deconvolution (inverse S Transform) expressions [9, 20].

$$f(x,y) = g_1(x,y) - \frac{\sigma_1^2}{4}\nabla^2 g_1(x,y) + \text{Higher Order Terms(H.O.T.s)} \tag{2.18}$$

$$f(x,y) = g_2(x,y) - \frac{\sigma_2^2}{4}\nabla^2 g_2(x,y) + \text{Higher Order Terms(H.O.T.s)} \tag{2.19}$$

The right hand sides in the above two equations express the same focused image, and equating them is called the Focus Equalization Technique. Equating the two right hand sides and squaring both sides to compute focus measures, and then neglecting the higher order terms, we obtain:

$$\iint_R \left[g_1(x,y) - \frac{\sigma_1^2}{4}\nabla^2 g_1(x,y)\right]^2 dxdy = \iint_R \left[g_2(x,y) - \frac{\sigma_2^2}{4}\nabla^2 g_2(x,y)\right]^2 dxdy \tag{2.20}$$

We use squaring to avoid adding positive and negative Laplacian quantities to

get a small sum that is sensitive to noise. We integrate over an image region R to avoid needing pixel correspondence to solve for depth. We need only image region correspondence for the region R. Larger the size of image region R, smaller is the sensitivity of the solution to errors in establishing accurate region correspondence. Increasing the size of R also reduces spatial-resolution of depth-map in 3D shape recovery applications.

Next consider an equally blurred image $g_3(x, y)$ obtained from $g_1(x, y)$ and $g_2(x, y)$:

$$g_3(x, y) = g_1(x, y) * h_2(x, y) = h_1(x, y) * h_2(x, y) * f(x, y) \tag{2.21}$$

$$g_3(x, y) = g_2(x, y) * h_1(x, y) = h_1(x, y) * h_2(x, y) * f(x, y) \tag{2.22}$$

$$g_1(x, y) * h_2(x, y) = g_2(x, y) * h_1(x, y) \tag{2.23}$$

$$g_3(x, y) = g_1(x, y) + \frac{\sigma_2^2}{4}\nabla^2 g_1(x, y) + H.O.T.s \tag{2.24}$$

$$g_3(x, y) = g_2(x, y) + \frac{\sigma_1^2}{4}\nabla^2 g_2(x, y) + H.O.T.s \tag{2.25}$$

The right hand sides in Eq.(2.24) and Eq.(2.25) express the same defocused image $g_3(x, y)$, and equating them is called the Blur or Defocus Equalization Technique (BET/DET) [14]. Equating the two right hand sides, squaring both sides, integrating to compute focus measures, and then neglecting higher order terms, we obtain:

$$\iint_R \left[g_1(x, y) + \frac{\sigma_2^2}{4}\nabla^2 g_1(x, y)\right]^2 dxdy = \iint_R \left[g_2(x, y) + \frac{\sigma_1^2}{4}\nabla^2 g_2(x, y)\right]^2 dxdy \tag{2.26}$$

Subtracting Eq.(2.26) from Eq.(2.20) and simplifying, we obtain

$$G = \sigma_1^2 - \sigma_2^2 = 8 \times \frac{\iint\limits_R g_1(x,y)\nabla^2 g_1(x,y)dxdy - \iint\limits_R g_2(x,y)\nabla^2 g_2(x,y)dxdy}{\iint\limits_R (\nabla^2 g_1(x,y))^2 dxdy + \iint\limits_R (\nabla^2 g_2(x,y))^2 dxdy} \tag{2.27}$$

Note the similarity between Eq.(2.17) and Eq.(2.27) which forms the basis of RDFD for computing the quantity $\sigma_1^2 - \sigma_2^2$. It is the same quantity used in DFD-STM [9,20], but computed using a different approximation over a small image region, and hence more robust. DFD based on STM uses a local cubic approximation for images which results in the Laplacians of the two images to be equal, i.e. $\nabla^2 g_1 = \nabla^2 g_2$. But, RDFD uses a better approximation allowing the two Laplacians to be different, i.e. $\nabla^2 g_1 \neq \nabla^2 g_2$. If we substitute $\nabla^2 g_1 = \nabla^2 g_2$ in Eq.(2.27) of RDFD, then Eq.(2.27) becomes similar to Eq.(2.17) except for integration. Eq.(2.27) can be solved by substituting for σ_1 in terms of σ_2 as follows.

A good approximation to the relation between normalized σ_i for $i = 1, 2$, in terms of the camera parameters for the i-th camera setting is [9]:

$$\sigma_i = m_i u^{-1} + c_i \tag{2.28}$$

where

$$m_i = -\frac{D_i s_0}{2\sqrt{2}} \quad \text{and} \quad c_i = -\frac{D_i s_0}{2\sqrt{2}}\Big[\frac{1}{f_i} - \frac{1}{s_i}\Big] \tag{2.29}$$

and s_i is the lens to image detector distance, s_0 is a known magnification constant, D_i is the aperture diameter, f_i is the focal length, and u is the object distance. Eliminating u in the two equations for σ_i for $i = 1, 2$, σ_1 can be expressed in terms of

σ_2 and known camera constants (obtained by calibration) α and β as:

$$\sigma_i = \alpha\sigma_2 + \beta \tag{2.30}$$

where

$$\alpha = \frac{m_1}{m_2} \quad \text{and} \quad \beta = c_1 - c_2\frac{m_1}{m_2} \tag{2.31}$$

Substituting Eq.(2.30) in Eq.(2.17) or Eq.(2.27), we obtain a quadratic equation:

$$(\alpha^2 - 1)\sigma_2^2 + 2\alpha\beta\sigma_2 + \beta^2 - G = 0 \tag{2.32}$$

which can be solved to obtain σ_2, and then the object distance u using Eq.(2.28). If the camera aperture diameter is not changed (i.e. $D_1 = D_2$), then $\alpha = 1$ and the equation becomes linear with a unique solution. In all our experiments, aperture diameter remains the same and therefore $\alpha = 1$, and a unique solution is obtained for σ_2. In the experiments, we use a lookup table obtained through calibration to solve for σ_2 and u after computing the quantity G from images using Eq.(2.27). Experimental results comparing the DFD with RDFD implemented directly using Eq.(2.27) is presented later. In the next section we present another interesting variation of RDFD.

2.4.1 RDFD based on Green's Identity

Image noise is dramatically amplified in the computed derivatives of images, and it becomes progressively hopeless as the order of image derivatives increases. Eq.(2.27) for implementing RDFD involves the computation of second-order image derivatives

(Laplacians) which yield very noisy estimates. In order to reduce the effects of noise, a new technique was used to compute the right hand side of Eq.(2.27). It uses the first-order image derivatives which are much less noise sensitive in comparison with the second-order derivatives (Laplacian). The well-known Greene's identity in Calculus is used to reduce the reliance on computing second order image derivatives by replacing some terms with equivalent terms involving only the first-order derivatives. Greene's identity is

$$\iint_R g\nabla^2 g = \int_B g(\vec{\nabla} g \cdot \vec{n}) - \int_B (\vec{\nabla} g \cdot \vec{\nabla} g) \tag{2.33}$$

where R is an image region and B is the boundary of R, $\vec{n}$ is the unit outward normal vector to the boundary, and $\vec{\nabla} g$ denotes the gradient vector of the blurred image $g(x, y)$. This identity is widely used in the theory of partial differential equations. In the case of one dimension this result can be derived as follows. Assume $g : [a, b] \to \Re$ be a twice differentiable function from an interval $[a, b]$ to the real numbers. An expression of the derivative of the product term $(g \cdot g')'$ can be rearranged to obtain

$$g \cdot g'' = (g \cdot \ g')' - (g')^2 \tag{2.34}$$

Integrating,

$$\begin{aligned} \int_a^b g \cdot g'' dx &= \int_a^b (g \cdot g')' dx - \int_a^b (g')^2 dx \\ &= g(b)g'(b) - g(a)g'(a) - \int_a^b (g')^2 dx \end{aligned} \tag{2.35}$$

Applying the two-dimensional equivalent of this formula to Eq.(2.27), we obtain

$$G = \sigma_1^2 - \sigma_2^2 = 8 \times \frac{\int\limits_B g_1(\vec{\nabla} g_1 \cdot \vec{n}) - \iint\limits_R \vec{\nabla} g_1 \cdot \vec{\nabla} g_1 - \int\limits_B g_2(\vec{\nabla} g_2 \cdot \vec{n}) + \iint\limits_R \vec{\nabla} g_2 \cdot \vec{\nabla} g_2}{\iint\limits_R (\nabla^2 g_1)^2 + \iint\limits_R (\nabla^2 g_2)^2} \tag{2.36}$$

In the numerator of the above equation, terms in the original equation (Eq.(2.27)) with area integrals involving image Laplacians (second-order derivatives) have been replaced by terms with boundary and area integrals of only the first-order image derivatives. Computationally, it may be more robust, but this conclusion is not certain because boundary integral of first-order derivatives may be as noise sensitive as area integral of second-order derivatives, especially when there is image shift due to camera shake or motion. Experimental results did not show significant improvement of accuracy, but the method is included here as future improvements may be possible.

2.4.2 Experiments

Many experiments were carried-out on both simulation and real image data to evaluate the performance of RDFD in relation to DFD. Some results on real image data are presented here. Two planar test objects- Checkerboard and BigLetter– shown in Fig.2-3 were used.

These objects were placed at 9 different distances in the range 300mm to 600mm from a digital camera. The digital camera had a computer controlled stepper motor for changing the lens position to focus at different distances. The lens step positions ranged from Step 0 to Step 1500. For each object and distance, two images of the

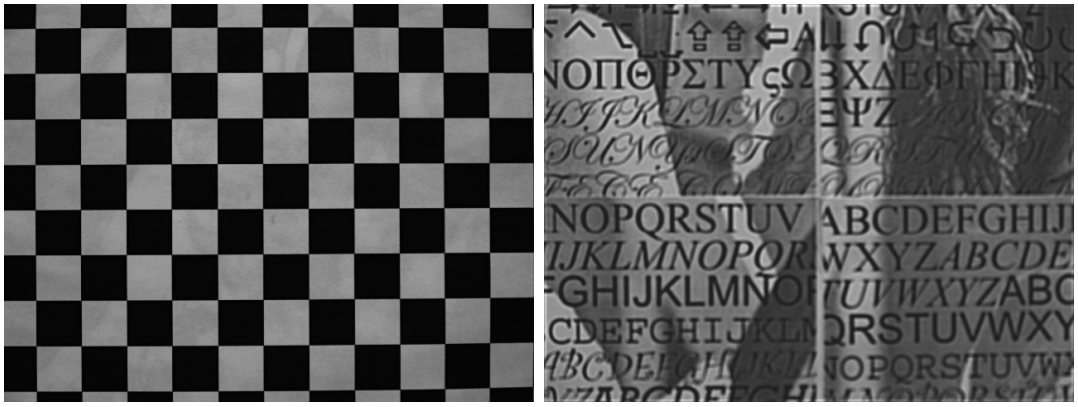


Figure 2-3: Test objects. Left: "Checkerboard". Right: "BigLetter".

test object were captured, one with the lens at step position 800, and another at step position 1100. Aperture diameter was not changed. The image captured at Step 1100 was shifted or scaled by different amounts to simulate the effect of camera/object motion or camera/hand shake. Note that image scaling simulates the effect of relative motion between the camera and the object along the direction of view (optical axis) as this motion results in changes in image magnification. The shifting and scaling amounts were limited to be small as in the case of actual applications. The image captured at Step 800 was not changed. The resulting images (size 600x480) were used for estimating the blur parameter σ and object distance for autofocusing. This estimate was done using both the RDFD and the DFD (STM) methods. The results are shown in Fig.2-4 and Fig.2-5.

In Fig.2-4(a) and (b), horizontal axis represents the reciprocal of object distance in meters, and the corresponding blur parameter σ is plotted along the vertical axis in pixel units. For an ideal thin-lens camera, this plot is expected to be roughly linear, but for practical cameras some deviation from linear behavior is usual. In Fig.2-4, the plots labeled "DFD by 0" and "RDFD by 0" represent the results for σ when there

is no image shift. This is the reference plot and other plots should be compared with this plot. For example, the plot labeled "DFD by 10" and "RDFD by 10" correspond to shifting the image at Step 1100 by 10 pixels. We see that the plot labeled "DFD by 10" is quite far from the plot labeled "DFD by 0", thus indicating a large error in the presence of 10 pixel image shift. This error for DFD generally increases as the shift increases. Therefore the DFD technique is very sensitive to image shifts. On the other hand, the plots labeled "RDFD by 0" and "RDFD by 10" almost coincide at different distances thus indicating that RDFD is robust with respect to 10 pixel image shifts. This is true even up to 20 pixel image shifts for the objects considered in our experiments.

Fig.2-4(c) and Fig.2-4(d) show the results in the case of scale/magnification changes for the image captured at lens Step 1100. The magnification was changed by up to 3 percent in the range [0.97, 1.03] where a factor of 1.0 represents no change. This simulates change in magnification due to different amounts of relative motion between the camera and the object. This motion is expected to occur in the elapsed time interval between the capture of the first image at lens step 800 and the second image at step 1100. For DFD, Fig.2-4(c) shows the deviation of computed σ with respect to the case when there is no magnification change (as shown in Fig.2-4(a) with label "DFD by 0"). Similarly, Fig.2-4(d) shows the results for RDFD. We see that the errors for DFD are usually much more than for RDFD leading to the conclusion that RDFD is robust against small changes in scaling.

Fig.2-5 shows the results for the other object BigLetter. The interpretation of the plots is similar to that for Fig.2-4. Again, the conclusion is the same, that RDFD is

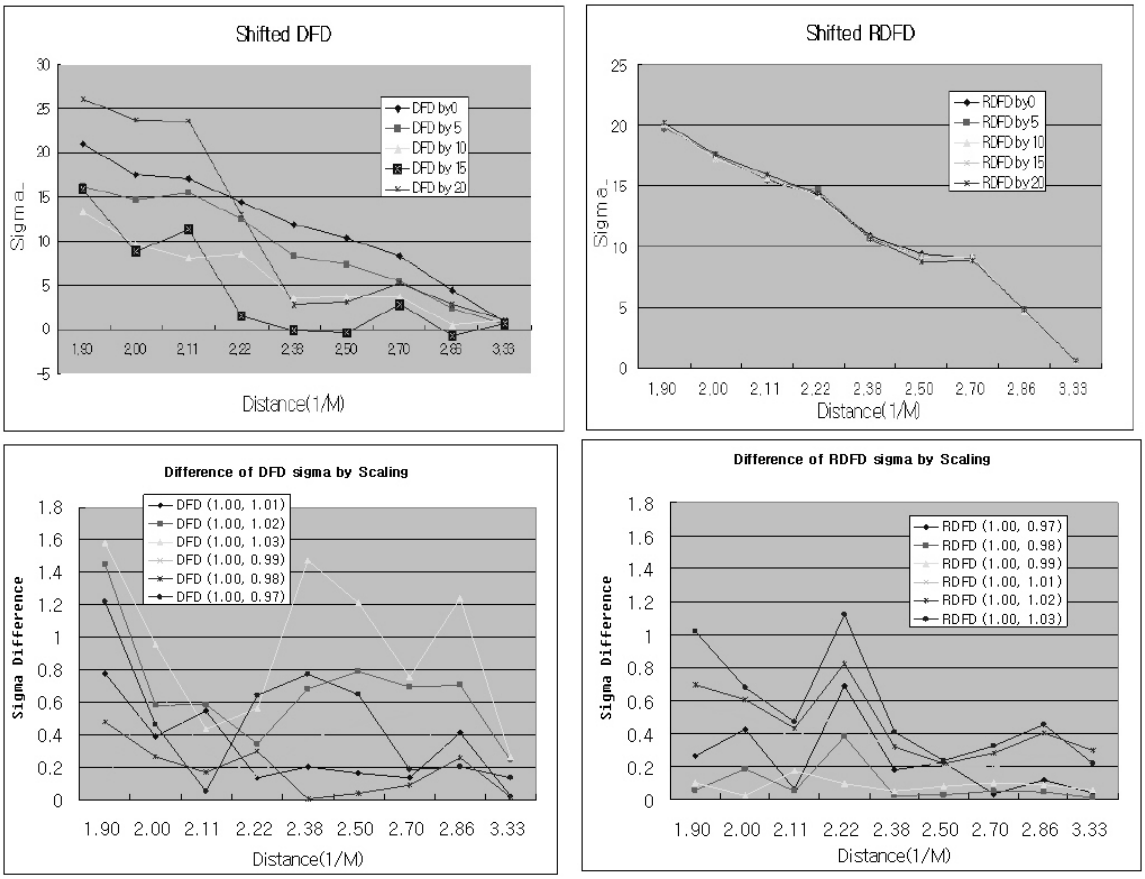


Figure 2-4: Results of DFD and RDFD with "Checkerboard". (a) Top left: DFD results with image shift. (b) Top right: RDFD results with image shift. (c) Bottom left: DFD result with scaling. (d) Bottom right: RDFD results with scaling.

far more robust with respect to image shifts and scale changes in comparison with DFD. As expected, the actual errors are not the same as in Fig.2-4. This indicates that the actual errors are dependent on the object pattern and contrast.

A limited set of experiments were carried out for the second version of RDFD that uses Eq.(2.36). The new version did not improve the results in comparison with the first version. In some cases it was worse than the first version. So these results are not presented.

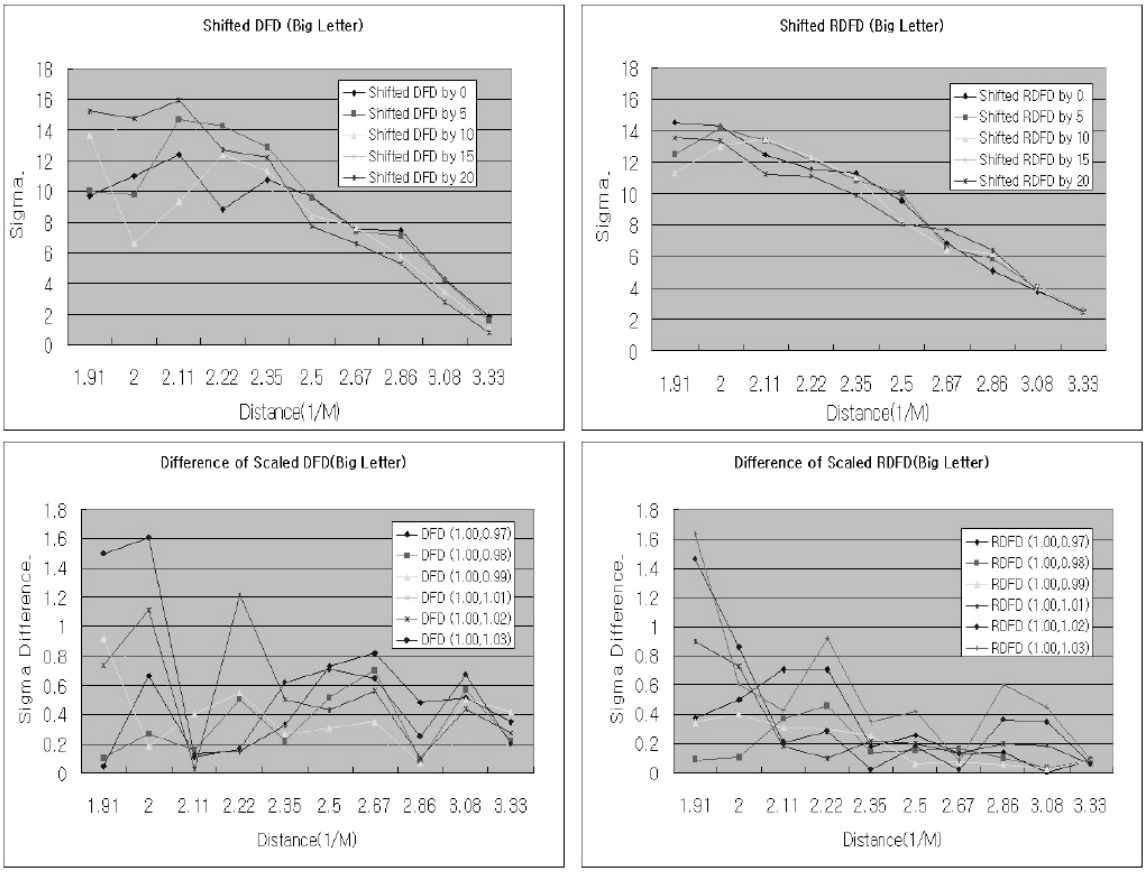


Figure 2-5: Results of DFD and RDFD with "BigLetter". (a) Top left: DFD results with image shift. (b) Top right: RDFD results with image shift. (c) Bottom left: DFD result with scaling. (d) Bottom right: RDFD results with scaling.

2.5 Two Variations of DFD: FET and BET

Two other variations of DFD were investigated. Both are closely related to DFD STM [9, 14, 20] but vary in some detail. The first one, Focus Equalization Technique (FET), is based on solving the following equation obtained by equating the right hand sides of Eq.(2.18) and Eq.(2.19):

$$\mathbb{FET}: \qquad 4(g_1(x,y) - g_2(x,y)) = \sigma_1^2 \nabla^2 g_1(x,y) - \sigma_2^2 \nabla^2 g_2(x,y) \tag{2.37}$$

The second one, Blur Equalization Technique (BET) [14], is based on solving

the following equation obtained by equating the right hand sides of Eq.(2.24) and Eq.(2.25):

$$\mathbb{BET}: \qquad 4(g_1(x,y) - g_2(x,y)) = \sigma_1^2 \nabla^2 g_2(x,y) - \sigma_2^2 \nabla^2 g_1(x,y) \tag{2.38}$$

The above equations are solved by substituting σ_2 with $\alpha\sigma_1 + \beta$ (with $\alpha = 1$) and solving for $\alpha = 1$. Effect of noise is reduced by computing $\alpha = 1$ at only those pixels where the Laplacian magnitude is more than a preset threshold. $\alpha = 1$ is averaged over a small image window to obtain a better estimate. These two methods give results that are comparable or sometimes better than the DFD STM, and they are both sensitive to image shifts as is the case with DFD STM. But these two methods are of interest as alternatives to DFD in the absence of image shifts.

2.6 Relation between Depth from Focus (DFF) and the Blur Parameter σ of Depth from Defocus (DFD)

DFF techniques [6] are based on computing a focus measure of blurred images. An example of a focus measure is the energy (i.e. integral of square) of image intensity or energy (integral of square) of derivatives of image intensity. The local convolution expression of STM can be used to derive a relation between these focus measures and the blur parameter on which the DFD techniques are based. By STM we obtain the

following expression for the focus measure based on image energy:

$$\iint_R g^2(x,y)dxdy = \iint_R \Big[f(x,y) + \frac{\sigma^2}{4}\nabla^2 f(x,y) + H.O.T.s\Big]^2 dxdy \tag{2.39}$$

Similarly, for the focus measure based on the energy of image gradient magnitude, we obtain

$$\iint_R (g_x^2 + g_y^2)dxdy = \iint_R \Big[f_x^2 + f_y^2 + \frac{\sigma^2}{4}(\nabla^2 f_x + \nabla^2 f_y) + H.O.T.s\Big]^2 dxdy \tag{2.40}$$

And for the focus measure based on the energy of image Laplacian we obtain

$$\iint_R (\nabla^2 g)^2 dxdy = \iint_R \Big[\nabla^2 f + \frac{\sigma^2}{4}\nabla^2(\nabla^2 f) + H.O.T.s\Big]^2 dxdy \tag{2.41}$$

In all the above three cases, the focus measure of the blurred image can be expressed in terms of the focused image and the blur parameter sigma as a polynomial of σ^2

$$FM(\sigma^2) = F_0 + F_2\sigma^2 + F_4\sigma^4 + H.O.T.s \tag{2.42}$$

where F_0, F_2, and F_4 are constants for a given focused image. For example, we obtain

$$\begin{aligned}\iint_R g^2(x,y)dxdy &= \iint_R f^2(x,y)dxdy + \frac{\sigma^4}{16}\iint_R [\nabla^2 f(x,y)]^2 dxdy \\ &\quad + \frac{\sigma^2}{2}\iint_R f(x,y)\nabla^2 f(x,y)dxdy + H.O.T.s\end{aligned} \tag{2.43}$$

Eq.(2.42) is a good approximation for small blur (i.e. small σ) which implies that

the camera parameter setting is close to the focus setting. This equation establishes an explicit relation between DFF and DFD techniques. It also provides a polynomial model for fitting the focus measures to a function of camera parameters. Such polynomial fitting is often done close to the focus setting (therefore σ will be small) to determine the value of the camera parameter setting that maximizes the focus measure.

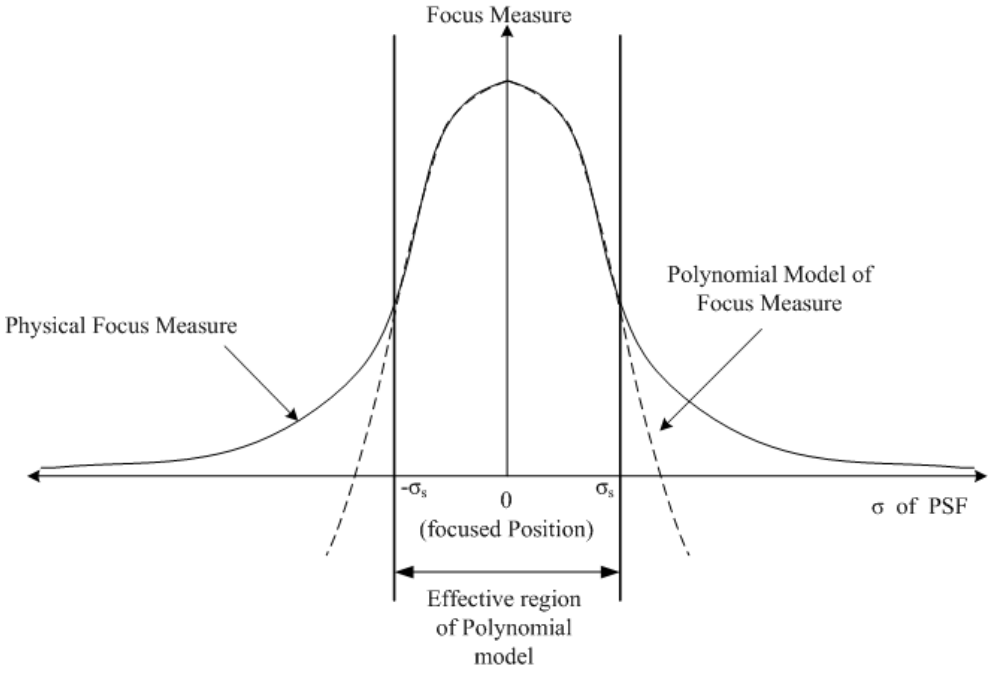


Figure 2-6: Polynomial model of Focus Measure

Fig. 2-6 illustrates the relation between a physical focus measure and its polynomial model, Eq.(2.42). σ_s denotes boundary in which polynomial model can approximate focus measure. The solid line represents actual focus measure, and the dotted line is the plot of polynomial model of focus measure.

We carried out several experiments on simulation data and real image data to verify that the peak position of focus measure predicted by the polynomial model

agrees with the peak position of actual computed focus measures. One such plot on real image data is shown in Fig. 2-8. We observe that the positions of the maximum focus measures coincide.

2.7 Summary

A new spatial domain depth-from-defocus technique– RDFD - is presented for autofocusing in digital cameras. It is adapted to be robust against image shifts due to camera/object motion and camera/hand shake. Its performance has been compared with a spatial domain DFD technique and its advantages in the presence of image shift and scale change have been demonstrated. RDFD shares some characteristics of both DFF and spatial domain DFD techniques. Investigation of RDFD leads to the derivation of an explicit relation between DFF and DFD techniques.

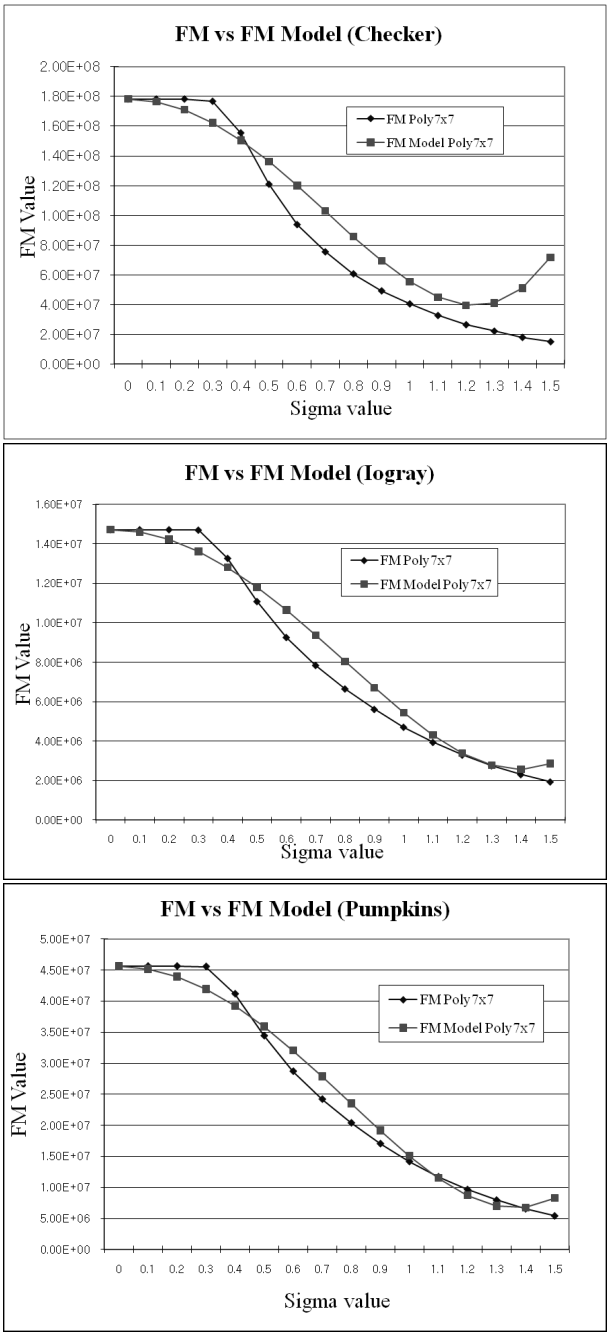


Figure 2-7: Polynomial Model of focus Measure. (a) Top: Checkerboard image, (b) Middle: Iogray image, (c) Bottom: Pumpkin image

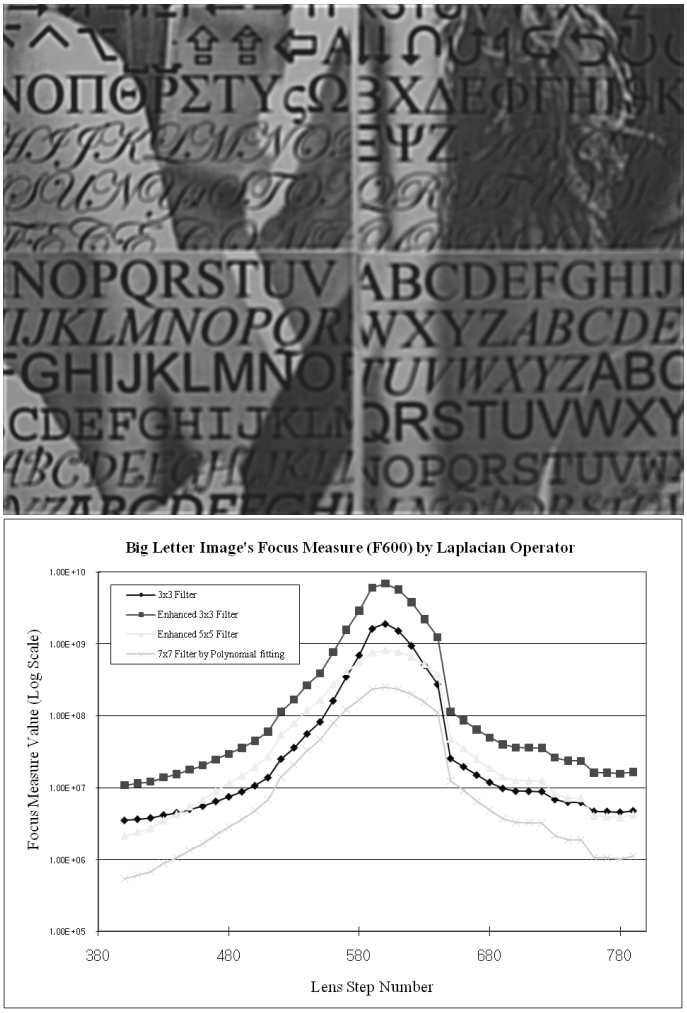


Figure 2-8: Focus Measure with Real Image. (a) Top: BigLetter image, (b) Bottom: FM of image in (a)

Chapter 3

Restoration of One-Dimensional Shift-Variant Blur

Fourier transform provides a useful mathematical tool to invert convolution. Therefore, blurred images that can be modeled as convolution of the original image with the shift invariant point-spread function (PSF) use FFT for deblurring. However, in many applications PSF is not shift invariant and its blurred image should be considered as the output due to a shift variant PSF. Many approaches have been proposed for shift-variant deblurring. In this chapter we investigate one approach based on the recently proposed Rao Transform (RT). This approach permits complete localization which enables parallel computation. Further, since RT does not restrict the shape of PSFs, e.g. only Gaussian PSF, Cylindrical PSF, or a rotationally symmetric PSF, etc., it is applicable to arbitrary PSFs. RT provides a closed-form solution with the weak assumption that the focused image at each point can be charaterized locally by a polynomial. It is a computationally efficient approach that can be easily implemented

in parallel, an therefore it is suitable for real-time appplications.

In this chapter the RT approach for one-dimensional shift-variant image restoration is tested and verified. The theory of one-dimensional RT, its advantages in comparison with related approaches, and experimental results are presented. Experiments include both simulated and real image data. Experimental results show that the new approach is effective and efficient. Examples of one-dimensional shift-variant blur are motion blur when a camera moves and one-dimensional barcode scanner. A noise sensitivity analysis is also presented for the new approach. RT for two-dimensional images is presented in chapter 4. RT can also be extended to higher dimensionsional signals [27].

3.1 Problem of Modeling of Shift Variant System

A conventional shift-variant image blurring model in the one-dimensional continuous domain is

$$g(x) = \int_r^s k(x,t)f(t)dt \tag{3.1}$$

Here x and t are real variables, $f(x)$ is an unknown real valued function representing the focused image that we need to solve for, and $g(x)$ and $k(x,t)$ are known (or given) real valued functions. $g(x)$ represents the blurred image and $k(x,t)$ represents the conventional shift-variant blurring kernel or Point Spread Function (SV-PSF). r and s are real constants. All functions here are assumed to be continuous, square-integrable, and differentiable up to some order as needed.

In the discrete domain, a standard matrix equation corresponding to the contin-

uous domain equation above can be set up and solved. However, this would result in a huge matrix corresponding to $k(x,t)$ and inverting it would be computationally exorbitant.

RT proposes a novel and efficient way to restate the above integral equation, Eq.(3.1), in an exactly equivalent form and without loss of any generality, using the change of variable

$$u = s - t \tag{3.2}$$

Note that the new variable u is a function of both x and t instead of just t. This novel idea, although simple, has eluded researchers in the past, and it facilitates localizing the problem completely and deriving a closed-form solution.

With the above change of variable, we obtain

$$g(x) = \int_{x-r}^{x-s} k(x,u) f(u) du \tag{3.3}$$

which can be rewritten as

$$g(x) = \int_{x-s}^{x-r} k(x, x-t) f(x-t) dt \tag{3.4}$$

The above Eq.(3.4) is named *Rao Transform* (RT) of $f(x)$. $f(x-t)$ in Eq.(3.4) can be expressed as a Taylor series expansion as

$$f(x-t) \approx \sum_{n=0}^{N} \frac{(-1)^n}{n!} t^n \frac{d^n f(x)}{dx^n} \tag{3.5}$$

Thus, Eq.(3.4) can be rewritten as

$$\begin{aligned} g(x) &= \int_{x-s}^{x-r} k(x, x-t) f(x-t) dt \\ &\approx \int_{x-s}^{x-r} k(x, x-t) \Big(\sum_{n=0}^{N} \frac{(-1)^n}{n!} t^n \frac{d^n f(x)}{dx^n} \Big) dt \\ &\approx \sum_{n=0}^{N} \frac{d^n f(x)}{dx^n} \frac{(-1)^n}{n!} \int_{x-s}^{x-r} t^n k(x, x-t) dt \end{aligned} \tag{3.6}$$

Therefore

$$g(x) \approx \sum_{n=0}^{N} k_n(x) f^{(n)}(x) \tag{3.7}$$

where

$$\begin{aligned} k_n(x) &\equiv \frac{(-1)^n}{n!} \int_{x-s}^{x-r} t^n k(x, x-t) dt \\ f^{(n)} &\equiv f^{(n)}(x) \equiv \frac{d^n f(x)}{dx^n} \end{aligned} \tag{3.8}$$

$k_n(x)$ has the form of weighted moment of kernel at point x, and $g(x)$ is rewritten as the sum of image derivatives multiplied by $k_n(x)$. Shift variant image blurring model is transformed into a differential equation as shown in Eq.(3.8).

If kernel $k(x, x-t)$ is an even function with respect to t (i.e. $k(x, x-t) = k(x, x+t)$) then, all odd moments of the kernel are zero.

$$k_n(x) = 0, \qquad \text{if } n = 1, 3, 5, \cdots \tag{3.9}$$

Since PSF is considered to be rotationally symmetric in many image applications,

this simplifies problem and saves computational task.

In practical applications, both $k_n(x)$ and $f^{(n)}(x)$ typically approach zero with increasing n. Let

$$f^{(m)}(x) = 0 \qquad \text{for} \quad m > N \tag{3.10}$$

Now, using suitable notation, we can derive from Eq.(3.7) an expression for the m-th order derivative of $g(x)$ for $m = 0, 1, 2, \ldots N$, as:

$$\begin{aligned} g^{(m)} &\equiv g^{(m)}(x) \equiv \frac{d^m g(x)}{dx^m} \\ &= \sum_{n=0}^{N} \sum_{p=0}^{m} C_p^m \; k_n^{(m-p)} \; f^{(n+p)}(x) \; T(n+p) \end{aligned} \tag{3.11}$$

where

$$k_n^{(m-p)}(x) = \frac{d^{(m-p)}}{dx^{(m-p)}} k_n(x) \tag{3.12}$$

and the function

$$T(n+p) = \begin{cases} 1 & \text{for } n+p \leq N \\ 0 & \text{otherwise} \end{cases} \tag{3.13}$$

ensures that terms with $f^{(n+p)}(x)$ in Eq.(3.11) are set to zero when $n + P > N$.

Eq.(3.11) can be simplified by grouping terms with respect to $f^{(n)}(x)$ to obtain

$$g^{(m)}(x) = \sum_{n=0}^{N} k_{m,n}(x) f^{(n)}(x) \qquad \text{for} \;\; m = 0, 1, 2, 3, \ldots, N \tag{3.14}$$

where

$$k_{m,n}(x) = \sum_{p=0}^{min(m,n)} \; C_p^m \; k_{n-p}^{(m-p)}(x) \tag{3.15}$$

Eq.(3.14) and Eq.(3.15) can be derived by noting the relation between the terms of $g^{(m)}(x)$ derived from $g^{(m-1)}(x)$ by applying the derivative operator for $m = 1, 2, \ldots, N$. These equations can be solved symbolically by using one equation to express an unknown in terms of the other unknowns, and substituting the resulting expression into the other equations to eliminate the unknown. Thus, both the number of unknowns and the number of equations are reduced by one. By this elimination process, we can eliminate the derivatives of $f(x)$ for $n = 0, 1, \cdots, N$. By repeating this process, we obtain $f^{(n)}(x)$ expressed in terms of $g^{(n)}(x)$ and coefficients. We may rewrite the equations as

$$f^{(m)}(x) = \sum_{n=0}^{N} k'_{m,n}(x) g^{(n)}(x) \qquad \text{for} \quad m = 0, 1, 2, 3, \ldots, N \tag{3.16}$$

where $k'_{m,n}$ denotes the coefficient of $g^{(n)}(x)$ for m-th derivative of $f(x)$.

The symbolic solution at point x is $f(x) = f^{(0)}(x)$, and we obtain $f^{(0)}(x)$ as

$$\begin{aligned} f^{(0)} &= f^{(0)}(x) = f(x) = \lim_{N\to\infty} \sum_{n=0}^{N} k'_{0,n}(x) g^{(n)}(x) \\ &\approx \sum_{n=0}^{N} k'_{0,n}(x) g^{(n)}(x) \end{aligned} \tag{3.17}$$

3.1.1 Matrix Form of RT for One-Dimension

We can express the above derived equations in a vector-matrix form. The major advantage of vector-matrix form of RT is the ease of inverting the equations and to apply error minimization techniques. Inversion of equation can be simplified and implemented by well developed techniques in linear algebra, such as Gaussian elimi-

nation and matrix decomposition. Also, error minimization techniques such as least square method and regularization method can be easily combined with RT. We can estimate error by a statistical analysis.

Eq.(3.14) can be written in vector-matrix form as:

$$\begin{bmatrix} g^{(0)} \\ g^{(1)} \\ \vdots \\ g^{(N)} \end{bmatrix} = \begin{bmatrix} k_{00} & k_{01} & \cdots & k_{0N} \\ k_{10} & k_{11} & \cdots & k_{1N} \\ \vdots & \vdots & \ddots & \vdots \\ k_{N0} & k_{N1} & \cdots & k_{NN} \end{bmatrix} \begin{bmatrix} f^{(0)} \\ f^{(1)} \\ \vdots \\ f^{(N)} \end{bmatrix} \tag{3.18}$$

The above equation can be written in a more compact form using vector and matrix variables as:

$$\mathbf{g}_x = \mathbf{K}_x \, \mathbf{f}_x \tag{3.19}$$

$\mathbf{g}_x$ and $\mathbf{f}_x$ are $(N+1) \times 1$ column vectors, and $\mathbf{K}_x$ is a square matrix of size $(N+1)\times(N+1)$. The subscript x in the above equation makes explicit the dependence of the vector/matrix on x. In this equation, note that the components of the column vectors are not the discrete sampled values of $g(x)$ and $f(x)$ at different points x, but they are the different order derivatives of $g(x)$ and $f(x)$ at the same point x. Therefore, in practical applications, the size of the matrices will be small (3 to 6), but this matrix equation will have to be solved many times, once at each (discrete) value of x. At each point x, $\mathbf{g}_x$ and $\mathbf{K}_x$ will have to be computed from known data around the point x.

We can obtain a symbolic solution for $f(x) = f^{(0)}(x)$ from Eq.(3.18) or (3.19)

through a successive Gaussian elimination and back substitution method. This symbolic solution is useful in practical applications as the value of N is typically very small, around 3 to 6. Assuming that the matrix $\mathbf{K}_x$ is non-singular, the solution can be written in matrix form as:

$$\mathbf{f}_x = \mathbf{K}_x^{-1}\ \mathbf{g}_x = \mathbf{K}'_x\ \mathbf{g}_x \tag{3.20}$$

In explicit algebraic form, elements of $\mathbf{K}'_x$, $\mathbf{f}_x$, and $\mathbf{g}_x$ are respectively denoted by K'_{ij}, $f^{(i)}(x)$, and $g^{(i)}(x)$.

If $k_n(x)$, $g^{(n)}(x)$, and $f^{(n)}(x)$ approach zero in $n \leq N$ and N is always same at every points x, then, Eq.(3.20) has same symbolic elements at every points x. Once we obtain a symbolic solution for $f(x) = f^{(0)}(x)$, solution at point x is obtained just by substituting actual value of x. This simplifies and saves computational task.

In practical applications, in order to reduce the effects of noisy data on the solution, it is easy to use a standard regularization technique to solve Eq.(3.19) for $\mathbf{f}_x$. For example, a spectral filtering technique based on the Singular Value Decomposition (SVD) of $\mathbf{K}_x$ such as the Truncated SVD or Tikhonov regularization [21] can be used.

3.1.2 Example

Let the shift-variant kernel be a Gaussian SV-PSF where the parameter σ is a function of $(x - t)$ denoted by $\sigma(x - t)$ (clarification on notation: it is not a product of σ and

$(x-t))$, we let

$$k(x, x-t) = \frac{1}{\sqrt{2\pi}\sigma(x-t)} \exp\Big(- \frac{(x-(x-t))^2}{2\sigma^2(x-t)}\Big)$$

Within the interval of significant support for the above filter, for example, let $k(x, x-t) \approx 0$ for $|t| \geq 2\sigma(x-t)$, a useful approximation in practical applications is that $k(x, x-t)$ is given by a truncated Taylor-series expansion up to some order M. With this approximation, denoting the derivative of $\sigma(x)$ with respect x by $\sigma_x(x)$ and setting $N = 2$, and $M = 1$, we obtain the following solution corresponding to Eq.(3.19) and Eq.(3.20):

$$\mathbf{K}_x = \begin{bmatrix} 1 & 2\sigma(x)\sigma_x(x) & \frac{1}{2}\sigma^2(x) \\ 0 & 1 & 3\sigma(x)\sigma_x(x) \\ 0 & 0 & 1 \end{bmatrix}$$

and $\mathbf{K}'_x = \mathbf{K}_x^{-1}$ is given by

$$\begin{bmatrix} 1 & -2\sigma(x)\sigma_x(x) & -\frac{1}{2}\sigma^2(x)(1-12\sigma_x^2(x)) \\ 0 & 1 & -3\sigma(x)\sigma_x(x) \\ 0 & 0 & 1 \end{bmatrix}$$

To our knowledge, a closed-form solution for the restored image $f(x)$ such as above has not been provided in previous literature (See [26, 27]).

3.2 Advantage of RT

Unlike other approaches [21,22,26], the new approach provides a *completely localized solution* in the following sense-the solution $f(x)$ at a point x is expressed in terms of the values of the known function $g(x)$ *at the same point* x, and also the characteristics (or moments with respect to t) of the *localized kernel at the same point* x. In problems such as shift-variant image deblurring, this facilitates rapid convergence and accuracy of the series expansion in Eq.(3.17) that provides the solution. For the same reason, it permits a fine-grain parallel computation. The solutions at different points x can be computed in parallel using local data of the observed/measured function $g(x)$ (whose local derivatives are used). At each point x, elements of a small (3x3 to 6x6 in practice, depending on data noise and solution accuracy) matrix $\mathbf{K}_x$ are computed and inverted. In practical applications, this matrix $\mathbf{K}_x$ often has a very simple form and can be computed very efficiently. For example, in shift-variant image deblurring example presented in the previous section, it is upper triangular and can be computed fast, either using analytic expressions, or, because the kernel is symmetric with respect to t, i.e. $k(x, x-t) = k(x, x+t)$, and has compact support with respect to t, i.e. it satisfies the condition $k(x, x-t) \approx 0$ for $|t| \geq A$ for some constant A which is much smaller than the size of the domain of integration or definition of $f(x)$ (this is the case when the blurring model is based on geometric optics).

The solution at each point x will be *compatible* with the solutions at near-by points (up to some order derivatives of the solution). Therefore the *local solutions* can all be sewed or synthesized *seamlessly* (without glitches or discontinuities) at

the borders to obtain a *global solution* to the original problem. The new method provides an analytic solution suitable for both theoretical analysis and numerical implementation. Numerical methods based on the new method are likely to offer significant computational savings [9]. The new approach can be naturally extended from one-dimensional to multi-dimensional problems.

Shift-variant deblurring is an ill-posed inverse problem and therefore a regularization approach such as the Truncated Singular Value Decomposition (TSVD) or Tikhonov regularization [21] could be used for solving Eq.(3.19) in the new approach to obtain smooth and stable restoration. Experimental investigation of incorporating regularization is an important research topic, as well as comparing the computational performance of the new approach and its variations with existing approaches in practical applications.

A special case of shift-variant deblurring addressed by the new approach is shift-invariant deblurring or deconvolution. This special case is obtained when the kernel k in Eq.(3.1) depends only on $(x-t)$. In this case, the solution provided by Eq.(3.20) and Eq.(3.17) is the same as that provided by the spatial domain deconvlution result provided in [7, 9].

3.3 Noise Sensitivity Analysis of Inverse RT Filter

In this section we derive expressions for noise sensitivity of image restoration based on RT. Expressions for local noise variance (or noise power) at a pixel in the restored image is obtained in terms of local noise variance in the input image. The restored

image at each pixel is computed as the weighted summation (linear combination) of pixel values in the input image in a small window that is used to estimate image derivatives. The weights or the coefficients of the inverse RT filter are all assumed to be noise-free. Only the input image data is assumed to be noisy. The noise variance at different pixels can be different as the filter coefficients change from pixel to pixel in shift-variant blurring/deblurring. The noise variance at a pixel in the restored image is useful in determining the degree of reliability of the deblurred image at that pixel. Clearly, unlike shift-invariant deblurring or deconvolution, this reliability can vary from pixel to pixel.

Let $g(m,n)$ be the blurred noise free discrete image and $\eta\ (m,n)$ be the additive noise. The noisy blurred digital image recorded by the camera is

$$g_\eta(m,n) = g(m,n) + \eta(m,n) \tag{3.21}$$

The noise $\eta(m,n)$ at different pixels are assumed to be independent, identically distributed, random variables with zero mean and standard deviation σ_n. The window size of the spatial domain filter is taken to be $(2L+1)\times(2L+1)$ and the image size is taken to be $M \times N$. First, we derive expressions for noise in the estimation of image derivatives based on convolution filters. This result will then be used to obtain noise variance in shift-variant image restoration based on RT. Convolution filtered image for estimating image derivatives can be expressed as

$$
\begin{aligned}
g_{fil,\eta}(m,n) &= \sum_{k,l=-N}^{L} w(k,l) g_{\eta}(m+k,n+l) \\
&= g_{fil}(m,n) + \eta_{fil}(m,n) \qquad (3.22)
\end{aligned}
$$

where

$$
g_{fil}(m,n) = \sum_{k,l=-L}^{L} w(k,l) g(m+k,n+l)
$$

and

$$
\eta_{fil}(m,n) = \sum_{k,l=-L}^{L} w(k,l) \eta(m+k,n+l)
$$

$g_{fil,n}(m,n)$ denotes noisy derivative of blurred image and $w(k,l)$ denotes the spatial filter for computing the desired derivative. The local Power $\mathbf{P}$ of $g_{fil,n}(m,n)$ can be written as

$$
\begin{aligned}
\mathbf{P}(g_{fil,\eta}) &= (g_{fil,\eta})^2 = (g_{fil}(m,n) + \eta_{fil}(m,n))^2 \qquad (3.23) \\
&= \mathbf{P}(g_{fil}) + \mathbf{P}(\eta_{fil}) + 2 g_{fil}(m,n) \eta_{fil}(m,n)
\end{aligned}
$$

where

$$
\mathbf{P}(g_{fil}) = (g_{fil}(m,n))^2 \text{ and } \mathbf{P}(\eta_{fil}) = (\eta_{fil}(m,n))^2 \qquad (3.24)
$$

The expected value (mean or average) of power of the filtered image $E\{\mathbf{P}(g_{fil})\}$

is

$$E\{\mathbf{P}(g_{fil})\} = \sum_{k_1,l_1=-L}^{L} \sum_{k_2,l_2=-L}^{L} w(k_1,l_1)w(k_2,l_2)E\{\eta(m+k_1,n+l_1)\eta(m+k_2,n+l_2)\} \tag{3.25}$$

If $k_1 \neq k_2$ or $l_1 \neq l_2$, then

$$E\{\eta(m+k_1,n+l_1)\eta(m+k_2,n+l_2)\} = 0 \tag{3.26}$$

If $k_1 = k_2$ or $l_1 = l_2$, then

$$E\{\eta(m+k_1,n+l_1)\eta(m+k_2,n+l_2)\} = E\{\eta^2(m+k_1,n+l_1)\} = \sigma_n^2 \tag{3.27}$$

Therefore, we get

$$E\{\mathbf{P}(\eta_{fil})\} = \sum_{k_1,l_1=-L}^{L} \sum_{k_2,l_2=-L}^{L} w^2(k_1,l_1)\sigma_n^2 = \mathbf{W}_e\sigma_n^2 \tag{3.28}$$

where

$$\mathbf{W}_e = \sum_{k,l=-L}^{L} w^2(k,l) \tag{3.29}$$

Therefore,

$$E\{\mathbf{P}(\eta_{fil})\} = \mathbf{P}(g_{fil}) + \mathbf{W}_e\sigma_n^2 \tag{3.30}$$

The expected value of local power $E\{\mathbf{P}(\eta_{fil})\}$ is the sum of expected local power of two parts– noise-free filtered image and the filtered noise. The local power of filtered noise is product of the energy of the filter and noise variance. For example,

the Laplacian operator is defined as

$$\bigtriangledown^2 = \begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix} \tag{3.31}$$

The expected noise power of this Laplacian filter is $20\sigma_n^2$ since $\mathbf{W}_e = 4 \times 1^2 + (-4)^2 = 20$.

Noise power in one-dimensional RT can be estimated similar to the two-dimensional case above in Eqs.(3.21) - (3.30); the blurred image is restored using Eq. (3.17) which is a weighted sum of the derivatives of the input blurred image. The weights are determined by the blurring kernel. The blurred image derivatives are obtained by spatial differentiation filters such as the Laplacian operator. Let every window for 1-D image derivative have a size of $1 \times (2L + 1)$. Then, the window can be specified as

$$W^{(n)} = [w_{n,-L} \quad w_{n,-L+2} \quad \cdots \quad w_{n,0} \quad \cdots \quad w_{n,L-1} \quad w_{n,L}] \tag{3.32}$$

where $W^{(n)}$ denotes the differentiation window of order of n. Now Eq.(3.17) can be

rewritten as

$$\begin{aligned} f_{retro}(x) &\approx \sum_{n=0}^{N} k'_{0n}(x) g^{(n)}(x) \\ &= \sum_{n=0}^{N} k'_{0n}(x) \left(\sum_{i=-L}^{L} w_n(i) g(x+i) \right) \\ &= \sum_{i=-L}^{L} \sum_{n=0}^{N} L_n(x) w_n(i) g(x+i) \\ &= \sum_{i=-L}^{L} \Phi(x,i) g(x+i) \end{aligned} \tag{3.33}$$

where $f_{retro}(x)$ denotes the restored 1-D image and

$$L_n(x) = k'_{0n}(x), \quad \text{for all i} \tag{3.34}$$

$$\Phi(x,i) = \sum_{n=0}^{N} L_n(x) w_n(i) \tag{3.35}$$

Now, by Eq.(3.28) - (3.30), the noise power in restoration would be

$$\mathbf{E}\left[\mathbf{p}(\eta_{fil}(x))\right] = \sum_{i=-L}^{L} (\phi(x,i))^2 \sigma_n^2 \tag{3.36}$$

Thus we can estimate the expected local noise power in a deblurred image obtained using inverse RT from Eq. (3.36) and (3.30). This expected power is a measure of the reliability of the restored image result and it can vary from pixel to pixel depending on the shift-variant blurring kernel.

Example

RT Image restoration with truncated Taylor Series expansion up to $M = 1$ and

$N = 2$ is

$$f(x) = g^{(0)}(x) - g^{(1)}(x)h_2^{(1)}(x) - g^{(2)}\left(\frac{1}{2}h_2^{(0)}(x) - \frac{3}{2}(h_2^{(1)}(x))^2\right) \tag{3.37}$$

Let the differentiation window with $L = 1$ be

$$\begin{aligned} W^{(0)} &\equiv [w_0(-1), w_0(0), w_0(1)] = [0,\ 1,\ 0]\,, \\ \frac{\partial}{\partial x} = W^{(1)} &\equiv [w_1(-1), w_1(0), w_1(1)] = [0,\ -1,\ 1]\,, \quad \text{and} \\ \frac{\partial^2}{\partial x^2} = W^{(2)} &\equiv [w_2(-1), w_2(0), w_2(1)] = [1,\ -2,\ 1] \end{aligned} \tag{3.38}$$

$$L_0(x) = 1, \quad L_1(x) = -h_2^{(1)}(x), \quad L_2(x) = -\left(\frac{1}{2}h_2^{(0)}(x) - \frac{3}{2}(h_2^{(1)}(x))^2\right) \tag{3.39}$$

$$\phi(x, i) = \sum_{n=0}^{N} L_n(x)w_n(i) \qquad \text{for i = -1, 0 1} \tag{3.40}$$

Substituting Eq.(3.40) into Eq.(3.36) noise sensitivity of RT at each pixel can be estimated.

3.4 Experiment

Simulations and some real experiments have been carried-out. In this section we present some experimental result for image restoration. In section 3.4.1, we analyze the performance of RT when a sinusoidal wave is the input to a shift-variant system,

and RT restores blurred barcode image with Gaussian SV-PSF in section 3.4.2.

3.4.1 Restoration of Sinusoidal Pattern

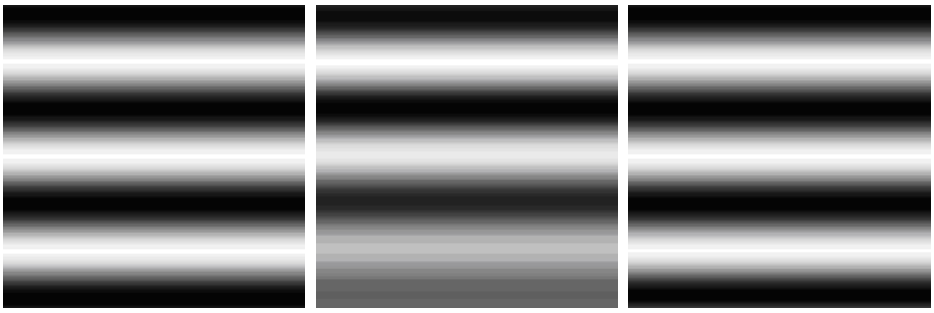


Figure 3-1: Shift-variant Blurred, Restored. Cross section plots are shown below. Input function = $sin(x)$, $N = 8$, $M = 2$, $x_{max} = 10$, $x_{min} = -10$, Sampling Frequency=200, Gaussian PSF: $\sigma(x) = 0.5 + 0.1x$.

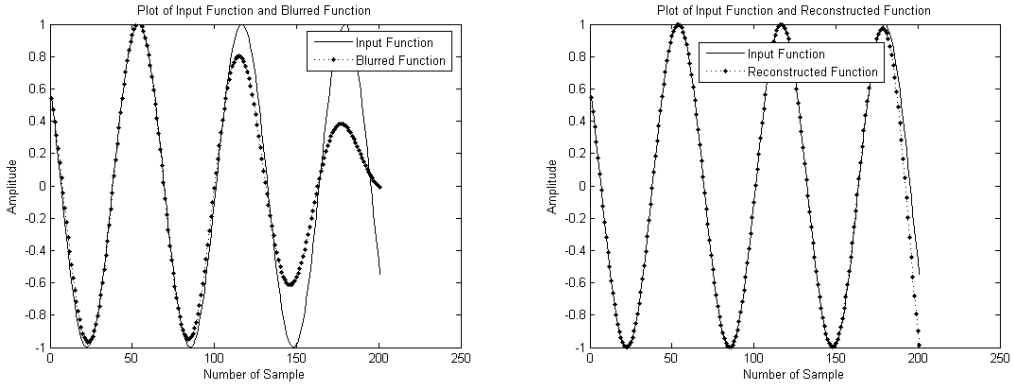


Figure 3-2: Cross section plots of Fig.3-1: The left-side plot shows blurred sine wave, and the right-side plot shows restoration result.

First we present the result of a simulation experiment. The unknown function f was chosen to be a polynomial of a certain order (e.g. $7x^5+6x^4-2x^3-6x^2+5x-1$) or a Sine function of a certain period (e.g. $\sin(x)$), then the kernel k was chosen to be one of Gaussian or rect (and a Cylindrical in the 2D case) with a Taylor series expansion up to order $M = 1$ *or* 2. The order N of the polynomial was varied (3 to 8) and the

period of the Sine function was varied ($\sin(x)$ to $\sin(2x)$) in different experiments. The spread parameter sigma of the SV-PSF in the different cases was varied linearly (e.g. $s = 0.5 + 0.1x$). The analytic expressions for the blurred image and the restored image were plotted in an interval (e.g. $x_{min} = -10$ to $x_{max} = 10$ with 200 sampled points). As expected, when the unknown function was a polynomial, the solution for $f(x)$ was exact. However, in the case of sine functions, due to truncation of the series expansion, as expected, the solution had small errors. This error increased when the ratio of the parameter sigma to the period of the sine wave increased. The error was small up to a ratio of around 0.2. See Fig.3-2. Experimental results on 2D image data is presented in next chapter.

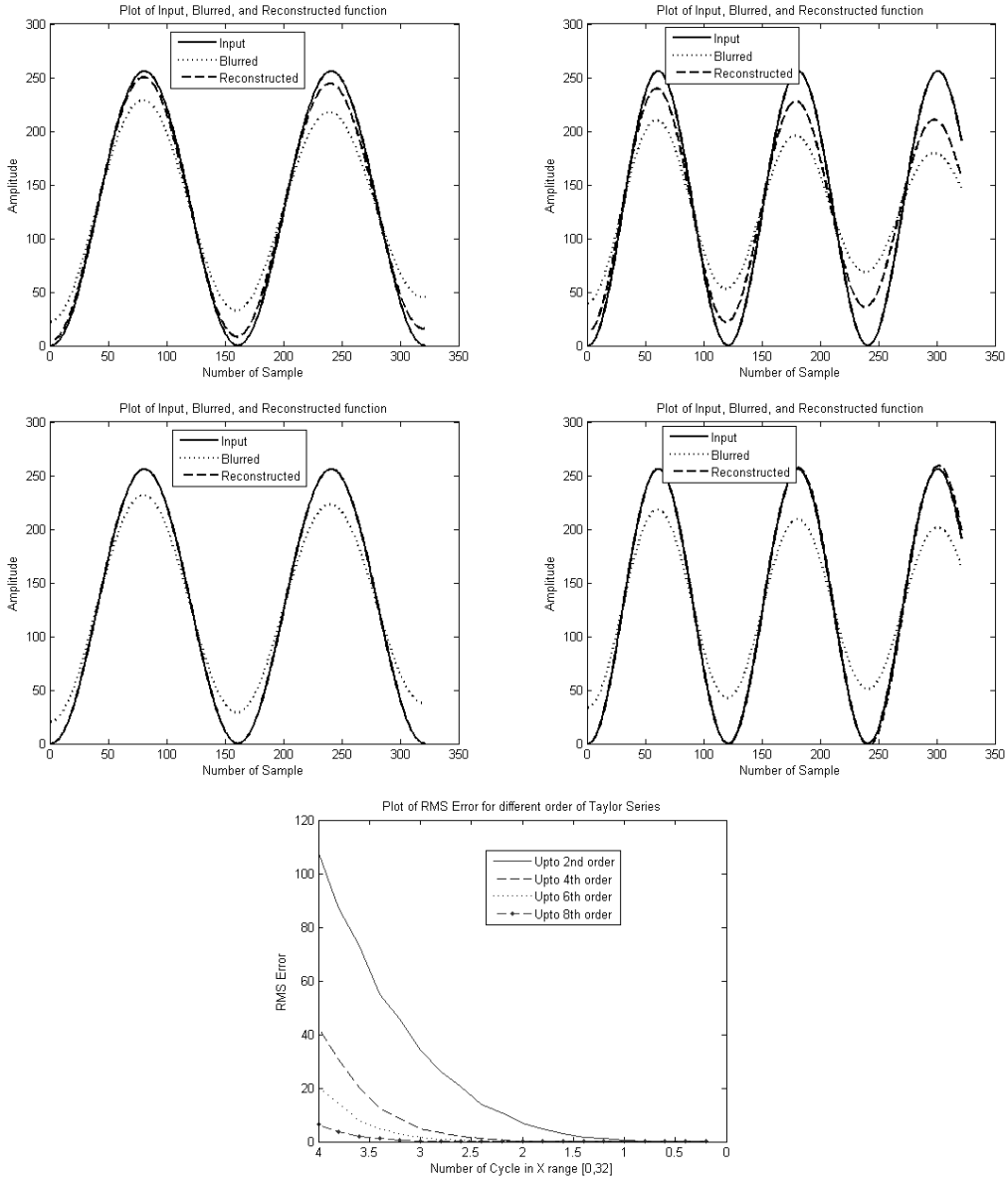


Figure 3-3: Simulation of 1-D blur with two sinusoidal waves: Top: $N = 2$ and $M = 2$, Middle: $N = 4$ and $M = 2$, Bottom: Input signal frequency vs. Taylor series order

In order to investigate the error due to truncation of Taylor Series expansion at different order terms, another simulation was performed with order $N = 2$ or 4 and $M = 2$ for two sinusoidal input waves. The input signal was

$$\begin{aligned} f(x) &= 128 - 128(cos(\alpha x)) \quad \text{with } \alpha = \pi/8 \text{ or } \pi/6 \\ \sigma(x) &= 1.5 + 0.002x \quad \text{for} \quad x \in [0, 32] \end{aligned}$$

It was sampled at 10 samples per unit interval. As we expected, RT with higher order of Taylor series expansion gave more accurate reconstruction results (See Fig.3-3). Higher the frequency of input signal, more the restoration error. This is because higher frequency signals have more energy in the higher order terms in Taylor-series expansion. In other words, higher frequency signals require higher order of Taylor Series expansion for same level of restoration quality.

3.4.2 Experiment on Barcode Image with Gaussian SV-PSF

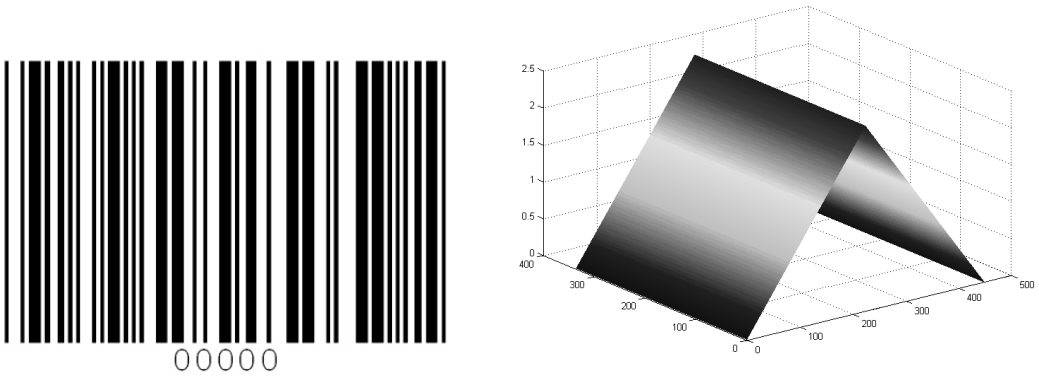
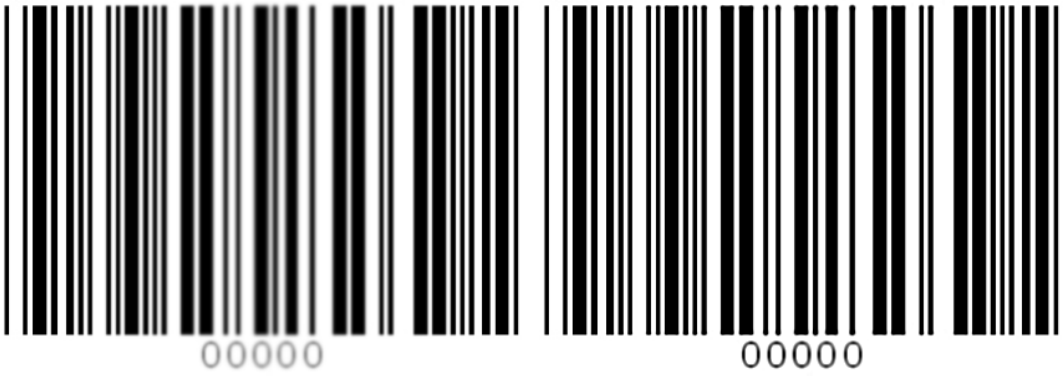


Figure 3-4: Simulation of 1-D blur with barcode image. Left: Input focused Image. Right: plot of parameter sigma of the shift-variant Gaussian PSF.

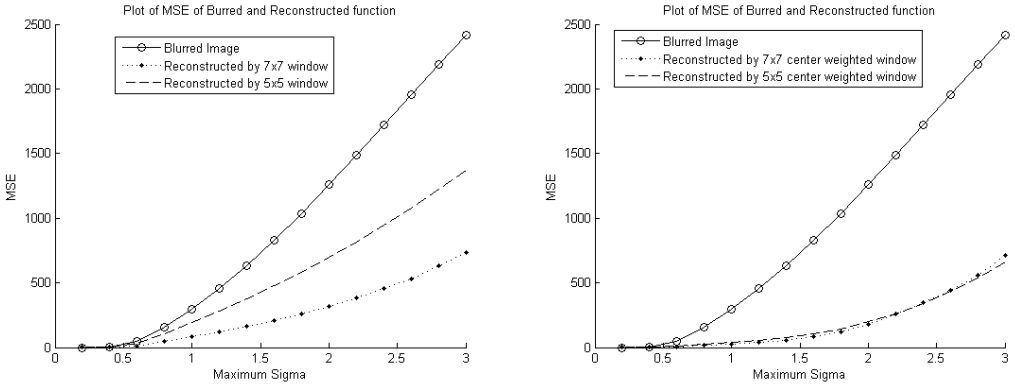


Left:Blurred Image with $\sigma_{max} = 1.5$ Right:Image reconstructed from the left image

Left:Blurred Image with $\sigma_{max} = 2.5$ Right:Image reconstructed from the left image

Figure 3-5: Shift-variant blurred and restored images. The parameter σ of Gaussian PSF is varied linearly along horizontal axis. σ has a maximum value at the image center.

Simulation experiments were carried-out on restoring 1-D barcode images with shift-variant blur. The barcode image was blurred by a Gaussian SV-PSF whose σ was varied as shown in Fig.3-4. For example, $\sigma(x) = \sigma_{max} - \frac{\sigma_{max}}{M} \left| x - \frac{M}{2} \right|$ where M denotes the horizontal size of image and σ_{max} denotes the maximum value of σ at the center. The barcode image size was 425×313. σ_{max} was 1.5 or 2.5 along the vertical line at the image center. The blurred image was restored with Taylor Series Expansion of order up to $M = 1$ and $N = 2$. The window size for computing image derivatives was 7×7 (with L=3). The second differentiation filter employed was based on fitting a cubic polynomial to image data by a Least Square Method.



Left:Window without center weighting Right:Window with center weighting

Figure 3-6: Plots of Mean Square Error: Gaussian Blur

We compare the MSE (Mean Square Error) for the blurred and the restored images for 2 different sizes of differentiation window with/without center weighting. The windows used for comparison have sizes of 5×5 and 7×7. Every differentiation window in Fig.3-6 was derived by cubic polynomial fitting and the corresponding center weighted window was obtained by center-weighting the coefficients, by 5,3,1, for 5×5, and 7,5,3,1, for 7×7, respectively from center to the border. In our experiment the filter window obtained by weighted polynomial fitting for derivative gave better restoration than that with general polynomial fitting. The MSE of reconstructed images using center weighted 7×7 window were around 0.15 times that of blurred image with $\sigma_{max} = 2$, but MSE using non-weighted 7×7 window was around 0.25 times that of the blurred image. Center weighted windows were more sensitive to noise (computed using Eq. (3.36)) than un-weighted windows.

3.5 Summary

In this chapter We have presented the theory of RT for one-dimensional shift-variant image deblurring, its advantage, noise sensitivity analysis, and experimental results. RT was found to be effective for inversion of one dimensional shift-variant system. The performance of one dimensional RT was verified by simulations and experiments with various inputs such as sinusoidal pattern and simulated barcode images. Experimental results are found to be satisfactory in terms of improvement in image quality and mean square error (MSE). One dimensional RT has also been applied to 2D deblurring to obtain good results in the following chapter 4. RT is completely localized, computationally efficient, and permits fine-grain parallel processing. It is not limited to any specific shape of PSF such as Gaussian PSF, and it is therefore extendable to deblurring images with optical aberrations such spherical and coma aberrations.

Chapter 4

Shift-Variant Image Deblurring: Two-dimensional Case

Restoration of a shift-variant blurred image is in general a two-dimensional problem as ooposed to the one-dimensional problem considered in the previous chapter. Shift-variant image blur may be due to misfocused three-dimensional object, imperfection of camera lens system such as lens abberation, and motion of objects/camera. As in the one-dimensional case, previous approaches for shift-variant image restoration have serious limitations in terms of computational needs, accuracy of restoration, restrictions on the type of PSFs, and parallel implementation. In the two-dimensional case, these problems become far more serious and difficult to solve. However, the complete localization feature of RT in modeling the blurring and deblurring process offers advantages in the restoration of two-dimensional images. In this chapter, we investigate the theory, application, and performance evaluation of RT in restoring shift-variant blurred images.

4.1 Brief Review of Prior Art

Many approaches for a shift-variant system attempt to transform or approximate the shift-variant system as a shift-invariant system in small image blocks or sections. This facilitates the use of well-known techniques for shift-invariant or convolution deblurring. One such approach is based on image warping or coordianate transformation [51,54,57,58]. In this approach an image is geometrically warped or transformed or distorted in such a way that the result of warped image can be approximated by a shift-invariant blurred image. Such a warped image is restored by well developed techniques for shift-invariant system. The problem of approaches based on image warping is that this approach is applicable to only a few special cases with certain shapes of blurring PSF.

Approaches based on the assumption of piecewise shift-invariance divides an image into small blocks or sections. The size of image sections have to be carefully selected so that variation of PSF within a section should be small enough to be negligible. Every image section is restored separately using techniques for shift-invariant deblurring. Then, the full restored image is obtained by sewing restored sections together like a mosaic. One problem of this method is discontinuity between sections in the restored image, and the second problem of this is that this method is valid only when the PSF varies slowly. The faster the PSF changes with image position, the more the error in the restored image.

Another method directly models shift-variant blurring as a matrix equation (e.g. $\mathbf{g} = \mathbf{R}\mathbf{f}$) [21, 24, 26]. Focused clear image $\mathbf{f}$ is obtained by inversion of kernel matrix

R. Singular value decomposition (SVD) is a useful method for this matrix inversion. But this matrix approach to modeling shift-variant or space-variant blurring for two dimensional images are not fully localized and computationally exorbitant. Therefore some approaches attempt to reduce computational cost by adopting the sectioning method in this matrix approach. A large shift-variant blurred image is divided into many smaller image blocks of size $K \times K$ where K is around 32 or 64 and restored separately and then synthesized. A two-dimensional image block is converted into a very large $K^2 \times 1$ one-dimensional vector $\mathbf{b}$ by rearranging the columns of the image block vertically one below another. The shift-variant blurring kernel or point spread function (SV-PSF) is specified by a huge $K^2 \times K^2$ matrix $\mathbf{A}$. The focused image vector $\mathbf{x}$, again represented as a large $K^2 \times 1$ column vector, is obtained by solving the matrix equation $\mathbf{Ax} = \mathbf{b}$. Matrix inversion for one image block costs $O(K^6)$, and matrix inversion is required once for each image block. The major problem with this matrix approach is the very high computational complexity, even when sectioning method is adopted. Moreover, this formulation of the problem does not exploit the natural locality and limited support domain of the physical blurring kernel and spatial smoothness of focused images.

In contrast to the above mentioned methods, two dimensional RT models the blurring process in a completely local form that naturally exploits the spatial locality and limited support domain of blurring kernels (SV-PSFs) and formulates the problem in terms of the derivatives of an analytic approximation to the focused image, which, unlike raw pixel data, exploits spatial smoothness in images. As a consequence, in a typical practical application, the new approach reduces computation significantly,

and provides a new theory. In this sense, the new approach represents a fundamental theoretical and computational advance. Further, RT is also relevant to solving integral/differential equations and shape from defocus [27, 28].

4.2 Theory of RT for Two-dimension

In the continuous domain, the shift-variant point spread function (SV-PSF), the focused image, and the corresponding blurred image are represented by $h(x, y, \alpha, \beta)$, $g(x, y)$, and $f(x, y)$ respectively. The conventional blurring model uses the global form SV-PSF k, and is given by

$$g(x, y) = \int_a^b \int_c^d k(x, y, u, v) f(u, v) du dv \tag{4.1}$$

A localized kernel $h(x, y, u, v)$ corresponding to $k(x, y, u, v)$ can be defined as [27]

$$h(x, y, u, v) = k(x + u, y + v, x, y) \tag{4.2}$$

A completely localized blurring model which is exactly equivalent to Eq.(4.1) can be obtained by the change of variables $u' = x - u$ and $v' = y - v$. The resulting expression defines two-dimensional *Rao Transform* (RT):

$$g(x, y) = \int_{x-b}^{x-a} \int_{y-d}^{y-c} h(x - u, y - v, u, v) f(x - u, y - v)\ du dv \tag{4.3}$$

Next we derive the inverse of RT to solve the above integral equation for the

focused image f. The m-th partial derivative with respect x and the n-the partial derivative with respect to y of a function will be denoted by the superscript (m, n) for the function:

$$f^{(m,n)} = \frac{\partial^m}{\partial x^m}\frac{\partial^n}{\partial y^n} f(x, y) \tag{4.4}$$

$$g^{(m,n)} = \frac{\partial^m}{\partial x^m}\frac{\partial^n}{\partial y^n} g(x, y) \tag{4.5}$$

$$h^{(m,n)} = \frac{\partial^m}{\partial x^m}\frac{\partial^n}{\partial y^n} h(x, y) \tag{4.6}$$

and the moment of $h(x, y)$ is represented as

$$h_{p,q}^{(m,n)} = h_{p,q}^{(m,n)}(x, y) = \int_{x-b}^{x-a}\int_{y-d}^{y-c} u^p v^q h^{(m,n)}(x, y, u, v)\ dudv \tag{4.7}$$

for $m, n, p, q = 0, 1, 2, \cdots$. Note that, for all SV-PSFs, by definition (follows from the conservation of light energy),

$$h_{0,0}^{(0,0)} = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} h(x, y, u, v)\ dudv = 1, \qquad \text{for all } (x, y) \tag{4.8}$$

and therefore, all derivatives of $h_{0,0}^{(0,0)}$ with respect to x and y are zero. Also, although unnecessary for a theoretical development of the method, due to its practical utility, we shall assume that h is symmetric with respect to u and v, i.e. $h(x, y, u, v) = h(x, y, |u|, |v|)$, which is the case for 2D Gaussian, cylindrical and rect functions. In this case, derivatives with respect to x and y are also symmetric, i.e

$$h^{(m,n)}(x, y, u, v) = h^{(m,n)}(x, y, |u|, |v|) \tag{4.9}$$

Using the relations $u^p v^q = -(-u)^p v^q$ when p is odd and $u^p v^q = -u^p(-v)^q$ when q is odd, we find that $h_{p,q}^{(m,n)} = 0$ when p is odd or q is odd.

Using the above notation, the truncated Taylor series expansion of $f(x-u, y-v)$ around (x, y) up to order N and $h(x-u, y-v, u, v)$ around the point (x, y, u, v) up to order M will be used below. For example, we express

$$h(x-u, y-v, u, v) \;\approx\; \sum_{m=0}^{M} a_m \sum_{j=0}^{m} C_j^m \; u^{m-j} \; v^j \; h^{(m-j,j)} \tag{4.10}$$

where c_i^n and c_j^m denote the binomial coefficients and

$$a_m = (-1)^m / m! \tag{4.11}$$

Substituting the truncated Taylor-series expansions of h and f into RT in Eq.(4.3) and simplifying, we get

$$g(x, y) \;\approx\; \sum_{n=0}^{N} a_n \; \sum_{i=0}^{n} \; C_i^n \; f^{(n-i,i)} \sum_{m=0}^{M} a_m \; \sum_{j=0}^{m} \; C_j^m \; h_{m+n-i-j,i+j}^{(m-j,j)} \tag{4.12}$$

The above equation can be rewritten as

$$g(x, y) \approx \sum_{n=0}^{N} \sum_{i=0}^{n} \; S_{n,i} \; f^{(n-i,i)} \tag{4.13}$$

where

$$S_{n,i} = a_n \; C_i^n \sum_{m=0}^{M} a_m \sum_{j=0}^{m} \; C_j^m \; h_{m+n-i-j,i+j}^{(m-j,j)} \tag{4.14}$$

Eq.(4.13) above is in a *completely localized form* in the sense that it expresses the blurred image g at (x, y) in terms of the derivatives of the focused image f and the moments of the blurring kernel h *at the same point* (x, y). This is a basic new result that facilitates local inversion of the blurring process exactly at the point (x, y).

We can now write expressions for the various partial derivatives of order (p, q) of g with respect to (x, y), as

$$g^{(p,q)} \approx \sum_{n=0}^{N} \sum_{i=0}^{n} \frac{\partial^p}{\partial x^p} \frac{\partial^q}{\partial y^q} \Big[S_{n,i} \; f^{(n-i,i)} \Big] \; T(n + P + q) \tag{4.15}$$

where

$$T(n+p+q) = \begin{cases} 1 & \text{if } \; n+p+q \leq N \\ 0 & \text{otherwise} \end{cases} \tag{4.16}$$

assures that terms with derivatives of f of order greater than N are set to zero, for $p + q = 0, 1, 2, \cdots, N$. Note that

$$\begin{aligned} S_{n,i}^{(p,q)} &= \frac{\partial^p}{\partial x^p} \frac{\partial^q}{\partial y^q} \; S_{n,i} \\ &= a_n \; C_i^n \sum_{m=0}^{M-(p+q)} a_m \; \sum_{j=0}^{m} \; C_j^m \; h_{m+n-i-j,i+j}^{(m-j+p,j+q)} \end{aligned} \tag{4.17}$$

The above equation for $g^{(p,q)}$ for $p, q = 0, 1, 2, \cdots, N$, and $0 \leq p + q \leq N$ constitute $(N+1)(N+2)/2$ equations in as many unknowns $f^{(p,q)}$. The system of equations for

$g^{(p,q)}$ can be expressed in a vector-matrix form as

$$\begin{bmatrix} g^{(0,0)} \\ g^{(1,0)} \\ \vdots \\ g^{(0,N)} \end{bmatrix} = \begin{bmatrix} r_{00} & r_{01} & \cdots & \\ k_{10} & k_{11} & \cdots & \\ \vdots & \vdots & \ddots & \vdots \\ & \cdots & & \end{bmatrix} \begin{bmatrix} f^{(0,0)} \\ f^{(1,0)} \\ \vdots \\ f^{(0,N)} \end{bmatrix} \tag{4.18}$$

or

$$\mathbf{g}_{x,y} = \mathbf{R}_{x,y}\mathbf{f}_{x,y} \tag{4.19}$$

where the subscripts (x, y) make explicit the dependence of the vectors/matrix on (x, y). $\mathbf{R}_{x,y}$ is the RT coefficient matrix of size $(N + 1)(N + 2)/2$ rows and columns. This matrix equation can be solved to obtain $f^{(p,q)}$, and in particular, $f^{(0,0)}$ by inverting the kernel matrix $\mathbf{R}_{x,y}$. The solution can be written in the form $\mathbf{f}_{x,y} = \mathbf{R}'_{x,y}\mathbf{g}_{x,y}$ where $\mathbf{R}'_{x,y} = \mathbf{R}^{-1}_{x,y}$ and the solution for $f^{(0,0)}$ can be expressed as

$$f(x, y) = f^{(0,0)} = \sum_{n=0}^{N} \sum_{i=0}^{n} S'_{n,i}\ g^{(n-i,i)} \tag{4.20}$$

The solution above needs to be computed at each pixel (x, y). As estimating accurate image derivatives is difficult in practice, a regularization approach such as the Truncated Singular Value Decomposition (TSVD) [27] or Tikhonov regularization can be used to solve Eq.(4.18)/(4.19) above to obtain a smooth and stable solution for f.

In practice we find that the truncation of the Taylor series expansion of the kernel h is useful even for small values of M, typically $M = 1$ or at most 2. However, error

introduced by the truncation of the Taylor series of an analytic approximation to the focused image f depends on two factors. The first is image noise, e.g. quantization noise in 8 bits/pixel images, which limits the use of image derivatives to an order of $N = 2$ or $N = 3$ to be the maximum. The second factor is the size of the region of significant support of the blurring kernel h, which is roughly equal to the size of the maximum blur circle. This constraint can be specified roughly by

$$h(x - u, y - v, u, v) \approx 0 \qquad \text{for } |u| > R \text{ or } |v| > R \tag{4.21}$$

where R could be the radius of the maximum blur circle size (note: the value of the SV-PSF is always positive) or 2σ for a Gaussian. The useful maximum value of R is typically limited by the value of N, because, within the region of significant support of the SV-PSF, the truncated Taylor series expansion of the focused image should be a good approximation to the actual focused image. Otherwise the product term in RT, $h(x - u, y - v, u, v)f(x - u, y - v)$, in the blurring model of Eq.(4.3) cannot be approximated well by the product of the truncated Taylor-series expansions of h and f in Eq.(4.12). In summary, in practical applications, N is limited to be around 3 due to image noise, which in turn limits the maximum blur circle size R for which the new method yields acceptable error in restoration. In our experiments, we find that for an 8 bits per pixel image, quantization and other noise limits N to around 2, and a value of $N = 2$ in turn limits the maximum blur circle diameter to be around 7 pixels. This is still a very useful and practical approach that provides good restoration results and new insights into the nature and structure of the shift-variant deblurring problem.

4.2.1 Example

We present a solution for the case of $N = 2$ and $M = 1$, $a = c = -\infty$ and $b = d = \infty$. for the case of a 2-D SV-PSF. In this case, Eq.(4.13) becomes

$$g^{(0,0)} = f^{(0,0)} \; + \; f^{(1,0)} \; h_{2,0}^{(1,0)} \; + \; f^{(0,1)} \; h_{0,2}^{(0,1)} \; + \; \frac{1}{2} f^{(2,0)} \; h_{2,0}^{(0,0)} \; + \frac{1}{2} \; f^{(0,2)} \; h_{0,2}^{(0,0)} \quad (4.22)$$

and, the system equations for its $g^{(x,y)}$ in vector-matrix form is

$$\mathbf{g}_{xy} = \mathbf{R}_{xy}\mathbf{f}_{xy} \equiv \begin{bmatrix} g^{(0,0)} \\ g^{(1,0)} \\ g^{(0,1)} \\ g^{(2,0)} \\ g^{(0,2)} \\ g^{(1,1)} \end{bmatrix} = \begin{bmatrix} 1 & h_{2,0}^{(1,0)} & h_{0,2}^{(0,1)} & \frac{1}{2}h_{2,0}^{(0,0)} & \frac{1}{2}h_{0,2}^{(0,0)} & 0 \\ 0 & 1 & 0 & \frac{3}{2}h_{2,0}^{(1,0)} & \frac{1}{2}h_{0,2}^{(1,0)} & h_{0,2}^{(0,1)} \\ 0 & 0 & 1 & \frac{1}{2}h_{2,0}^{(0,1)} & \frac{3}{2}h_{0,2}^{(0,1)} & h_{2,0}^{(0,1)} \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} f^{(0,0)} \\ f^{(1,0)} \\ f^{(0,1)} \\ f^{(2,0)} \\ f^{(0,2)} \\ f^{(1,1)} \end{bmatrix} \quad (4.23)$$

The above Eq.(4.22) gives a method of computing the output blurred image $g(x, y)$ given the input focused image $f(x, y)$. This equation could be used in Computer Graphics for rendering a realistic image generated by a camera with limited depth of field. Eq.(4.20) and Eq.(4.23) give the following *completely localized* explicit solution for $f^{(0,0)}$ at a point (x, y) in terms of the derivatives of g and moments of the derivatives

of h at the same point (x, y) as below

$$\begin{aligned} f^{(0,0)} &= g^{(0,0)} - g^{(1,0)}\, h_{2,0}^{(1,0)} - g^{(0,1)}\, h_{0,2}^{(0,1)} \\ &\quad + g^{(2,0)} \left(\frac{3}{2} (h_{2,0}^{(1,0)})^2 + \frac{1}{2} h_{0,2}^{(0,1)}\, h_{2,0}^{(0,1)} - \frac{1}{2} h_{2,0}^{(0,0)} \right) \\ &\quad + g^{(0,2)} \left(\frac{3}{2} (h_{0,2}^{(0,1)})^2 + \frac{1}{2} h_{0,2}^{(1,0)}\, h_{2,0}^{(1,0)} - \frac{1}{2} h_{0,2}^{(0,0)} \right) \end{aligned} \tag{4.24}$$

Further simplification of the above equation is possible when the kernel is rotationally symmetric (e.g. $h_{2,0}^{(1,0)} = h_{0,2}^{(0,1)}$, and $h_{2,0}^{(0,0)} = h_{0,2}^{(0,0)}$). The above equation gives an explicit, closed-form, formula for restoring an image blurred by a shift-variant symmetric point spread function. Such a closed-form solution is new and represents a basic theoretical advance. Closed-form expressions for $h_{p,q}^{(m,n)}(x, y)$ in the case of 2D Gaussian, Cylindrical, and rect SV-PSFs are given in [27].

4.2.2 Computational Complexity

The computational complexity of a conventional method (e.g. SVD) method is $O(K^6)$ for an image of size $K \times K$. In comparison, the computational complexity of the RT approach is $O(K^2 N^6)$ because the computations are dominated by the inversion of $\mathbf{R}_{x,y}$ of size $O(N^2)$ at K^2 pixels. Therefore, for $K = 32$ and $N = 4$, the computational advantage is a factor of 256. Clearly this is a significant improvement and this increases for larger K.

4.3 Experiments

We restored simulated blurred images and real blurred images. The real images were captured by two different cameras: cell-phone camera (Samsung SCH-U740) and normal commercial camera (Olympus E-1 camera). We have compared restoration quality in the sense of Root Mean Square (RMS) error of images. Image restoration based on general RT for simulated image improved RMS error by 20-40 percent and the result of real image showed clear and distinct improvement.

4.3.1 Simulation Results

First, we have simulated shift-variant blur and its restoration based on RT with E-1 camera image. The maximum blur parameter σ was varied from 0.6 to 3 pixel, and the PSF is assumed to be a Gaussian function. Blur parameter σ has maximum value near center and linearly decreases to 0. Fig.4-3 shows the variation of blur parameter. Some restoration results and plots of RMS error are presented in Fig.4-1 and Fig.4-2. Fig.4-2 shows that restoration by RT reduces RMS error up to around 37 percent for People image, around 36 percent for Alphabet image, and around 22 percent for Letter image. Blurred image with blur parameter value 2.3 produces the largest ratio between RMS of blurred image and that of restored image. As shown in Fig.4-2, RMS error of blurred letter image increases faster than other image, and its image restoration produces less improvement than other image.

Figure 4-1: Restoration by RT with shift-variant PSF. Focused position is at 500 step number. First column is blurred image and second column is restored image. First row: Blurred by $\sigma_{max} = 1.0$ pixel. Second row: Blurred by $\sigma_{max} = 2.0$ pixel. Third row: Blurred by $\sigma_{max} = 3.0$ pixel.

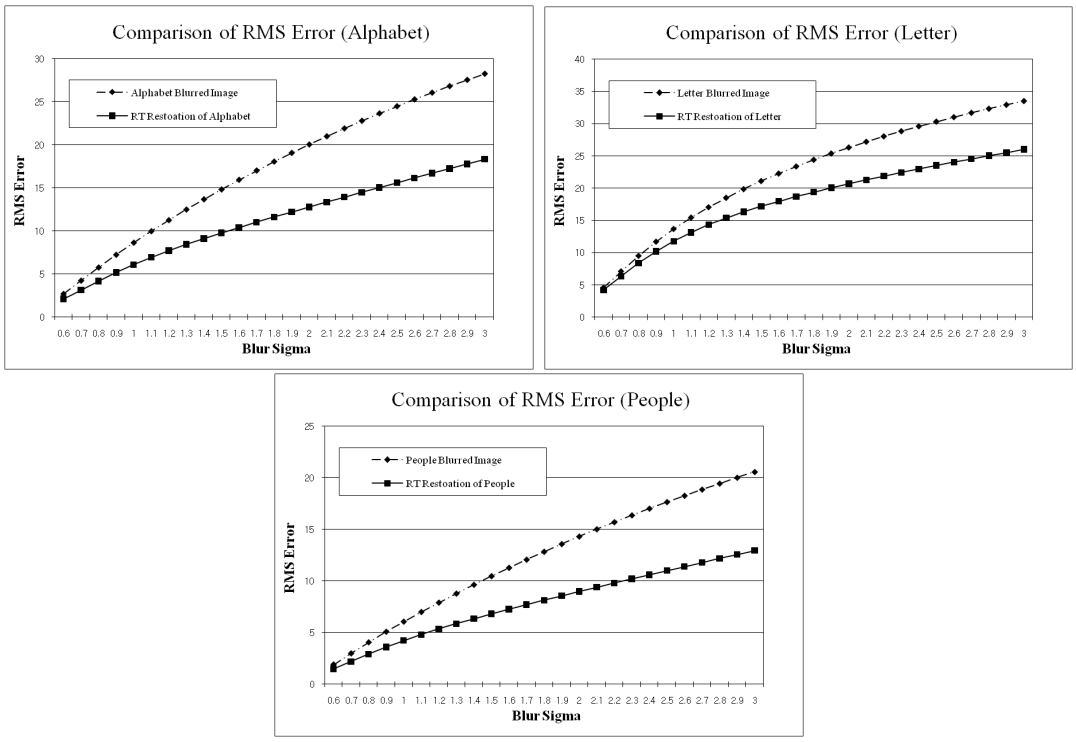


Figure 4-2: RMS Error of RT Restoration. $0.6 \leq \sigma \leq 2.0$

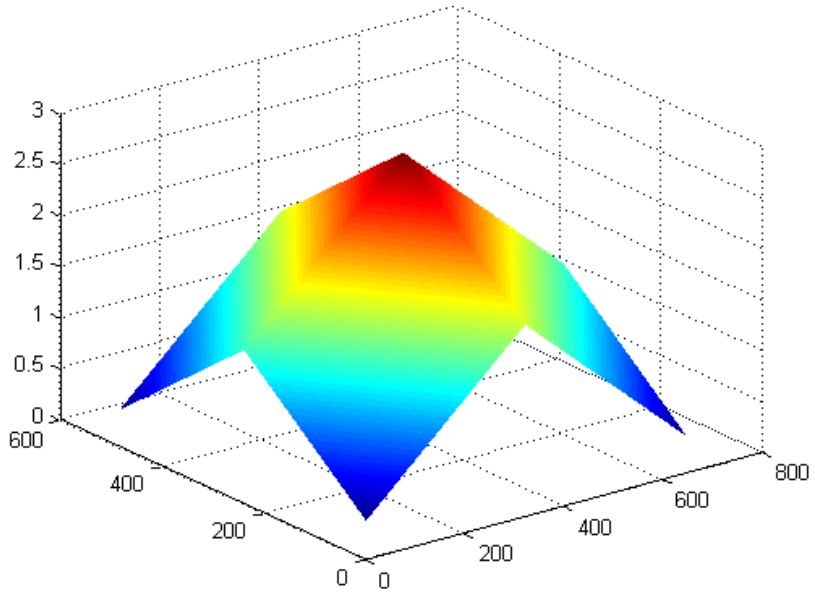


Figure 4-3: Sigma Plot with $\sigma_{max} = 3.0$ pixel

We have conducted experiments with another type of shift-variant blur where blur parameter linearly varies only along X axis like inclined planar object. Fig.4-4 shows the results of a simulation experiment using a Gaussian PSF where the blur parameter sigma was about 2.8 pixels at the center and decreased linearly to about 0.2 pixels with distance near the edges. Image size was 469×188.

Figure 4-4: Restoration of blurred image with shift-variant Gaussian PSF. (a) Top: Input focused image (source: NASA website), (b) Middle : Computed shift-variant Gaussian blurred image, (c) Bottom: Result of shift-variant image restoration by RT.

4.3.2 Results on Real Images

Fig.4-5 shows the results of experiments on real image data. A slanted planar object with printed characters was imaged by E-1 camera. The SV-PSF was estimated to be a Gaussian with sigma decreasing linearly from about 2.5 pixels at the left edge to 0.5 pixels near the center and then increasing back to 2.5 pixels near the right edge. Image size was 640×480.

Fig.4-6 shows experimental results of RT restoration for cell phone image with shift-variant blur. Blur parameter was estimated to be 2.7 near right edge of number plate and to be 1.35 near left edge of the plate. The cell phone camera was placed at around 50 cm away from the center of planar object for capturing 640×480 image. Fig.4-7 shows another experimental result. The cell phone camera was placed at 40 cm away from the center of object, and the object was rotated by 60°. The estimated blur parameter has a maximum at 350 pixel position and increases linearly to around 2.55 near the left edge and around 2.25 near the right edge.

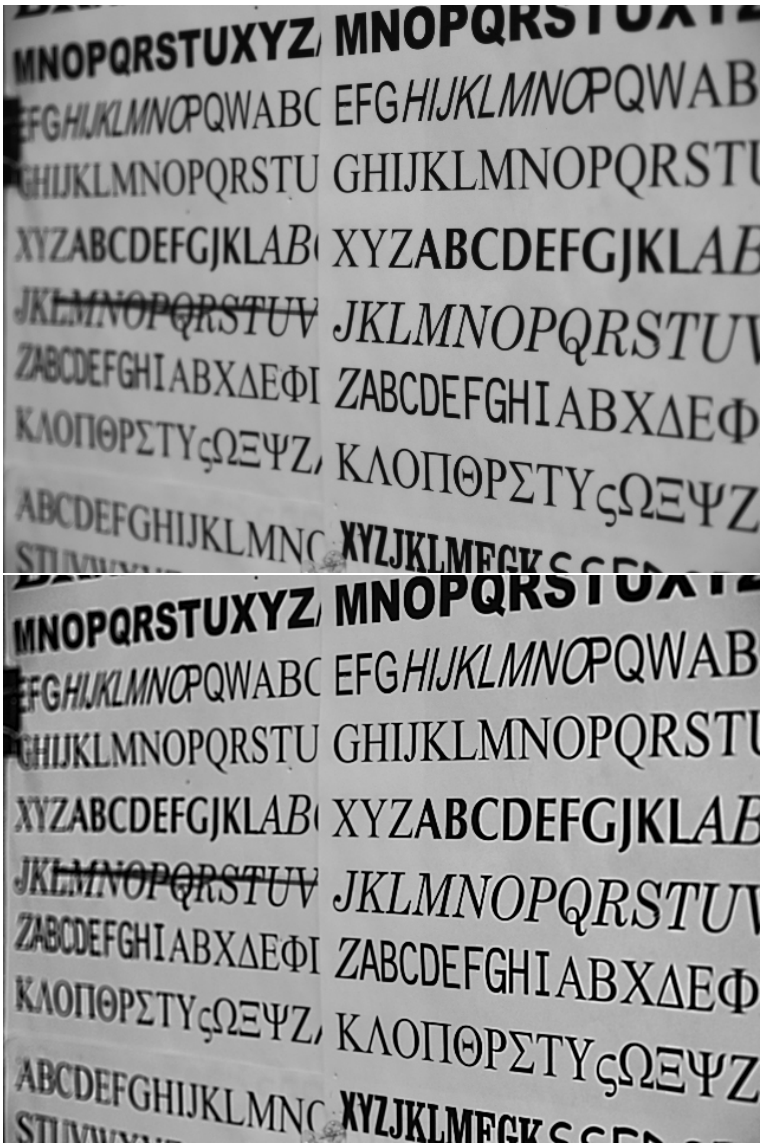


Figure 4-5: Restoration using RT on E-1 camera image. (a) Top: Real image of a horizontally slanted planar object with shift-variant blur, (b) Bottom: Result of shift-variant image restoration by RT.

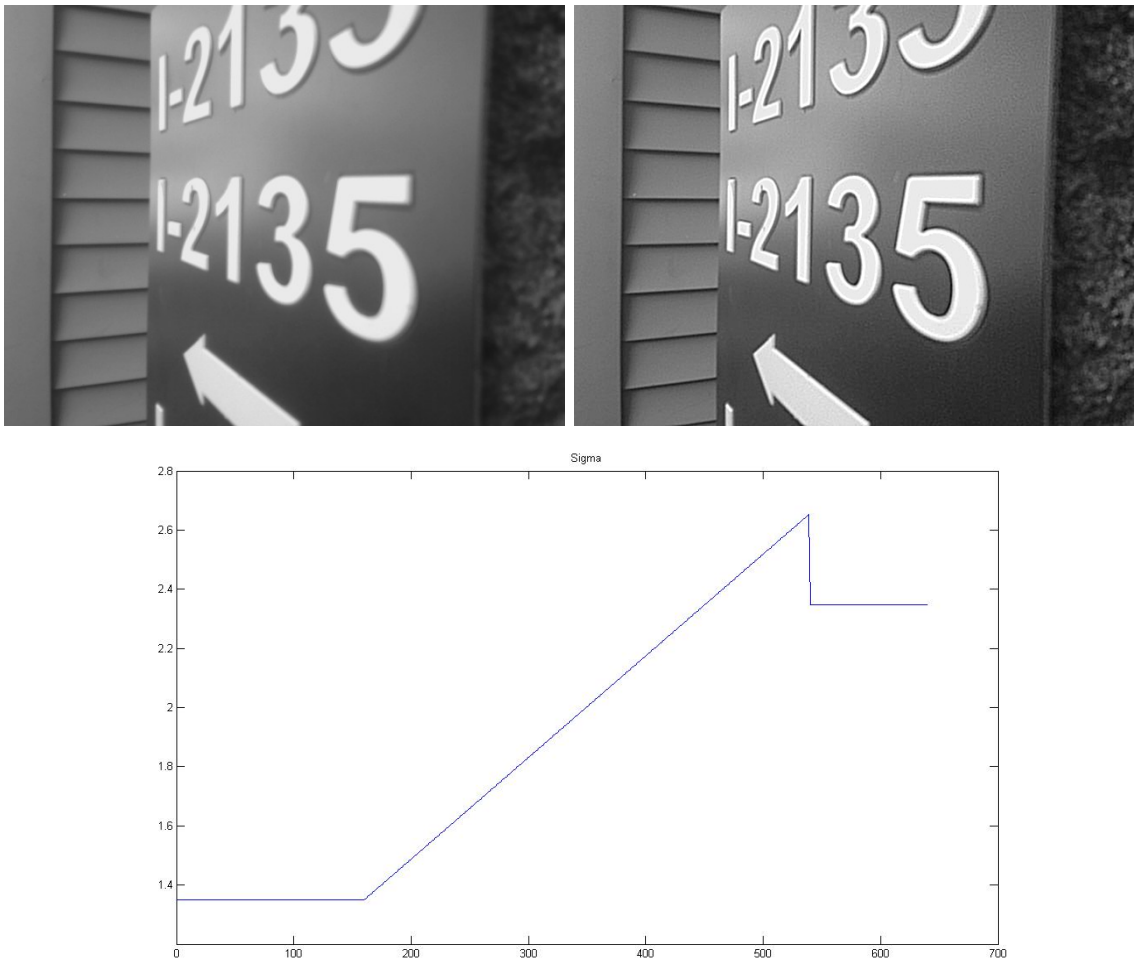


Figure 4-6: Restoration by RT with inclined Cell-phone Image. (a) Left top: Input Cell-phone Image, (b) Right top: Restored Image, (c) Bottom: Estimated σ plot.

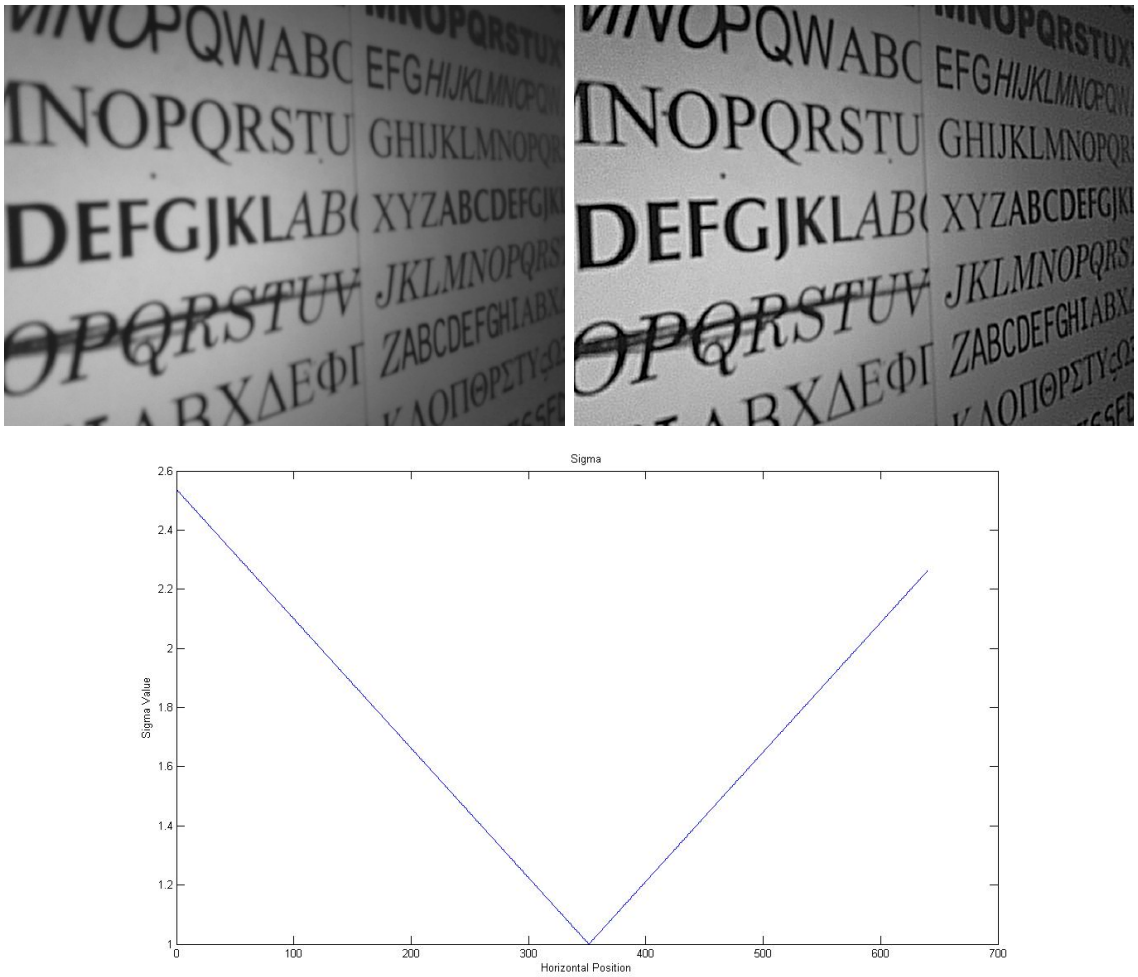


Figure 4-7: Restoration by RT with inclined Cell-phone Image. (a) Left top: Input Cell-phone Image, (b) Right top: Restored Image, (c) Bottom: Estimated σ plot.

4.3.3 Experiment on a Cylindrical Object

Shift-variant blurring arises when a camera in an endoscope captures image of a cylindrical organ. The the surface normal of the cylindrical surface makes a very steep angle with the direction of view of the camera. This dramatically increases the degree of shift-variant blurring. A simulation experiment was carried-out to generate and restore an image in this case base on RT. This experiment is also relevant to a camera mounted on a moving vehicle that images the road ahead as seen by a driver. The surface normal of the road makes a steep angle the direction of view of the driver.

We assumed a small camera passing through a cylindrical object. Image formation of this case can be illustrated by pinhole camera model as shown in Fig.4-8.

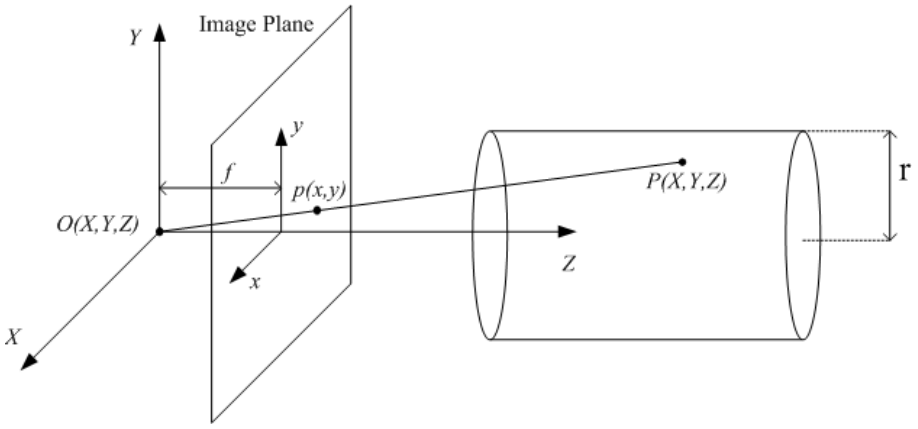


Figure 4-8: Pinhole camera model and cylindrical Object

If the radius of cross section of a cylinder is r and a camera is placed at the center of the cylinder, then, the blur circle radius is estimated by Eq.(2.4) as

$$R(x,y) = s\frac{D}{2}\Big[\frac{1}{f} - \frac{\sqrt{x^2+y^2}}{rf} - \frac{1}{s}\Big] \tag{4.25}$$

where blur circle radius R is a function of image coordination (x, y).

Similarly, the blur circle radius of planar object is obtained as

$$R(x,y) = s\frac{D}{2}\Big[\frac{1}{f} - \frac{f - bx - cy}{af} - \frac{1}{s}\Big] \tag{4.26}$$

where the planar object is defined as $Z = a + bX + cY$.

We obtained blur circle radius at every pixel by Eq.(2.4), and we simulated blurred images of cylindrical object with Gaussian SV-PSF by Eq.(2.6), then, we restored these blurred images using RT. The results show the effectiveness of RT in image restoration.

The cylinder images used in this simulation have a size of 512×512 as shown in Fig.4-9(b). Fig.4-9(a) shows the texture of image when we unfold the cylinder. Every rectangle in Fig.4-9(a) has a size of 8 pixel, and the length of cylinder is infinite. We assume that camera is placed at the center of cross section of the cylinder.

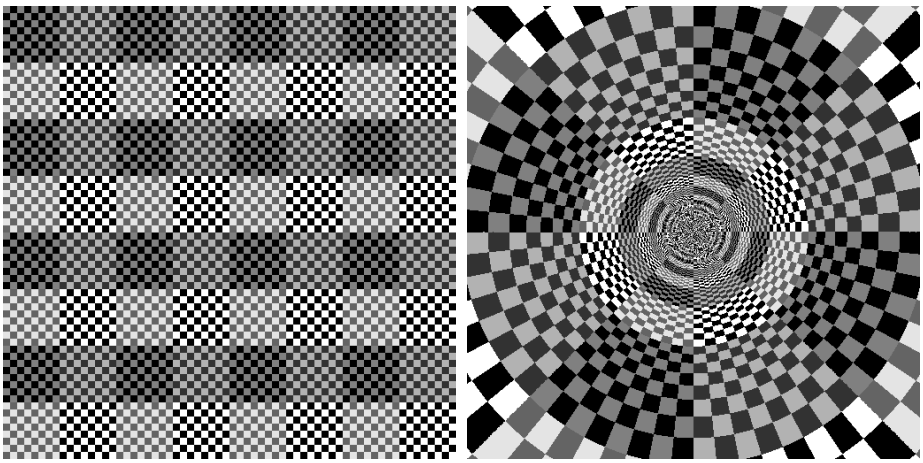


Figure 4-9: Cylinder image for pinhole camera model. (a) Left: Planar image pattern, (b) Right: Cylinder image

In this simulation the camera parameters are $f = 1.96cm$, $D = 3.0cm$, and $S_0 = 1.96cm$. CCD pixel size is 10 μm. Fig.4-10 shows object depth which is projected

onto the CCD sensor. We set a threshold of 500 cm on the depth value because the center of the CCD plane corresponds to a depth of infinity. We considered several different valuesof S which denotes the distance between lens and image sensor. The blurred images are shown in the left column in Fig.4-11, and restoration results are shown in the right column in Fig.4-11. 5x5 differential window is used for RT image restoration of these blurred images. The result of restoration shows improvement in terms of image quality and RMS error in areas away from the center.

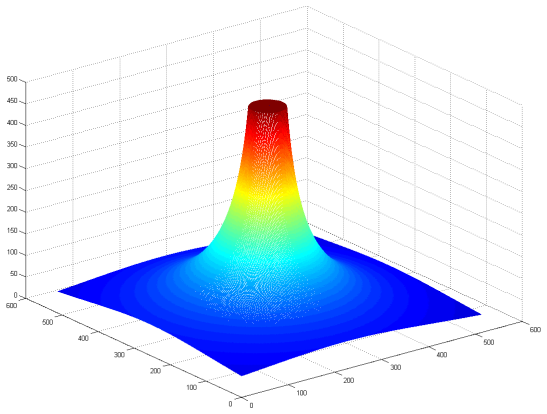


Figure 4-10: Depth of Fig. 4-9 (b)

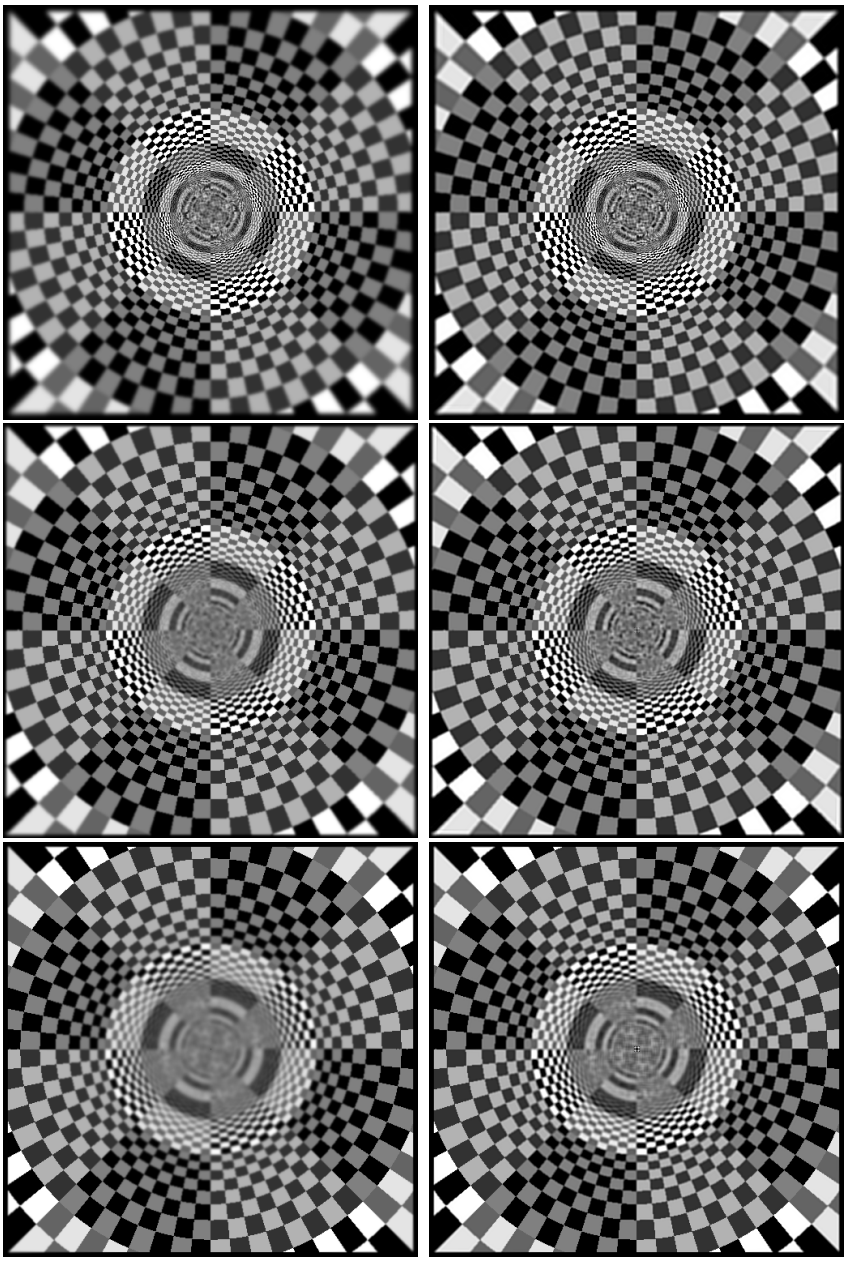


Figure 4-11: Image restoration result of Cylinder Image. (a) Left top: Blurred image with S = 1.98, (b) Right top: Restored image of (a), (c) Left middle: Blurred image with S = 2.00 (d) Right middle: Restored image of (c), (e) Left bottom: Blurred image with S = 2.02 (f) Right bottom: Restored image of (e)

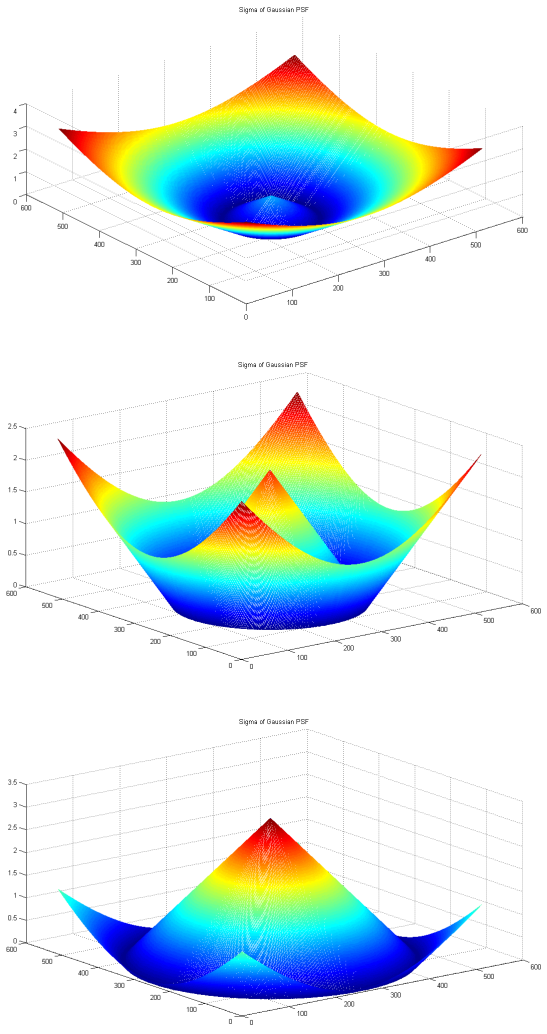


Figure 4-12: Estimated blur circle radius of Fig. 4-11 (a) Top: Radius of blur circle with S = 1.98, (b) Middle: Radius of blur circle with S = 2.00, (c) Bottom: Radius of blur circle with S = 2.02

4.3.4 Experiment on Spatial Differential windows derived by Polynomial Fitting

Image restoration based on RT involves estimating spatial derivatives of the blurred image at each pixel. The window size for estimating the derivatives should be comparable to the size of the blur circle radius at each pixel. A cubic polynomial can be fitted using a least square minimization approach in a square window to estimate the derivatives of the focused image. In this section, we study the effect of using different window sizes in estimating the image derivatives. In this experiment we have derived 5×5, 7×7, and 9×9 differential window using cubic polynomial fitting. A 3×3 window has only 9 samples but a cubic polynomial requires 10 samples, and therefore a filter cannot be derived in this case. Therefore we use the following filters shown in Eq.4.27 and 4.28.

$$\frac{d}{dx} = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} \tag{4.27}$$

$$\frac{d^2}{dx^2} = \begin{bmatrix} 1/7 & -2/7 & 1/7 \\ 5/7 & -10/7 & 5/7 \\ 1/7 & -2/7 & 1/7 \end{bmatrix} \tag{4.28}$$

Iogray image and Moon images shown in Fig.4-13 are used as sample images in this experiment. These images are blurred by Gaussian PSF with SV-blur parameter. The plot of blur parameter shapes are shown in Fig.4-3. Taylor series of RT is expanded up to order 4, and image derivatives were obtained by differentiation windows having size 3×3, 5×5, and 7×7.

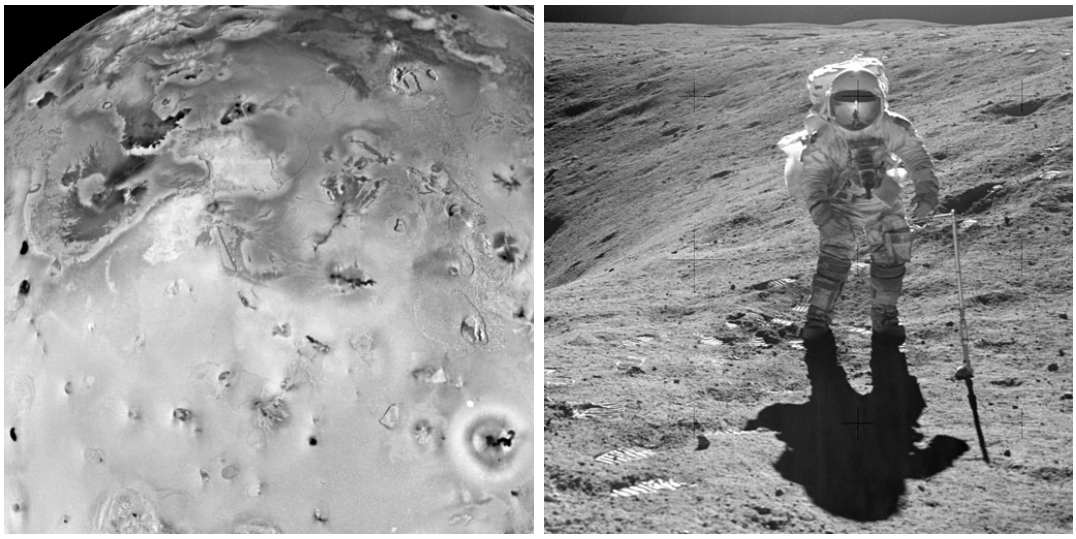


Figure 4-13: Sample Images used for Simulation of spatial window. (a) Left: Iogray image, (b) Right: Moon image

Image restoration with the 3×3 window produces better results than other windows up to a maximum blur parameter value of 2. Then it gives a more noisy result

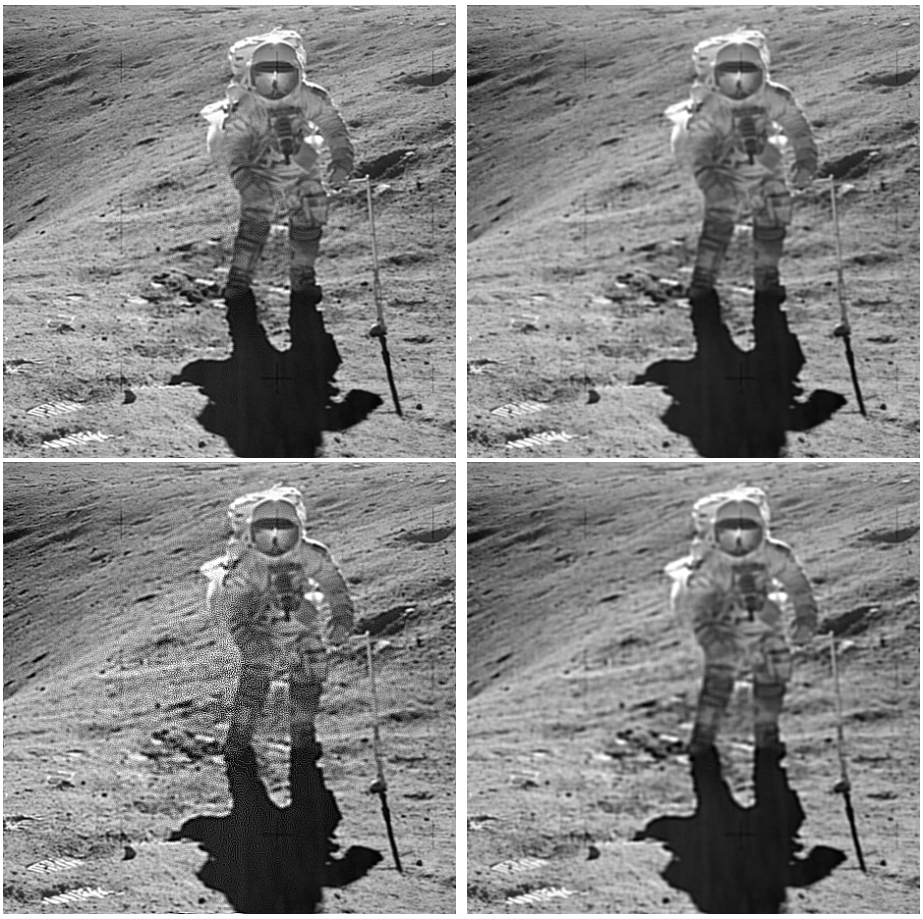


Figure 4-14: Result of RT with Moon image. (a) Left top: RT using 3×3 window RT with $\sigma_{max} = 2.0$, (b) Right top: RT using 5×5 window $\sigma_{max} = 2.0$, (c) Left bottom: RT using 3×3 window RT with $\sigma_{max} = 3.0$, (d) Right bottom: RT using 5×5 window $\sigma_{max} = 3.0$

when the maximum blur parameter is larger than 2. While Fig.4-14(a) shows less RMS error than Fig.4-14(b) by 18 percent, RMS error of Fig.4-14(c) is 70 percent larger than that of of Fig.4-14(d). This RMS error is larger than even that of blurred image. Furthermore, image restoration using 3×3 for Iogray image produced worse results. RMS error of Fig.4-15(c) is 3.3 times larger than that of Fig.4-15(d).

We also considered image restoration of a real image using these differential filters. Real images captured by an E-1 camera is estimated to have a linearly varying σ shown in Fig.6-27.

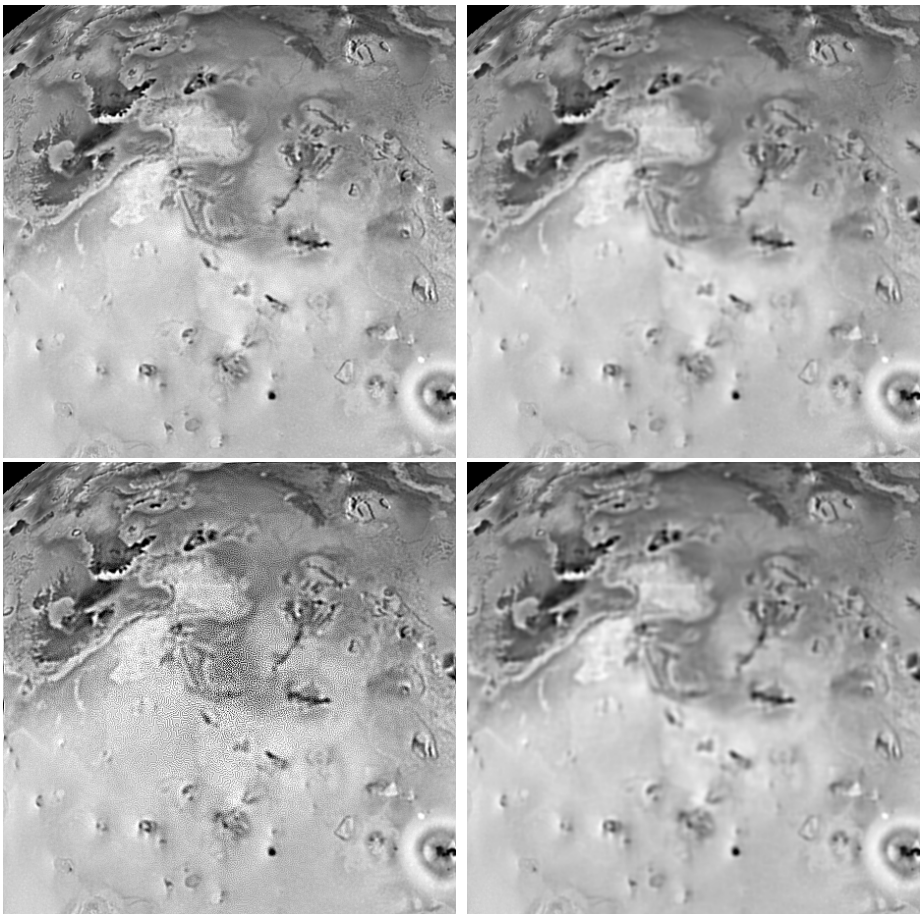


Figure 4-15: Result of RT with Iogray image. (a) Left top: RT using 3×3 window RT with $\sigma_{max} = 2.0$, (b) Right top: RT using 7×7 window $\sigma_{max} = 2.0$, (c) Left bottom: RT using 3×3 window RT with $\sigma_{max} = 3.0$, (d) Right bottom: RT using 7×7 window $\sigma_{max} = 3.0$

Experimental results are shown in Fig.4-16(a), Fig.4-16(b), and Fig.4-16(c). In general, best results are obtained when the filter window size is roughly comparable to the blur circle size.

Figure 4-16: The restoration result of inclined planar image. (a) Top: the restored Image using 3×3 spatial filter, (b) Middle: the restored image using 5×5 window (c) Bottom: the restored image using 7×7 window.

4.4 Summary

In this chapter we have investigated two dimensional RT for shift-variant deblurring. We have presented the basic theory, algorithm, computational complexity analysis, and experimental results. We verified the performance of two dimensional image restoration based on RT by restoring many shift-variant blurred images. Test images included simulated blurred images and real images. Simulation experiments included linearly varying and non-linearly varying SV-PSF. In these experiments image restoration based on RT was found to be computationally efficient and sufficiently accurate.

Chapter 5

Motion and Defocus Blur

In this chapter we consider shift-invariant or convolution motion blur and motion blur combined with defocus blur. First we derive a completely localized expression for shift-invariant motion blur along an arbitrary direction and an expression for deblurring. Next the case of shift-invariant motion blur combined with shift variant defocus blur is considered. Expressions for blurring and deblurring based on RT are presented. Results of simulation experiments are presented.

5.0.1 Shift-Invariant Motion Blur

We now derive an expression for motion deblur with uniform motion. Uniform motion blur is considered to be shift-invariant so that it can be modeled by forward S-Transform related to RT, and deblurring can be modeled by inverse S-Transform (see section 2.3 and [7, 9]).

Blurring due to uniform motion with velociy $\mathbf{v} = (v_x, v_y)$ mm/sec along a direction

θ_1 with respect to the x-axis can be modeled by

$$g(x,y) = (1/T)\int_{-T/2}^{T/2} f(x - v_x t, y - v_y t)dt \tag{5.1}$$

where T is the time period of camera exposure, and $f(x,y)$ and $g(x,y)$ respectively denote clear image without motion and motion blurred image. Note that Eq.(5.1) involves integration with respect to time t when motion occurs while x and y are spatial variables. If $l = |\mathbf{v}|t$ is the displacement in time t and $L = |\mathbf{v}T|$ is the total displacement during the exposure T, then using change of the variable of integration from t to l, and noting that $dl = |\mathbf{v}|dt$ and $L = |\mathbf{v}|T$, we obtain

$$g(x,y) = \int_{-L/2}^{L/2} f(x - l\cos\theta_1, y - l\sin\theta_1)h(l)dl \tag{5.2}$$

where $h(l) = 1/L$ models PSF of motion as shown in Fig.5-1,

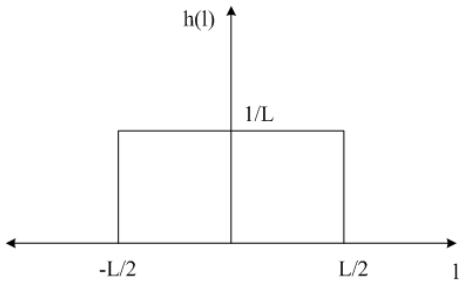


Figure 5-1: Model of Motion PSF

Expanding $f(x - \cos\theta_1 l, y - \sin\theta_1 l)$ by Taylor series and substituting $h(l)$ into

Eq.(5.2), we get

$$\begin{aligned} g(x,y) &= \int_{-L/2}^{L/2} \left(\sum_m \sum_n \frac{(-l)^{m+n}}{m!n!} (\cos\theta_1)^m (\sin\theta_1)^n f^{m,n}(x,y) \right) \frac{1}{L} dl \\ &= \sum_m \sum_n \frac{(-1)^{m+n}}{m!n!} M_{m,n} f^{m,n}(x,y) \end{aligned} \tag{5.3}$$

where

$$M_{m,n} = \frac{1}{L} \int_{-L/2}^{L/2} l^{m+n} \cos^m\theta_1 \sin^n\theta_1 dl \tag{5.4}$$

and $f^{m,n}(x,y)$ denotes $(m,n)^{th}$ derivative of $f(x,y)$. If $m+n$ is odd, then $M_{m,n}$ is always zero, and if $m+n$ is even, $M_{m,n}$ is non-zero. Thus, we get

$$M_{m,n} = \begin{cases} \frac{L^{m+n}}{2^{m+n}(m+n+1)} \cos^m(\theta_1) \sin^n(\theta_1), & \text{for } n+m \text{ even} \\ 0, & \text{otherwise} \end{cases} \tag{5.5}$$

If we expand Taylor series up to 3rd order, we get

$$\begin{aligned} M_{1,0} &= M_{0,1} = 0 \\ M_{2,0} &= \frac{L^2}{12} \cos^2(\theta_1) \\ M_{0,2} &= \frac{L^2}{12} \sin^2(\theta_1) \\ M_{1,1} &= \frac{L^2}{6} \cos(\theta_1)\sin(\theta_1) \\ M_{3,0} &= M_{2,1} = M_{1,2} = M_{3,0} = 0 \end{aligned} \tag{5.6}$$

Note that $M_{m,n}$ in Eq.(5.3) is equivalent to the moment of kernel, $h_{m,n}$, in the general forward S-Transform formula Eq.(5.7).

$$g(x,y) = \sum_m \sum_n \frac{(-1)^{m+n}}{m!n!} f^{m,n}(x,y)\ h_{m,n} \tag{5.7}$$

Therefore, by substituting Eq.(5.6) into inverse S-Transfrom up to order 3, we obtain a motion deblur formula as

$$\begin{aligned} f(x,y) &= g^{0,0}(x,y) - \frac{L^2}{24}\cos^2(\theta_1) g^{2,0}(x,y) - \frac{L^2}{24}\sin^2(\theta_1) g^{0,2}(x,y) \\ &\quad -\frac{L^2}{12}\cos(\theta_1)\sin(\theta_1) g^{1,1}(x,y) \end{aligned} \tag{5.8}$$

The expression above is novel but equivalent to more conventional formulations of this problem.

5.0.2 Experiments on Shift-invariant Motion Deblur

The images used in this simulation experiments are shown in Fig.5-2. Focused or clear images are blurred by motion filters with various motion lengths and angles of 0° degree and 45° degree from the x axis, and the motion blurred images are restored by Eq.(5.8). For implementing the inverse S-Transform we use the differentiation filters derived by cubic polynomial fitting. Here some simulation results are shown in Fig.5-3 - Fig.5-9. The restored images are somewhat improved when the motion length is less than 7 pixel. The MSE of restoration with 5 pixel-length motion blur was around half that of the original blurred image.

Figure 5-2: Clear Images used for motion deblur. Left: Checkerboard, Middle: Moon, Right: Iogray

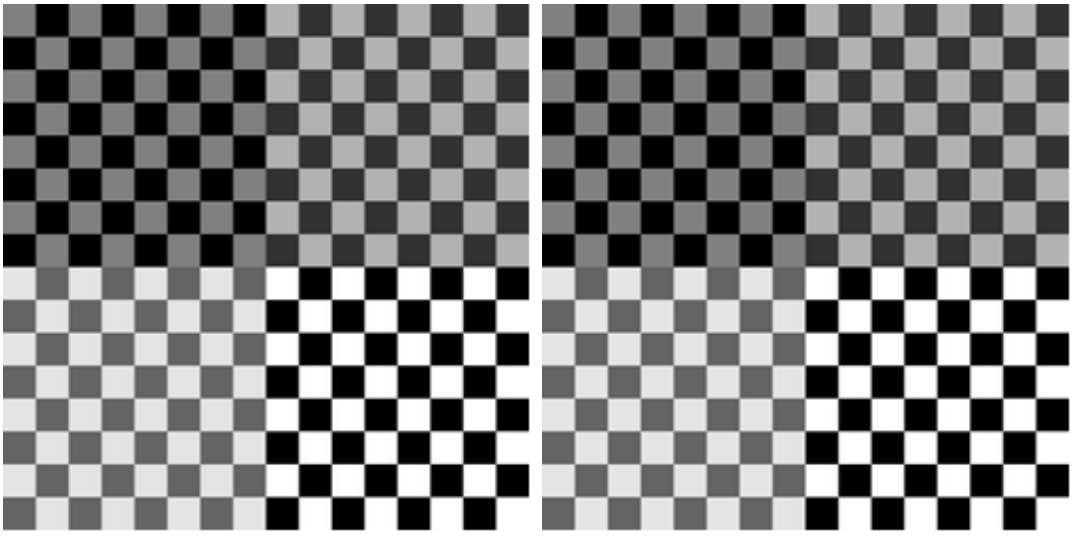


Figure 5-3: Image Restoration (Motion Length=2 and Motion Angle=0°). Left: Blurred Image, Right: Restored Image

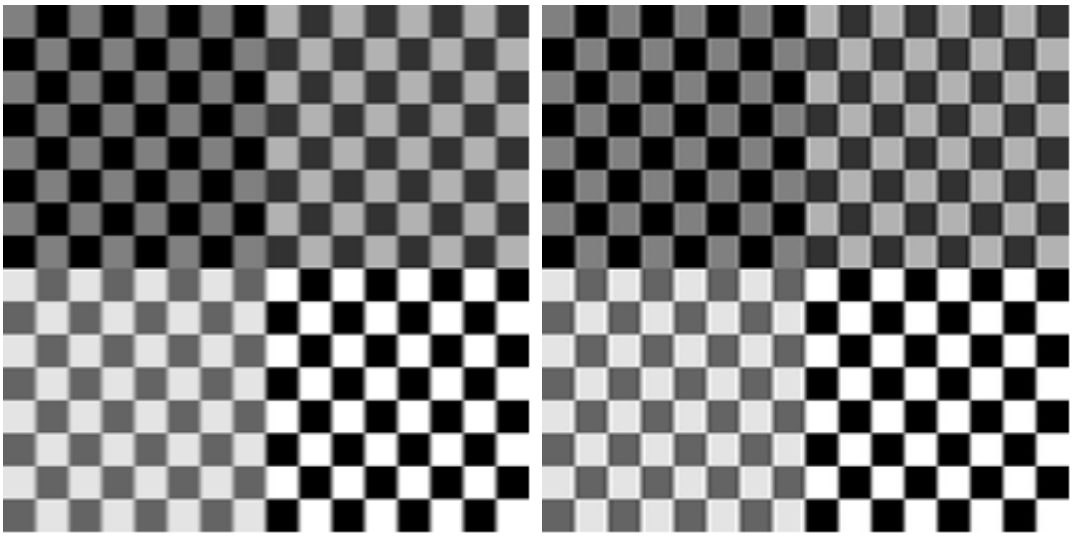


Figure 5-4: Image Restoration (Motion Length=6 and Motion Angle=0°). Left: Blurred Image, Right: Restored Image

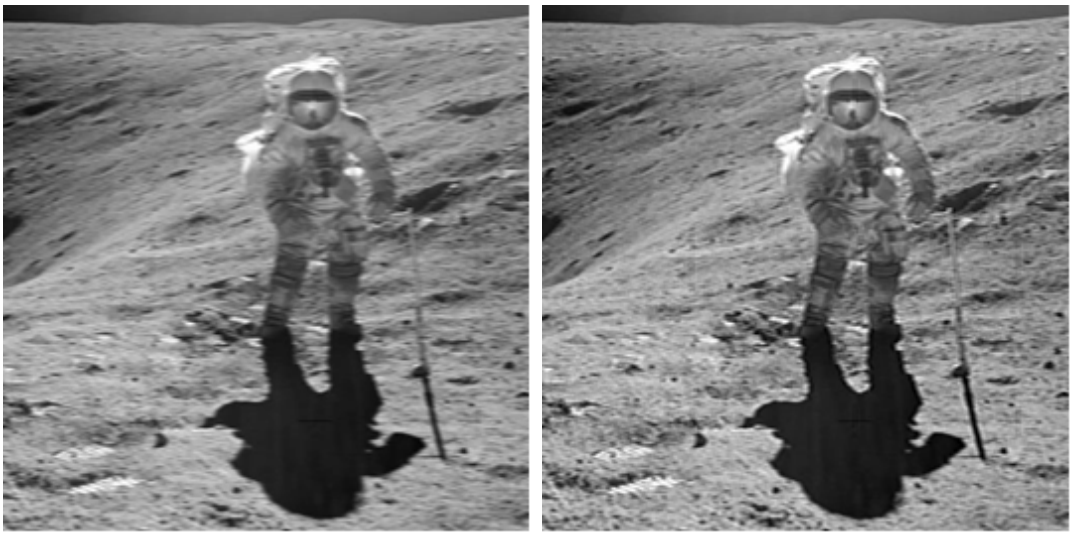


Figure 5-5: Image Restoration (Motion Length=6 and Motion Angle=0°). Left: Blurred Image, Right: Restored Image

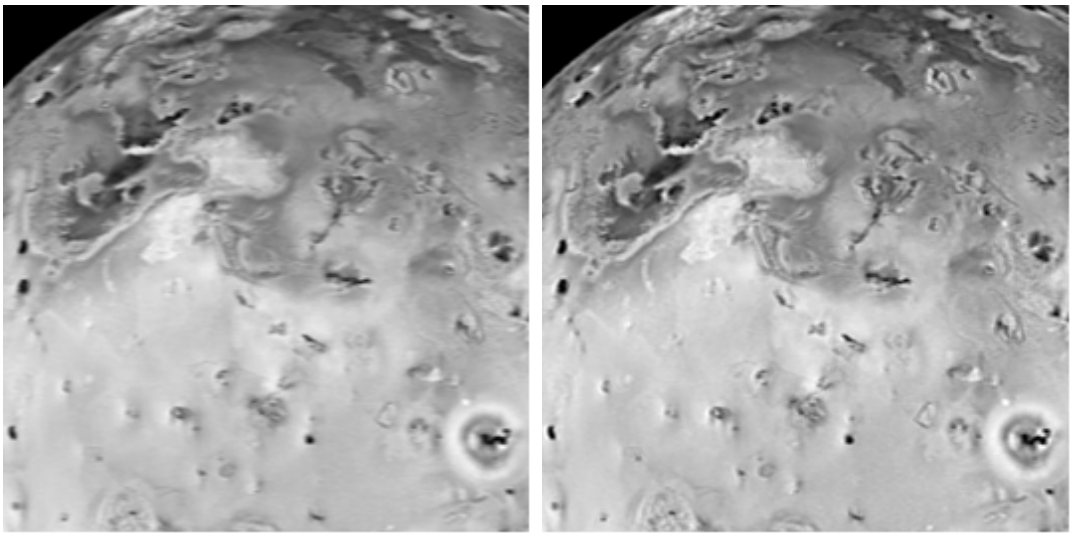


Figure 5-6: Image Restoration (Motion Length=6 and Motion Angle=0°). Left: Blurred Image, Right: Restored Image

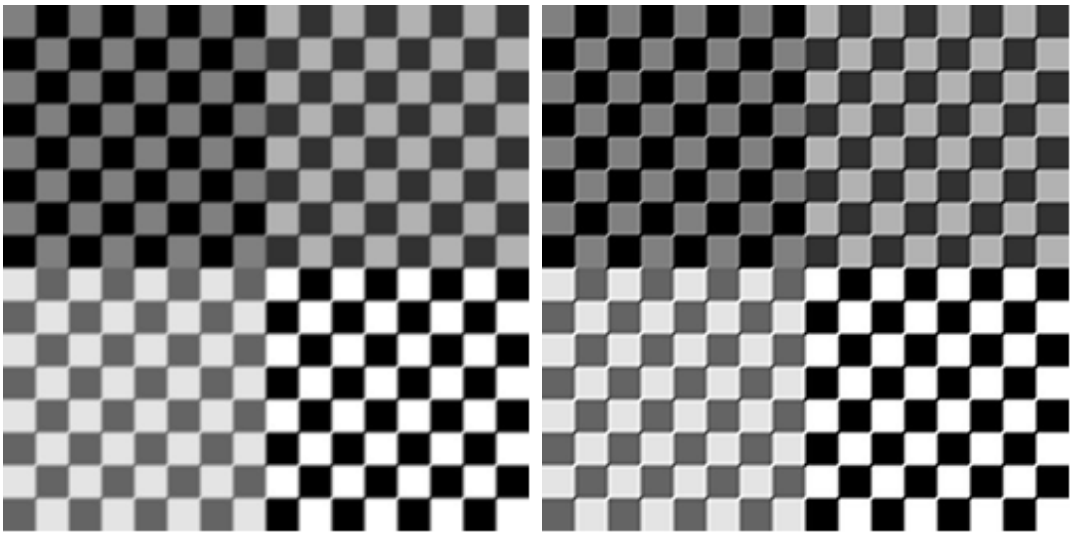


Figure 5-7: Image Restoration (Motion Length=6 and Motion Angle=45°). Left: Blurred Image, Right: Restored Image

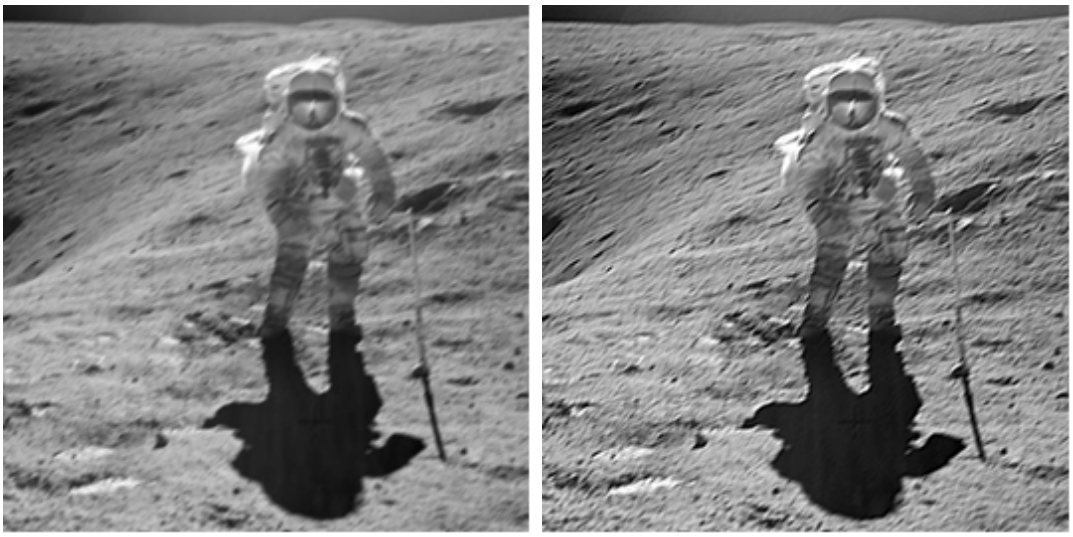


Figure 5-8: Image Restoration (Motion Length=6 and Motion Angle=45°). Left: Blurred Image, Right: Restored Image

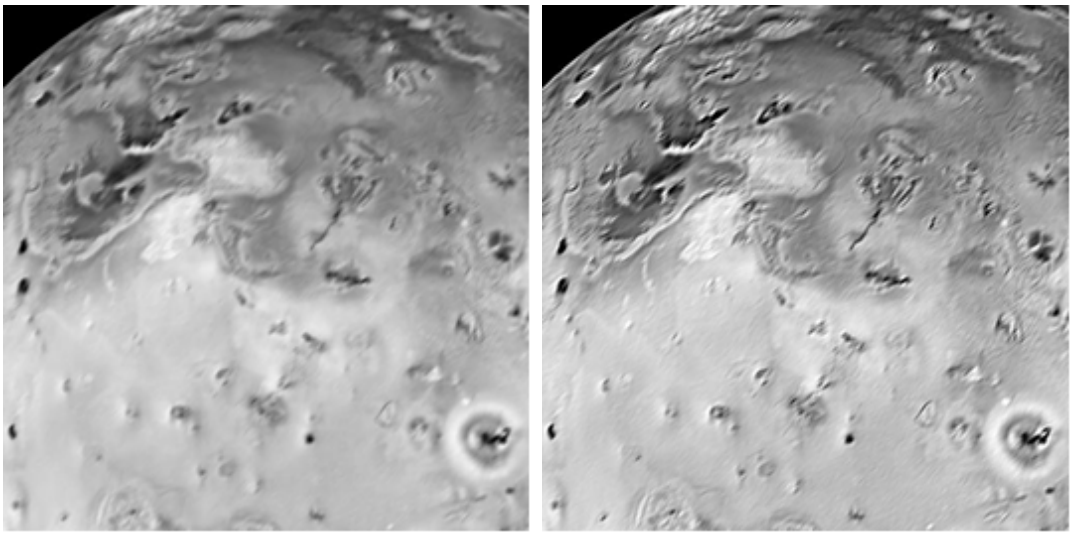


Figure 5-9: Image Restoration (Motion Length=6 and Motion Angle=45°). Left: Blurred Image, Right: Restored Image

5.0.3 Multiple Blur Model

In this section we present a theory for deblurring images with multiple blur containing both motion blur as well as defocus blur. Motion blur is restricted to be shift-invariant and defocus blur is permitted to be shift-variant. This type of problem arises when a camera moves perpendicular to its direction of view. This theory can be easily extended to more general case where the motion is also shift-variant, but its practical utility may be limited due to the complexity of the problem.

Suppose that a camera senses a shift-variant defocused image and the target image moves with uniform speed resulting in motion blur. This case of multiple blur can be considered as a sequential blurring process where the shift-variant defocus blur follows shift-invariant motion blur. Therefore, this can be modeled as a focused image first blurred by a defocus blur, and the resulting image is then blurred by motion blur. Image restoration can be implemented by first deblurring the motion blur and then deblurring the defocus blur. It is also possible to carry out deblurring in a single step using a single inverse filter.

Suppose an object moves by length L when the camera shutter is open, and its motion is at an angle θ_1. As shown in section 5.0.1, this motion blurred image can be expressed in terms of RT. Let $g(x, y)$ be blurred image and $f(x, y)$ be the clear image. Then, $g(x, y)$ can be written as

$$g(x,y) = \sum_m \sum_n \frac{(-1)^{m+n}}{m!n!} M_{m,n} f^{m,n}(x,y) \tag{5.9}$$

where

$$M_{m,n} = \frac{1}{L} \int_{-L/2}^{L/2} l^{m+n} \cos^m \theta_1 \sin^n \theta_1 dl \tag{5.10}$$

Let $u(x, y)$ be the multi-blurred image obtained by blurring $g(x, y)$ with shift-variant defocus PSF $h(x, y)$. Then, RT in this case is

$$u(x, y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} h(x - \alpha, y - \beta, \alpha, \beta) g(x - \alpha, y - \beta) d\alpha d\beta \tag{5.11}$$

By Taylor series expansion $g(x - \alpha, y - \beta)$ can be expanded as

$$g(x - \alpha, y - \beta) = \sum_i \sum_j \frac{1}{i!j!} (-\alpha)^i (-\beta)^j g^{(i,j)} \tag{5.12}$$

where

$$g^{(i,j)} = \frac{d^i}{dx^i} \frac{d^j}{dy^j} g(x, y)$$

Substituting Eq.(5.9) into Eq.(5.12), we can obtain

$$g(x - \alpha, y - \beta) = \sum_i \sum_j \frac{1}{i!j!} (-\alpha)^i (-\beta)^j \sum_m \sum_n \frac{(-1)^{m+n}}{m!n!} M_{m,n} f^{(m+i,n+j)} \tag{5.13}$$

where

$$g^{(i,j)} = \sum_m \sum_n \frac{(-1)^{m+n}}{m!n!} M_{m,n} f^{(m+i,n+j)} \tag{5.14}$$

In deriving the above equation $M_{m,n}$ are assumed to be constant with respect to (x, y) as the motion blur is shift-invariant. This implies that motion is uniform and L and θ_1 are constant with respect to (x, y). If the motion blur is shift-variant, then the theory here can be extended to that case by taking the derivatives of the product $M_{m,n} f^{(m,n)}$ above by considering both terms of the products to be dependent on (x, y).

Approximating the image locally by a cubic polynomial, Taylor series expansion can be truncated at order 3. Therefore, motion blurred image derivative $g^{(i,j)}$ can be expressed as

$$\begin{aligned} g^{(i,j)} &= f^{(i,j)} + \frac{L^2}{12}\cos^2\theta_1 f^{(i+2,j)} + \frac{L^2}{12}\sin^2\theta_1 f^{(i,j+2)} \\ &\quad + \frac{L^2}{6}\sin\theta_1\cos\theta_1 f^{(i+1,j+1)} \end{aligned} \tag{5.15}$$

If the shift-variant defocus PSF is a 2D Gaussian function and is expanded up to order 1 by Taylor series, then the PSF is rotationally symmetric and its odd moments are zero:

$$h_{k,l}^{(i,j)} = h_{k,l}^{(i,j)} = 0 \qquad \text{if } k \text{ is odd or } l \text{ is odd} \tag{5.16}$$

Now we can simplify Eq.(5.11) as

$$u(x,y) = g^{(0,0)} + h_{2,0}^{(1,0)} g^{(1,0)} + h_{0,2}^{(0,1)} g^{(0,1)} + \frac{1}{2} h_{2,0}^{(0,0)} g^{(2,0)} + \frac{1}{2} h_{0,2}^{(0,0)} g^{(0,2)} \tag{5.17}$$

where

$$
\begin{aligned}
g^{(0,0)} &= f^{(0,0)} + \frac{L^2}{12}\cos^2\theta_1 f^{(2,0)} + \frac{L^2}{12}\sin^2\theta_1 f^{(0,2)} + \frac{L^2}{6}\cos\theta_1\sin\theta_1 f^{(1,1)}, \\
g^{(1,0)} &= f^{(1,0)} + \frac{L^2}{12}\cos^2\theta_1 f^{(3,0)} + \frac{L^2}{12}\sin^2\theta_1 f^{(1,2)} + \frac{L^2}{6}\cos\theta_1\sin\theta_1 f^{(2,1)}, \\
g^{(0,1)} &= f^{(0,1)} + \frac{L^2}{12}\cos^2\theta_1 f^{(2,1)} + \frac{L^2}{12}\sin^2\theta_1 f^{(0,3)} + \frac{L^2}{6}\cos\theta_1\sin\theta_1 f^{(1,2)}, \\
g^{(2,0)} &= f^{(2,0)}, \quad g^{(0,2)} = f^{(0,2)}, \quad \text{and} \quad g^{(1,1)} = f^{(1,1)}
\end{aligned} \tag{5.18}
$$

Eq.(5.18) is obtained by setting image derivatives of order higher than 3 to be zero.

Substituting Eq.(5.18) into Eq.(5.17) we can obtain an expression for $u^{(0,0)}$ in terms of a linear combination of coefficients and image derivatives of $f(x, y)$. Then, we can formulate a linear equation with one coefficient matrix and two column vectors of image derivatives as shown in Eq.(4.18). We get a solution for $f(x, y)$ by inverting this coefficient matrix. Once we get a closed-form solution for $f(x, y)$ by inverting the coefficient matrix, we can obtain a clear image $f(x, y)$ by just substituting moments of SV-PSF into the closed-form solution. This operation offers significant speed-up in the restoration process.

5.0.4 Experiments on Multiple Blur Model

Many experiments were conducted to restore multiple blur images having shift-variant defocus and shift-invariant motion blur. Some of the results are presented in this section. In this experiment, 3 different images– checker, Iogray, and Moon were used. Every image has a size of 469×469, and the PSF is assumed to be an SV-Gaussian

function. Parameter σ of Gaussian PSF is taken to vary linearly from the borders of the image to their center as shown in Fig.4-3. Maximum sigma σ_{max} is set to be 1,2, and 3. Object is assumed to have motion with a displacement of 3,5, and 7 pixels. Motion angle with horizontal axis is 0°, 45°, and 90°. Fig.5-10 and Fig.5-11 show the result of RT image restoration with defocus and motion blur. These restoration results show improved image quality in the sense of RMS error and clarity. In the case of $\sigma_{max} = 2$, approximately 15 percent of RMS error is improved. Fig.5-10(b) and Fig.5-10(d) show an improvement of around 16 percent in RMS error. Restorations for $\sigma_{max} = 3$ gives similar reduction in RMS error. Fig.5-11(b) and Fig.5-11(f) have improvement by around 13 percent of RMS error. However, image with motion length 7 shows unstable results since the motion length is larger than the differentiation window size. Nevertheless, larger window is not an appropriate solution to improve restoration since larger window also averages neighboring pixels.

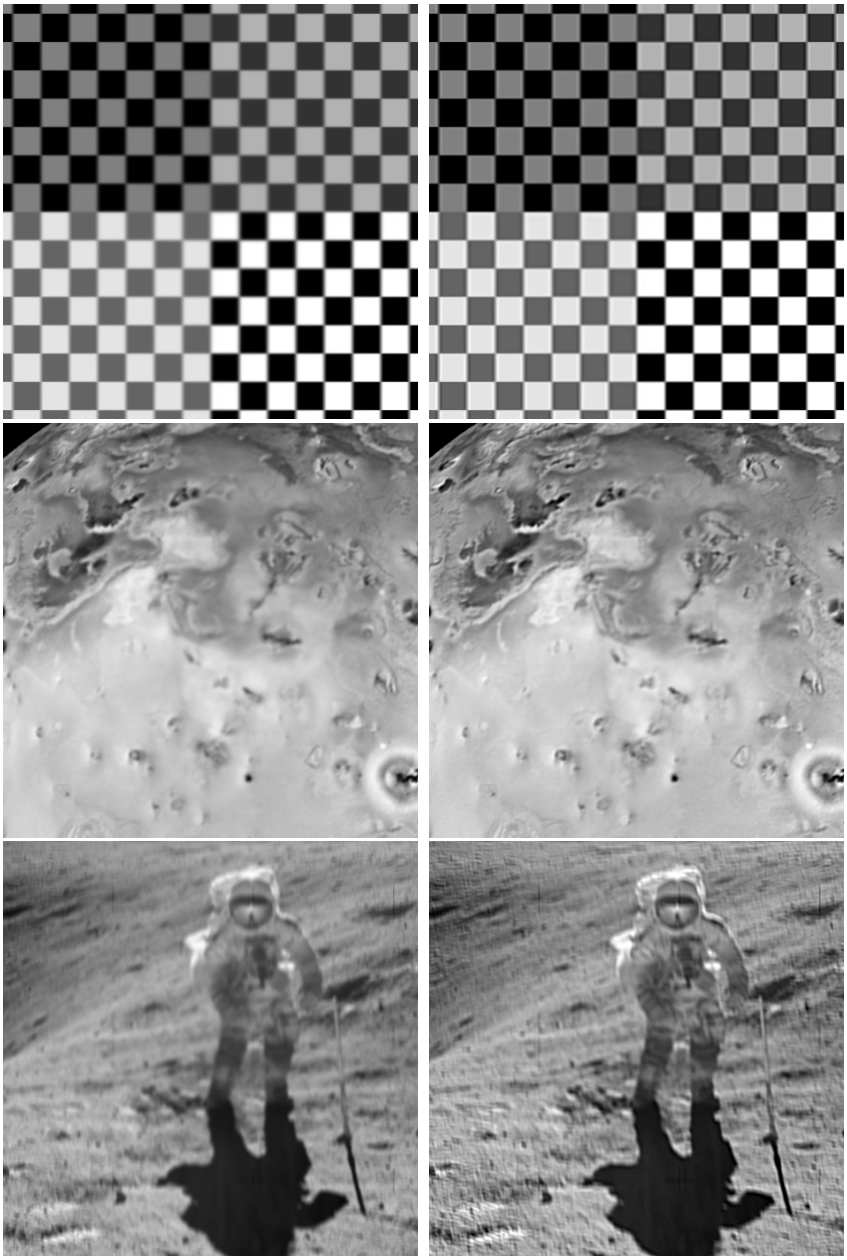


Figure 5-10: Image restoration result of multiple blur. (a) Left top: Blurred Checker image with $\sigma_{max} = 2$, motion length = 5, and angle = 0° (b) Right top: Restored image of (a), (c) Left middle: Blurred Iogray image with $\sigma_{max} = 2$, motion length = 3, and angle = 45° (d) Right middle: Restored image of (c), (e) Left bottom: Blurred Moon image with $\sigma_{max} = 2$, motion length = 7, and angle = 90° (f) Right bottom: Restored image of (e)

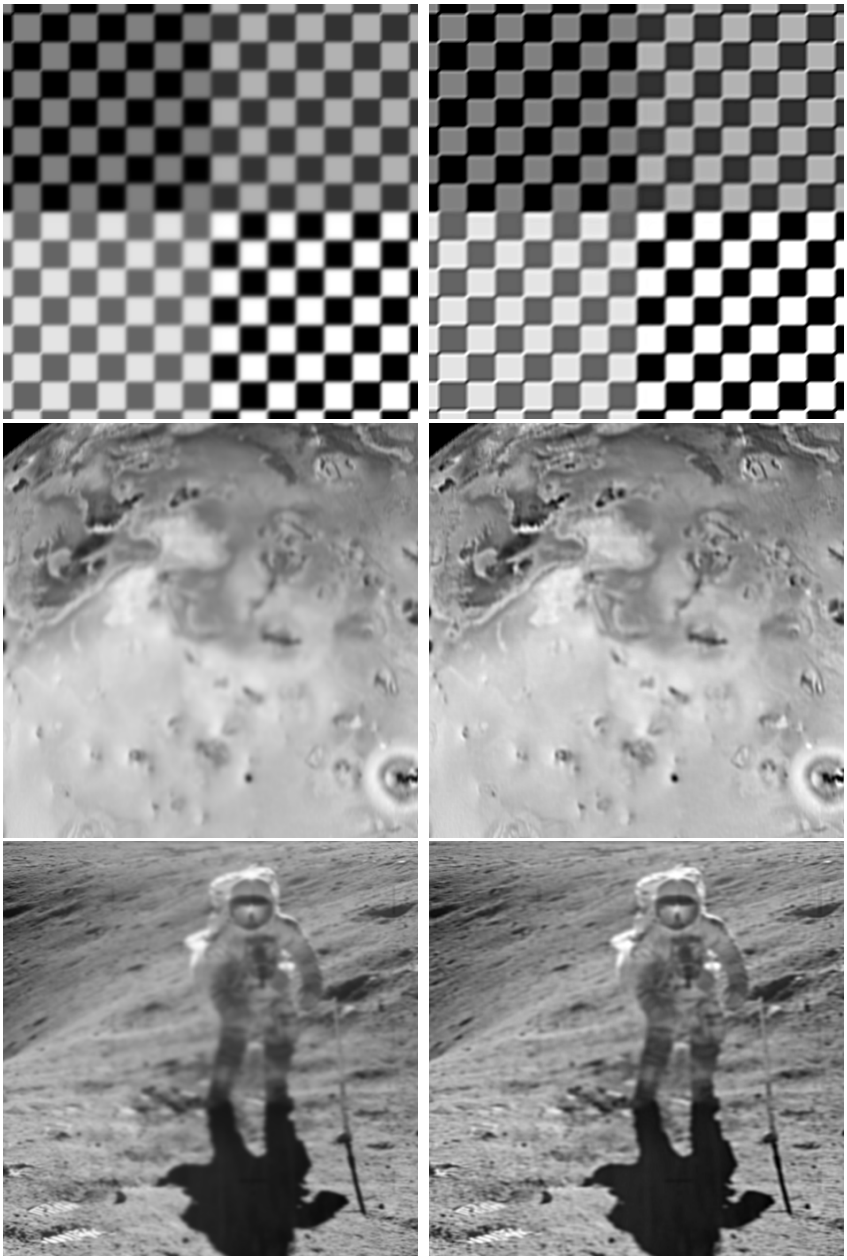


Figure 5-11: Image restoration result of multiple blur. (a) Left top: Blurred Checker image with $\sigma_{max} = 3$, motion length = 7, and angle = 45° (b) Right top: Restored image of (a), (c) Left middle: Blurred Iogray image with $\sigma_{max} = 3$, motion length = 5, and angle = 90° (d) Right middle: Restored image of (c), (e) Left bottom: Blurred Moon image with $\sigma_{max} = 3$, motion length = 3, and angle = 0° (f) Right bottom: Restored image of (e)

Chapter 6

Comparison of Matrix Approaches and RT for Image Deblurring

6.1 Introduction

In this chapter, some standard matrix approaches to shift-variant deblurring are summarized and their performance is compared with RT based method. The matrix approaches include Singular Value Decomposition (SVD) and Conjugate Gradient Method (CGM). SVD and RT methods are considered both with and without regularization. Truncated SVD and SVD/RT with Tikhonov regularization are considered.

6.2 Matrix approach to shift-variant imaging system

6.2.1 Matrix formation

Sectioning method models the blur process to be shift-invariant or convolution. The general 1-D convolution can be written as

$$b(s) = \int_{-\infty}^{\infty} p(s-t)x(t)dt \tag{6.1}$$

Let x and p in Eq.(6.1) be discrete data and consist of five elements. Then Eq.(6.1) can be rewritten as a matrix-vector multiplication:

$$\begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \\ b_5 \end{bmatrix} = \begin{bmatrix} p_5 & p_4 & p_3 & p_2 & p_1 & & & & \\ & p_5 & p_4 & p_3 & p_2 & p_1 & & & \\ & & p_5 & p_4 & p_3 & p_2 & p_1 & & \\ & & & p_5 & p_4 & p_3 & p_2 & p_1 & \\ & & & & p_5 & p_4 & p_3 & p_2 & p_1 \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \\ x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ y_1 \\ y_2 \end{bmatrix} \tag{6.2}$$

where w and y represent pixels in the original scene that are actually outside the field

of view. w and y can be removed if the boundary condition of input is given. There are three boundary condition in imaging system, such as zero boundary condition, periodic boundary condition, and reflective boundary conition.

Zero Boundary Conditions:

Imaging system has zero boundary condition when $w = y = 0$. Thus, Eq.(6.2) can be written as

$$\begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \\ b_5 \end{bmatrix} = \begin{bmatrix} p_3 & p_2 & p_1 & & \\ p_4 & p_3 & p_2 & p_1 & \\ p_5 & p_4 & p_3 & p_2 & p_1 \\ & p_5 & p_4 & p_3 & p_2 \\ & & p_5 & p_4 & p_3 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{bmatrix} \tag{6.3}$$

Periodic Boundary Conditions:

Periodic boundary condition is specified by setting $w_1 = x_4$, $w_2 = x_5$, $y_1 = x_1$, and $y_2 = x_2$, and Eq.(6.2) becomes

$$\begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \\ b_5 \end{bmatrix} = \begin{bmatrix} p_3 & p_2 & p_1 & p_5 & p_4 \\ p_4 & p_3 & p_2 & p_1 & p_5 \\ p_5 & p_4 & p_3 & p_2 & p_1 \\ p_1 & p_5 & p_4 & p_3 & p_2 \\ p_2 & p_1 & p_5 & p_4 & p_3 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{bmatrix} \tag{6.4}$$

Reflexive Boundary Condition:

Reflextive boundary condition happens when $w_1 = x_1$, $w_2 = x_1$, $y_1 = x_5$, and $y_2 = x_4$, and Eq.(6.2) becomes

$$\begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \\ b_5 \end{bmatrix} = \begin{bmatrix} p_3+p_4 & p_2+p_5 & p_1 & & \\ p_4+p_5 & p_3 & p_2 & p_1 & \\ p_5 & p_4 & p_3 & p_2 & p_1 \\ & p_5 & p_4 & p_3 & p_2+p_1 \\ & & p_5 & p_4+p_1 & p_3+p_2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{bmatrix} \tag{6.5}$$

These three matrix formulations for the 1-dimensional convolution, Eq.(6.3)-(6.5), can be extended to the 2-dimensional case, but an $M \times N$ 2-D image needs to be rearranged as a column vector of size $MN \times 1$ [21]. The kernel matrix of a 2-dimensional PSF $\mathbf{P}$ is a complicated $MN \times MN$ matrix. However, since we assume the PSF to be shift-invariant in the sectioning method, $MN \times MN$ matrix $\mathbf{P}$ has the form of a Toeplitz matrix and it can be expressed as a Kronecker product of two $M \times N$ matrices [22].

6.3 Kernel Matrix Inversion

6.3.1 Singular Value Decomposition

Singular value decomposition (SVD) is an important factorization of a rectangular real or complex matrix, with many applications in signal processing and statistics [83]. The purpose of SVD is to compute the inverse matrix. Let $\mathbf{b}$ denote the blurred image,

$\mathbf{x}$ denote the focused input image, and $\mathbf{A}$ is the kernel matrix that models the blur process. The matrix form of the blurring model is

$$\mathbf{b} = \mathbf{A}\mathbf{x} \tag{6.6}$$

To invert $\mathbf{A}$ we decompose it using Singular Value Decomposition (SVD):

$$\mathbf{A} = \mathbf{U}\Sigma\mathbf{V}^T \tag{6.7}$$

where $\mathbf{U}$ and $\mathbf{V}$ are orthogonal matrices satisfying $\mathbf{U}^T\mathbf{U} = \mathbf{I}_N$ and $\mathbf{V}^T\mathbf{V} = \mathbf{I}_N$ and Σ is a diagonal matrix with singular values

$$\Sigma = \begin{bmatrix} \sigma_1 & & & \\ & \sigma_2 & & \\ & & \ddots & \\ & & & \sigma_N \end{bmatrix}$$

where $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_N \geq 0$. By substituting Eq.(6.7) into Eq.(6.6) we get the focused image as

$$\mathbf{b} = \mathbf{U}\Sigma\mathbf{V}^T\mathbf{x}$$

$$\mathbf{x} = \mathbf{V}\Sigma\mathbf{U}^T\mathbf{b} \tag{6.8}$$

We can rewrite Eq.(6.8) as

$$\mathbf{x} = \sum_{i=1}^{N} \frac{u_i^T \, b}{\sigma_i} \, v_i \tag{6.9}$$

SVD allows us to invert arbitrary non-singular matrix but it is difficult to apply it in image processing due to the high cost of computation. One alternatives is to decompose the kernel matrix into Kronecker product of two smaller matrices [21–24, 69]. The Kronecker product, denoted by $\otimes$, is an operation on two matrices of arbitrary size resulting in a block matrix. It gives the matrix of the tensor product, and its inversion is also a Kronecker product of two inverse matrices. Let matrix $\mathbf{A}$ and $\mathbf{B}$ be $M \times N$ and $P \times Q$ matrices, then $\mathbf{A} \otimes \mathbf{B}$ is an $MP \times NQ$ block matrix and

$$(\mathbf{A} \otimes \mathbf{B})^{-1} = \mathbf{A}^{-1} \otimes \mathbf{B}^{-1} \tag{6.10}$$

The SVD of a Kronecker product can be expressed in terms of the SVDs of the matrices that form it; if

$$\mathbf{A} = \mathbf{A}_r \otimes \mathbf{A}_c \tag{6.11}$$

and

$$\mathbf{A}_r = \mathbf{U}_r \mathbf{\Sigma}_r \mathbf{V}_r^T \quad \text{and} \quad \mathbf{A}_c = \mathbf{U}_c \mathbf{\Sigma}_c \mathbf{V}_c^T \tag{6.12}$$

then

$$\begin{aligned} \mathbf{A} &= \mathbf{A}_r \otimes \mathbf{A}_c = (\mathbf{U}_r \mathbf{\Sigma}_r \mathbf{V}_r) \otimes (\mathbf{U}_c \mathbf{\Sigma}_c \mathbf{V}_c) \\ &= (\mathbf{U}_r \otimes \mathbf{U}_c)(\mathbf{\Sigma}_r \otimes \mathbf{\Sigma}_c)(\mathbf{V}_r \otimes \mathbf{V}_c)^T \end{aligned} \tag{6.13}$$

6.3.2 Conjugate Gradient Method

The conjugate gradient method is an iterative algorithm for the numerical solution of particular systems of linear equations that are *symmetric* and *positive definite*. If the shift-invariant point spread function (SV-PSF) is rotationally symmetric like a Gaussian function or a cylindrical function, the blurring kernel matrix will be symmetric and conjugate gradient method can be applied to kernel inversion.

We say that two non-zero vectors u and v are *conjugate* with respect to $\mathbf{A}$ if

$$\mathbf{u}^T \mathbf{A} \mathbf{v} = 0 \tag{6.14}$$

Suppose that $\mathbf{p}_k$ is a sequence of n mutually conjugate directions. Then the $\mathbf{p}_k$ form a basis, so we can expand the solution $\mathbf{x}_*$ of Eq.(6.6) in this basis.

$$\begin{aligned} \mathbf{x}_* &= \sum_{i=1}^{n} \alpha_i \mathbf{p}_i \\ \mathbf{p}_k^T \mathbf{b} &= \mathbf{p}_k^T \mathbf{A} \mathbf{x}_* = \sum_{i=1}^{n} \alpha_i \mathbf{p}_k^T \mathbf{A} \mathbf{p}_i = \alpha_k \mathbf{p}_k^T \mathbf{A} \mathbf{p}_k \\ \alpha_k &= \frac{\mathbf{p}_k^T \mathbf{b}}{\mathbf{p}_k^T \mathbf{A} \mathbf{p}_k} \end{aligned} \tag{6.15}$$

α_k in Eq.(6.15) denote the coefficients of basis, which can be obtained by an iterative method like *Gram-Schmidt orthonormalization*. Let $\mathbf{r}_k$ be the residual at the k_{th} step

$$\mathbf{r}_k = \mathbf{b} - \mathbf{A}\mathbf{x}_k \tag{6.16}$$

Eq.(6.16) gives following Eq.(6.17)

$$\begin{aligned} \mathbf{p}_{k+1} &= \mathbf{r}_k - \sum_{i \leq k} \frac{\mathbf{p}_i^T \mathbf{A}\mathbf{r}_k}{\mathbf{p}_i^T \mathbf{A}\mathbf{p}_i}\mathbf{p}_i \\ \mathbf{x}_{k+1} &= \mathbf{x}_k + \alpha_{k+1}\mathbf{p}_{k+1} \end{aligned} \tag{6.17}$$

with

$$\alpha_{k+1} = \frac{\mathbf{p}_{k+1}^T \mathbf{r}_k}{\mathbf{p}_{k+1}^T \mathbf{A}\mathbf{p}_{k+1}} \tag{6.18}$$

Preconditioned Conjugate Gradient method is designed for fast convergence of the conjugate gradient method by assuming $\mathbf{x}_0$ as a well-known vector, usually vector $\mathbf{b}$ is chosen as $\mathbf{x}_0$ in image deblurring.

6.3.3 Complexity of SVD

The SVD of an $M \times N$ matrix is typically computed by a two-step procedure. In the first step, the matrix is reduced to a bidiagonal matrix. This takes $O(MN^2)$ floating-point operations. The second step is to compute the SVD of the bidiagonal matrix. This step can only be done with an iterative method (as with eigenvalue algorithms). However, in practice it suffices to compute the SVD up to a certain precision costing $O(N)$. Thus, the first step is more expensive, and the overall cost

is $O(MN^2)$ floating-point operations. If the matrix is a square martix with $M = N$, then the cost of SVD is $O(M^3)$. As we mentioned earlier, kernel matrix for an $M \times M$ image is $M^2 \times M^2$ matrix and its inverse costs $O(M^6)$. If the kernel matrix can be expressed as the Kronecker product of two matrices, inverse of Kronecker product requires two SVD operations and 1 Kronecker product operation with two $M \times M$ matrices according to Eq.(6.10); two SVDs cost $O(2M^3)$ and Kronecker product operation costs $O(M^4)$. Thus, the overall cost is $O(M^4 + 2M^3)$. Roughly inverse operation of Kronecker product is M^2 times faster than the general SVD. For example, if image size is 500×500, the size of kernel matrix is 250000×250000. The overall cost of general SVD is of order 1.5625×10^{16}, and Kronecker product method is of order 2.5×10^9. General SVD operation is 5.537×10^7 times slower than FFT operation and 2.5×10^5 times slower than inverse operation of Kronecker product; FFT operation costs around 2.8219×10^4, and inverse operation Kronecker product costs about 2.5×10^9.

6.4 Image Restoration based on RT and SVD

Here, we compare the performance of image restoration based on the RT method and two matrix approaches – sectioning method and Kronecker product. The images used in this experiment are blurred by a shift-variant Gaussian PSF. We chose the SV-PSF to have a minimum near the border and maximum near the image center. Fig.6-1 shows two 469×469 images used in this experiment (kernel size is 219961×219961). PSF is assumed to be piecewise shift-invariant or convolution in 67×67 image blocks.

Blur parameter in a block is approximated to be the average of the original blur parameters in that block. RT with order 2, Eq.(6.19), was used for image restoration in this experiment.

$$
\begin{aligned}
f^{(0,0)} \;=\; & g^{(0,0)} - \; g^{(1,0)}\; h_{2,0}^{(1,0)} \; - g^{(0,1)}\; h_{0,2}^{(0,1)} \\
& + g^{(2,0)} \left(\frac{3}{2}\,(h_{2,0}^{(1,0)})^2 \; + \; \frac{1}{2}\, h_{0,2}^{(0,1)}\; h_{2,0}^{(0,1)} \; - \; \frac{1}{2} h_{2,0}^{(0,0)} \right) \\
& + g^{(0,2)} \left(\frac{3}{2}\,(h_{0,2}^{(0,1)})^2 \; + \; \frac{1}{2}\, h_{0,2}^{(1,0)}\; h_{2,0}^{(1,0)} \; - \; \frac{1}{2}\, h_{0,2}^{(0,0)} \right)
\end{aligned}
\tag{6.19}
$$

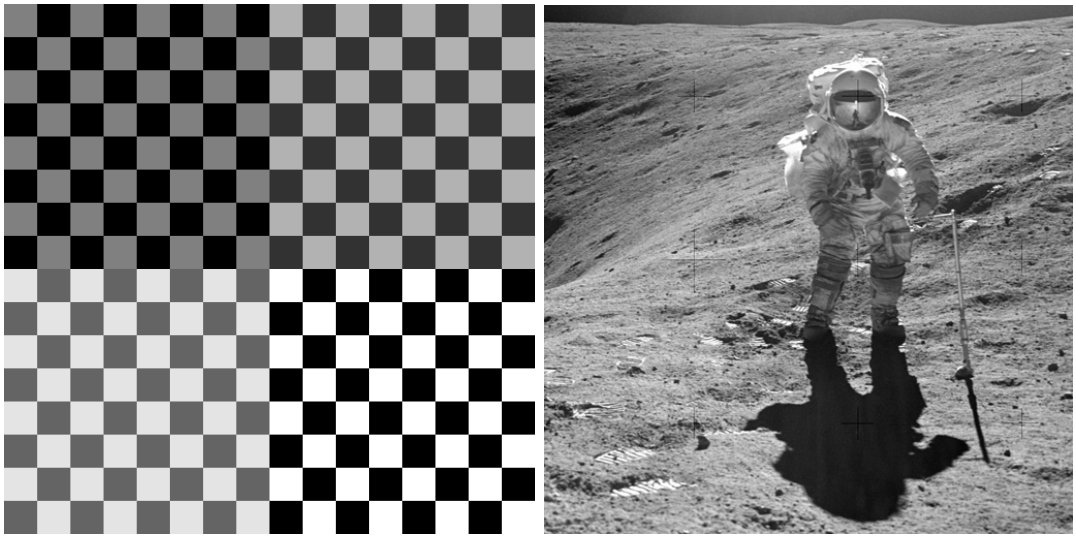


Figure 6-1: Sample image for simulation. (a) Left: "Checkerboard", (b) Right: "Moon".

First we compare the computational costs of Kronecker SVD and RT. We used a desktop personal computer wth an Intel Pentium processor (clock speed 1.7GHz). The restoration algorithm was implemented in Matlab. No particular effort was made to optimize the source code to reduce computations. The restoration algorithm was run 50 times in a loop to improve the estimate of computational cost. The average times

of Kronecker SVD and RT were around 1.68 seconds and 0.11 seconds respectively. RT restoration is around 16 times faster than Kronecker SVD.

Many experiments were carried-out with various blur parameters. Some of the experimental results are presented here. The blur parameter σ in the experiments has a maximum value in the range of 0.5 to 3.0 pixels near the center of the image and a minimum value of 0 at the corner of the image. Fig.6-2(a) shows the variation of σ where $\sigma_{max} = 1$. SVD restoration was carried-out based on Fig.6-2(b) and RT restoration was performed using sigma shown in Fig.6-2(a)

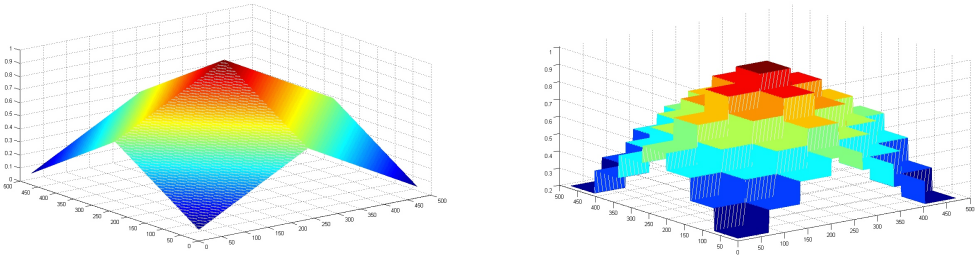


Figure 6-2: Plot of $\sigma_{max} = 1$. (a) Left: σ for RT, (b) Right: σ for SVD.

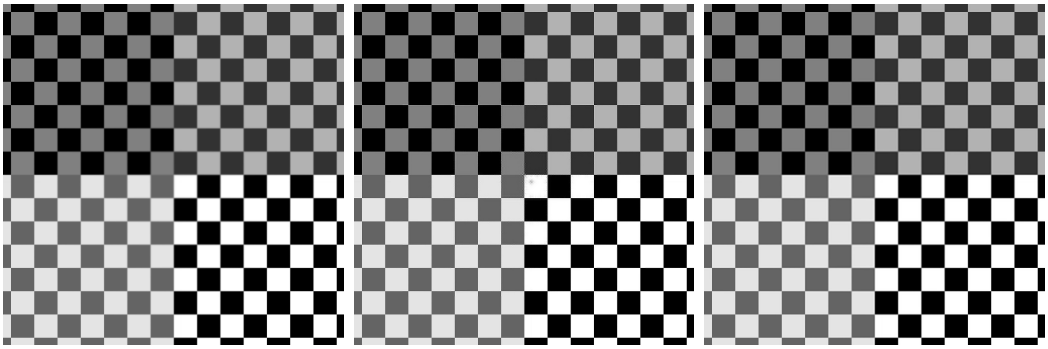


Figure 6-3: Simulation results of "Checkerboard" with $\sigma_{max} = 1$. (a) Left top: Blurred image , (b) Right top: Restored image using plain SVD without truncation or regularization, (c) Bottom: Restored image using RT.

Fig.6-3, Fig.6-4, Fig.6-5, and Fig.6-6 show the results of SVD and RT restoration. SVD without some form of regularization is rarely used in practice. The results shown

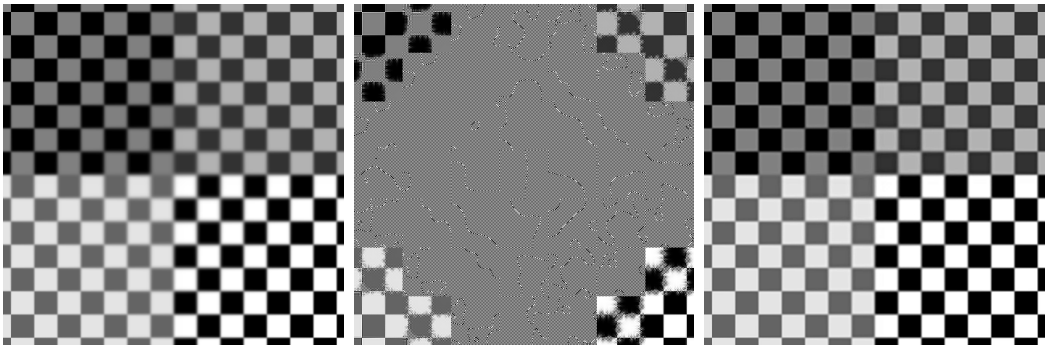


Figure 6-4: Simulation results of "Checkerboard" with $\sigma_{max} = 3$. (a) Left: Blurred image , (b) Middle: Restored image using plain SVD without truncation or regularization, (c) Right: Restored image using RT.

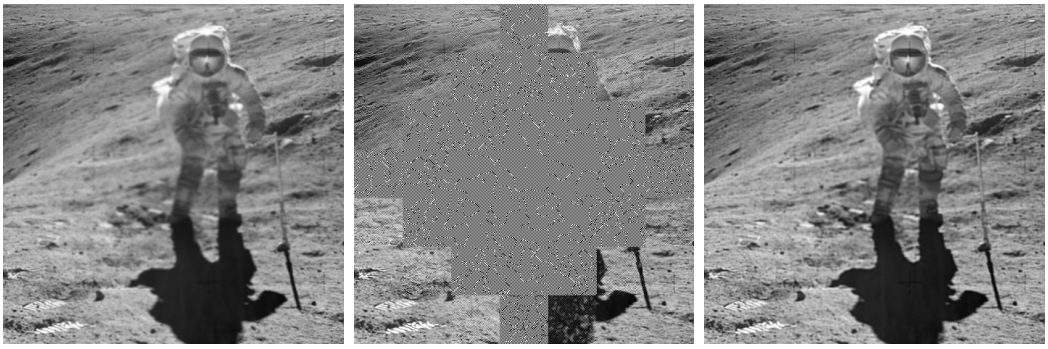


Figure 6-5: Simulation results of "Moon" with $\sigma_{max} = 2$. (a) Left: Blurred image , (b) Middle: Restored image using plain SVD without truncation or regularization, (c) Right: Restored image using RT.

here are for SVD without regularization. These results in fact show the need for and importance of regularization, as the results of SVD fail when $\sigma > 1$ near the center of the image in two of the examples. The main cause of these failures is the ill-posedness of the problem which becomes worse as the blur increases. The central area of Fig.6-4(b) and Fig.6-6(b) have very poor restoration. As the blur increases, the magnitude of the singular values drop rapidly, an indication of the higher noise sensitivity, as shown in Fig.6-8. This leads to a steep drop in the accuracy of image restoration. In this case, RT gives better results due to its local nature of computations and the truncation of Taylor series of focused image at third order terms in deriving it. This

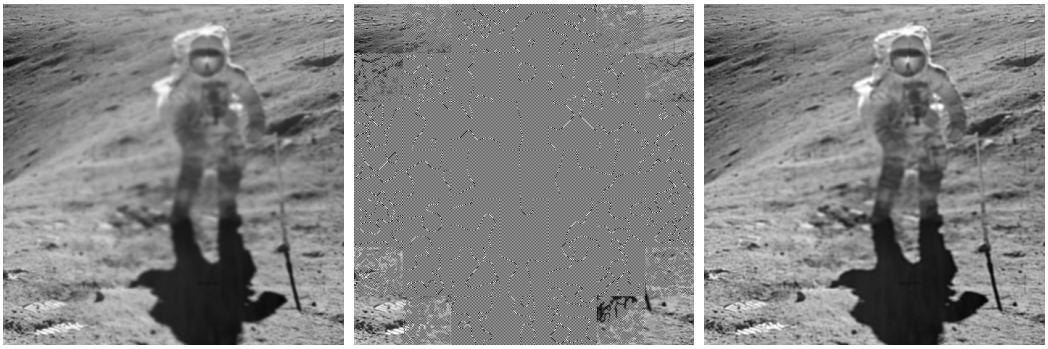


Figure 6-6: Simulation results of "Moon" with $\sigma_{max} = 3$. (a) Left: Blurred image , (b) Middle: Restored image using plain SVD without truncation or regularization, (c) Right: Restored image using RT.

truncation is an implicit form of regularization.

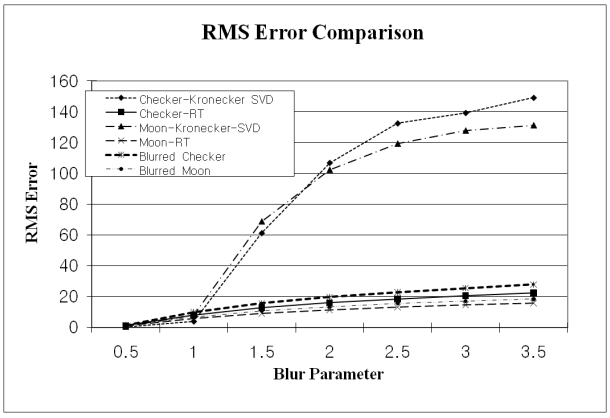


Figure 6-7: RMS error comparison of SVD and RT

Fig.6-7 shows RMS error of SVD and RT. RT restoration reduced the RMS error of the original blurred image with respect to the focused image by around 20 percent for the Checker image and around 16 percent for the Moon image. SVD restoration shows good performance up to $\sigma = 1$, but RMS error of SVD increases steeply in the range of $1 < \sigma < 2.5$. Since singular values divide blurred image as in Eq.(6.9), image restoration with small singular values amplify noise and error. Fig.6-8 shows the average of singular values for PSFs with different blur parameters. Larger value of blur parameter yields smaller average singular values. Average of singular value

is around 0.164 at $\sigma = 1$ and around 0.05 at $\sigma = 2$. Therefore noise and truncation errors become roughly 6 and 20 times larger.

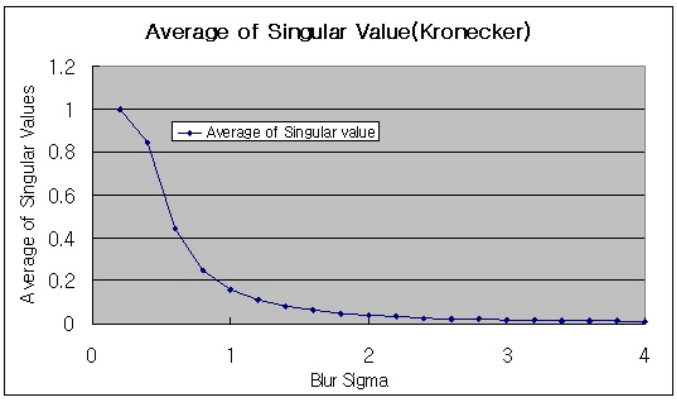


Figure 6-8: Plot of average Singular Value of PSF.

6.5 Image Restoration based on Conjugate Gradient Method

Image restoration experiments based on Conjugate Gradient (CG) method were performed under the same conditions as in Section 6.4. Computational time of CG method was calculated as the average over a large number of iterations. The average time for each iteration was found to be 8 seconds in comparison with 0.18 seconds for RT which is 44 times less than CG. Clearly, RT would be preferable in real-time applications.

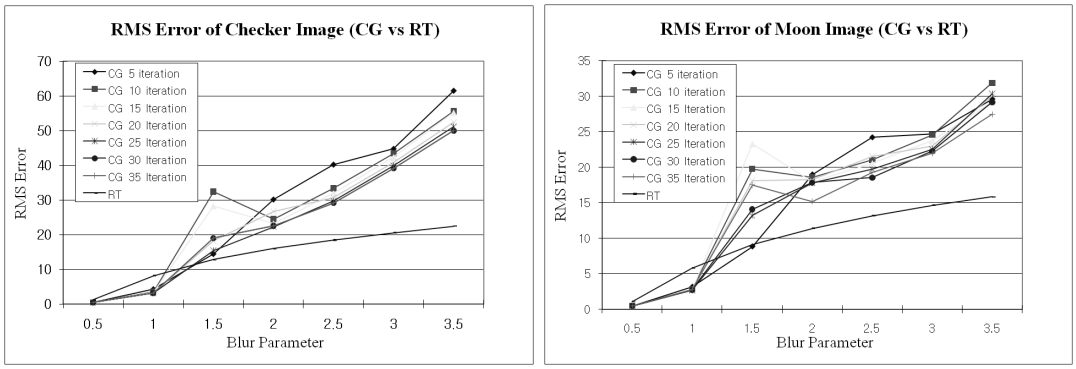


Figure 6-9: RMS error comparison of Conjugate Gradient and RT. Left: Checkerboard image. Right: Moon image.

RMS error of image restoration based on CG method is presented in Fig.6-9. Image restoration based on CG produces worse results than RT for large blur image. Like Kronecker SVD, CG is strongly affected by ill-posedness of the problem when the blur is large, e.g. $\sigma > 1$. After a certain number of iterations, additional iterations do not reduce the RMS error significantly.

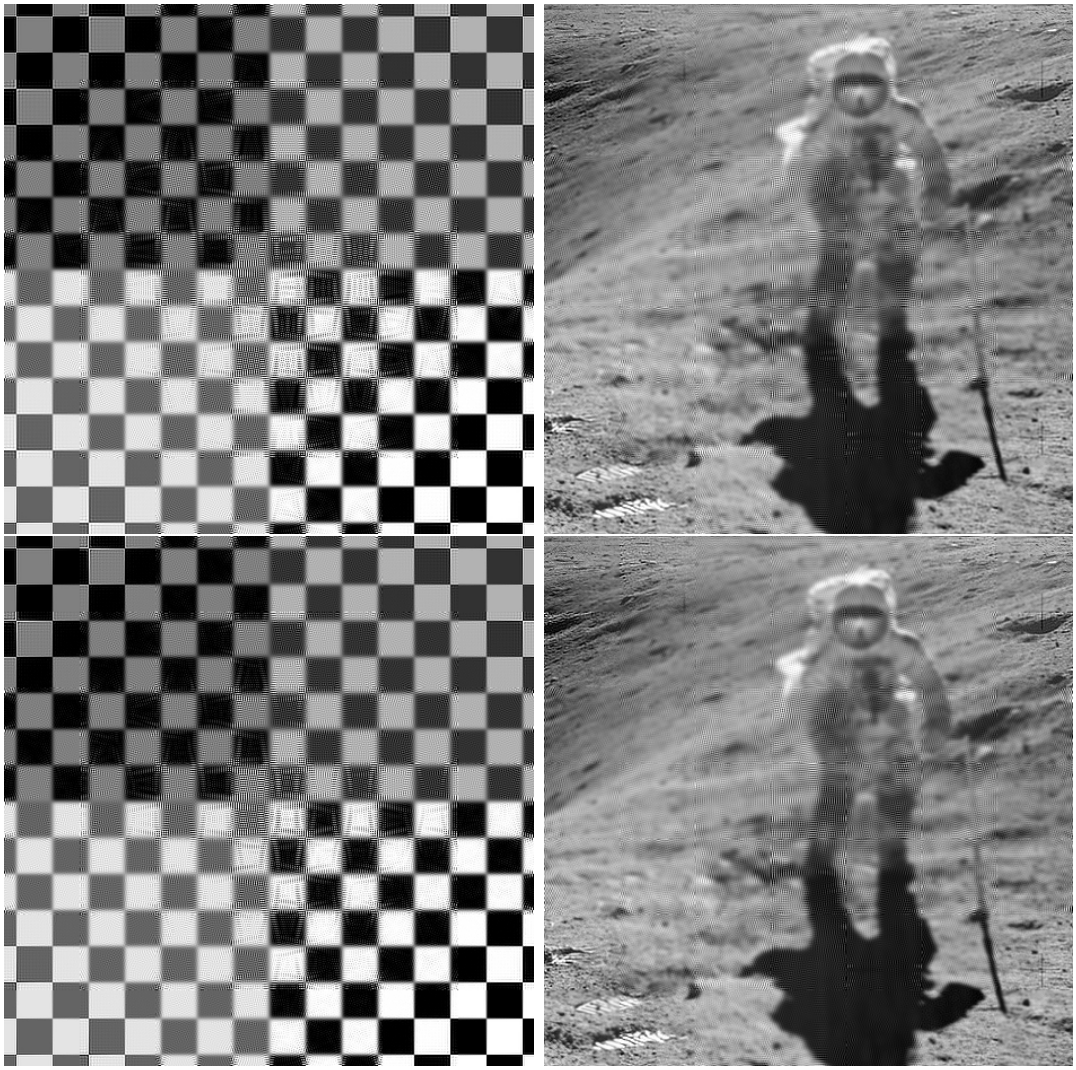


Figure 6-10: Restoration result of Conjugate Gradient with $\sigma_{max} = 3$ Top: 15 iterations. Bottom: 35 iterations.

6.6 Regularization techniques

Results in the previous sections show that plain SVD and CG do not perform well in image restoration. This is due to the inherent ill-posedness of the image deblurring problem which becomes worse with increasing blur. There is a need for regularization which imposes the restriction of smoothness on the restored images. Tikhonov method and Truncation method are commonly used as regularization methods for ill-posed

problems [21]. In this section we describe Tikhonov regularization and Truncated SVD for image restoration based on RT and SVD.

6.6.1 Regularization of SVD

Recall that the general form of inverse SVD takes the form

$$\mathbf{x} = \sum_{i=1}^{N} \frac{u_i^T \; b}{\sigma_i} \; v_i \tag{6.20}$$

Since singular value σ_i appears in the denominator in Eq.(6.20), small σ_i causes the SVD model to be highly sensitive to truncation error and noise. Thus, regularization of such small σ is required for stability.

Regularization is carried-out by multiplication with a filter factor ϕ_i as

$$\mathbf{x}_{filt} = \sum_{i=1}^{N} \phi_i \; \frac{u_i^T \; b}{\sigma_i} \; v_i \tag{6.21}$$

Truncated Singular Value Decomposition (TSVD) truncates smaller singular values less than regularization parameter α for stable result.

$$\phi_i = \begin{cases} 1 & \text{for } \sigma_i \geq \alpha \\ 0 & \text{Otherwise} \end{cases}$$

On the other hand, Tikhonov regularization reduces the noise error and truncation

error by assigning a suitable value to the regularization parameter α in:

$$\phi_i = \frac{\sigma_i^2}{\sigma_i^2 + \alpha^2} \qquad \text{for } i = 1, 2, \cdots, N$$

A regularized solution can be obtained by solving Eq.(6.22)

$$\min_{\mathbf{x}} \{\|\mathbf{b} - \mathbf{A}\mathbf{x}\|_2^2 + \alpha^2 \|\mathbf{x}\|_2^2\} \tag{6.22}$$

The optimal regularization parameter α is usually unknown. In practical problems it is often determined by an ad hoc method. Possible approaches to decide optimal α are discrepancy principle, cross-validation, L-curve method, etc. [43, 73].

6.6.2 Regularization of RT

In this section we derive an expression for Tikhonov regularization for image restoration based on RT. Matrix form of RT at a pixel is obtained in terms of image derivatives and moments of PSF. To apply Tikhonov regularization to RT we need to express system equation of RT in a vector-matrix form. Recall that the system equation of RT in vector-matrix form is expressed in Chapter 4 as

$$\mathbf{g}_{xy} = \mathbf{R}_{xy}\mathbf{f}_{xy}, \quad \text{or} \quad \mathbf{g} = \mathbf{R}\mathbf{f} \tag{6.23}$$

The equation above already includes truncation regularization as it was derived using the truncation of the Taylor-series expansion of $\mathbf{f}$. This truncation limits the size of

all matrices above. In addition to the truncation regularization, Tikhonov regularization which typically imposes somoothness on some of the derivatives of $\mathbf{f}$ is presented below. Note that regularization in RT is completely local because the image restoration at each pixel is carried-out separately. In other approaches, the entire restored image is computed together in one step.

As discussed in section 6.6.1, general Tikhonov Regularization takes the form

$$\min_{\mathbf{f}}\{\|\mathbf{b}-\mathbf{A}\mathbf{x}\|_2^2+\alpha^2\|\mathbf{D}\mathbf{x}\|_2^2\} \tag{6.24}$$

where $\mathbf{D}$ is a carefully chosen regularization matrix, often an approximation to a derivative. By substituting RT matrices, $\mathbf{g}$, $\mathbf{R}$, and $\mathbf{f}$, into $\mathbf{b}$, $\mathbf{A}$, and $\mathbf{x}$ in Eq.(6.24), we get the Tikhonov regularization formula for RT as

$$\min_{\mathbf{f}}\{\|\mathbf{g}-\mathbf{R}\mathbf{f}\|_2^2+\alpha^2\|\mathbf{D}\mathbf{f}\|_2^2\} \tag{6.25}$$

If the regularization matrix $\mathbf{D}$ is approximated to be the Laplacian operator, then $\mathbf{D} = [\ 0\ 0\ 0\ 1\ 1\ 0\ ]$. Similarly $\mathbf{D} = [\ 0\ 1\ 1\ 0\ 0\ 0\ ]$ represents gradient magnitude. By setting the derivative of the minimization function in Eq.(6.25) to zero, it follows that the problem is mathematically equivalent to solving the norm equation. Since partial derivative of Eq.(6.24) is zero at the minimum, the closed-form of regularization is obtained as

$$(\mathbf{R}^T\mathbf{R}+\alpha^2\mathbf{D}^T\mathbf{D})^{-1}\mathbf{f}=\mathbf{R}^T\mathbf{g} \tag{6.26}$$

From this Eq.(6.26) the Tikhonov solution $\mathbf{f}_{\alpha,D}$ is given by the closed-form expression

$$\mathbf{f}_{\alpha,D} = (\mathbf{R}^T\mathbf{R} + \alpha^2\mathbf{D}^T\mathbf{D})^{-1}\mathbf{R}^T\mathbf{g} \tag{6.27}$$

In the experiments, we approximate the regularization matrix $\mathbf{D}$ to be the Laplacian operator for RT regularization.

6.6.3 Experiments on Regularization of SVD

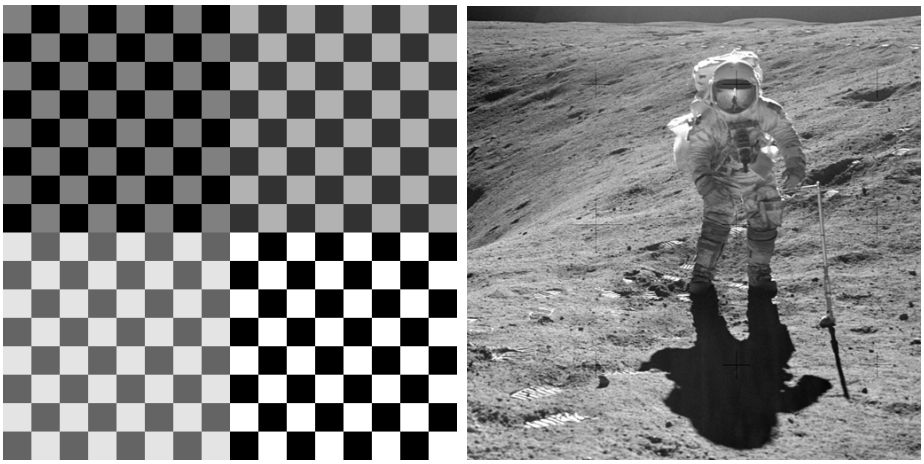


Figure 6-11: Sample image for simulation. (a) Left: "Checkerboard", (b) Right: "Moon".

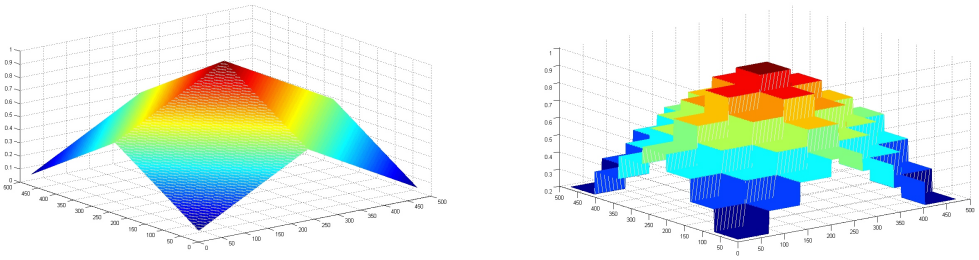


Figure 6-12: Plot of blur parameter σ with $\sigma_{max} = 1$. (a) Left: Original σ, (b) Right: Approximated σ for SVD.

Images with shift-variant Gaussian blur were restored using the regularized SVD

method. As in section 6.4, sample images in Fig.6-11 were blurred by blur parameter σ (see Fig.6-12-(a)), and then, we restored these blurred images by TSVD method and Tikhonov method. To get optimal regularization parameter, L-curve method and Cross-Validation method were tried, but, possibly due to large blur and kernel approximation, these methods failed to determine an appropriate parameter as shown in Fig.6-13. Image section where $\sigma > 1$ fails to determine an appropriate parameter.

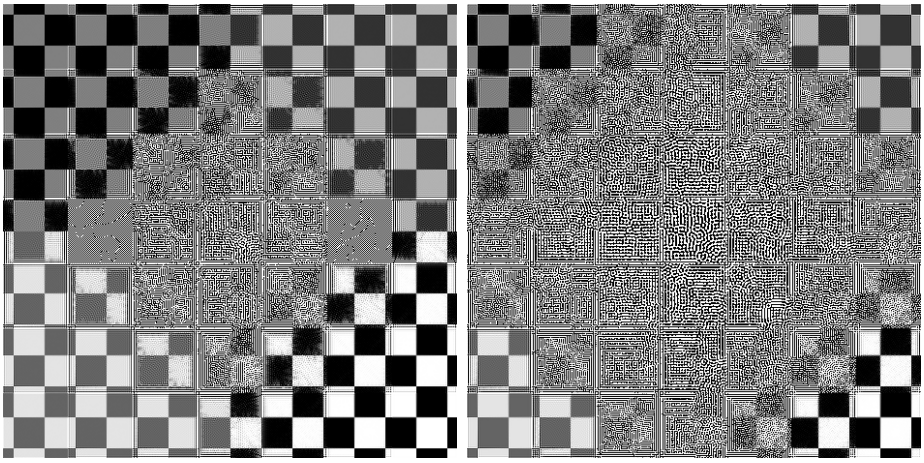


Figure 6-13: Restored image by Truncated SVD with L-curve method. (a) Left: $\sigma_{max} = 2.0$, (b) Right: $\sigma_{max} = 3.0$.

Therefore, regularization parameter was determined by hundreds of trials with various parameters in this experiment. For this experiment we increased blur parameter by 0.5 from 1.0 upto 3.5, and regularization parameter was increased by 0.001 starting from 0.001. We determined the optimal regularization parameter to be the one with the lowest RMS error.

Next we present the result of Tikhonov regularized SVD for checker image. Fig.6-15 shows the results for blurred checkerboard image with $\sigma_{max} = 2.0$. The optimal regularization parameter was estimated to be 0.032. For comparison, we also present

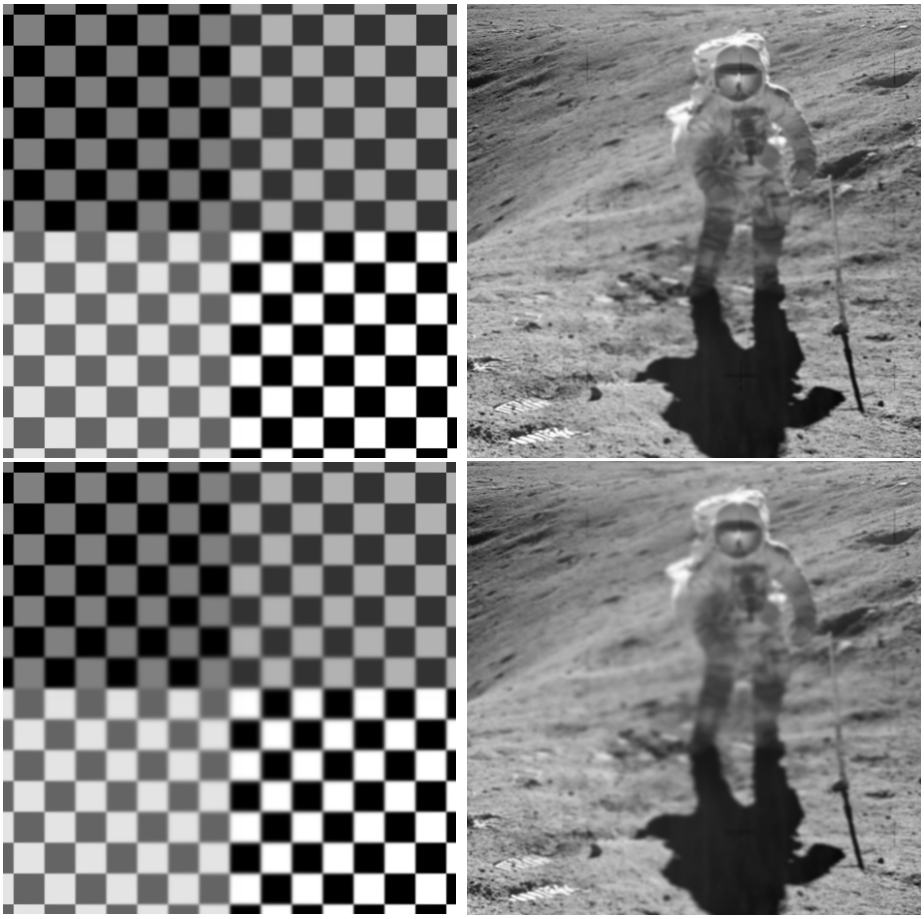


Figure 6-14: Blurred images. (a) Top: Blurred images by blur parameter $\sigma_{max} = 2.0$, (b) Bottom: Blurred images by blur parameter $\sigma_{max} = 3.0$.

the result of restoration with regularization parameter 0.001. The result shows that regularization offers improved image restoration by SVD. When $\sigma_{max} = 2.0$, the RMS error of Tikhonov regularized SVD is around 1/5 of RMS error of general SVD for Moon image and around 1/2 for checkerboard image. When $\sigma_{max} = 3.0$, RMS error of Tikhonov regularization for the Moon image is around 72 percent less than that of the general SVD. Tikhonov regularization for checkerboard produces half the RMS error of the general SVD when $\sigma_{max} = 3.0$. However, when $\sigma_{max} = 0.5$, regularization does not give any improvement in result since most of the singular values of SVD are bigger than the regularization parameter.

Truncated SVD produces results similar to that of Tikhonov regularization. In this case, checkerboard is improved by around 55 percent and moon image by around 80 percent in the sense of RMS error. When $\sigma_{max} = 2.0$, optimal regularization parameter of Truncated SVD for the moon image was estimated to be 0.039 and its RMS error was around 9.3.

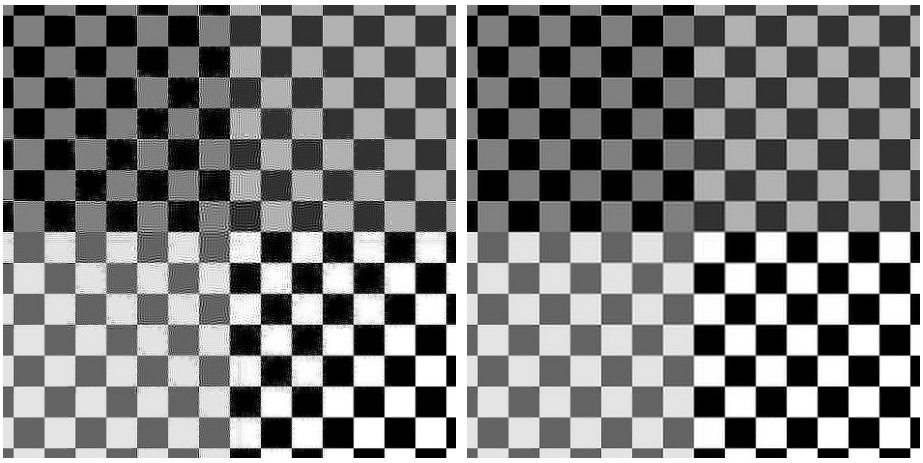


Figure 6-15: Regularization result of Tikhonov SVD with $\sigma_{max} = 2.0$. (a) Left: Restored image by Tikhonov SVD with parameter 0.001, (b) Right: Restored image by Tikhonov SVD with parameter 0.032

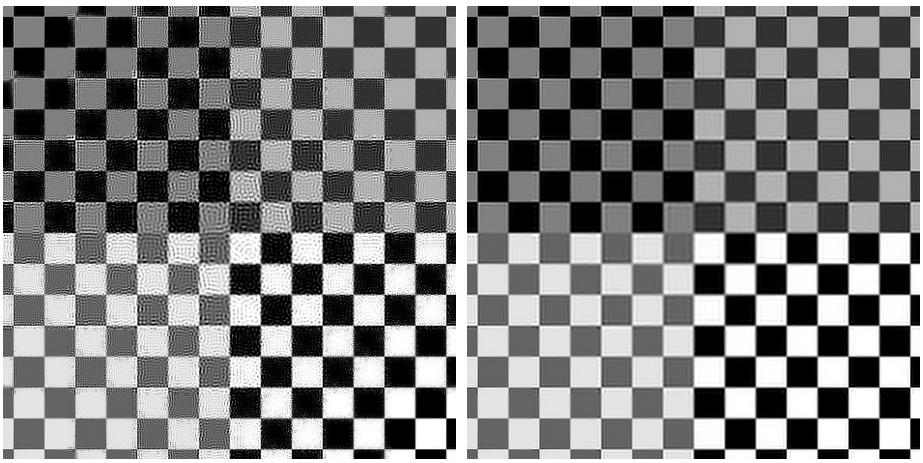


Figure 6-16: Regularization result of Tikhonov SVD with $\sigma_{max} = 3.0$. (a) Left: Restored image by Tikhonov SVD with parameter 0.001, (b) Right: Restored image by Tikhonov SVD with parameter 0.032

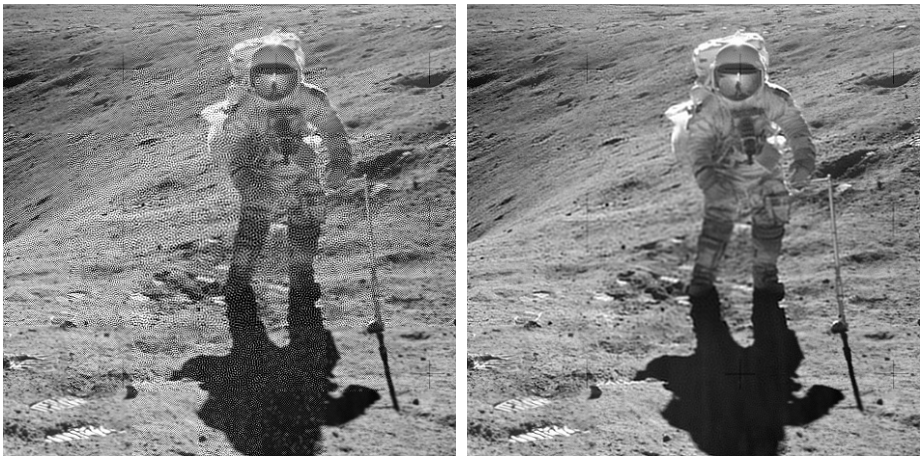


Figure 6-17: Regularization result of Tikhonov SVD with $\sigma_{max} = 2.0$. (a) Left: Restored image by Tikhonov SVD with parameter 0.001, (b) Right: Restored image by Tikhonov SVD with parameter 0.031

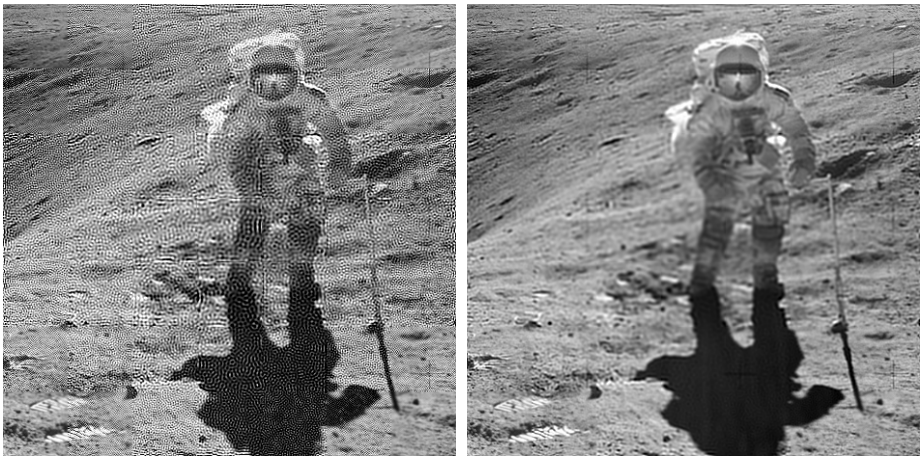


Figure 6-18: Regularization result of Tikhonov SVD with $\sigma_{max} = 3.0$. (a) Left: Restored image by Tikhonov SVD with parameter 0.001, (b) Right: Restored image by Tikhonov SVD with parameter 0.031

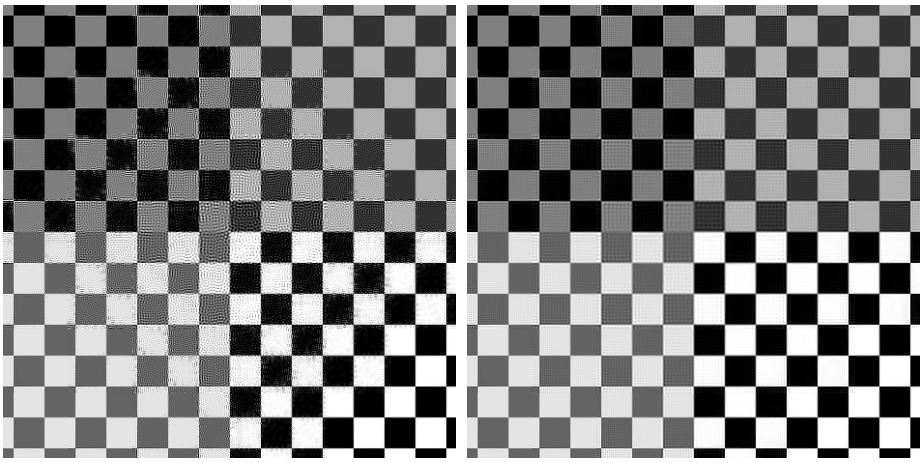


Figure 6-19: Regularization result of TSVD with $\sigma_{max} = 2.0$. (a) Left: Restored image by TSVD with parameter 0.001, (b) Right: Restored image by TSVD with parameter 0.027

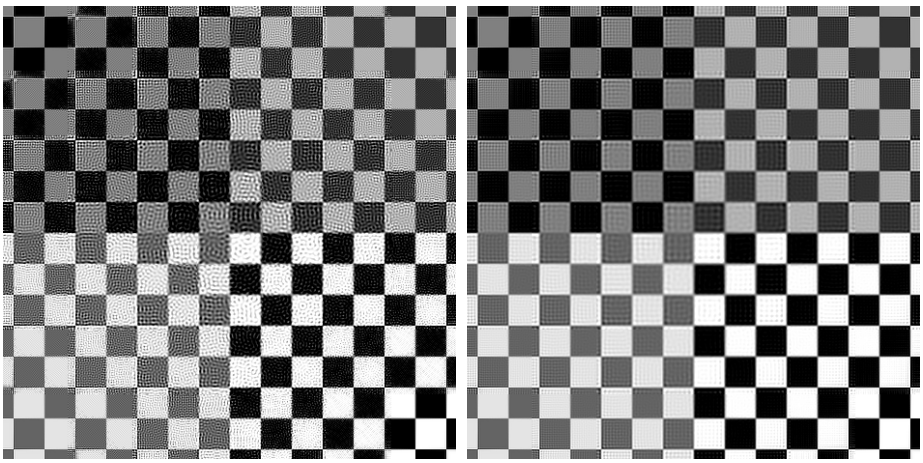


Figure 6-20: Regularization result of TSVD with $\sigma_{max} = 3.0$. (a) Left: Restored image by TSVD with parameter 0.001, (b) Right: Restored image by TSVD with parameter 0.023

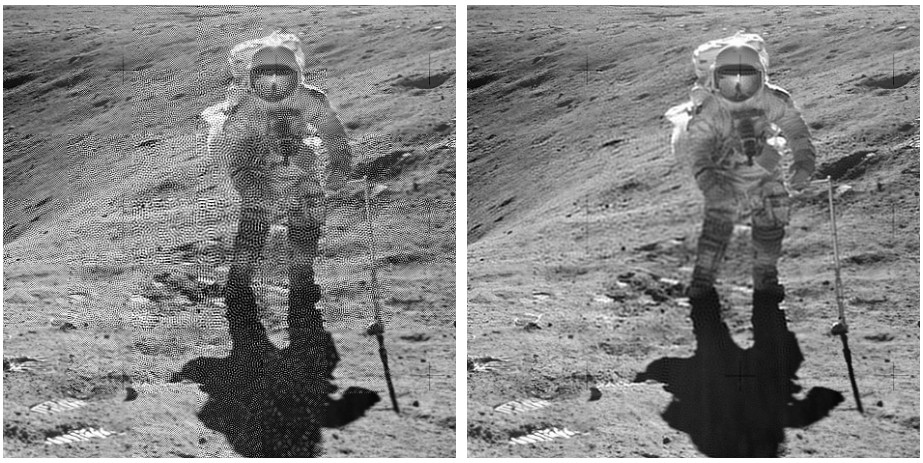


Figure 6-21: Regularization result of TSVD with $\sigma_{max} = 2.0$. (a) Left: Restored image by TSVD with parameter 0.001, (b) Right: Restored image by TSVD with parameter 0.039

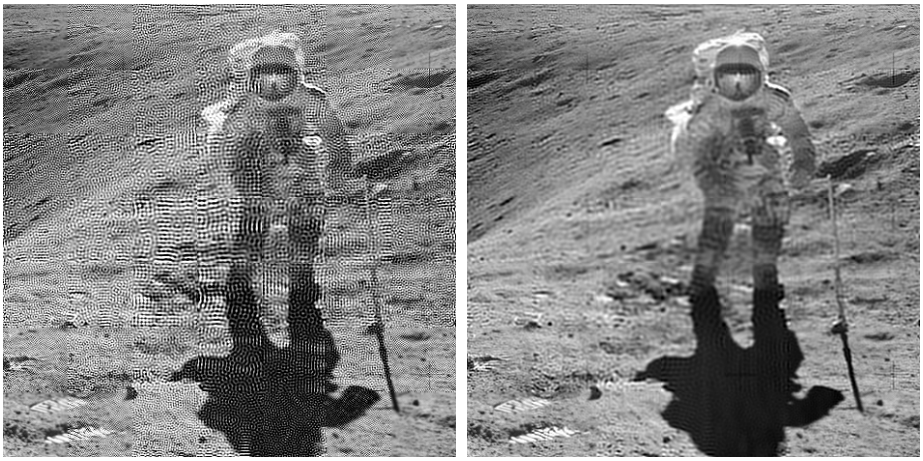


Figure 6-22: Regularization result of TSVD with $\sigma_{max} = 3.0$. (a) Left: Restored image by TSVD with parameter 0.001, (b) Right: Restored image by TSVD with parameter 0.038

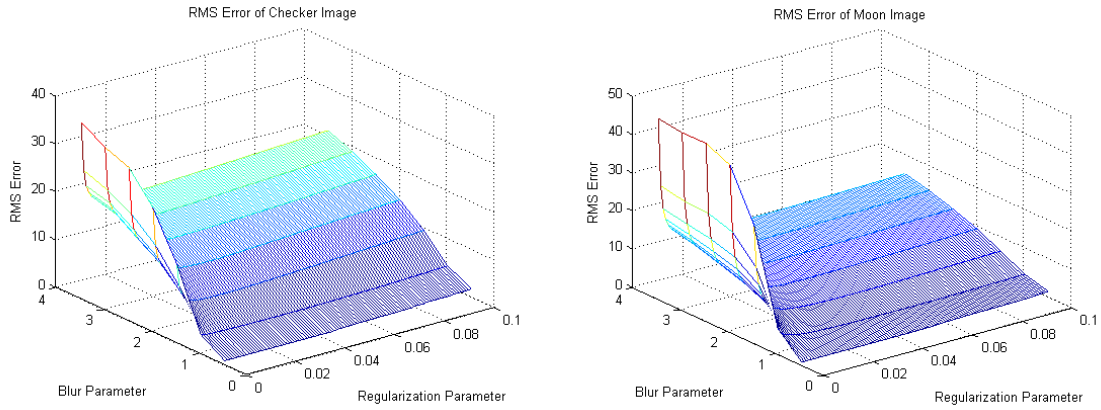


Figure 6-23: RMS error of Tikhonov SVD with various regularization parameter. (a) Left: Checkerboard image, (b) Right: Moon image.

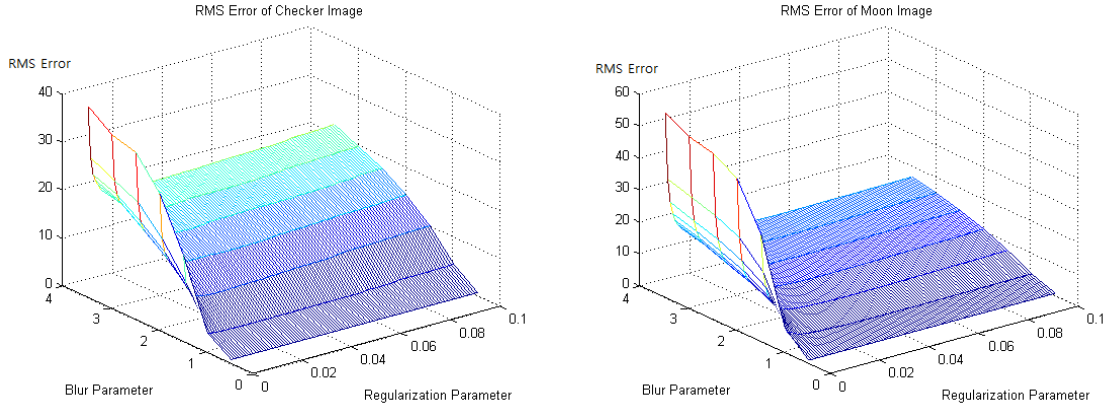


Figure 6-24: RMS error of Truncated SVD with various regularization parameter. (a) Left: Checkerboard image, (b) Right: Moon image.

6.6.4 Experiments on Regularization of RT

Experimental results of image restoration based on regularized RT as in Eq.(6.27) are presented here. The images used and the blur parameter are the same as in section 6.6.3. We consider Taylor series expansion terms of order up to 3 in RT. The size of filter windows for estimating image derivatives was 7x7. Fig.6-25 shows the result of regularized RT with regularization parameter 0.05. Restoration based on general RT without regularization is shown in the left column in Fig.6-25 for comparison.

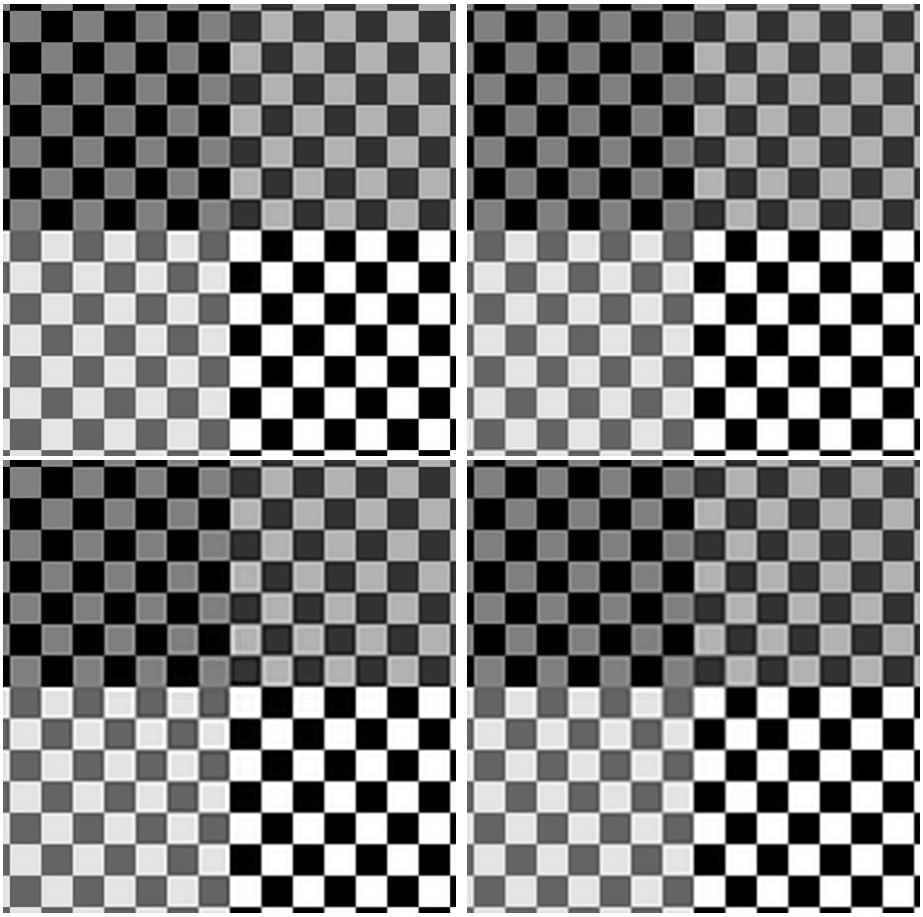


Figure 6-25: Regularization result of RT with Checkerboard image (a) Left top: General RT (no regularization) with $\sigma_{max} = 2.0$, (b) Right top: Regularized RT with $\sigma_{max} = 2.0$ and regularization parameter 0.05, (c) Left bottom: General RT with $\sigma_{max} = 3.0$, (d) Right bottom: Regularized RT with $\sigma_{max} = 3.0$ and regularization parameter 0.05

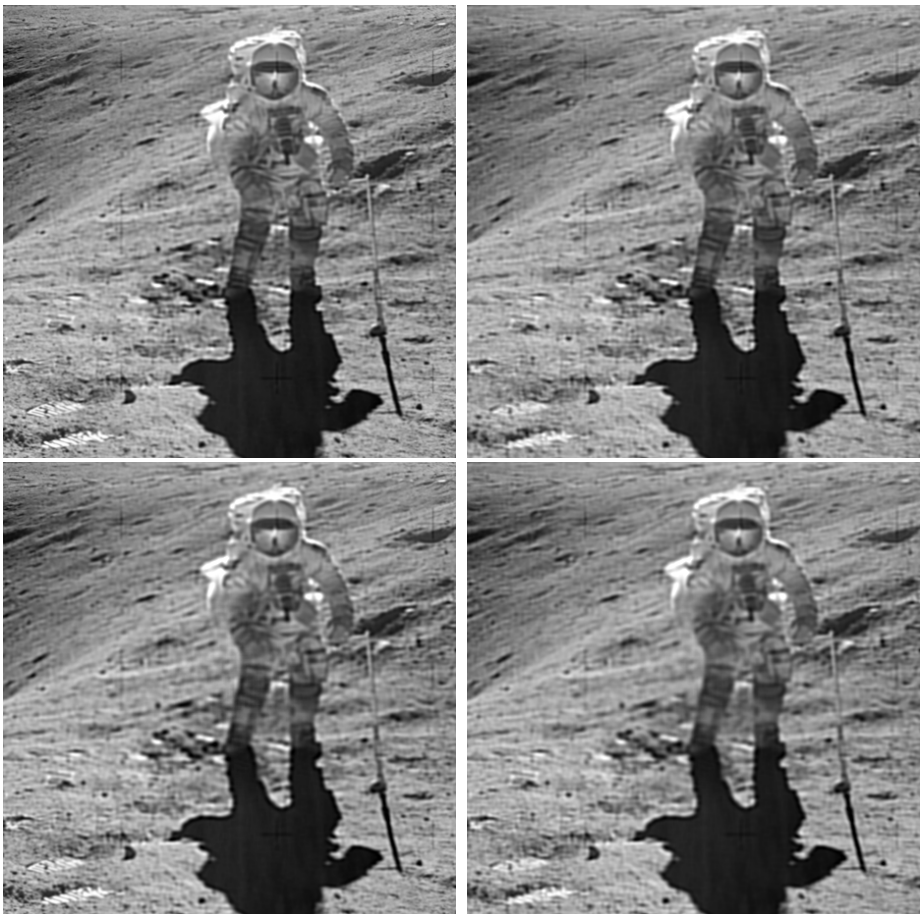


Figure 6-26: Regularization result of RT with Moon image (a) Left top: General RT with $\sigma_{max} = 2.0$, (b) Right top: Regularized RT with $\sigma_{max} = 2.0$ and regularization parameter 0.05, (c) Left bottom: General RT with $\sigma_{max} = 3.0$, (d) Right bottom: Regularized RT with $\sigma_{max} = 3.0$ and regularization parameter 0.05

We carried-out regularization for RT with a real image captured by Olympus E-1 camera. The image captured has a size of 640×480. The shift-variant PSF was assumed to be a Gaussian. The blur parameter was estimated to be as shown in Fig.6-27(b). This image of an inclined image was restored using Eq.(6.27).

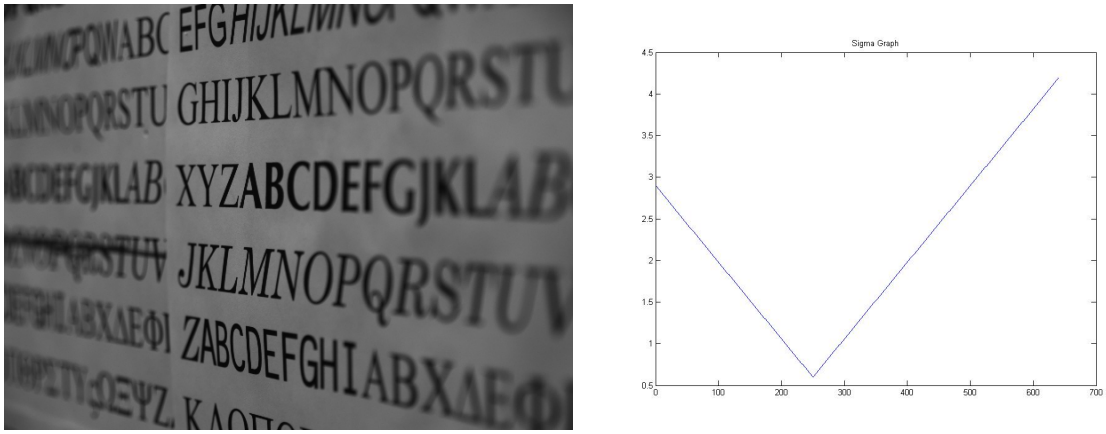


Figure 6-27: Test image and estimated σ. (a) Left: The image of a slanted plane with shift-variant blur captured by Olympus E-1 camera and used in the experiments. (b) Right: A plot of the estimated Gaussian blur parameter σ along the horizontal axis.

Figure 6-28: Restored image with/without regularization based on RT. (a) Top: restored Image without regularization, (b) Middle: restored and regularized image with $\alpha = 0.15$ (c) Bottom: restored and regularized image with $\alpha = 0.35$.

6.7 Summary of experimental results

Kronecker SVD appraoch without regularization uses piecewise convolution approximation in small image blocks. It is computationally expensive, around 16 times more than RT in the experiments. Also, if the blur is not small ($\sigma > 1.0$), the RMS error is much more than RT. Conjugate gradient method was also far more computationally expensive than RT, around 44 times more in the experiments, and it was also worse than RT in terms of RMS error. In the SVD method with regularization, finding the optimal regularization parameter is a problem. After using a nearly optimal value for the regularization parameter, the results of image restoration are found to be good, but computational cost remains high. RT inherently includes regularization in the form of Taylor-series truncation in its implementation. Additional Tikhonov regularization can be used where it computes a smoother solution by penalizing large variations in the derivatives of the focused image. In both cases, RT gives good restoration results at a relatively low computational cost.

Chapter 7

Conclusion

In this thesis we have addressed two closely related topics: a novel robust autofocusing technique (RDFD) and a new image restoration technique based on RT for shift-variant system. We discussed their theory, and verified their performance by many simulation and actual experiments.

RDFD is a new autofocusing technique that is robust against camera/object motion. Since RDFD shares some characteristics of both DFF and spatial domain DFD techniques, it provises computational efficiency like DFD and robustness against discrepancy between sampled images like DFF. Experimental results confirm that RDFD is consistent in the presence of camera/object motion. Investigation of RDFD leads to the derivation of an explicit relation between DFF and DFD techniques, which is a new polynomial model of Focus Measure (FM). The new polynomial model approximates FM with respect to blur parameter σ such that DFD can predict FM value corresponding to a given σ. The polynomial model provides a good approximation to FM if σ is in the range $0 < \sigma < 1.3$. RDFD provides new insights into both DFD

and DFF.

We presented the theoretical basis of RT for shift-variant systems in one dimension and two dimension, and we discussed image restoration based on RT. Also, we compared the performance of RT with related approaches that incorporate regularization. RT provides a very useful mathematical model to formulate a shift-variant system and invert it. One of the most important advantages of RT is its computational efficiency in comparison with other techniques and the fact that it does not have any restriction on the shape of the shift-variant kernel. Many related approaches solve the shift-variant deblurring problem only when its kernel is restricted in some way. For example, sectioning method requires variation of the PSF to be slow enough that it is negligible in each image section. The matrix approach becomes computationally exorbitant unless it is a special case where the kernel matrix is a Toeplitz matrix or a Kronecker product. Another substantial advantage of RT is that it is completely localized so that it can be easily implemented in parallel. Moreover, regularization can be incorporated into RT for reducing noise sensitivity. New insights into image blurring and image restoration processes provided by RT theory and algorithms can guide future research in this area.

Appendix A

Derivation of Differentiation Filters by Polynomial Fitting

In this section we derive differentiation filters for estimating image derivatives using polynomial fitting.

Let a given blurred image be an n-th order polynomial given by

$$g(x, y) = \sum_{i+j=0}^{i+j=n} a_{i,j} x^i y^j \tag{A.1}$$

The pixel where we want to estimate the derivative is taken to be the origin of the image domain, and N neighboring pixels including the origin is used as a sample set for polynomial fitting. The sample set is specified by an $N \times 1$ column vector. Suppose that the extracted sample image is a square window of size 7×7, then the

column vector is

$$\mathbf{g} = \left[\begin{array}{ccccccc} g(-3,-3), & g(-3,-2), & \cdots, & g(-3,3), & g(-2,-3), & \cdots, & g(3,3) \end{array} \right]^T \quad \text{(A.2)}$$

where $\mathbf{g}$ denotes sample column vector.

Since an n-th order polynomial has $n(n+1)/2$ coefficients, coefficient column vector is an $\frac{n(n+1)}{2} \times 1$ vector. Let the coefficient column vector $\mathbf{C}$ of the n-th order polynomial be specified by

$$\mathbf{C} = \left[\begin{array}{ccccccccc} a_{00}, & a_{10}, & a_{01}, & a_{20}, & a_{11}, & a_{02}, & a_{30}, & a_{21}, & \cdots, & a_{0n} \end{array} \right]^T \quad \text{(A.3)}$$

where $a_{i,j}$ denotes the coefficient of $x^i y^j$ term. The relation between sample vector $\mathbf{g}$ and coefficient vector $\mathbf{C}$ is expressed as

$$\mathbf{g} = \mathbf{vC} \quad \text{(A.4)}$$

where

$$\mathbf{v} = \begin{bmatrix} 1 & x|_{x=-3} & y|_{y=-3}, & x^2|_{x=-3}, & \cdots & xy^{n-1}|_{x=-3,y=-3}, & y^n|_{x=-3,y=-3} \\ 1 & x|_{x=-3} & y|_{y=-2}, & x^2|_{x=-3}, & \cdots & xy^{n-1}|_{x=-3,y=-2}, & y^n|_{x=-3,y=-2} \\ \vdots & & & \vdots & & & \vdots \\ 1 & x|_{x=-3} & y|_{y=3}, & x^2|_{x=-3}, & \cdots & xy^{n-1}|_{x=-3,y=3}, & y^n|_{x=-3,y=3} \\ 1 & x|_{x=-2} & y|_{y=-3}, & x^2|_{x=-2}, & \cdots & xy^{n-1}|_{x=-2,y=-3}, & y^n|_{x=-2,y=-3} \\ \vdots & & & \vdots & & & \vdots \\ 1 & x|_{x=3} & y|_{y=3}, & x^2|_{x=3}, & \cdots & xy^{n-1}|_{x=3,y=3}, & y^n|_{x=3,y=3} \end{bmatrix} \tag{A.5}$$

One row of matrix $\mathbf{v}$ in Eq.(A.5) corresponds to one pixel. The equation above was solved for the polynomial coefficients $\mathbf{C}$ using the pseudo-inverse of $\mathbf{v}$ as $\mathbf{C} = \mathbf{v}^{+}\mathbf{g}$ where $\mathbf{v}^{+} = (\mathbf{v}^T\mathbf{v})^{-1}\mathbf{v}^T$.

Since the pixel position where we calculate derivatives is the origin in the image domain, $x = 0$ and $y = 0$, the i-th derivative with respect to x and j-th derivative with respect to y is obtained as

$$g^{(i,j)}(x,y)\Big|_{x=0,y=0} = \frac{\partial^{(i,j)} g(x,y)}{\partial x^i y^j}\Big|_{x=0,y=0} = i!\ j!\ a_{ij} \tag{A.6}$$

Since $\mathbf{v}$ is independent of the image pixel values in $\mathbf{g}$, computing the pseudo-inverse $\mathbf{v}^{+}$ of $\mathbf{v}$ is required only once. The result can be used to estimate the polynomial coefficients at each pixel in the form of a convolution filter. The 7×7 differentiation

filters thus obtained by fitting a cubic polynomial are listed in Eq.(A.7)-(A.10)

$$\frac{d}{dx} = \begin{bmatrix} 0.0380 & -0.0210 & -0.0244 & 0.00 & 0.0244 & 0.0210 & -0.0380 \\ 0.0125 & -0.0380 & -0.0329 & 0.00 & 0.0329 & 0.0380 & -0.0125 \\ -0.0028 & -0.0482 & -0.0380 & 0.00 & 0.0380 & 0.0482 & 0.0028 \\ -0.0079 & -0.0516 & -0.0397 & 0.00 & 0.0397 & 0.0516 & 0.0079 \\ -0.0028 & -0.0482 & -0.0380 & 0.00 & 0.0380 & 0.0482 & 0.0028 \\ 0.0125 & -0.0380 & -0.0329 & 0.00 & 0.0329 & 0.0380 & -0.0125 \\ 0.0380 & -0.0210 & -0.0244 & 0.00 & 0.0244 & 0.0210 & -0.0380 \end{bmatrix} \tag{A.7}$$

$$\frac{d}{dy} = \begin{bmatrix} 0.0380 & 0.0125 & -0.0028 & -0.0079 & -0.0028 & 0.0125 & 0.0380 \\ -0.0210 & -0.0380 & -0.0482 & -0.0516 & -0.0482 & -0.0380 & -0.0210 \\ -0.0244 & -0.0329 & -0.0380 & -0.0397 & -0.0380 & -0.0329 & -0.0244 \\ 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 \\ 0.0244 & 0.0329 & 0.0380 & 0.0397 & 0.0380 & 0.0329 & 0.0244 \\ -0.0210 & 0.0380 & 0.0482 & 0.0516 & 0.0482 & 0.0380 & 0.0210 \\ .0380 & -0.0125 & 0.0028 & 0.0079 & 0.0028 & -0.0125 & -0.0380 \end{bmatrix} \tag{A.8}$$

$$\frac{d^2}{dx^2} = \begin{bmatrix} 0.0170 & 0.00 & -0.0102 & -0.0136 & -0.0102 & 0.00 & 0.0170 \\ 0.0170 & 0.00 & -0.0102 & -0.0136 & -0.0102 & 0.00 & 0.0170 \\ 0.0170 & 0.00 & -0.0102 & -0.0136 & -0.0102 & 0.00 & 0.0170 \\ 0.0170 & 0.00 & -0.0102 & -0.0136 & -0.0102 & 0.00 & 0.0170 \\ 0.0170 & 0.00 & -0.0102 & -0.0136 & -0.0102 & 0.00 & 0.0170 \\ 0.0170 & 0.00 & -0.0102 & -0.0136 & -0.0102 & 0.00 & 0.0170 \\ 0.0170 & 0.00 & -0.0102 & -0.0136 & -0.0102 & 0.00 & 0.0170 \end{bmatrix} \tag{A.9}$$

$$\frac{d^2}{dy^2} = \begin{bmatrix} 0.0170 & 0.0170 & 0.0170 & 0.0170 & 0.0170 & 0.0170 & 0.0170 \\ 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 \\ -0.0102 & -0.0102 & -0.0102 & -0.0102 & -0.0102 & -0.0102 & -0.0102 \\ -0.0136 & -0.0136 & -0.0136 & -0.0136 & -0.0136 & -0.0136 & -0.0136 \\ -0.0102 & -0.0102 & -0.0102 & -0.0102 & -0.0102 & -0.0102 & -0.0102 \\ 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 \\ 0.0170 & 0.0170 & 0.0170 & 0.0170 & 0.0170 & 0.0170 & 0.0170 \end{bmatrix} \tag{A.10}$$

Bibliography

[1] J. M. Tenenbaum, "Accommodation in computer vision", PhD Dissertation, Stanford Univ. 1970.

[2] F. R. Boddeke, L. J. Van Vliet, H. Netten, and I. T. Young, "Autofocusing in microscopy based on the OTF and sampling", *Bioimaging 2*, 193-203 1994.

[3] J. L. Pech-Pacheco and G. Cristobal, "Diatom autofocusing in bright field microscopy: a comparative study," *Proc. Int. Conf. on Pattern Recognition*, Vol. 3, pp. 318-321, IEEE Computer Soc., Los Alamitos, CA, 2000.

[4] E. Krotkov, "Focusing", *international Journal of Computer Vision*, 1, pp. 223-237, 1987.

[5] S. K. Nayar, "Shape from Real-time Range Camera", *Proceeding of IEEE computer Society Conference on Computer Vision and Pattern Recognition*, San Diego, California, June 1992.

[6] M. Subbarao, T. S. Choi, and A. Nikzad, "Focusing Techniques", *Journal of Optical Engineering*, pp. 2824-2836 Nov. 1993.

[7] M. Subbarao and T. Wei, "Depth form Defocus and Rapid Autofocusing: A practical Approach", *Proceeding of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Champaign, Illinois, June 1992.

[8] M. Subbarao, and G. Surya, "Application of Spatial-domain Convolution/Deconvolution Transform for Determining Distance from Image Defocus", *SPIE Conference OE/Technology'92* , Vol. 1822, pp. 159-167, Boston, Nov. 1992.

[9] M. Subbarao and G. Surya, "Depth from Defocus: A Spatial Domain Approach", *International Journal of Computer Vision*, Vol. 13, No. 3, pp. 271-294, 1994.

[10] M. Subbarao, G. Surya, and T. Wei, "Focused Image Recovery from Two Defocused Images Recorded with Different Camera Settings", *Proc. IEEE Computer Society conference on Computer Vision and Pattern Recognition*, pp. 786-791, June 1994,

[11] M. Subbarao, Y. Kang, T. Xue, "Shift-variant image deblurring for machine vision: one-dimensional blur", *Proc. SPIE* vol. 7432, 2009.

[12] A. P. Pentland, "A New Sense for Depth of Field", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. PAMI-9, No. 4, pp. 523-531, 1987.

[13] A. P. Pentland, T. Darrell, M. Turk, and W. Huang, "A simple real-time range camera", *IEEE conference on Computer Vision and Pattern Recognition*, pp. 256-261, 1989.

[14] T. Xian, "Three-Dimensional Modeling and Autofocusing Technology for New Generation Digital Cameras", Ph. D. thesis, Dept. of Electrical and Computer Engineering, SUNY at Stony Brook, May 2006.

[15] M. Subbarao, "Parallel depth recovery by changing camera parameters", *Second International Conference on Computer Vision*, pp. 149-155, Florida, Dec. 1988.

[16] M. Subbarao, "Efficient depth recovery through inverse optics", Machine Vision for Inspection and Measurement, Academic press, Boston, 1989, pp. 101-126.

[17] P. Favaro, and S. Soatto, "A geometric approach to shape from defocus", IEEE Trans. on PAMI, Vol. 27, Issue 3, pp. 406-417, March 2005.

[18] S. Chaudhuri and A. N. Rajagopalan, "Depth from Defocus: A Real Aperture Imaging Approach", Springer-Verlag, New York, 1999.

[19] T. Wei, "Three-Dimensional Machine Vision using Image Defocus", Ph.D. Thesis, Dept. of Electrical Engineering, SUNY at Stony Brook, Dec. 1994.

[20] G. Surya, "Three-Dimensional Scene Recovery from Image Defocus", Ph.D. Thesis, Dept. of Electrical Engineering, SUNY at Stony Brook, Dec. 1994.

[21] P. C. Hansen, J. G. Nagy, and D.P. O'Leary, "Deblurring Images; Matrices, Spectra, and Filtering", SIAM, 2006.

[22] J. Kamm and J. G. Nagy, "Kronecker Product and SVD Approximations in Image Restoration", Linear Algebra Appl., 284 , pp. 177-192, 1998.

[23] J. Kamm and J. G. Nagy, "Kronecker Product and SVD Approximations for Separable Spatially Variant Blurs", Advanced Signal Processing Algorithms, Architectures, and Implementations VIII, Franklin T. Luk, Editor, 3461, pp. 358-369, 1998.

[24] J. G. Nagy and D. P. O'Leary, "Restoring Images Degraded by Spatially-Variant Blur", SIAM Journal on Scientific Computing, 19 , pp. 1063-1082, 1998.

[25] M. Subbarao, T. Wei, and G. Surya, "Focused Image Recovery from Two Defocused Images Recorded with Different Camera Settings", *IEEE Transactions on Image Processing* , Vol. 4, No. 12, Dec. 1995.

[26] A. K. Jain, Fundamentals of Digital Image Processing, Section 8.9, pages 299 to 30, Prentice-Hall, Inc., 1989, ISBN 0-13-336165-9.

[27] M. Subbarao, "*Rao Transforms: A New Approach to Integral and Differential Equations*", Second Edition, June 2007, self-published book, 130 pages, U.S. Copyright Registration No. TX 6-591-547, June 29, 2007, (First Edition U.S. Copyright Registration No. TX 6-195-821, June 1, 2005), can be purchased online at http://www.integralresearch.net.

[28] M. Subbarao, U.S. Patent Application No. 11/235724, Filed on 9/26/05, U.S. Patent Application No. 11/242191, Filed on 10/03/2005, and U.S. Patent Application No. 11/450024, 6/10/2006. See http://www.integralresearch.net.

[29] P. Meer and I. Weiss. "Smoothed differentiation filters for images". *Journal of Visual Communication and Image Representation*, 3:58.72, 1992.

[30] M. Subbarao and N, Gurumoorthy,"Depth Recovery from Blurred Edges", *Proc. IEEE Conf. CVPR*, pp. 498-503. 1988.

[31] P. Grossman,"Depth from Focus", *Pattern Recognition Letter*, 5 pp. 498-503, 1987

[32] M. Watanabe and S. Nayar, "Rational filters for passive depth from defocus", Tech. Rep. CUCS-035-95, Dept. of Computer Science, Columbia University, Sept. 1995.

[33] M. Watanabe and S. Nayar, "Rational filters for passive depth from defocus", *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pp 431-438, 1996.

[34] S. K. Nayar and Y. Nakagawa. "Shape from focus: An effective approach for rough surfaces", *Intl Conf. on Robotics and Automation*, pp. 218-225, 1990.

[35] Y. Xiong, and S. A. Shafer," Depth from focusing and defocusing", *Proc. IEEE Intl. Conf. on Computer Vision and Pattern Recognition*, New york, USA, 1993.

[36] Y. Xiong and S. A. Shafer, " Moment filters for high precision computation of focus and stereo", *Proc. of Intl Conf. on Robotics and Automation*, pp. 108-113, Aug. 1995.

[37] J. Ens and P. Lawrence, "A matrix based method for determining depth from focus", *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 600-609, June 1991.

[38] M. Gokstorp, "Computing depth from out-of-focus blur using a local frequency representation", *Proc. of Intl Conf. on Patt. Recog.*, pp. 600-609, Oct. 1994.

[39] W. N. Klarquist, W. S. Geisler, and A. C. Boviki, "Maximum-Likelihood Depth-from-Defocus for Active Vision", Proc. of the International Conference on Intelligent Robots and Systems, 1995.

[40] Y. Kang, X. Tu, S. Dutta, and M. Subbarao, "Robust Depth-from-Defocus for Autofocusing in the Presence of Image Shifts", *SPIE Conference OE/Technology'08*, San Diego, CA, Aug. 2008.

[41] M. Subbarao, Y. Kang, S. Dutta, and X. Tu , "Computational Algorithm and Analysis of A New Loclaized and Efficeient Approach to Shift-variant Image Deblurring", *2008 IEEE International Conference on Image Processing*, San Diego, CA, Oct. 2008.

[42] X. Tu, M. Subbarao, and Y. Kang, "Computational Algorithm and Analysis of A New Loclaized and Efficeient Approach to Shift-variant Image Deblurring", *2008 IEEE International Conference on Image Processing*, San Diego, CA, Oct. 2008.

[43] Hansen, P.C., 1998, "Rank-deficient and Discrete ill-posed problems", SIAM.

[44] C. T. Chen,"Linear system theory and design", Oxford Univ. Press, 1999.

[45] H. J. Trussel, and B. R. Hunt, "Image Restoration of Space-Variant Blurs by Sectioned Methods", *IEEE Trans. Acoustics, Speech Signal Process.* 26, pp. 608-609, 1978.

[46] A. Rosenfeld, and A. C. Kak, "Digital picture processing", Academic Press, 1982.

[47] M. I. Sezan, "An overview of convex projections theory and its applications to image recovery problems", *Ulrramicroscopy*, No. 40, pp. 5547, 1992.

[48] H. J . Trussell and S. Fogel, "Identification and Restoration of Spatially Variant Motion Blurs in Sequential Images", *IEEE Trans. Image processing*, Vol. 1, No. 1, Jan. 1992.

[49] M. I. Sezan and H. J. Trussell, "Prototype image constraints for set - Theoretic image restoration", *IEEE Trans. Signal processing*. Vol. 39, pp. 2275-2285, Oct. 1991.

[50] H. J. Trussell, "Convergence criteria for iterative restoration methods", *IEEE Trans. Acoust. Speech Signcil Processing*, Vol. ASSP-31, pp. 129-136, 1983.

[51] S. R. McNown and B. R. Hunt, "Approximate Shift-Invariance by Warping Shift-Variant Systems", *Proc. SPIE, Image Reconstruction and Restoration*, 2302, pp. 156-167, 1994.

[52] R. L. Lagendijk and J. Biemond, "Block-adaptive Image Identification and Restoration", *Acoustics, Speech, and Signal Processing, ICASSP-91.*, 1991 International Conference

[53] T. P. Costello and W. B. Mikhael, "Efficient restoration of space-variant blurs from physical optics by sectioning with modified Wiener filtering", *Digital Signal Processing, elsevier*, Vol. 13, Issue 1, pp. 1-22, Jan. 2003.

[54] A. Sawchuk, "Space-variant image restoration by coordinate transformations", *J. Opt. Soc. Am.*, 64, pp. 138-144, 1974.

[55] D. L. Angwin, "Image restoration using reduced order models", Ph.D. thesis, Rensselaer Polytechnic Inst., Dept. ECSE, 1989.

[56] A. F. Boden, D. C. Redding, R. J. Hanisch, and J. Mo, "Massively parallel spatially variant maximum likelihood restoration of Hubble Space Telescope imagery", J. Opt. Soc. Am. A 13, pp. 1537-1545, 1996.

[57] A. Sawchuk, "Space-Variant Image Motion Degradation and Restoration", *Proc. IEEE*, Vol.60, No.7, July 1972.

[58] G. M. Robbins and T. S. Huang, "Inverse Filtering for Linear Shift-Variant Imaging System", *Proc. IEEE*. Vol.60, No.7, July 1972.

[59] M. K. Ozkan, A. M. Tekalp, and M. I. Sezan. "POCS-based restoration of space-varying blurred images", *IEEE Trans. Image Process.* 3, pp. 450-454, 1994.

[60] N. S. Subotic and B. E. A. Saleh, "Optical Time-Variant Processing of Signals in the Mixed Time-Frequency Domain", Opt. Commun. 52, 4, 1984.

[61] B. E. A. Saleh and N. S. Subotic, "Time-Variant Filtering of Signals in the Mixed Time-Frequency Domain", *IEEE Trans. ASSP*, Vol. ASSP-33, No. 6, Dec. 1985.

[62] L. D. Jacobson and H. Wechsler, "Joint Spatial/Spatial Frequency Representations", Signal Proc. 14, 37, 1988.

[63] J. Bardsley, S. Jefferies, J. Nagy, and R. Plemmons, "Blind Iterative Restoration of Images with Spatially-Varying Blur", *Optics Express*, 14, pp. 1767-1782, 2006.

[64] R. Vio, J. Nagy, and W. Wams, "Multiple-image Composition and Deblurring with Spatially-variant PSFs", *Astron. Astrophys.*, 434, pp. 795-800, 2005.

[65] H. W. Engl, M. Hanke, and A. Neubauer, "Regularization of Inverse Problems". Kluwer Academic Publishers, Dordrecht, 2000.

[66] E. S. Angel and A. K. Jain, "Restoration of images degraded by spatially varying point spread functions by a conjugate gradient method", *Appl. Opt.* 17, 2186-2190, 1978.

[67] A. Rosenfeld, and A. C. Kak, "Digital picture processing", Academic Press, 1982.

[68] G. H. Golub and C. F. Van Loan."Matrix Computations", Johns Hopkins Univ. Press, 1996.

[69] J. G. Proakis, and D. G. Manolakis,"Digital signal processing", Prentice Hall, 1996.

[70] W.J. Smith, "Modern Optical Engineering", McGraw.Hill, New York, 1990.

[71] T. P. Costello and W. B. Mikhael, "One-Dimensional Comparison of Wiener Filtering and Richardson-Lucy Methods for Sectioned Restoration of Space-Variant Digital Images", *Trans. IEEE. Circuit and System: Funcdamental Theory and Applications*, vol. 49, No. 4, April 2002.

[72] T. P. Costello and W. B. Mikhael, "Restoration of digital images with known space-variant blurs from conventional optical systems", *Pro. SPIE, AeroSense*, Orlando, FL, Apr. 1999.

[73] H. C. Andrews, C. L. Patterson,"Singular value decompositions and digital image processing", *IEEE Trans. ASSP*, 1976.

[74] E. B. Moody, "Discrete orthogonal polynomial restoration of images degraded by spatially varying point spread functions", *Image Processing*, Pro. ICIP-94., Nov 1994, vol 2, pp. 680-684, 1994.

[75] J. L. Roberts. "Quadrature correlation pulse detector". US Patent No. 4038540, July 1977. Assignee: Honeywell Inc.

[76] W. P. Harthill and J. L. Roberts, "Quadrature correlation phase determining apparatus". US Patent No. 4071821, January 1978. Assignee: Honeywell Inc.

[77] L. B. Lucy, "An Iterative Technique for the Rectification of observed distrurbance", Astron. J. 79, 1974.

[78] W. H. Richardson, "Bayesian-based iterative method of image restoration", J. Opt. Soc. Am. Vol. 62, 1974.

[79] M. A. Bueno-Ibarra, J. Alvarez-Borrego, L. Acho and M. C. Chavez-Sanchez, "Fast autofocus algorithm for automated microscopes", *Optical Engineering*, 44(6), 063601, June 2005.

[80] A. N. Tikhonov and V. Y. Arsenin, "Solutions of Ill-Posed Problems", Winston, New York, 1977.

[81] P. C. Hansen,"Analysis of discrete ill-posed problems by means fo the L-curve", *SIAM Rev.*, 34, 1992.

[82] G. H. Golub and W. Kahan, "Calculating the singular values and pseudo-inverse of a matrix", J. Society for Industrial and Applied Mathematics: Series B, Numerical Analysis 2 (2): 205-224, 1965.

[83] G. Strang. "Introduction to Linear Algebra", Wellesley-Cambridge Press, 1998.

[84] P. A. Devijver and J. Kittler, "Pattern Recognition: A Statistical Approach", Prentice-Hall, 1982.