

# Parameters Estimation For Image Restoration

Ratnakar Dash



Department of Computer Science and Engineering  
National Institute of Technology Rourkela  
Rourkela – 769 008, India

# Parameters Estimation For Image Restoration

*Dissertation submitted in partial fulfillment of the requirements for the degree of*

Doctor of Philosophy

*in*

Computer Science and Engineering

*by*

Ratnakar Dash

(Roll- 507CS002)

*under the guidance of*

Prof. Banshidhar Majhi



Department of Computer Science and Engineering  
National Institute of Technology Rourkela  
Rourkela, Odisha, 769 008, India

March 2012

*dedicated to my Parents...*



Department of Computer Science and Engineering  
**National Institute of Technology Rourkela**  
Rourkela-769 008, Odisha, India.

**Dr. Banshidhar Majhi**  
Professor

March 6, 2012

## Certificate

This is to certify that the work in the thesis entitled ***Parameters Estimation For Image Restoration*** by ***Ratnakar Dash*** is a record of an original research work carried out by him under my supervision and guidance in partial fulfillment of the requirements for the award of the degree of Doctor of Philosophy in Computer Science and Engineering in the department of Computer Science and Engineering, National Institute of Technology Rourkela. Neither this thesis nor any part of it has been submitted for any degree or academic award elsewhere.

***Banshidhar Majhi***

# Acknowledgement

This dissertation, though an individual work, has benefited in various ways from several people. Whilst it would be simple to name them all, it would not be easy to thank them enough.

The enthusiastic guidance and support of *Prof. Banshidhar Majhi* inspired me to stretch beyond my limits. His profound insight has guided my thinking to improve the final product. My solemnest gratefulness to him.

I am also grateful to *Prof. Pankaj Kumar Sa* for his ceaseless support throughout my research work. My sincere thanks to *Prof. Rameswar Baliarsigh* for his continuous encouragement and invaluable advice.

It is indeed a privilege to be associated with people like *Prof. S. K. Rath, Prof. S.K.Jena, Prof. D. P. Mohapatra, Prof. A. K. Turuk, Prof. S.Chinara* and *Prof. B. D. Sahoo*. They have made available their support in a number of ways.

Many thanks to my comrades and fellow research colleagues. It gives me a sense of happiness to be with you all. Special thanks to *Arunanshu, Saroj, Hunny, Ashis, Sunita* whose involvement gave a new breath to my research.

Finally, my heartfelt thanks to my wife *Smita* for her unconditional love and support. Words fail me to express my gratitude to my beloved parents who sacrificed their comfort for my betterment.

*Ratnakar Dash*

# Abstract

Image degradation generally occurs due to transmission channel error, camera mis-focus, atmospheric turbulence, relative object-camera motion, etc. Such degradations are unavoidable while a scene is captured through a camera. As degraded images are having less scientific values, restoration of such images is extremely essential in many practical applications.

In this thesis, attempts have been made to recover images from their degraded observations. Various degradations including, out-of-focus blur, motion blur, atmospheric turbulence blur along with Gaussian noise are considered. Basically image restoration schemes are based on classical, regularisation parameter estimation and PSF estimation. In this thesis, five different contributions have been made based on various aspects of restoration. Four of them deal with spatial invariant degradation and in one of the approach we attempt for removal of spatial variant degradation.

Two different schemes are proposed to estimate the motion blur parameters. Two dimensional Gabor filter has been used to calculate the direction of the blur. Radial basis function neural network (RBFNN) has been utilised to find the length of the blur. Subsequently, Wiener filter has been used to restore the images. Noise robustness of the proposed scheme is tested with different noise strengths.

The blur parameter estimation problem is modelled as a pattern classification problem and is solved using support vector machine (SVM). The length parameter of motion blur and sigma ( $\sigma$ ) parameter of Gaussian blur are identified through multi-class SVM.

Support vector regression (SVR) has been utilised to obtain a true mapping of the images from the observed noisy blurred image. The parameters in SVR play a key role in SVR performance and these are optimised through particle swarm optimisation (PSO) technique. The optimised SVR model is used to restore the noisy blurred images.

Blur in the presence of noise makes the restoration problem ill-conditioned. The regularisation parameter required for restoration of noisy blurred image is discussed and for the purpose, a global optimisation scheme namely PSO is utilised

to minimise the cost function of generalised cross validation (GCV) measure, which is dependent on regularisation parameter. This eliminates the problem of falling into a local minima. The scheme adapts to degradations due to motion and out-of-focus blur, associated with noise of varying strengths.

In another contribution, an attempt has been made to restore images degraded due to rotational motion. Such situation is considered as spatial variant blur and handled by considering this as a combination of a number of spatial invariant blurs. The proposed scheme divides the blurred image into a number of images using elliptical path modelling. Each image is deblurred separately using Wiener filter and finally integrated to construct the whole image.

Each model is studied separately, and experiments are conducted to evaluate their performances. The visual as well as the peak signal to noise ratio (PSNR in dB) of restored images are compared with competent recent schemes.

***Keywords:*** Image restoration, out-of-focus blur, motion blur, SVM, SVR, multi-class SVM, blind image deconvolution, regularisation, spatial variant blur, point spread function.

# Contents

<b>Certificate</b>	<b>iii</b>
<b>Acknowledgement</b>	<b>iv</b>
<b>Abstract</b>	<b>v</b>
<b>List of Figures</b>	<b>vii</b>
<b>List of Tables</b>	<b>viii</b>
<b>1 Introduction</b>	<b>2</b>
1.1 Image Degradation Model . . . . .	4
1.1.1 Discrete Convolution Model for Image Degradation . . . . .	5
1.1.2 Blur Model . . . . .	6
1.1.3 Noise Model . . . . .	9
1.2 Image Restoration . . . . .	10
1.3 Literature Review . . . . .	13
1.3.1 Common Classical Image Restoration Techniques . . . . .	13
1.3.2 Blind Image Restoration Techniques . . . . .	17
1.3.3 Regularisation Approaches . . . . .	28
1.4 Motivation . . . . .	30
1.5 Thesis Layout . . . . .	30
<b>2 Motion Blur Parameters Estimation using Gabor filter and RBFNN</b>	<b>34</b>
2.1 Motion Blur Model . . . . .	35
2.2 Angle Estimation using Gabor Filter . . . . .	35
2.3 Length Estimation using RBFNN . . . . .	38
2.3.1 Radial Basis Function Neural Network . . . . .	38
2.4 Noise Robustness of the proposed method . . . . .	40
2.5 Simulation Results and Discussions . . . . .	41

2.6	Summary	47
<b>3</b>	<b>Blur Parameter Identification using SVM</b>	<b>52</b>
3.1	Support Vector Machine	52
3.1.1	An Overview of Multi-class SVM Approaches	54
3.2	SVM based blur identification	56
3.2.1	Blur Classification	58
3.3	Simulation Results and Discussion	59
3.4	Summary	64
<b>4</b>	<b>PSO based SVR for Blind Image Restoration</b>	<b>67</b>
4.1	Support Vector Regression	68
4.2	Influence of the parameters on the performance of SVR	70
4.3	Proposed PSO based SVR for Blind Image Restoration	71
4.3.1	Particle Swarm Optimisation	71
4.3.2	Parameter Optimisation of SVR	72
4.4	Experimental Results	72
4.5	Summary	77
<b>5</b>	<b>PSO Based Regularisation Parameter Estimation</b>	<b>80</b>
5.1	Regularised Image Restoration	80
5.2	Generalised Cross Validation	86
5.3	Regularisation Parameter Estimation using PSO	87
5.4	Results and Discussion	89
5.5	Summary	91
<b>6</b>	<b>Rotational Motion Deblurring using Elliptical Modelling</b>	<b>96</b>
6.1	Elliptical Motion Blur Analysis	98
6.2	Proposed Deblurring Method	100
6.2.1	Missing Pixel Interpolation	101
6.3	Simulation Results and Discussion	101
6.4	Summary	103
<b>7</b>	<b>Conclusions and Future Work</b>	<b>110</b>
	<b>Bibliography</b>	<b>113</b>
	<b>Dissemination of Work</b>	<b>124</b>

# List of Figures

1.1	Image degradation restoration model. . . . .	4
1.2	(a) PSF of motion blur. (b) Its frequency response. . . . .	8
1.3	(a) PSF of out-of-focus blur. (b) Its frequency response. . . . .	8
1.4	(a) PSF of Gaussian blur. (b) Its frequency response. . . . .	9
1.5	Blur identification method based on frequency domain nulls. . . . .	20
1.6	NASRIF algorithm. . . . .	23
2.1	(a) Original <i>Lena</i> image. (b) Blurred <i>Lena</i> image with $L = 20$ and $\theta = 30^\circ$ . (c) Spectrum of the blurred <i>Lena</i> image. (d) Frequency plot of PSF. . . . .	36
2.2	(a) Gabor filter mask with $\phi = 30^\circ$ . (b) Gabor filter mask with $\phi = 45^\circ$ . . . . .	37
2.3	Relation between blur length and Fourier feature for different images. . . . .	39
2.4	Radial basis function neural network (RBFNN) . . . . .	40
2.5	(a) <i>Cameraman</i> image degraded by motion blur with $L = 10$ , $\theta = 45^\circ$ and Gaussian noise with SNR = 25 dB. (b) Power spectrum of the blurred <i>Cameraman</i> image in (a). (c) <i>Cameraman</i> image degraded by motion blur with $L = 10$ , $\theta = 45^\circ$ and Gaussian noise with SNR = 40 dB. (d) Power spectrum of the blurred <i>Cameraman</i> image in (b). . . . .	41
2.6	Relationship between blur length and Fourier feature for noisy images. . . . .	42
2.7	Convergence characteristics of RBFNN. . . . .	43
2.8	(a) Original <i>Lena</i> image. (b) Blurred image with blur length $L = 15$ and $\theta = 30^\circ$ . (c) Restored using ML method (PSNR = 17.711 dB). (d) Restored after parameter estimation (PSNR = 24.4186 dB). . . . .	45
2.9	(a) Original <i>Tree</i> image. (b) Blurred image with blur length $L = 20$ and $\theta = 45^\circ$ . (c) Restored with ML method (PSNR = 17.047 dB). (d) Restored after parameter estimation (PSNR = 24.7106 dB). . . . .	46

2.10	(a) Original <i>Checkerboard</i> image. (b) Blurred image with blur length $L = 10$ and $\theta = 40^\circ$ . (c) Restored with ML method (PSNR = 17.33 dB). (d) Restored after parameter estimation (PSNR = 26.21 dB). . . . .	47
2.11	(a) Original <i>Stik</i> image, (b) Blurred image with blur length $L = 12$ and $\theta = 40^\circ$ , (c) Restored with ML method (PSNR = 19.53 dB), (d) Restored after parameter estimation (PSNR = 28.41 dB). . . . .	48
2.12	(a) Noisy blurred <i>Cameraman</i> image with $L = 30$ and $\theta = 40^\circ$ and Gaussian noise (SNR = 25 dB). (b) Noisy blurred <i>Cameraman</i> image with $L = 10$ and $\theta = 45^\circ$ and Gaussian noise (SNR = 30 dB). (c) Restoration results of (a) after parameter estimation (PSNR = 23.18 dB). (d) Restoration results of (b) after parameter estimation (PSNR = 25.21 dB). . . . .	49
2.13	(a) Noisy blurred <i>Vase</i> image with $L = 40$ and $\theta = 60^\circ$ and Gaussian noise (SNR = 25 dB). (b) Noisy blurred <i>Vase</i> image with $L = 40$ and $\theta = 60^\circ$ and Gaussian noise (SNR = 30 dB). (c) Restoration results of (a) after parameter estimation (PSNR = 20.58 dB). (d) Restoration results of (b) after parameter estimation (PSNR = 22.19 dB). . . . .	50
2.14	(a) Noisy blurred <i>Car</i> image. (b) Extracted <i>Number plate</i> image. (c) Restored with ML method (d) Restoration result of <i>Number plate</i> image after parameter estimation. . . . .	50
3.1	Optimal hyperplane classifying a two class problem . . . . .	54
3.2	Behaviour of a high frequency portion of <i>Cameraman</i> image under horizontal motion blur of different length:(a) True image (b) $L = 5$ (c) $L = 10$ . (d) $L = 20$ (e) True high frequency portion. (f) $L = 5$ (g) $L = 10$ (h) $L = 20$ . . . . .	57
3.3	Variance curve for different blur length for different images. . . . .	57
3.4	Variance plot of <i>Cameraman</i> image of size $128 \times 128$ degraded with motion blur of $L = 10$ . . . . .	58
3.5	Binary decision tree SVM for blur classification. . . . .	59
3.6	Standard images used for testing. . . . .	60
3.7	Restoration results for <i>Lena</i> image: (a) True image. (b) Motion blurred ( $L = 9$ ). (c) Restored after parameter estimation using RBFNN. (d) Restored after parameter estimation using multi-class SVM. . . . .	62

4.1	The soft margin loss setting for a linear SVM. . . . .	69
4.2	Flow chart of the proposed scheme for SVR parameter optimisation. . . . .	73
4.3	Restoration of <i>Lena</i> Image: (a) Original <i>Lena</i> image. (b) Motion Blurred <i>Lena</i> image (SNR = 40 dB). (c) Restored with PCA. (d) Restored with Maximum Likelihood. (e) Restored with SVR without parameter optimisation ( $C = 1$ ). (f) Restored with PSO based <i>Lena</i> SVR model . . . . .	75
4.4	Restoration of <i>Pepper</i> Image: (a) Original <i>Pepper</i> image. (b) Out-of-focus blurred <i>Pepper</i> image (SNR = 40 dB). (c) Restored with PCA. (d) Restored with Maximum Likelihood. (e) Restored with SVR without parameter optimisation ( $C = 1$ ). (f) Restored with PSO based <i>Lena</i> SVR model. . . . .	76
4.5	Restoration of <i>Canon</i> Image: (a) Blurred and noisy <i>Canon</i> image. (b) Restored with PCA. (c) Restored with Maximum Likelihood. (d) Restored with SVR without parameter optimisation ( $C = 1$ ). (e) Restored with PSO based <i>Lena</i> SVR model. . . . .	77
4.6	Restoration of <i>key</i> Image: (a) Blurred and noisy <i>Key</i> image. (b) Restored with PCA. (c) Restored with maximum likelihood. (d) Restored with SVR without parameter optimisation ( $C = 1$ ). (e) Restored with PSO based <i>Lena</i> SVR model. . . . .	78
5.1	Variation of MSE for different $\alpha$ at different noise conditions for <i>Lena</i> image. . . . .	82
5.2	Effect of $\alpha$ in regularised restoration: (a) Original <i>Cameraman</i> Image. (b) Noisy blurred <i>Cameraman</i> image with SNR = 40 dB. (c) Restored with inverse filter $\alpha = 0$ . (d) (e) (f) Restored with regularisation filter with ( $\alpha = 0.01$ ), ( $\alpha = 0.004$ ), ( $\alpha = 0.2$ ). . . . .	83
5.3	Effect of $\alpha$ in regularised restoration: (a) Original <i>Lena</i> Image. (b) Noisy Blurred <i>Lena</i> image with SNR = 30 dB. (c) Restored with inverse filter $\alpha = 0$ , (d) (e) (f) Restored with regularisation filter with ( $\alpha = 0.01$ ), ( $\alpha = 0.004$ ), ( $\alpha = 0.2$ ). . . . .	84
5.4	Effect of regularisation operator on restoration: (a) Noisy Blurred <i>Cameraman</i> image with SNR = 40 dB. (b) Restored image without any filtering i.e., $Q = 1$ $\alpha = 0.01$ . (c) Restored image with $Q$ as Laplacian operator and $\alpha = 0.01$ . . . . .	85
5.5	GCV function for degraded <i>Cameraman</i> image. . . . .	88
5.6	GCV function for degraded <i>Lena</i> image. . . . .	89

5.7	Restoration of <i>Cameraman</i> Image: (a) Original <i>Cameraman</i> Image. (b) Motion blurred and noisy image (SNR = 30 dB). (c) Restored with GCV ( $\alpha = 0.004$ ). (d) Restored after Arnoldi regularisation ( $\alpha = 0.01$ ). (e) Restored after PSO Based GCV minimisation ( $\alpha = 0.053$ ). . . . .	92
5.8	Restoration of ABC Image: (a) Original ABC image. (b) out-of-focus Blurred ABC image (SNR = 30 dB). (c) Restored with GCV ( $\alpha = 0.015$ ). (d) Restored after Arnoldi regularisation ( $\alpha = 0.054$ ). (e) Restored after PSO based GCV minimisation ( $\alpha = 0.093$ ). . . . .	93
5.9	Restoration of <i>Lena</i> image: (a) Original <i>Lena</i> image. (b) Out-of-focus blurred <i>Lena</i> image (SNR = 60 dB). (c) Restored with GCV ( $\alpha = 0.048$ ). (d) Restored after Arnoldi regularisation ( $\alpha = 0.006$ ). (e) Restored after PSO based GCV minimisation ( $\alpha = 0.004$ ). . . . .	94
6.1	Spatial variant blur model. . . . .	98
6.2	Elliptical motion blur model. . . . .	99
6.3	Missing pixel problem. . . . .	101
6.4	Restored image corrupted my missing pixels: (a) Restored image with out interpolation. (b) Restored image after interpolation. . . .	102
6.5	Restoration of <i>Lena</i> image: (a) True image. (b) Circularly blurred. (c) Elliptically blurred. (d) Restored image of (b). (e) Restored image of (c). . . . .	104
6.6	Restoration of <i>Stik</i> image: (a) True image. (b) Circularly blurred. (c) Elliptically blurred. (d) Restored image of (b). (e) Restored image of (c). . . . .	105
6.7	Restoration of <i>Vase</i> image: (a) True image. (b) Circularly blurred. (c) Elliptically blurred. (d) Restored image of (b). (e) Restored image of (c). . . . .	106
6.8	Restoration of <i>Chekerboard</i> image: (a) True image. (b) Circularly blurred. (c) Elliptically blurred. (d) Restored image of (b). (e) Restored image of (c). . . . .	107
6.9	(a) Variation of PSNR (dB) with respect to exposure time. (b) Variation of PSNR (dB) with respect to rotational speed. . . . .	108

# List of Tables

2.1	Estimated blur angle $\theta$ in noise free Situation. . . . .	44
2.2	Time (in sec) required to estimate blur angle ( $\theta$ ) using different methods. . . . .	44
2.3	Estimated blur length $L$ in noise free situation. . . . .	44
2.4	Estimated blur length $L$ and blur angle $\theta$ in noisy situation. . . . .	46
3.1	Blur identification performance of <i>Cameraman</i> multi-class SVM model for <i>Cameraman</i> test image. . . . .	61
3.2	Blur identification performance of <i>Cameraman</i> multi-class SVM model for <i>Lena</i> test image. . . . .	61
3.3	Blur identification performance of <i>Pepper</i> multi-class SVM model for Gaussian blurred <i>Pepper</i> test image. . . . .	63
3.4	Blur identification performance of <i>Pepper</i> multi-class SVM model for Gaussian blurred <i>Cameraman</i> test image. . . . .	63
3.5	Estimated blur length $L$ in noise free situations . . . . .	64
3.6	Blur length $L$ estimation in noisy situation . . . . .	64
4.1	PSNR (dB) comparison of restored <i>Lena</i> image using <i>Lena</i> SVR model and other schemes for different degradations. . . . .	75
4.2	PSNR (in dB) comparison of restored <i>Pepper</i> image using <i>Lena</i> SVR model and other schemes for different degradations. . . . .	76
4.3	PSNR (in dB) comparison of restored <i>Cameraman</i> image using <i>Pepper</i> SVR model and other schemes for different degradations. . . . .	77
5.1	PSNR of restored out-of-focus blurred and noisy images . . . . .	91
5.2	PSNR of restored out-of-focus blurred and noisy images . . . . .	91
6.1	PSNR (in dB) of restored images. . . . .	103
+		

# Chapter 1

## Introduction

# Chapter 1

## Introduction

The human visual system along with neural system enable a person to classify and recognise the objects. It processes the visual information and sends it to the human brain for identification. It allows us to observe our environment and work accordingly. The human visual system is highly complex and has drawn attention of the researchers over the last few decades.

Like human visual system, digital image processing involves the process of acquiring, manipulating, and analysing images through a computer [1, 2]. Images are created by a large number of physical devices, which includes camera, x-ray, electron microscope, ultrasound and are used in variety of applications. A digital image is generally expressed as a function of two-dimensional variable and mathematically represented as,

$$I = f(x, y) \tag{1.1}$$

where  $x, y$  are called spatial coordinates of the pixel location and  $f$  represents the amplitude of an image for a pixel at location  $(x, y)$ . The amplitude is called gray level or intensity [1, 3]. Digital image processing has many advantages in terms of cost, speed, flexibility etc. It has become the dominant method in use due to increasing performance of personal computers. Digital image processing is used in almost every discipline of science & engineering including medical, entertainment [4], and industry, military, civil etc. In each of the applications, the objective is to extract information about the scene being imaged. Depending on the nature of application, image processing can be classified into the following subareas [3].

- (i) Image Enhancement
- (ii) Image Restoration
- (iii) Image Compression
- (iv) Image Segmentation
- (v) Image Understanding.

*Image enhancement* techniques deal with improving the visual appearance of the image so that it will be more pleasing to the human eye. Some of the examples of image enhancement are histogram equalisation, unsharp masking, contrast stretching, etc. On the other hand, *image restoration* accentuates on retracing the original image as close as possible from the degraded observation. The restoration techniques assume a degradation model and design a filter to achieve an approximated version of the original image. The closeness of the restored image towards the true image depends on the accuracy of the model and the designed filter. Both enhancement and restoration techniques try to improve the appearance of the image using a filter. However, enhancement is more subjective in nature and depends on individuals perception [5, 6]. *Image compression* techniques deal with representing an image with least possible bits which can be reconstructed again without losing the intelligibility of the image. In *image understanding*, the content of an image is represented symbolically by using some attributes of the image [3]. The first three areas are lower level tasks. They take images as their input and outputs an image. *Image segmentation* and *understanding* are higher-level tasks in which it takes an image as its input and produces attributes or symbolic representation of the contents of input image as outputs [3]. In this thesis, investigation has been restricted to image restoration.

The rest of the chapter is organised as follows. The degradation model in its integral as well as discrete formulation is defined in Section 1.1. Image restoration applications and its importance are briefed in Section 1.2. Some of the existing literatures are reviewed in Section 1.3. The research motivation and its objective are formally stated in Section 1.4. Finally, Section 1.5 outlines the layout of the thesis.

## 1.1 Image Degradation Model

The degraded image  $g(x, y)$  is obtained by applying the degradation operator  $H$  onto the image  $f(x, y)$  along with the additive noise  $\eta(x, y)$ . The degradation phenomenon is mathematically expressed as,

$$g(x, y) = H[f(x, y)] + \eta(x, y) \quad (1.2)$$

The objective of image restoration is to estimate  $f(x, y)$  from the observed image  $g(x, y)$  using the known value of  $H$  [7, 8]. The overall degradation and restoration model is shown in the Figure 1.1. The operator  $H$  may be linear or nonlinear.

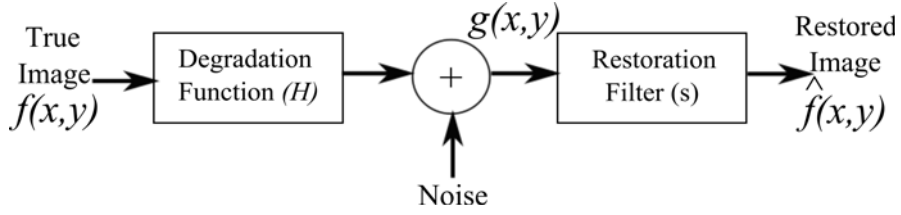


Figure 1.1: Image degradation restoration model.

Mostly, it is assumed to be linear which satisfies the principles of superposition and homogeneity [6]. The operator  $H$  is also considered to be space invariant or position invariant. An operator is said to be space invariant if the response at any point depends on the value at that point but not on the position of the point and is defined mathematically as,

$$H[f(x - \alpha, y - \beta)] = g(x - \alpha, y - \beta) \quad (1.3)$$

for all  $f(x, y)$  and any  $\alpha$  and  $\beta$ . In terms of impulse function  $f(x, y)$  is expressed as,

$$f(x, y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(\alpha, \beta) \delta(x - \alpha, y - \beta) d\alpha d\beta \quad (1.4)$$

Substituting (1.4) in (1.2) we get the observed image  $g(x, y)$  and is expressed as,

$$g(x, y) = H \left[ \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(\alpha, \beta) \delta(x - \alpha, y - \beta) d\alpha d\beta \right] + \eta(x, y) \quad (1.5)$$

As  $f(\alpha, \beta)$  is independent of  $x$  and  $y$ , using the linearity property, the  $g(x, y)$  can be expressed as,

$$g(x, y) = \left[ \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(\alpha, \beta) h(x, \alpha, y, \beta) d\alpha d\beta \right] + \eta(x, y) \quad (1.6)$$

where  $h(x, \alpha, y, \beta) = H[\delta(x - \alpha, y - \beta)]$  is called point spread function (PSF) in the optics. The expression given in (1.6) is called the Fredholm integral of the first kind. Since operator  $H$  is spatial invariant we have,

$$H[\delta(x - \alpha, y - \beta)] = h(x - \alpha, y - \beta) \quad (1.7)$$

and the degraded image is given as,

$$g(x, y) = \left[ \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(\alpha, \beta) h(x - \alpha, y - \beta) d\alpha d\beta \right] + \eta(x, y) \quad (1.8)$$

This expression is called the convolution integral in the continuous variable.

### 1.1.1 Discrete Convolution Model for Image Degradation

Images are generally constructed in two different ways; continuous and discrete. The process of converting a continuous image into a discrete format involves sampling, quantisation, and coding. Sampling should be done according to Nyquist criterion for an accurate reconstruction. The discrete model for a linear degradation caused due to blur and additive noise can be expressed as,

$$g(x, y) = \sum_{\alpha=1}^m \sum_{\beta=1}^n f(\alpha, \beta) h(x, y, \alpha, \beta) + \eta(x, y) \quad (1.9)$$

where  $f(x, y)$  represent the original image of size  $m \times n$  and  $h(x, y)$  represents the PSF of size  $p \times q$ . In the above formulation,  $\eta(x, y)$  is taken as additive noise introduced by the system and is assumed to be zero mean white Gaussian noise. Using spatial invariant property of PSF, the  $g(x, y)$  can be described as,

$$\begin{aligned} g(x, y) &= \sum_{\alpha=1}^m \sum_{\beta=1}^n f(\alpha, \beta) h(x - \alpha, y - \beta) + \eta(x, y) \\ &= f(x, y) * h(x, y) + \eta(x, y) \end{aligned} \quad (1.10)$$

where  $*$  denotes the two dimensional linear convolution. Using linear convolution rule, the size of  $g$  becomes  $M \times N$ , where  $M = m + p - 1$  and  $N = n + q - 1$ . If  $g, h, f, \eta$  are ordered lexicographically by stacking the rows of each matrix into a column vector, then the degraded image can be obtained through matrix multiplication as,

$$g = \mathbf{H}f + \eta \quad (1.11)$$

where  $f \in \mathbb{R}^{mn \times 1}$ ,  $g \in \mathbb{R}^{MN \times 1}$  and  $\mathbf{H} \in \mathbb{R}^{MN \times mn}$ . In the above formulation,  $\mathbf{H}$  becomes a block toeplitz with toeplitz block (BTTB) matrix. For an image of size  $1024 \times 1024$ , the size of  $\mathbf{H}$  would be  $1024^2 \times 1024^2$  which uses a large memory. This will also require to solve a system of 1,048,576 simultaneous linear equations to obtain  $f$  directly. The complexity can be significantly reduced by exploiting the periodic nature of  $\mathbf{H}$ . The linear convolution is converted to circular convolution using the periodicity.

After zero padding to  $g$  and  $f$ , the result of linear convolution is equal to circular convolution. In that process, the matrix  $H$  becomes a block circulant matrix with circulant blocks (BCCB) given as,

$$\mathbf{H} = \begin{bmatrix} \mathbf{H}(0) & \mathbf{H}(N-1) & \dots & \mathbf{H}(1) \\ \mathbf{H}(1) & \mathbf{H}(0) & \dots & \mathbf{H}(2) \\ \mathbf{H}(2) & \mathbf{H}(1) & \dots & \mathbf{H}(3) \\ \dots & \ddots & \ddots & \dots \\ \mathbf{H}(N-1) & \mathbf{H}(N-2) & \dots & \mathbf{H}(0) \end{bmatrix} \quad (1.12)$$

where each sub matrix  $\mathbf{H}(i)$  is a circulant matrix. Computational requirements for the blur operator  $\mathbf{H}$  can be significantly reduced by through these BCCB matrices.

### 1.1.2 Blur Model

Blurs are treated as low pass filters which smoothes out the abrupt changes in the gray level of an image. There are different analytical models used in nature to represent the shift invariant degradation model. The blur also decreases the image contrast by averaging the pixels. An image may be blurred in various ways including motion blur, out-of-focus blur, atmospheric turbulence blur etc. The spatially continuous PSF  $h(x, y)$  of any blur must satisfy the following constraints—

- $h(x, y)$  accepts only non negative values due to physics of underlying image formation process.
- The values in the PSF are real valued because images are real valued.
- The capturing process is modelled as a passive operation on the image data. No energy is absorbed or generated during the image formation process. For spatially continuous blur, it satisfies the following condition,

$$\int_{-\infty}^{\infty} \int_{-\infty}^{\infty} h(x, y) dx dy = 1 \quad (1.13)$$

In the discrete domain, it is expressed as,

$$\sum_{x=0}^{M-1} \sum_{y=0}^{N-1} h(x, y) = 1 \quad (1.14)$$

The characteristics of some common blurs are described below–

### ***Linear Motion Blur***

This blur results either due to object or camera motion during exposure. It is governed by two parameters, namely the length of motion ( $L$ ) and the angle of motion ( $\theta$ ). When a scene to be imaged translates with a relative velocity  $V$  in respect to the camera, the blur length  $L$  in pixels is  $L = VT_{exposure}$  where,  $T_{exposure}$  is the time duration of the exposure. The expression for PSF of the motion blur is given as,

$$h(x, y) = \begin{cases} \frac{1}{L} & \text{if } 0 \leq |x| \leq L \cos \theta; y = L \sin \theta, \\ 0 & \text{otherwise.} \end{cases} \quad (1.15)$$

When the blur angle is zero i.e.  $\theta = 0$ , it is called as horizontal motion blur. The discrete version of PSF is given by,

$$h(m, n; L) = \begin{cases} \frac{1}{L} & \text{if } m = 0, |n| \leq \left\lfloor \frac{L-1}{2} \right\rfloor, \\ \frac{1}{2L} \{ (L-1) - 2 \left\lfloor \frac{L-1}{2} \right\rfloor \} & \text{if } m = 0, |n| \leq \left\lceil \frac{L-1}{2} \right\rceil, \\ 0 & \text{elsewhere.} \end{cases} \quad (1.16)$$

The Figure. 1.2 shows the PSF of motion blur and its spectrum.

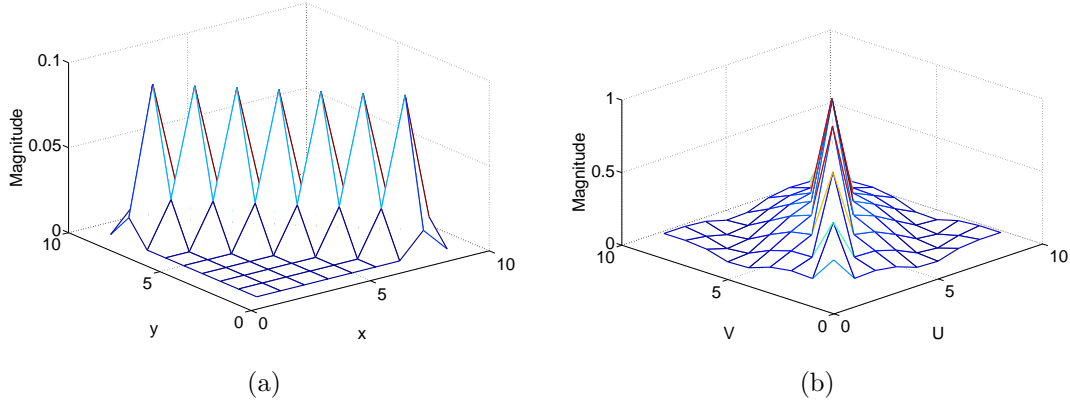


Figure 1.2: (a) PSF of motion blur. (b) Its frequency response.

### ***Out-of-Focus Blur***

When a three dimensional scene is projected onto a two dimensional plane, some part of the scene may not be focused properly. For a circular aperture camera, the image of a point source is a small disk known as circle of confusion. The strength of defocus depends on the focal length and distance between the object and camera. The PSF of the out-of-focus blur is given as,

$$h(x, y) = \begin{cases} \frac{1}{\pi R^2} & \text{if } \sqrt{x^2 + y^2} \leq R, \\ 0 & \text{otherwise.} \end{cases} \quad (1.17)$$

where  $R$  is the radius. The PSF for out-of-focus and its frequency domain plot

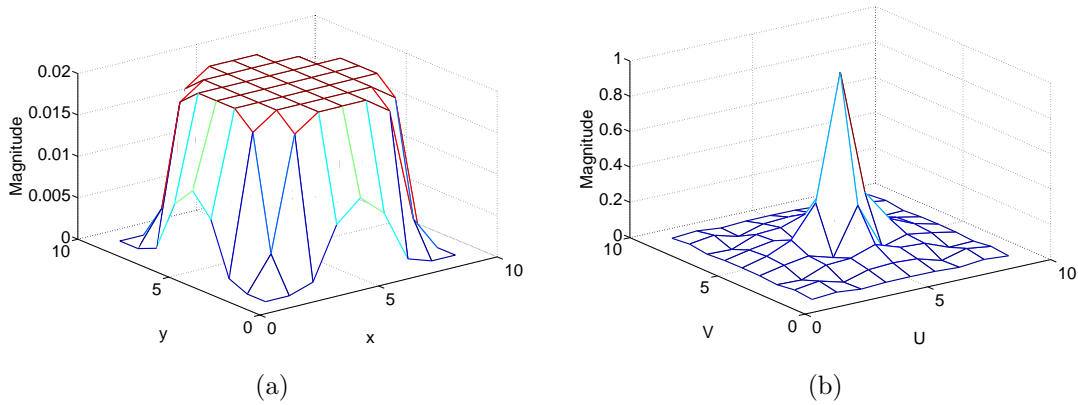


Figure 1.3: (a) PSF of out-of-focus blur. (b) Its frequency response.

are depicted in the Figure. 1.3.

### Atmospheric Turbulence Blur

This blur occurs due to turbulent atmosphere of the earth. It is mainly encountered in remote sensing applications because of variations in the wind velocity. This leads to change in the refractive index of a layer which distorts the image to be observed. It is effectively modelled as a Gaussian PSF with standard deviation  $\sigma$ . The spatial domain and frequency domain plot of the turbulence blur are shown in the Figure. 1.4. The PSF of the turbulence blur is given as,

$$h(x, y) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right) \quad (1.18)$$

Here  $\sigma$  decides the severity of the blur. In this thesis, the above blurs are

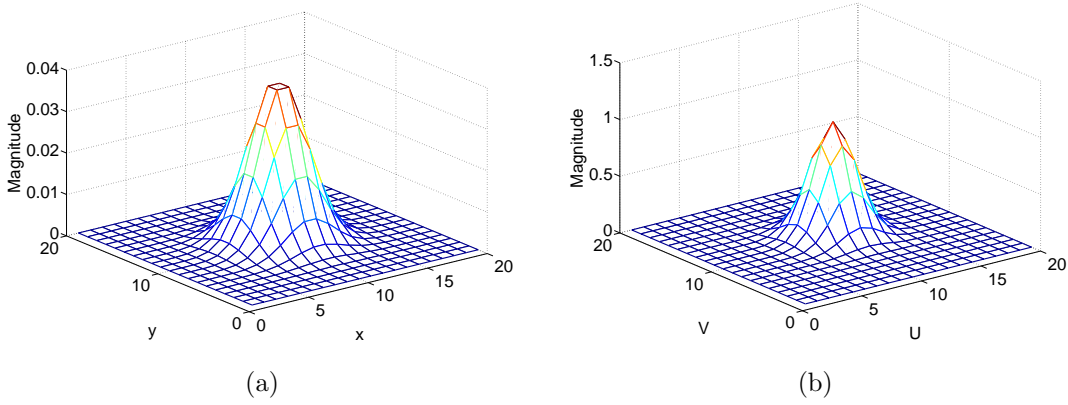


Figure 1.4: (a) PSF of Gaussian blur. (b) Its frequency response.

considered independently and attempts have been made to restore the images from the corresponding degraded images.

### 1.1.3 Noise Model

Images are contaminated by different types of noise. Most common types of noise are impulsive and Gaussian noise, which affect the image at the time of acquisition due to noisy sensors. Noise also contaminates the image during transmission due to channel errors. Although there are different noise models, our investigation confines to dealing with blur in the presence of Gaussian noise which is the most common scenario in practical applications [9]. The Gaussian noise model is described in detail below.

### *Gaussian Noise*

The Gaussian noise is a random variable and is expressed as,

$$\eta_G(x, y) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right). \quad (1.19)$$

It is characterised by its variance  $\sigma^2$ . The noisy image  $f_G(x, y)$  is stated as addition of original  $f(x, y)$  with the noise term. It is given as,

$$f_G(x, y) = f(x, y) + \eta_G(x, y) \quad (1.20)$$

## 1.2 Image Restoration

Due to imperfections in the image formation process and the imaging device, the observed image often represents the degraded version of the original image. The corrections of these imperfections are mandatory in many of the subsequent image processing and vision tasks. Different types of degradations exist in the nature which includes noise, blur, geometrical degradations, illuminations etc. In this thesis, an effort has been made on removing the blur and noise from degraded images. Due to enormous applications of image restoration, researchers have gained interest to work in this area. The research on image restoration started in 1950s with astronomical imaging when scientists of United States of America and Soviet Union were involved in producing images of the Earth and the solar system. The images were degraded versions of the original images due to substandard imaging environment, spinning and the tumbling of the space craft. To retrieve the meaningful information from the degraded images, image restoration techniques were used. It is not a surprise to see that digital image restoration is used in astronomical imaging even today. Ground based imaging systems were also subject to blurring due to change in refractive index of the atmosphere [10].

Image restoration also plays an important role in medical imaging. It has been used to remove film-grain noise in X-ray images, angiography images and additive noise in magnetic resonance images [11–15]. It has applications to quantitative auto radiography (QAR) in which image is obtained by exposing X-ray film to a

radio active specimen. Though image restoration has been successfully applied, but still has a scope for improvement in quality and resolution.

Image restoration has also received attention in media where old movies and picture are corrected in order to obtain a good quality picture which includes removal of scratches from the deteriorated films. Another important application of image restoration is in the field of image and video coding. The techniques used to increase the coding efficiency and to reduce the bit rates of the coded images create blocking artifacts. Image restoration has been successfully used as a post processing step after decompression to eliminate the blocking artifact resulted due to coarse quantisation of transformed coefficients [16–20]. In addition, digital image restoration is used in many other applications. Printing applications use restoration techniques to ensure high-quality halftone reproductions of continuous image [21]. Defense applications may also require restoration such as a guided missile which takes distorted images due to the pressure difference around a camera mounted on the missile. Looking at its wide-spread application areas in almost every field, it finds an important place in this technological world. Thus, even though several suggestions have been made, the field of image restoration still remains an active field of research.

The problem of restoring the original image  $f(x, y)$  requires deconvolution of the PSF  $h(x, y)$  with the observed image  $g(x, y)$ . In most of the situations, PSF is assumed to be known prior to restoration. This is known as classical image restoration. Variety of well known techniques such as inverse filtering, Wiener filtering, least square filtering, recursive Kalman filtering are available [22–26]. Some of the well known classical image restoration techniques are discussed in Section 1.3.1.

In practical situations, PSF is unknown and very little information is available about the original image. Thus, it is required to obtain the original image directly from the observed image using partial information or no information about the blurring process. The process of recovering the true image from the degraded observation without having information about the blurring phenomenon is known as blind deconvolution. But in practice, some partial information should be known for successful image restoration. There are many factors which motivates the use

of blind deconvolution. In many cases, it is costly and dangerous to have prior information about the imaged scene. In some applications like remote sensing and astronomical imaging, it is difficult to model the image because the scenes have never been imaged before. In such cases blur characteristics is also not accurately predicted. The other motivation behind the use of blind deconvolution is that the physical requirements for image quality improvement is unrealisable. For example in space applications, the physical weight of a high resolution camera exceeds practical constraint. In X-ray imaging, the intensity of beam cannot be increased in order to get good image quality because it is hazardous to health. In such situations, it is difficult to measure the PSF of the imaging system and classical image restoration cannot be used. In such situations, blind deconvolution is used for image restoration.

The blind restoration algorithms may broadly be classified into two categories. In one category, PSF is identified first (Apriori Estimation) and then utilised to restore the image using any of the classical restoration technique [27–32]. The second category of algorithms estimate the PSF and restore the image simultaneously (Simultaneous Estimation) [33–37]. In this thesis two apriori methods have been proposed to estimate the blur parameters for subsequent image restoration. Another attempt has been made for simultaneous restoration with parameter estimation. A brief review of different blind restoration techniques have been discussed in Section 1.3.

In image restoration, the performance of the filter depends on the signal to noise ratio of the output. For the purpose of measuring the performance of the algorithms, an objective parameter, the peak signal to noise ratio (PSNR in dB) is often used which is defined as,

$$\text{PSNR (dB)} = 10 \log_{10} \left( \frac{255^2}{\text{MSE}} \right) \quad (1.21)$$

where MSE is defined as,

$$\text{MSE} = \frac{1}{MN} \sum_{x=1}^M \sum_{y=1}^N \left( f(x, y) - \hat{f}(x, y) \right)^2 \quad (1.22)$$

Another similar measure known as, improved signal to noise ratio (ISNR) is often

used that determines how much SNR is improved with respect to the degraded image by the restoration filter.

## 1.3 Literature Review

This section describes a brief review on different classical and blind image restoration techniques. Some techniques assume blur along with Gaussian noise present in the image whereas others assume the presence of blur in the image.

### 1.3.1 Common Classical Image Restoration Techniques

This section assumes that PSF is known prior to restoration. A number of methods exist to remove the blur from the observed image  $g(m, n)$  using a linear filter. The restored image  $\hat{f}$  from a given blurred image is given by

$$\begin{aligned}\hat{f}(m, n) &= g(m, n) * h(m, n) \\ &= \sum_{k=0}^{M-1} \sum_{l=0}^{N-1} g(k, l) h(m-k, n-l)\end{aligned}\quad (1.23)$$

where  $*$  denotes the deconvolution which represents the inverse of the convolution. In the frequency domain, this can be expressed as

$$\hat{F}(u, v) = G(u, v) H(u, v) \quad (1.24)$$

where  $\hat{F}$  denotes the estimated image in spectral domain.  $G(u, v)$  and  $H(u, v)$  are the blurred image and PSF in frequency domain respectively.

#### ***Inverse Filtering***

Inverse filter [1] uses the inverse of the PSF as an impulse response. It is difficult to implement the filter in the image domain. An estimate of the transform of the original image is obtained by dividing the transform of the degraded image  $G(u, v)$  by the degradation function. The division is performed on individual elements. Using the value of  $G(u, v)$

$$\hat{F}(u, v) = F(u, v) + \frac{N(u, v)}{H(u, v)} \quad (1.25)$$

The inverse filter requires PSF prior to restoration. It gives good results if there is no noise. When there is no noise, second term of the above equation vanishes and the restored image is identical to the original image. The values of  $H(u, v)$  is zero at some selected frequencies. When the noise is associated with the image, noise is amplified at those frequencies and the result is dominated by the amplified noise.

### ***Least Square Restoration***

To alleviate the problem of noise sensitivity of inverse filter, numbers of least square filters have been developed. Two commonly used least squares filters are Wiener filter and constrained least square filter.

*Wiener filter:* [1] It is a spatial invariant filter which uses the minimum mean square error criterion. The PSF is chosen such that mean square error between the restored image and true image is minimised i.e. MSE defined in (1.22) is minimised. The solution to the minimisation problem is called Wiener filter in which restored image in frequency domain is defined as,

$$\hat{F}(u, v) = \left[ \frac{H^*(u, v)}{|H(u, v)|^2 + S_\eta(u, v) / S_f(u, v)} \right] G(u, v) \quad (1.26)$$

where  $S_\eta(u, v)$  and  $S_f(u, v)$  are the power spectrum of the noise and the original image respectively. The restored image in spatial domain is obtained by inverse Fourier transform of  $\hat{F}(u, v)$ . If there is no noise added to the degraded image, then the noise power spectrum vanishes and the Wiener filter reduces to an inverse filter. When power spectrum of the original image  $S_f(u, v)$  is unknown, the Equation (1.26) is approximated as,

$$\hat{F}(u, v) = \left[ \frac{1}{H(u, v)} \frac{|H(u, v)|^2}{|H(u, v)|^2 + K} \right] G(u, v) \quad (1.27)$$

where  $K$  is a constant and is chosen interactively.

*Constrained Least Square Filtering:* The difficulties of the Wiener filter (computation of power spectrum of original image) and inverse filter (noise amplification) are overcome by constrained least square filter. Noise sensitivity problem is removed by using a smoothness measure for optimal restoration. The

restoration is constrained by the parameters of the problem. The criterion function  $C$  is minimised and is defined as,

$$C = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} [\nabla^2 f(x, y)]^2 \quad (1.28)$$

subject to the constraint

$$\|g - H\hat{f}\|^2 = \|\eta\|^2 \quad (1.29)$$

where  $\|x\|$  is the Euclidean norm and  $\hat{f}$  is the estimate of the true image. The solution of the above optimisation problem is given as,

$$\hat{F}(u, v) = \left[ \frac{H^*(u, v)}{|H(u, v)|^2 + \alpha |Q(u, v)|} \right] G(u, v) \quad (1.30)$$

where parameter  $\alpha$  is adjusted to satisfy the constraint defined in (1.29) and  $Q(u, v)$  is the Fourier transform of the Laplacian operator which is typically chosen as,

$$q(x, y) = \begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix} \quad (1.31)$$

In this filtering, the value of  $\alpha$  is selected manually to yield good results for high and medium noise conditions. Both filters produce almost equal results for low noise conditions. The constrained least square filter outperforms Wiener filter when  $\alpha$  is optimum. The parameter  $\alpha$  is a scalar quantity whereas the value of  $K$  in Wiener filter is the ratio of two unknown quantities whose value is seldom constant.

### ***Lucy-Richardson Deconvolution (L-R)***

This algorithm has been independently proposed by Lucy [38] and Richardson [39]. The L-R algorithm is an iterative technique that maximises a Poisson statistics image model likelihood function. The L-R algorithm performs the following steps.

1. An initial approximation of the restored image  $\hat{f}_0$  is made. Typically, the observed image  $g$  is taken as  $\hat{f}_0$
2. The approximation is convolved with the PSF as,

$$\varphi_n = h * \hat{f}_n \quad (1.32)$$

3. A correction factor is calculated depending on the ratio of the blurred image and output of the last step as,

$$\phi_n = \overset{\leftarrow}{h} * \frac{g}{\varphi_n} \quad (1.33)$$

where  $\overset{\leftarrow}{h}$  is the PSF in reverse order.

4. The new restored image is given by

$$\hat{f}_{n+1} = \hat{f}_n \cdot \phi_n \quad (1.34)$$

where  $\cdot$  denotes the pixel by pixel multiplication in spatial domain

steps 2-4 are iteratively performed till an acceptable image quality is obtained.

### ***Recursive Approaches*** [10]

Recursive approaches are advantageous because they allow spatial adaptivity to incorporate in the filter model. These approaches also require less memory storage than direct or iterative approach. Discrete Kalman filter is recursive equivalent of the Wiener filter. Kalman filter is based on autoregressive (AR) modelling of the prior statistical knowledge of  $f$ . Using state space representation, the global state vector for an image model at any pixel position is represented as,

$$\begin{aligned} \underline{f}(m, n) = [ & f(m, n), f(m, n-1), \dots, \\ & f(m-1, N), f(m-1, N-1), \\ & \dots, f(m-M+1, n-M+1)]^T \end{aligned} \quad (1.35)$$

where  $M \times M$  is the image size. Image model is then defined as

$$\underline{f}(m, n) = A \underline{f}(m, n-1) + \underline{w}(m, n) \quad (1.36)$$

where  $\underline{w}(m, n) = [1, 0, 0, \dots, 0]^T w(m, n)$ ,  $w(m, n)$  is the zero mean Gaussian distributed with standard deviation  $\sigma$  and  $A$  is an  $M \times M$  prediction matrix. The Kalman filter prediction and update equations are given by

*Prediction:*

$$\underline{\hat{f}}^+(m, n) = A \underline{\hat{f}}(m, n-1) \quad (1.37)$$

$$P^+(m, n) = AP(m, n-1)A^T + R_{\underline{w}\underline{w}} \quad (1.38)$$

update:

$$\underline{\hat{f}}(m, n) = \underline{\hat{f}}^+(m, n) + K(m, n)[\underline{y}(m, n) - H\underline{\hat{f}}^+(m, n)] \quad (1.39)$$

$$P(m, n) = [I - K(m, n)H] P^+(m, n) \quad (1.40)$$

$$K(m, n) = P^+(m, n)H^T[HP^+(m, n)H^T + R_{nn}]^{-1} \quad (1.41)$$

where

$$P^+(m, n) = E \left\{ (\underline{f}(m, n) - \underline{\hat{f}}^+(m, n))(\underline{f}(m, n) - \underline{\hat{f}}^+(m, n))^T \right\} \quad (1.42)$$

and

$$P(m, n) = E \left\{ (\underline{f}(m, n) - \underline{\hat{f}}(m, n))(\underline{f}(m, n) - \underline{\hat{f}}(m, n))^T \right\} \quad (1.43)$$

In this model observed noise and model noise is assumed to be zeros mean Gaussian process with  $R_{nn} = Enn^T$  and  $R_{ww} = Ew^T$ .

### 1.3.2 Blind Image Restoration Techniques

Blind image deconvolution problem has some important characteristics [35] which are given as,

1. The true image and PSF must be irreducible. An irreducible signal is one which cannot be exactly expressed as the convolution of two or more component signals.
2. The restored image is not the exact true image, rather it is a scaled, shifted version of the original image. That is,

$$\hat{f}(x, y) = Kf(x - D_x, y - D_y) \quad (1.44)$$

where  $\hat{f}(x, y)$  is the estimated image and  $K, D_x, D_y$  are the real constants.

3. The solution of the problem is not unique.

4. There is always a poor compromise between the complexity, convergence and portability of the algorithms used for blind deconvolution.

Few of the long list of blind deconvolution techniques for images are described below.

### ***Motion Blur Identification using Statistical Measures***

Moghaddam and Jamzad [27] proposed a method which uses two dimensional cepstrum of the blurred image  $g(x, y)$  to identify the blur length and angle of motion blur. The cepstrum is defined as,

$$C(g(x, y)) = F^{-1}(\log|F(g(x, y))|) \quad (1.45)$$

The blur angle is estimated from the inverse tangent of the slope of the straight line connecting origin to the first negative peak in the cepstrum. The length of that line gives the motion blur length. When the signal to noise ratio of the image is low, it is difficult to estimate the blur parameters. For noisy situations, they suggested the use of bispectrum. The discrete bispectrum of the  $i^{th}$  segment of the blurred image  $B_i(k, l)$  for one dimension case ( $l = 0$ ) can be defined as

$$\begin{aligned} B_i(k, 0) &= |F_i(k)H(k) + W_i(k)|^2 [F_i(0) + W_i(0)] \\ &= |F_i(k)H(k)|^2 F_i(0)H(0) + \dots |W_i(k)|^2 W_i(0) \end{aligned} \quad (1.46)$$

where  $F_i(k)$  and  $W_i(k)$  are the Fourier transforms of  $i^{th}$  segment of the original image and noise respectively.  $H(k)$  denotes the Fourier transform of the PSF.

### ***Steerable Filter***

Steerable filters [40] are oriented filters that can be given any orientation through a linear combination of set of basis filters. This is used to detect the edges in an image. Steerable filter is applied to the power spectrum of the blurred image to detect the direction of motion. The second derivative of the Gaussian function is used as the basis function. The circular symmetric Gaussian function in two dimensions is given as

$$G(x, y) = e^{-(x^2+y^2)} \quad (1.47)$$

The  $L_2$  norm of response of the filter for different values of blur angle is calculated.  $\theta$  with the largest  $L_2$  norm gives the estimate of the blur angle.

### ***Radon Transform***

The radon transform for a given function  $f(x, y)$  is defined as,

$$R_f(x, \theta) = \int_{-\infty}^{\infty} f(x \cos \theta - y \sin \theta, x \sin \theta + y \cos \theta) dy \quad (1.48)$$

This can be elucidated as integration of  $f$  over a line in  $\mathbb{R}^2$  of distance  $x$  to the origin and at an angle  $\theta$  to the  $y$ -axis. Radon transform of blurred image is found by varying  $\theta$  from  $0^\circ$  to  $360^\circ$ . The angle corresponding to the maximum value of  $R$  gives the blurring angle.

However, this method requires the image to be square size with black background. For rectangular images, this method leads to wrong estimation of  $\theta$ . When the background is not black, or when there are objects close to the boundary of the image, the sharp edges will cause additional lines in the spectral domain at  $0^\circ$ .

### ***Hough Transform***

The anisotropy in the spectrum of the blurred image can be used to determine the angle, which is perpendicular to the motion direction. Hough transform [28] is used to detect the orientation of the lines in the spectrum treating it as an image. The Hough transform divides the parameter space into accumulator cells. The curve  $r$  for a given point  $(x, y)$  in Hough transform can be expressed as,

$$r = x \cos \phi + y \sin \phi \quad (1.49)$$

For each point in the image, the corresponding curve is entered in the accumulator by incrementing the count in each cell along the curve. Hough transform returns the accumulator array in which the maximum value corresponds to blur direction. This transform produces best result when there is a single line in the spectrum which is highly unlikely situation.

### ***Blur Identification based on Frequency Domain Zeros***

The observed image in frequency domain is expressed as,

$$G(u, v) = F(u, v)H(u, v) \quad (1.50)$$

Zeros of  $G(u, v)$  are collectively the zeros of  $F(u, v)$  and  $H(u, v)$ . If we assume a parametric form of PSF and given the zeros of  $H(u, v)$ , the parametric values can be uniquely determined. The commonly encountered PSF like motion blur and out-of-focus blur are completely characterised by their frequency domain zeros [30]. Figure 1.5 shows the block diagram of frequency domain zero method for PSF identification. Due to its low computational complexity and reliability

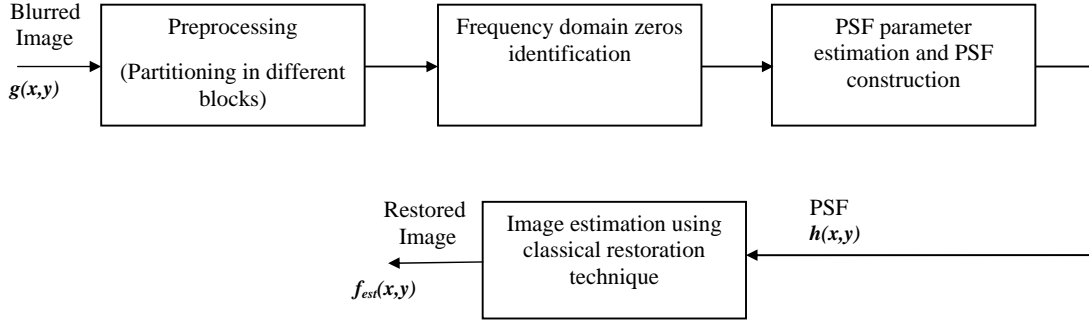


Figure 1.5: Blur identification method based on frequency domain nulls.

it has become popular and successful. But this method fails to identify the frequency domain zeros if SNR is low. Another limitation of this method is that the parametric form of the PSF must be known. For real time applications like astronomical imaging and medical imaging, where PSF may be Gaussian in nature. The frequency domain zero technique for blur parameter identification cannot be used for Gaussian blur.

### ***ARMA Parameters Estimation based Blur Identification***

Blur identification based on ARMA parameter estimation involves modelling the true image as a two dimensional auto regressive (AR) process and PSF as a two dimensional moving average (MA) process. Using this technique, the blurred image is represented as auto regressive moving average process (ARMA). ARMA parameters are estimated to identify the true image and PSF. Several authors have proposed techniques like maximum likelihood (ML) [41–43], generalised cross validation (GCV) [44], neural networks [45], to estimate the ARMA parameters. ML and GCV methods are widely used in image processing applications. AR

model of the true image is represented as,

$$f(x, y) = \sum_{\substack{(l,m) \in R_a \\ (l,m) \neq (0,0)}} a(l, m) f(x - l, y - m) + v(x, y) \quad (1.51)$$

where  $a(0, 0) = 1$  and  $v(x, y)$  is the modelling error and is zero mean homogeneous noise process. The AR model parameters are estimated to minimise the variance of  $v(x, y)$ . The image can also be represented by lexicographical ordering of the two dimensional matrices as,

$$f = Af + v \quad (1.52)$$

Assuming linear degradation model, the degraded image  $g(x, y)$  is expressed as,

$$g(x, y) = \sum_{(l,m) \in R_h} h(l, m) f(x - l, y - m) + n(x, y) \quad (1.53)$$

where  $R_h$  is the finite support of PSF  $h(x, y)$  and  $n(x, y)$  is the additive noise of the imaging system. Using matrix vector notation, the observed image can be written as,

$$g = Hf + \eta \quad (1.54)$$

Rearranging the above equations it can be written as,

$$g = H(I - A)^{-1}v + \eta \quad (1.55)$$

where  $I$  is the identity matrix. Blind image deconvolution technique using ARMA process is used to estimate the parameters  $a(l, m)$  and  $h(l, m)$  from the observed image.

### ***Iterative Blind Deconvolution***

Ayers and Dainty [34] proposed a method called iterative blind deconvolution (IBD) which doesn't assume parametric form for either image or blur. Their method estimates the image and blur simultaneously in an iterative manner by adding different constraints on each. This method comes under the class of nonparametric deterministic image constraint image restoration. It assumes some deterministic constraints such as non-negativity, known finite support, blur invariant edges. The PSF constraints are non-negativity and known finite support.

At the start of iteration, a random initial guess is taken for the true image. Subsequently, the algorithm alternates between the spatial and Fourier domain after adding known constraints on image and PSF. The negative valued pixels within region of support are replaced with zero and nonzero pixels outside the region of support with the background pixel value. In Fourier domain, the PSF (Image) is estimated by using the FFT of the degraded image and PSF estimate. In  $i^{th}$  iteration, the frequency domain estimate of the PSF and image is given by,

$$H_i(u, v) = \frac{G(u, v) \hat{F}_{i-1}^*(u, v)}{\left| \hat{F}_{i-1}(u, v) \right|^2 + \alpha / |H_{i-1}(u, v)|^2} \quad (1.56)$$

$$F_i(u, v) = \frac{G(u, v) \hat{H}_{i-1}^*(u, v)}{\left| \hat{H}_{i-1}(u, v) \right|^2 + \alpha / |F_{i-1}(u, v)|^2} \quad (1.57)$$

where  $*$  denotes the complex conjugate and  $\alpha$  determines the energy of the additive noise. The IBD method is popular due to its low computational complexity. The major limitation of this method is that uniqueness and convergence property is uncertain and the algorithm is unreliable. The performance of the algorithm also depends on the initial estimate of the true image and it exhibits instability.

### ***Simulated Annealing***

Macallum has proposed simulated annealing (SA) for blind deconvolution [46]. In this technique, he made the same assumption on PSF as given in IBD method. It minimises the following multimodal cost function defined as,

$$J\left(\hat{f}(x, y), \hat{h}(x, y)\right) = \sum_{\forall(x, y)} \left[ \hat{f}(x, y) * \hat{h}(x, y) - g(x, y) \right]^2 \quad (1.58)$$

The PSF and image both are assumed to be positive and have known finite support. Global minimisation of  $J$  is achieved by varying the parameters. For some parameter values if  $J$  decreases then it is accepted. Otherwise if it increases, it is accepted with a probability  $p = \exp(\frac{-\Delta J}{T_k})$  where  $\Delta J$  is the change in the cost function and  $T_k$  is the temperature parameter. Temperature parameter  $T_k$  is reduced with the iteration in order to reach the global minimum of  $J$ . This algorithm provides sensible results and is reliable unlike IBD algorithm. The limitation of the algorithm is slow convergence to the global minimum. The

convergence of the algorithm depends on how  $T_k$  is reduced. The algorithm is computationally complex for large sized images.

### ***NAS-RIF***

*The nonnegativity and support constraints recursive inverse filtering* (NAS-RIF) was developed by Kundur [36] to overcome the poor convergence problem of IBD and computational complexity problem of SA. NAS-RIF algorithm models the image restoration problem as an optimisation problem. In addition to assumptions made in IBD and SA, it also assumes that the PSF is absolutely summable i.e.  $\sum_{\forall(x,y)} |h(x,y)| < \infty$  and its inverse ( $h^{-1}(x,y)$ ) is also absolutely summable. It consists of a variable finite impulse response (FIR) filter  $u(x,y)$  to which blurred image  $g(x,y)$  is given as an input. The output of this filter gives the estimate of the true image. The block diagram of the algorithm is shown in the Figure 1.6.

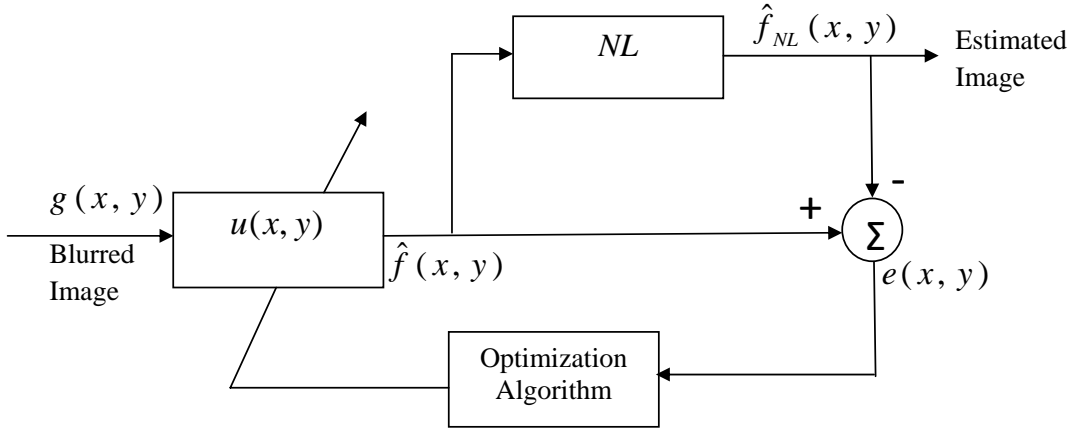


Figure 1.6: NASRIF algorithm.

The nonlinear filter does a non expansive mapping to project the estimated image into the space which represents the known characteristics of the true image. The difference between the projected image and estimated image is used as the error measure. This error is used to update the coefficients of the FIR filter. The advantage of this algorithm is that it doesn't require PSF to be of known finite extent.

### ***Defocus Parameter Estimation using Wavelet Transform***

Lin et al. [29] have proposed an method to estimate the out-of-focus bur parameter estimation using wavelet transform. The ratio of wavelet coefficients of first and second level of an image has a nonlinear relationship with the defocus blur parameter. This relationship is used to train a neural network in the training phase. During testing phase, wavelet coefficients of the blurred image for the two levels are provided to the trained neural network which gives the defocus parameter as its output. The ratio decreases as the amount of defocus increases.

### ***Blind Deconvolution using SVR***

Li et al. [47] have used support vector regression (SVR) to restore images directly from degraded images without having information about the degradation function. They obtained an optimised mapping from a neighbourhood in a degraded image to the central pixel in the original image. A training set  $(X_i, y_i)$  is constructed from the blurred image and true image. A  $7 \times 7$  window is chosen in the vicinity of the pixel in the blurred image  $g(x, y)$  and converted to a vector of  $49 \times 1$ . The corresponding pixel  $f(x, y)$  of the true image is taken as the target pixel. The window is moved from top to bottom and samples are created. During restoration, same window is used to create the attribute. The trained SVR model predicts the pixel value. Thus the image is restored on a pixel by pixel basis. An apriori information about the blur and noise can enhance the performance of this algorithm. Another advantage of their algorithm is that it can generalise to new images degraded with different types of blur and is robust to parameter selection.

### ***PCA based Blind Image Restoration***

In another work, Li et al. [37] have proposed a blind restoration technique for atmospheric turbulence degraded images based on principal component analysis (PCA). They suggested multichannel image restoration using PCA. It is also shown that single channel blind image deconvolution is also possible through creating an ensemble of blurred images from a single blurred image using shifting technique. Variance of an image decreases with the blurring. So variance maximisation of the blurred image is equivalent to restore the high frequency

component. PCA is an orthogonal transformation technique used to reduce the dimensionality of the dataset to 1. PCA transforms the data into a new orthogonal coordinate system where the first coordinate is the first principal component i.e. the eigen vector associated with the maximum eigen value. The eigen value points the importance of eigen vector to the variance of the data. PCA based blind image deconvolution is fast and robust to noise. However, improper PSF support can introduce artifacts in the restoration result.

### ***Kullback-Leibler Divergence Approach***

Recently Seghouane [48] has proposed new algorithm for maximum likelihood blind image restoration. In their work, he modelled the original image and additive noise as multivariate Gaussian processes with unknown covariance matrices. In this algorithm blur, noise and image parameters are estimated by alternating minimisation of the Kullback-Leibler divergence between a model family of probability distributions defined using the linear image degradation model and a desired family of probability distributions. The algorithm has the advantage of providing closed form expressions for the parameters to be updated.

### ***Dependent Component Analysis for Blind Image Restoration***

Du and Kopriva [49] suggested dependent component analysis (DCA) to restore images without having the information about the noise statistics or blur parameter. They have applied DCA technique to restore atmospheric turbulence blurred images. The use of DCA doesn't require the sources to be independent. DCA basically finds a transform  $T$  that can improve the statistical independence between the sources but leave the basis matrix unchanged and is defined as,

$$T(G) = T(AS) \cong AT(S) \quad (1.59)$$

### ***Probabilistic Modelling for Motion Deblurring***

Shan et al. [50] have proposed an unified probabilistic model for both blur kernel estimation and unblurred image restoration. They solve a maximum a posteriori (MAP) problem by advanced iterative optimisation technique which alternates between the blur kernel refinement and image restoration until convergence. Their

work uses a model of the spatially random distribution of the image noise. This model helps to suppress the artifacts commonly encountered in image restoration. A smoothness constraint is enforced in the image in the areas of low contrast. An advanced optimisation scheme is employed which allows computationally intensive steps in the frequency domain.

### ***Multilayer Neural Network for Blur Identification***

Multilayer multivalued neural network (MLMVN) consists of multivalued neurons, have complex valued weights and activation function as a function of the argument of a weighted sum. MLMVN have been used by [32] to identify blur type and its parameters. A derivative free learning algorithm is used for training the feed forward neural network. Each of the neuron in the output layer represents one blur type and they are considered as one class. The output neurons classify the blur and its parameters and rejects other blurs. The MLMVN based blur identification is computationally fast and cheap.

### ***Image Restoration using $C_p$ and MSE***

The choice of regularisation parameter is crucial for the quality of the restored image [7]. Two estimators such as closed form approximation to the minimum of  $C_p$  selection criterion and minimum of mean square error criterion (MSE) have been proposed in their work. Using  $C_p$  criterion, regularisation parameter is chosen by minimising an unbiased estimate of a total squared error risk which is defined as,

$$\begin{aligned} C_p(\lambda) &= \left\| g - H \hat{f}_\lambda \right\|^2 + 2\sigma_\omega^2 \text{tr}(H_\lambda) \\ &= \left\| (I - H_\lambda) g \right\|^2 + 2\sigma_\omega^2 \text{tr}(H_\lambda) \end{aligned} \quad (1.60)$$

where  $H_\lambda = H(H^T H + \lambda Q^T Q)^{-1} H^T$ ,  $\hat{f}_\lambda$  is the regularised estimated image and  $\sigma_\omega^2$  is the variance of i.i.d noise added to the image.  $H$  is the blurring matrix. Using MSE criterion,  $\hat{f}_\lambda$  is made as close as possible to original image  $f$ . The risk criterion for estimation is given as,

$$MSE(\lambda) = E\{(f_\lambda - f)^2\} \quad (1.61)$$

where expectation is taken with respect to the probability of distribution of additive noise. The optimum value of  $\lambda$  is that value which minimises the above equation. Both the estimators depends on the unknown image  $f$  and variance of the noise  $\sigma_\omega$ .

### ***Local Neural Approach to Semi-blind Image Restoration***

I.Gallo et al. [51] proposed an iterative strategy based on neural learning. This work uses a local error function derived from the conventional global constrained error measure. It also assigns a separate regularisation parameter to each pixel based on local gray level variance. In their work, they treated the pixels of the restored image as synaptic weights. The weights are updated in each iteration during learning to reduce the output error measurement.

### ***Image Restoration using Kurtosis Minimisation***

Image restoration using kurtosis minimisation (KM) has been suggested by Li and Simske [52] for atmospheric turbulence blurred images. They have used the concept that kurtosis of an image increases with extent of blurring. Phase structure has been utilised to analyse the impact of blurring on kurtosis. Blur parameter is estimated after setting the search space on a trial and error basis. For each of the estimated parameter, the image is deblurred using a classical image restoration technique. The restored image with minimum kurtosis is selected and blur parameter corresponding to that image is considered as the identified parameter. This is summarised as,

$$\lambda_k = \arg \left\{ \min_{\lambda \in \Omega} k \left( \hat{f}(\lambda) \right) \right\} \quad (1.62)$$

KM method is based on statistics of a given image and may not give good results for any image.

### ***Dispersion Minimisation for Blind Image Deconvolution***

Vural and Sethares [53] have presented an non linear adaptive filtering based iterative technique for simultaneous blur identification and image restoration from a noisy blurred image. The restoration problem has been modelled as an

optimisation problem and tries to minimise the dispersion by using a cost function called constant modulus (CM). The noisy blurred image is passed through an adaptive FIR filter whose coefficients are updated to minimise the cost function. Their method is applicable to minimum as well as mixed phase blurs and is applied to six or less-bit gray scale images.

### ***Fast Identification of Motion Blur Parameters***

Recently Dobes et al. [54] have proposed a fast method of finding motion blur length and direction of blur. Their scheme constructs the PSF after parameter identification. It computes the power spectrum of the image gradient in the frequency domain. The power spectrum of the image gradient is applied to a band pass Butterworth filter to smooth the spectrum and remove the unwanted noise. The resulting pattern consists of parallel stripes and the directions stripes correspond to the blur angle. Thus the orientation of stripes is found using Radon transform. The distance between the neighbouring stripes corresponds to the blur length. Finally, the image is deblurred using the computed kernel and Lucy-Richardson algorithm.

### **1.3.3 Regularisation Approaches**

Addition of noise during the blurring process makes the deconvolution problem ill-posed. One of the most popular approach which is used to handle such a problem is regularisation [55–58]. The important issue in regularisation approach is proper selection of regularisation parameter which still remains as an active field of research. Variety of algorithms have been proposed in the literature to find the regularisation parameter to solve the restoration problem. Some of the important schemes are discussed below.

#### ***Adaptive Selection of Regularisation Parameter and Operator***

Wu et al. [59] have suggested a scheme to select regularisation parameter and operator based on the local noise variance. They have used the fact that local noise variance reflects the degree of noise contamination of local image. An uniform blur

operator  $d$  was used to blur the degraded image ( $g$ ) and is denoted as,

$$z = d * g \quad (1.63)$$

The noise variance in the degraded image ( $g$ ) is approximated as,

$$\sigma_n^2 = r\sigma_g^2 - \sigma_z^2 \quad (1.64)$$

where  $\sigma_n^2, \sigma_g^2, \sigma_z^2$  are the variances of noise ( $n$ ),  $g$  and  $z$  respectively and  $r$  is a correlation factor. The local noise variance are proportionally mapped into some region  $[v_{max}, v_{min}]$ . The mapped local noise variance is used as the local regularisation coefficient. They also compute the regularisation operator adaptively.

### ***GCV based Regularisation***

Reeves and Mereseru [56] have proposed the idea of GCV for regularisation parameter estimation. GCV is applied by considering each pixel of the blurred image as one set of data. For a fixed value of regularisation parameter, the restored image is obtained using all the pixels leaving one pixel. Then the restored image is reblurred again to predict the noisy blurred image pixel that has been left out in the restoration process. Each observation gives a different restored image. The regularisation parameter which minimises the mean square error over all the observations gives the optimum regularisation parameter.

### ***Regularisation Parameter Estimation in Total Variation Image Restoration***

Liao et al. [57] developed a fast image restoration method which selects the regularisation parameter automatically to restore noisy blurred images. The method exploits the generalised cross validation technique to determine the amount regularisation used in each restoration step. The regularisation parameter is updated in each iteration, which increases the closeness of the restored image towards the true image. The algorithm minimises an objective function in an alternative minimisation framework.

### *Arnoldi Process based Regularisation Parameter estimation*

Kai et al. [60] have exploited the GCV method for estimating the regularisation parameter. The large computational complexity involved in GCV technique is reduced using arnoldi process. The Arnoldi process factors the system matrix into Hessenberg matrix and orthogonal one. Arnoldi process has been used to efficiently compute the numerator and denominator of the GCV function. Their algorithm is basically suggested for super resolution image restoration.

## 1.4 Motivation

Keeping the research directions in view, it has been realised that there exists enough scope to improve the restoration performance. In this thesis, an effort has been made to remove blur from images. In particular, the objectives are narrowed to —

- (i) devise algorithms to find motion blur parameter for effective image restoration.
- (ii) restore blurred images with minimal apriori knowledge.
- (iii) utilise soft computing techniques for deblurring.
- (iv) estimate an optimum regularisation parameter for images degraded due to blur and noise.
- (v) restore images degraded with spatial variant blur .

## 1.5 Thesis Layout

Rest of the thesis is organised as follows —

**Chapter 2: Motion Blur Parameters Estimation using Gabor filter and RBFNN** In this chapter, two different schemes are proposed to estimate the motion blur parameters. Two dimensional gabor filter has been used to calculate the direction of the blur. Radial basis function neural networks (RBFNN) has

been utilised to find the length of the blur. Subsequently, classical restoration has been used to restore the image. Noise robustness of the proposed scheme is tested with different noise strengths.

**Chapter 3: Blur Parameter Identification using SVM** A blur identification technique through support vector classification is presented in this chapter. Support Vector machine (SVM) was originally developed for binary decision problems and has been extended to multi-class problems with the help of binary decision tree (BDT). The length of motion blur and the parameter of Gaussian blur is identified using Multiclass SVM. The PSF, thus identified, is used to determine the true image by deconvolving with the observed blurred image.

**Chapter 4: PSO based SVR for Blind Image Restoration** This chapter presents a swarm intelligence based parameter optimisation scheme for blind image restoration. Support vector regression (SVR) has been utilised to obtain a true mapping of the images from the observed noisy blurred image. The parameters of SVR are optimised through particle swarm optimisation (PSO) technique. Finally, restoration performance is compared with other competent schemes.

**Chapter 5: PSO Based Regularisation Parameter Estimation** Blur in the presence of noise makes the inversion problem ill-conditioned. The regularisation parameter required for restoration of noisy blurred image is discussed in this chapter. Generalised cross validation (GCV) criterion is used to estimate the regularisation Parameter. Trapping into the local minimum has been the major limitation of GCV error function. For the purpose an global optimisation scheme namely particle swarm optimisation (PSO) is utilised to minimise the cost function of GCV measure. The scheme adapts to various degradation phenomenon like motion and out-of-focus blur along with noise of various strengths.

**Chapter 6: Rotational Motion Deblurring using Elliptical Modelling** The degradation caused due to rotational motion of object or camera no longer remains as spatial invariant. Such situation arises in satellite imaging. In case of

spatial variant blur, more than one PSF act on the image and they are difficult to estimate. This chapter devotes restoration of images in such a phenomenon. The spatial variant blur is considered to be combination of spatial invariant blurs and each one is estimated using elliptical modelling. Finally Wiener filter has been used to estimate the true image.

**Chapter 7: Conclusions and Future Work** This chapter provides the concluding remarks with more emphasis on achievements and limitations of the proposed schemes. The scopes for further research are outlined at the end.

The contributions made in each chapter are discussed in sequel, which include proposed schemes, their simulation results, and the comparative analysis.

# Chapter 2

Motion Blur Parameters Estimation  
using Gabor Filter and RBFNN

## Chapter 2

# Motion Blur Parameters Estimation using Gabor filter and RBFNN

The goal of image restoration is to reconstruct an approximated version of the original image from a degraded observation. Image degradation occurs due to various reasons like camera mis-focus, atmospheric turbulence, camera or object motion, etc. The blurring in images due to motion is commonly encountered when there is a relative motion between the camera and the object. Motion deblurring has many applications such as satellite imaging, medical imaging, traffic control, etc. The motion may be linear or non-linear. The degradation due to motion can be modelled as a two dimensional linear shift invariant process.

In the present chapter, an attempt has been made to estimate the motion blur parameters using Gabor filter and radial basis function neural network (RBFNN). Exhaustive simulation has been carried out to evaluate the proposed scheme. The robustness of our scheme is tested by introducing additive Gaussian noise of varying strength to the blurred image. After motion blur parameter estimation, image is restored using classical restoration technique, namely Wiener filter [1].

The rest of the chapter is organised as follows. The blur model is described in Section 2.1. Proposed blur angle estimation using Gabor filter is presented in Section 2.2. Section 2.3 describes the proposed RBFNN method to determine the blur length. Noise robustness of the proposed scheme is outlined in Section 2.4. Section 2.5 deals with simulation and comparative analysis of the suggested scheme. Finally, Section 2.6 gives the concluding remarks.

## 2.1 Motion Blur Model

The PSF for motion blur can be described as,

$$h(x, y) = \begin{cases} 1/L & \text{if } \sqrt{x^2 + y^2} \leq L/2, \\ & \tan \theta = y/x \\ 0 & \text{otherwise} \end{cases} \quad (2.1)$$

where  $L$  is the blur length and  $\theta$  is the angle of motion blur. The restoration performance depends on the estimation of point spread function (PSF), which in turn is dependent on  $L$  and  $\theta$ . So accurate estimation of these parameters from a given motion blurred image is a challenging issue. The proposed algorithm estimates the parameters  $\theta$  and  $L$  separately. The PSF is constructed from the estimated parameters, and the conventional Wiener filter is used for restoration of the blurred image. The overall algorithm is described in *Algorithm 1*. Details of the parameter estimation are described in sequel in the following sections.

### **Algorithm 1**

**Input: Motion Blurred Image**

**Output: Restored Image**

Step 1. The blur angle ( $\theta$ ) is determined using Gabor filter.

Step 2. The blurred image is rotated in the direction opposite to the blur angle to obtain the equivalent horizontal blurred image.

Step 3. The blur length ( $L$ ) is estimated using RBFNN.

Step 4. PSF is constructed using the estimated blur parameters.

Step 5. The image is restored using the Wiener filter.

## 2.2 Angle Estimation using Gabor Filter

One of the important observations in motion blurred images is that its frequency spectrum shows dominant parallel lines which corresponds to the angle of blur.

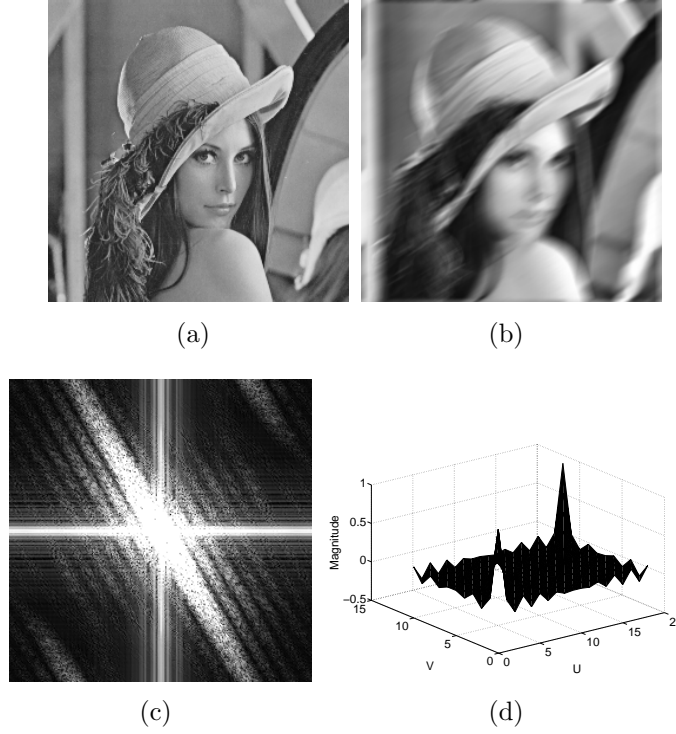


Figure 2.1: (a) Original *Lena* image. (b) Blurred *Lena* image with  $L = 20$  and  $\theta = 30^\circ$ . (c) Spectrum of the blurred *Lena* image. (d) Frequency plot of PSF.

This can be observed from *Lena* image blurred with an angle  $\theta = 30^\circ$  and  $L = 20$  as shown in Figure 2.1. So, any of the line detection algorithms can be used to determine the orientation of the parallel lines.

Gabor filters are Gaussian filters modulated by a sinusoidal wave. A good number of researchers has used Gabor filter bank to extract image features in applications like pattern recognition, image segmentation etc [61]. The two-dimensional Gabor filter is defined as,

$$G(x, y) = \frac{1}{2\pi\sigma_x\sigma_y} \exp \left[ -\frac{1}{2} \left( \frac{x^2}{\sigma_x^2} + \frac{y^2}{\sigma_y^2} \right) \right] \times \exp [-j\omega (x \cos \phi + y \sin \phi)] \quad (2.2)$$

where  $\sigma_x$  and  $\sigma_y$  is the standard deviation in  $x$  and  $y$  direction respectively.  $\phi$  and  $\omega$  represents the orientation and frequency of the Gabor filter.

Modulated Gaussian filters can be used to find the orientation in the patterns. The two-dimensional Gabor filter masks for different orientation are shown in Figure 2.2. The response of the Gabor filter varies with orientation parameter and has been used here to calculate the blur angle. The two-dimensional Gabor

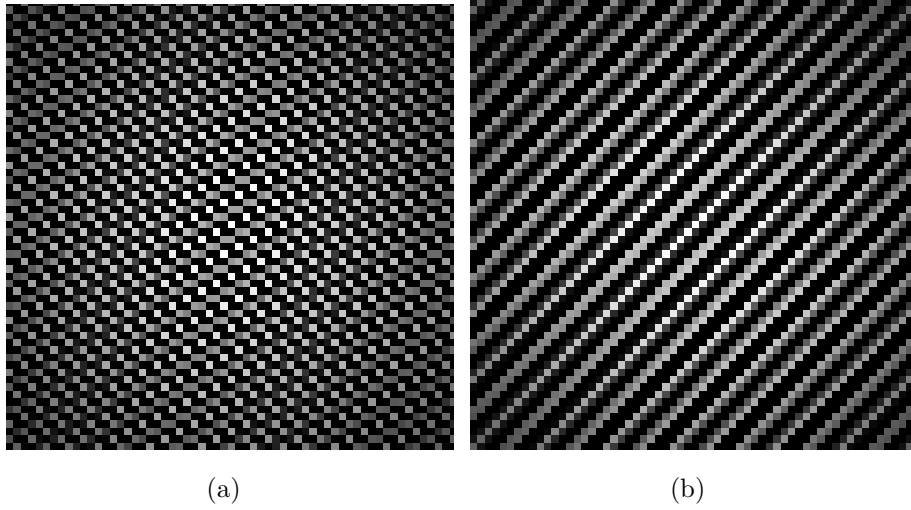


Figure 2.2: (a) Gabor filter mask with  $\phi = 30^\circ$ . (b) Gabor filter mask with  $\phi = 45^\circ$ .

filter is convolved with the spectrum of the blurred image to get the response at different orientations by keeping other parameters fixed. The detail of the angle estimation strategy is described below.

Pattern of the frequency response of the blurred image has been used to find the motion direction. As it can be clearly seen from the Figure 2.1(c), for a blur angle  $\theta$  the patterns are oriented at  $\alpha = \theta + 90$ . So, the orientation of the lines in the spectrum of the blurred image is directly related to the blur angle. Various line detection algorithms such as Hough transform, Radon transform can be used to detect the orientation of the line. However, Hough transform requires a threshold to identify points on the line [28]. This threshold is different for different images. Any small error in threshold could result in a large variation in estimation of the blur angle. To alleviate this problem, Gabor filter has been used to determine the blur angle. The response of the Gabor filter depends upon the frequency and orientation of the input image. Gabor filter with different orientation are convolved with the spectrum of the blurred image and  $L_2$  norm of the result is computed. Detail of the motion direction estimation is described in *Algorithm 2*.

**Algorithm 2**

**Input:** Motion Blurred Image

**Output:** Blur Angle ( $\theta$ )

Step 1. The spectrum of the blurred image is computed.

Step 2. Logarithm of the spectrum of the blurred image i.e.  $I = \log(G(u, v))$  is used as input to the Gabor filter.

Step 3. Gabor filter with different orientation ( $\theta$ ) are convolved with  $I$  to get different responses  $R(\theta)$ .

Step 4. For every  $\theta$ , the  $L_2$  norm of the result of the convolution is calculated. The blurring angle is then calculated as,

$$\hat{\theta}_{blur} = \arg \left\{ \max_{\theta} R(\theta) \right\}.$$

## 2.3 Length Estimation using RBFNN

The second parameter of motion blur is the length of the blur ( $L$ ). This describes how much distance the object or camera has moved during the exposure time. To predict the blur length of a particular blurred image, we employ a RBFNN with sum of the Fourier coefficients (SUMFC) of the corresponding blurred image as the input. Following observations with respect to SUMFC versus the blur length of various blurred images motivated us to utilise a nonlinear predictor RBFNN.

Fourier feature of an image is one of the simplest features in frequency domain and easy to determine using FFT algorithm. Standard images, including *Lena*, *Cameraman* have been blurred horizontally using different blur lengths. The SUMFCs of different blurred images have been computed and normalised between 0 and 1. Figure 2.3 depicts the relationship between SUMFCs and their corresponding blur lengths. It has been observed that there exists a nonlinear relation between these two parameters and it is true for all images. This non-linear behaviour is exploited to predict the blur length from SUMFC as an input in the RBFNN.

### 2.3.1 Radial Basis Function Neural Network

Neural networks have been used for many image processing applications [62]. Radial basis function neural network (RBFNN) has gained considerable attention

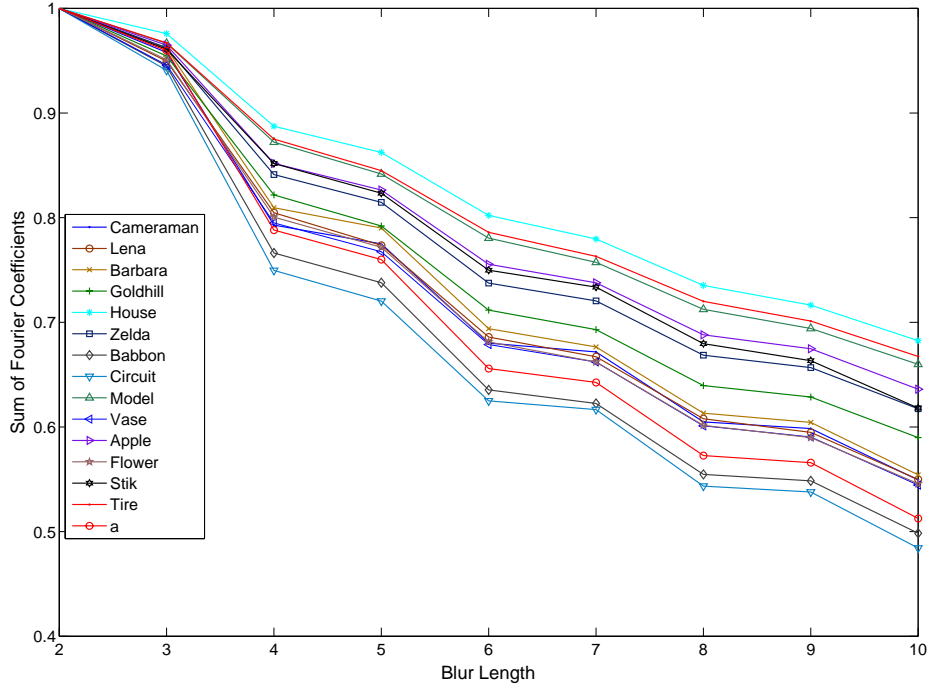


Figure 2.3: Relation between blur length and Fourier feature for different images.

as an alternate to multilayer perceptrons trained by the back propagation algorithm. The basis functions are embedded in a two-layer neural network, where each hidden unit implements a radial basis activation function. There are no weights connected between the input layer and the hidden layer. The output layer of the network computes the weighted sum of the outputs from the hidden layer. The input to a RBFNN is nonlinear while the output is linear. RBFNN is characterised by its localisation (centre) and activation hyperspace (activation function). The activation function used in a RBFNN is usually a localised Gaussian basis function. The structure of the network is shown in Figure 2.4.

In this work, the standard Gaussian nonlinear basis function is used and is defined as,

$$\phi_i(x) = \exp\left(-\frac{(x - c_i)^2}{2\sigma^2}\right) \quad (2.3)$$

where,  $c_i$  is the centre of  $i^{th}$  neuron and  $\sigma^2$  defines the variance. Euclidean distance is used as the distance metric to calculate the distance of the inputs from

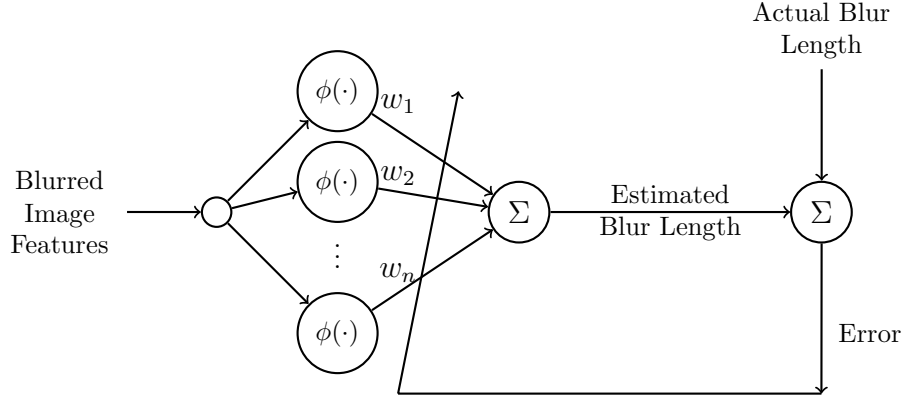


Figure 2.4: Radial basis function neural network (RBFNN)

the basis centre. The spread  $\sigma$  of all the Gaussian basis function are fixed, and a standard value of 0.1 is used. Levenberg Marquardt (LM) learning algorithm is applied to train the network . Using a set of input–output pair (SUMFC-L), the network parameters are optimised using LM. In order to determine the error, the actual output on the output layer is compared with the desired output. Depending on this error value, the weight matrix between the input-output layers is updated.

## 2.4 Noise Robustness of the proposed method

The proposed method for motion blur parameter estimation is also robust to noise. When there is noise associated with an image along with blur, parallel dark lines in the spectrum are not prominent. More lines further disappear as the amount of noise increases. The noisy blurred images and their spectrum for different SNR are shown in Figure 2.5. Authors in [63] have shown that denoising the image before deblurring can further degrade the result and also affects the precision of the algorithm. So the approach of denoising before deblurring is not used in practice. So in this proposed scheme, restoration is performed in the presence of noise.

It has been observed from the Figure 2.5 that for a low value of SNR i.e. 25 dB, the white lines are still present in the power spectrum. As reported in [27], direction of white bound in the spectrum of the noisy image matches with the direction of the motion blur as with noise free image. So, same strategy has been used for angle estimation as for noise free images. Gabor filter has been

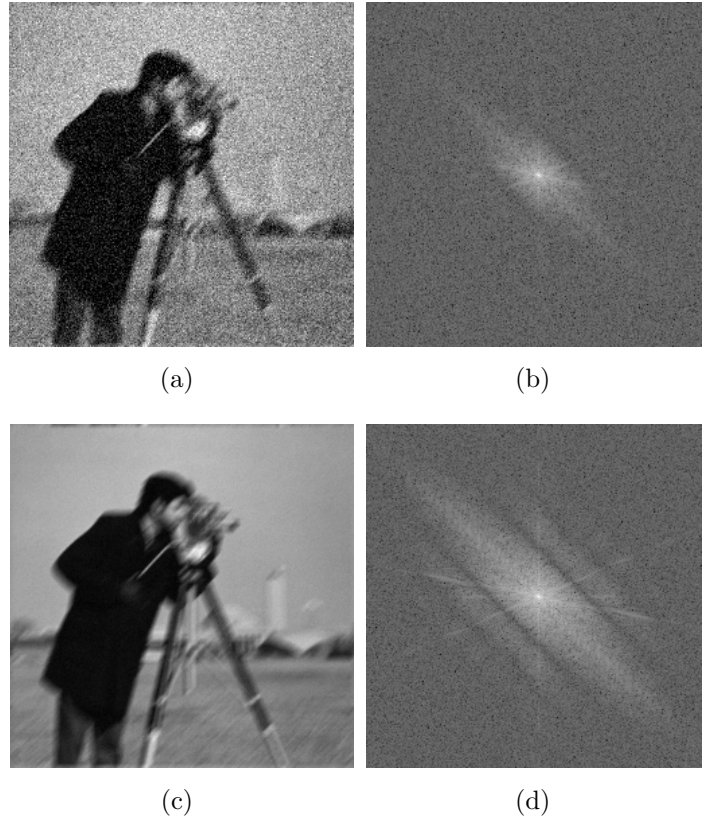


Figure 2.5: (a) *Cameraman* image degraded by motion blur with  $L = 10$ ,  $\theta = 45^\circ$  and Gaussian noise with SNR = 25 dB. (b) Power spectrum of the blurred *Cameraman* image in (a). (c) *Cameraman* image degraded by motion blur with  $L = 10$ ,  $\theta = 45^\circ$  and Gaussian noise with SNR = 40 dB. (d) Power spectrum of the blurred *Cameraman* image in (b).

used to find the direction of this white thick lines in the spectrum as described in Section 2.2. Identification of length of motion blur in noisy images is difficult due to absence of parallel lines in the spectrum [6]. The nonlinear relationship between Fourier feature and blur length increases when noise is added to the image as shown in Figure 2.6. This is due to the fact that noise introduces more frequency components. However, when the RBFNN is trained with known noisy samples, estimation accuracy is not greatly affected. The details description of training the RBFNN based blur length estimator is presented in the following section.

## 2.5 Simulation Results and Discussions

The proposed scheme works in two phases namely estimation of blur angle ( $\theta$ ) using Gabor transform and estimation of blur length ( $L$ ) using RBFNN.

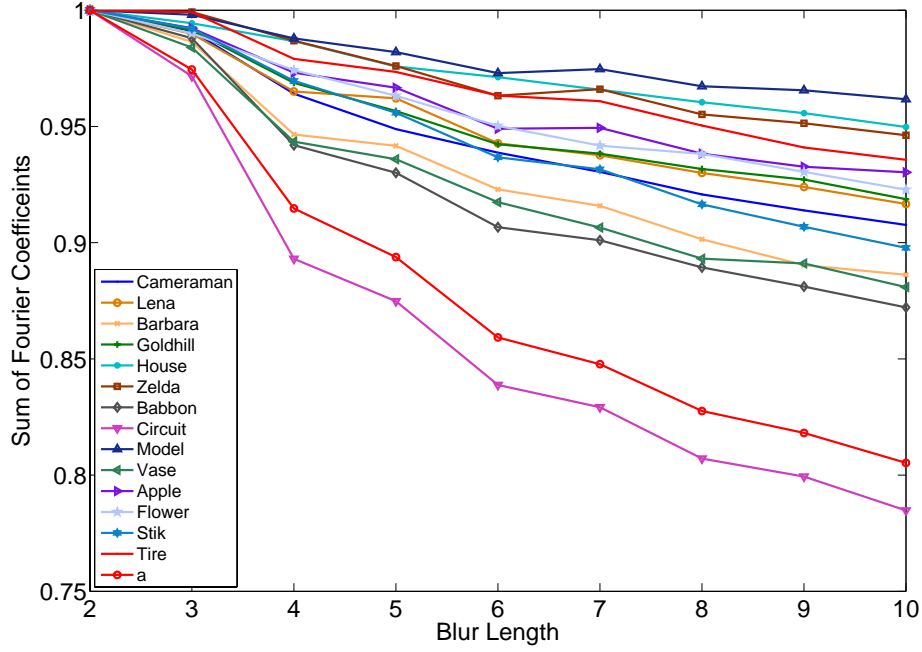


Figure 2.6: Relationship between blur length and Fourier feature for noisy images.

To validate the scheme, simulation has been carried out on standard images with various combinations of  $L$  and  $\theta$ . For comparative analysis, the degraded images are subjected to some of the existing schemes for restoration. The overall simulation work is described in four different experiments and detailed below.

**Experiment 1: Training of RBFNN length estimator.**

Fifteen Standard images including *Cameraman*, *Lena*, *Tree*, *Barbara*, *Baboon* of size  $(256 \times 256)$  etc. are motion blurred with blur length varying from 2 to 10 in a step of 1 pixel. Then standard Gaussian noise of different signal to noise ratio (SNR) is added to 8 images. The SUMFC for each blurred image is computed. So exhaustive 135 training samples (SUMFC vs.  $L$ ) are collected from noisy as well as noise free images. Out of the total samples, 108 samples have been used to train the proposed RBFNN length estimator using back propagation algorithm. Rest 27 samples have been used for testing purpose. The training process is continued till the error is less than 0.01. The training convergence characteristics in MSE (measured in dB) is shown in Figure 2.7. It has been observed that

training convergence is achieved after about 36 epochs. The trained RBFNN is used subsequently to estimate the blur length from degraded images.

***Experiment 2: Performance analysis in noise free images***

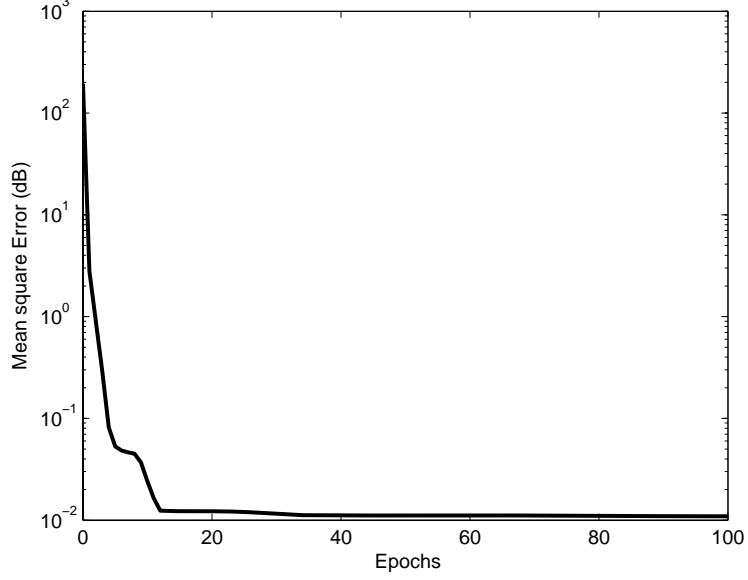


Figure 2.7: Convergence characteristics of RBFNN.

Images like *Tree*, *Lena*, *Cameraman* with different blur length and angle are taken as the input to the proposed estimation scheme. The Blur angle for each of these images is calculated using Gabor filter. The blur angle has also been computed with other methods [27, 64]. Table 2.1 shows the blur angle estimation with the proposed scheme along with other schemes. It has been observed that Radon transform is well suited for calculating the blur angle. However, it fails to estimate the accurate result when the blur angle is equal to  $45^\circ$ . The proposed Gabor filter angle estimator works satisfactorily for all the mentioned blur angles. The computation cost involved for estimating the blur angle is comparable with that of Rekleitis and Radon Transform and shown in Table 2.2. However, even the proposed scheme lacks in exact estimation of blur angle.

Once the blur angle is estimated, the same blurred image is rotated in the direction opposite to the blur angle to obtain the equivalent horizontal blurred image. This image is subjected to estimate the blur length using the trained RBFNN. SUMFCs for the equivalent blur images belonging to training set as

Table 2.1: Estimated blur angle  $\theta$  in noise free Situation.

Image	Original $\theta$	Radon [27] Transform	Rekleitis [64]	Proposed method
<i>Lena</i> (256x256)	$\theta = 30^\circ$	$\theta_{est} = 29^\circ$	$\theta_{est} = 28^\circ$	$\theta_{est} = 29^\circ$
<i>Tree</i> (256x256)	$\theta = 45^\circ$	$\theta_{est} = 40^\circ$	$\theta_{est} = 48^\circ$	$\theta_{est} = 46^\circ$
<i>Cameraman</i> (256x256)	$\theta = 40^\circ$	$\theta_{est} = 41^\circ$	$\theta_{est} = 43^\circ$	$\theta_{est} = 39^\circ$
<i>Chekerboard</i> (128x128)	$\theta = 40^\circ$	$\theta_{est} = 37^\circ$	$\theta_{est} = 38^\circ$	$\theta_{est} = 39^\circ$
<i>Stik</i> (64x64)	$\theta = 35^\circ$	$\theta_{est} = 32^\circ$	$\theta_{est} = 33^\circ$	$\theta_{est} = 36^\circ$

Table 2.2: Time (in sec) required to estimate blur angle ( $\theta$ ) using different methods.

Image	Original $\theta$	Radon [27] Transform	Rekleitis [64]	Proposed method
<i>Lena</i> (256x256)	$\theta = 30^\circ$	62	68	65
<i>Tree</i> (256x256)	$\theta = 45^\circ$	68	73	75
<i>Cameraman</i> (256x256)	$\theta = 40^\circ$	68	69	71
<i>Chekerboard</i> (128x128)	$\theta = 40^\circ$	57	55	53
<i>Stik</i> (64x64)	$\theta = 40^\circ$	50	48	50

well as for images outside the training set are computed and used as an input to RBFNN to estimate the corresponding blur length. The estimation accuracy is compared with the existing schemes and is shown in Table 2.3. It has been observed that estimated blur lengths are approximately close to the original values. The proposed scheme gives an estimation error of about 1 – 1.7 pixel in length and  $1^\circ$  in angle.

The PSF is constructed using the estimated parameters  $(\theta, L)$ . Subsequently, the blurred images have been restored using Wiener filter. The restored results

Table 2.3: Estimated blur length  $L$  in noise free situation.

Image	Original $L$	Steerable [65] filter	Rekleitis [64]	Proposed approach
<i>Lena</i> (256x256)	$L = 15$	$L_{est} = 18$	$L_{est} = 19$	$L_{est} = 14$
<i>Tree</i> (256x256)	$L = 20$	$L_{est} = 17$	$L_{est} = 18$	$L_{est} = 19$
<i>Cameraman</i> (256x256)	$L = 30$	$L_{est} = 27$	$L_{est} = 32$	$L_{est} = 29$
<i>Chekerboard</i> (128x128)	$L = 30$	$L_{est} = 27$	$L_{est} = 32$	$L_{est} = 29$
<i>Stik</i> (64x64)	$L = 30$	$L_{est} = 27$	$L_{est} = 32$	$L_{est} = 29$

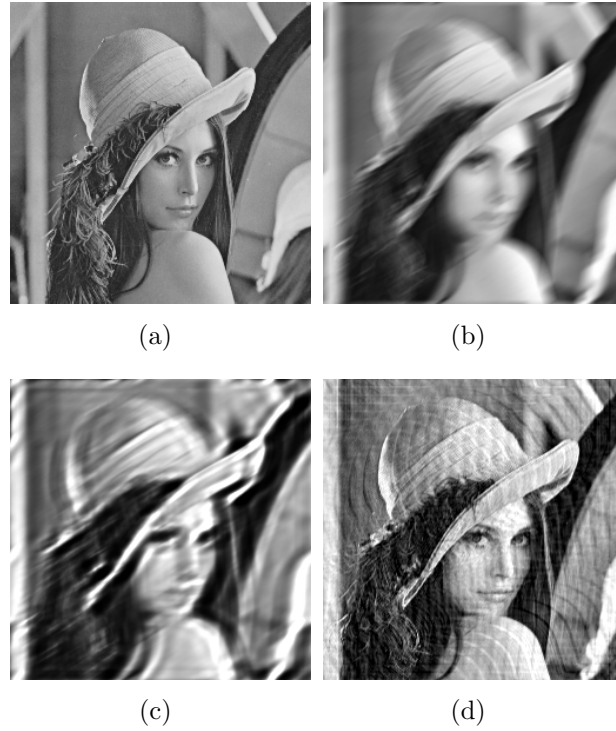


Figure 2.8: (a) Original *Lena* image. (b) Blurred image with blur length  $L = 15$  and  $\theta = 30^\circ$ . (c) Restored using ML method (PSNR = 17.711 dB). (d) Restored after parameter estimation (PSNR = 24.4186 dB).

have been compared with the maximum likelihood (ML) method available in MATLAB toolbox. The restoration results for *Lena*, *Tree*, *Checkerboard* and *Stik* images are shown in Figures 2.8, 2.9, 2.10 and 2.11 respectively. Peak signal to noise ratio (PSNR in dB) is used as the quantitative parameter to evaluate the proposed scheme. It has been observed that the proposed scheme performs better than the ML scheme in terms of both subjective and objective evaluation.

### ***Experiment 3: Performance study in noisy situations***

The objective of this experiment is to test the robustness of the proposed parameter estimation method. Standard images are blurred with different length and angle. Then additive Gaussian noise of different SNR (25dB and 30dB) is added to create noisy blurred images. Gabor filter and RBFNN have been utilised to estimate the blur parameters. The restoration results for two different noisy blurred images are shown in the Figures 2.12 and 2.13 respectively. It has been

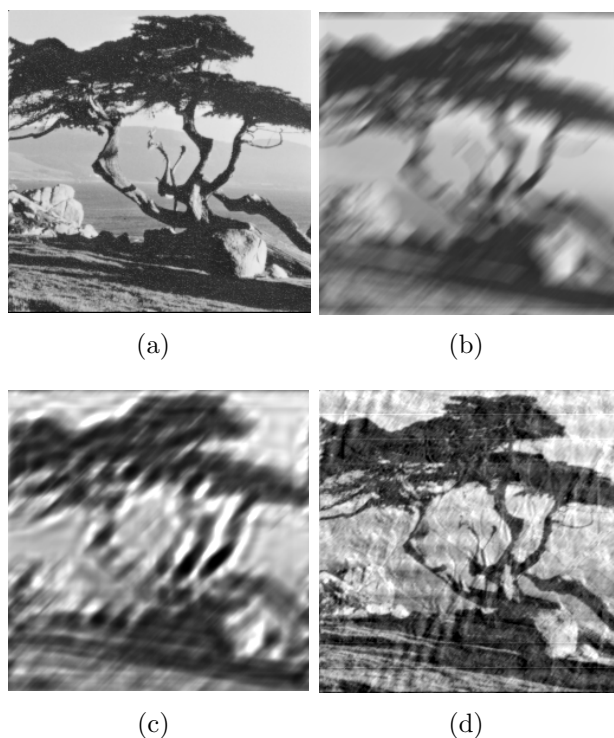


Figure 2.9: (a) Original *Tree* image. (b) Blurred image with blur length  $L = 20$  and  $\theta = 45^\circ$ . (c) Restored with ML method (PSNR = 17.047 dB). (d) Restored after parameter estimation (PSNR = 24.7106 dB).

observed that the restored results in noisy situations are also comparable to that of noise-free situations. The proposed method gives acceptable result upto 25dB SNR. Table 2.4 shows the estimated values of the blur parameters with different noisy situations.

Table 2.4: Estimated blur length  $L$  and blur angle  $\theta$  in noisy situation.

Image	Original $L$ and $\theta$	Estimated $L$ and $\theta$ (SNR = 25 dB)	Estimated $L$ and $\theta$ (SNR = 30 dB)
<i>Lena</i>	$L = 15, \theta = 30^\circ$	$L = 13.3, \theta = 31^\circ$	$L = 13.6, \theta = 31^\circ$
<i>Tree</i>	$L = 20, \theta = 45^\circ$	$L = 18.42, \theta = 46^\circ$	$L = 18.68, \theta = 46^\circ$
<i>Cameraman</i>	$L = 30, \theta = 40^\circ$	$L = 28.5, \theta = 39^\circ$	$L = 28.3, \theta = 39^\circ$
<i>Vase</i>	$L = 40, \theta = 60^\circ$	$L = 39, \theta = 59^\circ$	$L = 38.4, \theta = 59^\circ$

#### ***Experiment 4: Performance analysis with natural images***

To test the effectiveness of the proposed parameter estimation method on natural blurred images, an experiment has been conducted on photographic motion blurred image of the number plate of a car. A camera, which is mounted on

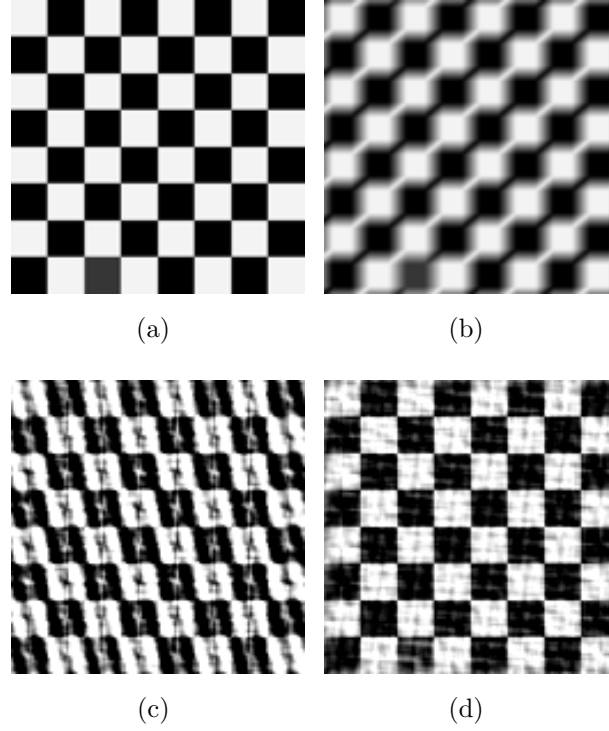


Figure 2.10: (a) Original *Checkerboard* image. (b) Blurred image with blur length  $L = 10$  and  $\theta = 40^\circ$ . (c) Restored with ML method (PSNR = 17.33 dB). (d) Restored after parameter estimation (PSNR = 26.21 dB).

a tripod, is positioned in front of car facing the number plate. When the shutter is released the arm of the tripod is moved with an arbitrary velocity, thereby capturing a motion blurred image whose PSF is unknown. The *number plate* image is subjected to the proposed and existing schemes. The blur length and angle of the number plate are estimated and found to be  $L = 20.2$  and  $\theta = 24^\circ$  respectively. It has been observed from the Figure 2.14 that the restoration performance of the proposed scheme is superior as compared to the existing scheme. PSNR comparison is not possible in this case as it requires the original image.

## 2.6 Summary

In this chapter, a deblurring scheme based on apriori blur parameter estimation from motion blurred images has been proposed. Subsequently, the estimated PSF from the parameters obtained are utilised for conventional restoration. To estimate the angle of blur, Gabor filter has been used whereas for length of blur,

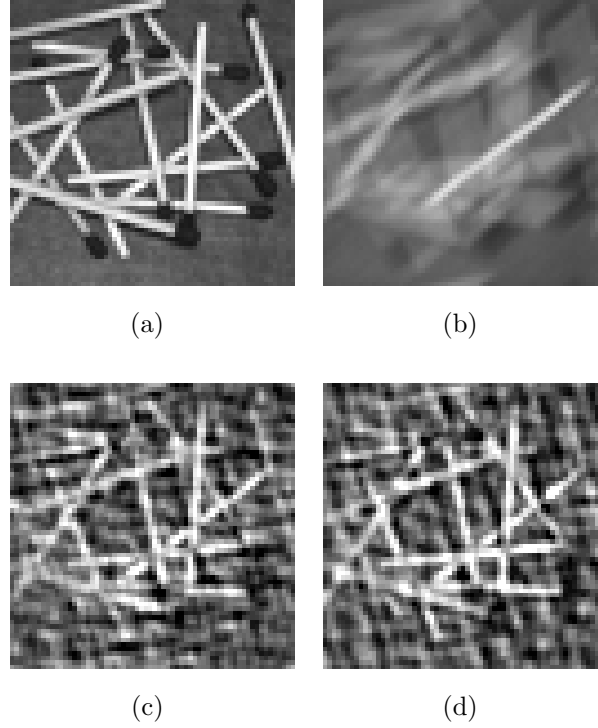


Figure 2.11: (a) Original *Stik* image, (b) Blurred image with blur length  $L = 12$  and  $\theta = 40^\circ$ , (c) Restored with ML method (PSNR = 19.53 dB), (d) Restored after parameter estimation (PSNR = 28.41 dB).

a RBFNN is utilised. Both the estimation schemes perform accurately even at the noisy conditions. This implicates the robustness of the proposed scheme. The simulation experiments have been conducted in MATLAB along with other competent schemes, and the results have been compared to show the efficacy of the scheme. Both standard and real time images have been included in simulation. It has been observed in all cases the proposed scheme has an upper hand in parameter estimation as well as restoration performance.

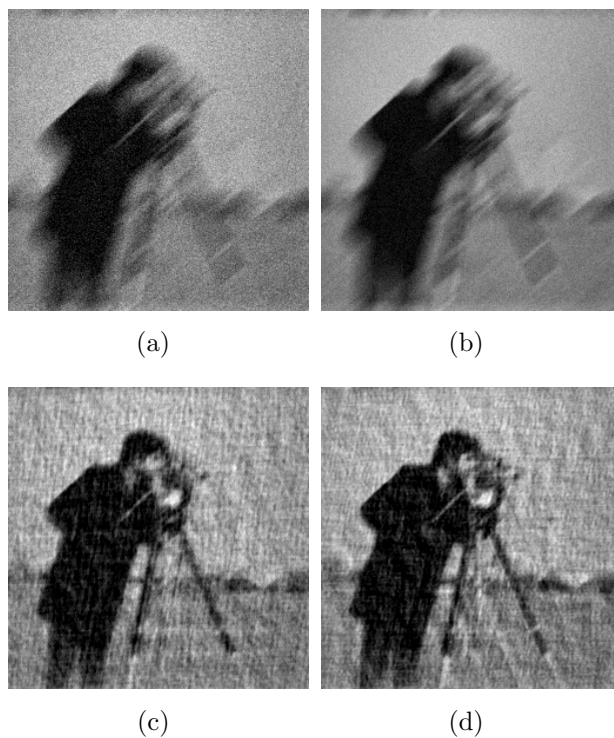


Figure 2.12: (a) Noisy blurred *Cameraman* image with  $L = 30$  and  $\theta = 40^\circ$  and Gaussian noise (SNR = 25 dB). (b) Noisy blurred *Cameraman* image with  $L = 10$  and  $\theta = 45^\circ$  and Gaussian noise (SNR = 30 dB). (c) Restoration results of (a) after parameter estimation (PSNR = 23.18 dB). (d) Restoration results of (b) after parameter estimation (PSNR = 25.21 dB).

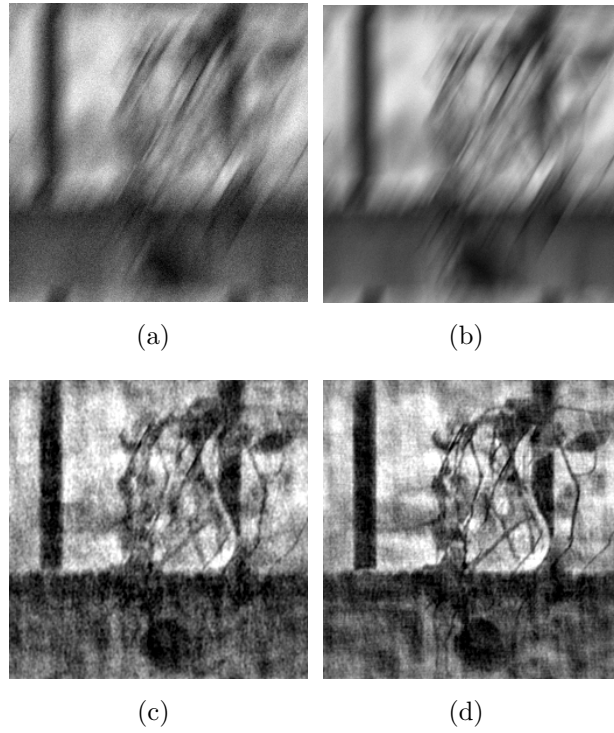


Figure 2.13: (a) Noisy blurred *Vase* image with  $L = 40$  and  $\theta = 60^\circ$  and Gaussian noise (SNR = 25 dB). (b) Noisy blurred *Vase* image with  $L = 40$  and  $\theta = 60^\circ$  and Gaussian noise (SNR = 30 dB). (c) Restoration results of (a) after parameter estimation (PSNR = 20.58 dB). (d) Restoration results of (b) after parameter estimation (PSNR = 22.19 dB).



Figure 2.14: (a) Noisy blurred *Car* image. (b) Extracted *Number plate* image. (c) Restored with ML method (d) Restoration result of *Number plate* image after parameter estimation.

# Chapter 3

Blur Parameter Identification  
using SVM

---

---

## Chapter 3

# Blur Parameter Identification using SVM

Images may be blurred by the imaging devices, or the medium through which the light propagates. The relative motion between the imaging device and/or the object also leads to image blurring. As discussed in the introductory chapter, the blurring process is assumed to be linear, and it is mathematically represented as the 2-D convolution between the original image and the degradation function also known as point spread function (PSF). The PSF parameters need to be identified prior to restoration. In this chapter, the length parameter of motion blur and sigma parameter of atmospheric turbulence blur have been identified using a multi-class support vector machine (SVM). Subsequently, the estimated blur parameter is employed to construct the PSF for restoration. SVM has been used to solve the estimation problem as a pattern classification problem.

The rest of the chapter is organised as follows. Support vector machine and different multi-class SVM approaches are described in Section 3.1. The proposed SVM based blur identification has been provided in Section 3.2. Experimental results and their discussion are detailed in Section 3.3. Finally Section 3.4 summarises the chapter.

### 3.1 Support Vector Machine

Support vector machine is a theoretically well motivated algorithm and originally developed at Bell laboratories by Vapnik [66]. SVM has been further modified by researchers [67] addressing the problem of large tasks, which consumes more

time and memory. SVMs have been successfully applied in many applications ranging from object detection and digit recognition [68,69], handwritten character recognition [70] information and image retrieval [71,72] and prediction [73–75]. Pattern Classification, automatic recognition are critical issues in decision making process [76]

For a binary classification problem having the training data points  $(x_i, y_i), i = 1, 2, \dots, m$ ,  $x_i \in R^p$  and  $y_i \in \{-1, 1\}$ , the SVM constructs a hyperplane which separates the data into two classes. The point  $x$  which lies on the hyper plane satisfies (3.1).  $W$  is normal to the hyperplane. The SVM tries to reduce the shortest distance between the separating hyperplane and the closest points to the hyperplane. There are many possible classifiers, which can classify the data into two classes. However, SVM constructs an optimal hyperplane which maximises its distance from the nearest data point of each class. This is shown in the Figure 3.1. Points on the dotted line are called support vectors.

$$W \cdot x + b = 0 \quad (3.1)$$

The above equation helps to find a decision boundary for a linearly separable data. For a nonlinear separable data, SVM creates a hyperplane by mapping the data set into a higher dimensional plane to convert it as a linear problem. This mapping to a high dimensional plane is achieved using a kernel function. SVM solves a convex optimisation problem through quadratic programming with linear constraints.

The SVM classifies the data into two classes by separating them into two categories. SVM models are equivalent to the classical multilayer perceptron (MLP) neural network where weights are optimised in order to classify the test data accurately. Traditional neural network solves a non-convex unconstrained minimisation problem. SVM models also use a training data set  $(x_i, y_i)$  like MLP and update the weights. However, it utilises a kernel function and weights of the networks are found by solving a quadratic programming problem with linear

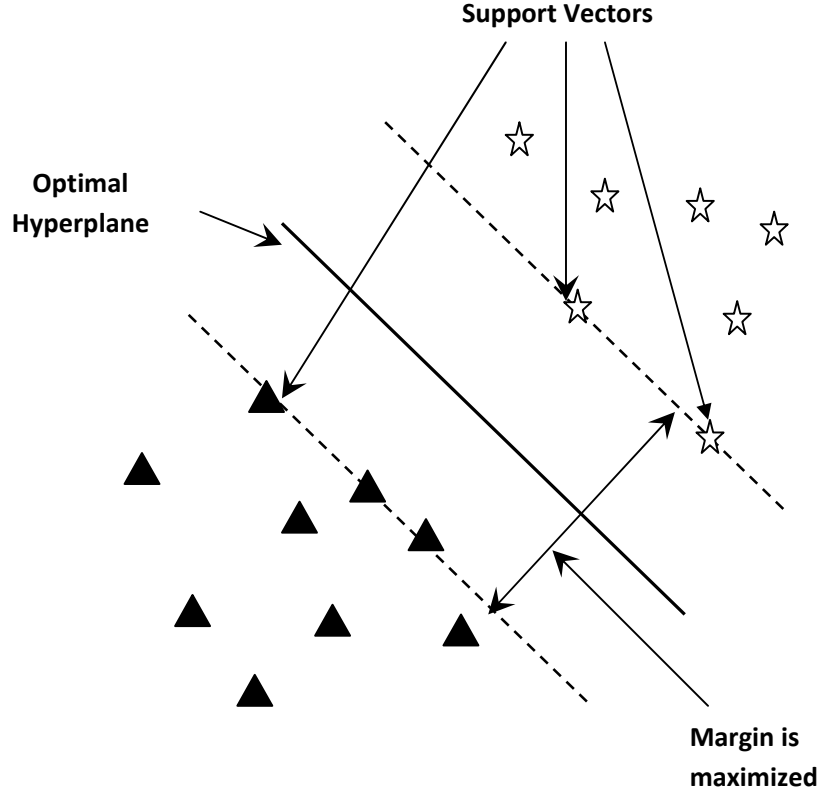


Figure 3.1: Optimal hyperplane classifying a two class problem

constraints. SVM solves the following optimisation problem

$$\min_{W, b, \xi, \xi^*} \frac{1}{2} W^T W + C \sum_{i=1}^N (\xi_i + \xi_i^*) \quad (3.2)$$

$$\text{Subject to } \begin{cases} y_i - (W^T \phi(x_i) + b) \leq \varepsilon + \xi_i \\ (W^T \phi(x_i) + b) - y_i \leq \varepsilon + \xi_i^* \\ \xi_i, \xi_i^* \geq 0; i = 1, 2, \dots, N \end{cases}$$

where  $\varepsilon$  is the error tolerance and  $\xi, \xi_i^*$  are the slack variables.

### 3.1.1 An Overview of Multi-class SVM Approaches

Initially, SVM has been developed for two-class classification problem. It is also called dichotomies decision problem. Later, SVM was also used to perform

multi-class classification. However, more study on multi-class SVM is required for its efficient utilisation in important applications. Some authors solve  $k$ -class classification problem with  $k$  number of SVMs. Authors in [77] solve the multi-class problem as a single optimisation problem. However, this approach is computationally expensive. Number of schemes has been proposed to solve the same problem as several binary problems which decompose the multi-class problem into a number of two-class problems [78]. Several widely used decomposition techniques are available, which are briefly discussed below.

### One-against one

In this technique,  $k(k-1)/2$  number of two-class classifiers are constructed using all combinations of the  $k$  classes for a  $k$ -class problem. All the classifiers are trained by taking the samples of the first class as positive and second class as negative. Then all these classifiers are combined using a winner take algorithm which uses a majority voting scheme. When a new sample is tested, it is tested with each of the hyperplanes and vote count of the class for that sample is noted. The sample is assigned to the class whose vote count is highest. For training, samples from only two classes out of  $k$  classes are used. So the number of samples required for training is less as compared with other decomposition schemes. The disadvantage of this method is that when  $k$  is large, a testing sample has to be tested with all the classes. This increases the testing time of the sample.

### One-against rest

One-against-all SVMs were first introduced by Vapnik [66]. For a  $k$ -class ( $k > 2$ ) problem,  $k$  numbers of hyperplanes are constructed. The  $j^{th}$  SVM is trained by taking the samples of  $j^{th}$  class as positive example and all others as negative examples. In the testing phase, the sample is presented to all the  $k$ -classifiers. The sample is assigned a class using the winner take all principle which assigns the class according to the maximum output among all  $k$ -classes. The limitation of this technique is that every classifier has to be trained with all available samples, which increase the training complexity. The performance of one-against-all SVMs was improved by the same author in 1998 [79].

### Directed Acyclic Graph SVM (DAGSVM)

This technique also uses  $k(k-1)/2$  number of classes and the training procedure is same as one-against-one. It is introduced by authors in [80]. During the testing phase, the algorithm takes a decision based on a rooted binary directed acyclic graph [80]. This method creates a model for each pair of the classes. When this model classifies two classes  $k_1$  and  $k_2$ , it doesn't vote for class  $k_1$  rather it gives vote against  $k_2$ . After this point it rejects all the models, which involves class  $k_2$ . After each classification by one of the models, one more class can be discarded. Therefore, after  $k-1$  steps, one class is left out, which becomes the prediction for the current testing sample. It gives a faster testing than one-against-one scheme having the same accuracy.

### SVM using Binary Decision Tree

This method has been developed by Madzarov et al. [81] which uses multiple SVMs arranged in a binary tree like structure. Each node in the tree represents a SVM which is used to train two classes. The algorithm measures the similarity between the remaining samples, and the two classes used for training. Using this similarity measure, all samples in the node are allotted to two sub nodes. This process is repeated till each node consists of samples from only one class. This technique has been utilised for motion blur and Gaussian blur parameter classification.

## 3.2 SVM based blur identification

The blurring process is same as low pass filtering where the filter output lacks high frequency components. Edges in an image are high frequency components and play a major role in image analysis. The high frequency regions in a *Cameraman* image for different motion blur length are shown in Figure 3.2. It is evident from Figure 3.2 as the blur length increases, edges become smoother. The variance parameter also decreases with the blurring strength in a blurred image. Changes of variance with blur length for different images are shown in Figure 3.3. Variance parameter has been used as one of the criteria to select the samples from the blurred image.

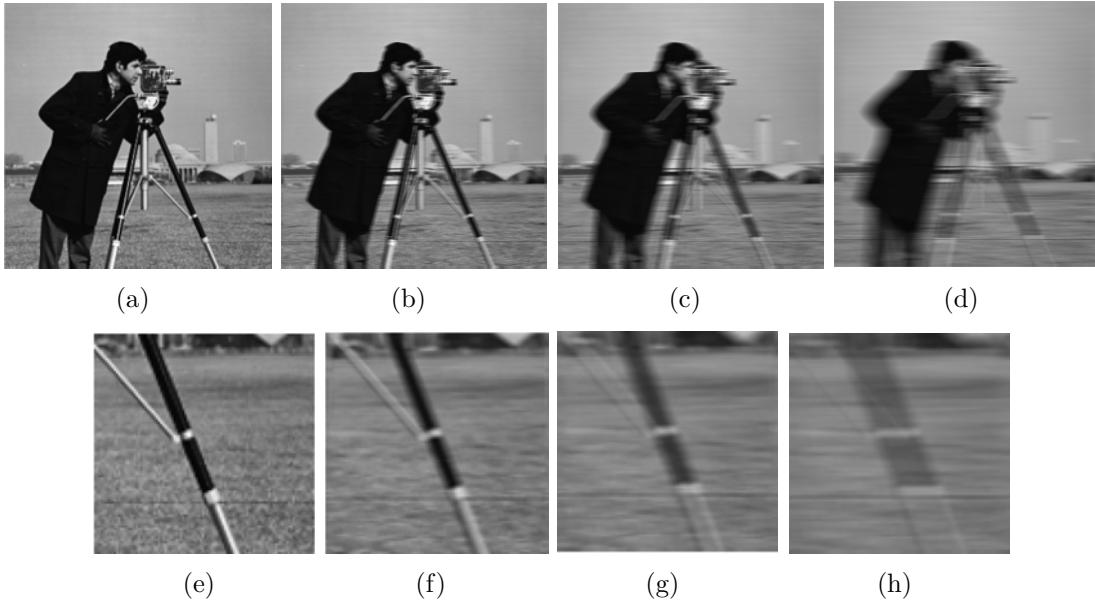


Figure 3.2: Behaviour of a high frequency portion of *Cameraman* image under horizontal motion blur of different length: (a) True image (b)  $L = 5$  (c)  $L = 10$ . (d)  $L = 20$  (e) True high frequency portion. (f)  $L = 5$  (g)  $L = 10$  (h)  $L = 20$ .

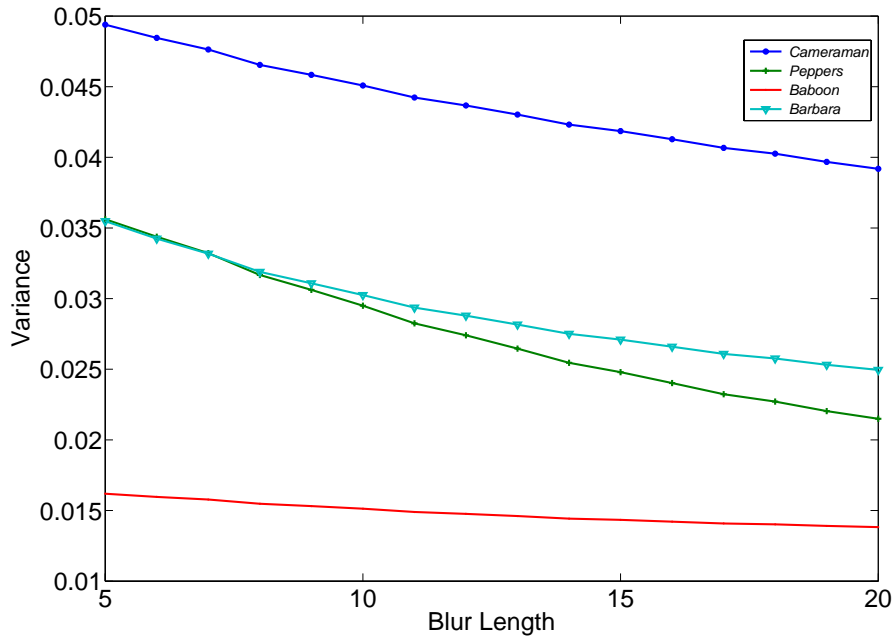


Figure 3.3: Variance curve for different blur length for different images.

The local variance of the blurred *Cameraman* image is computed by taking a window of size  $7 \times 7$  and the plot of centre pixel versus variance is shown in

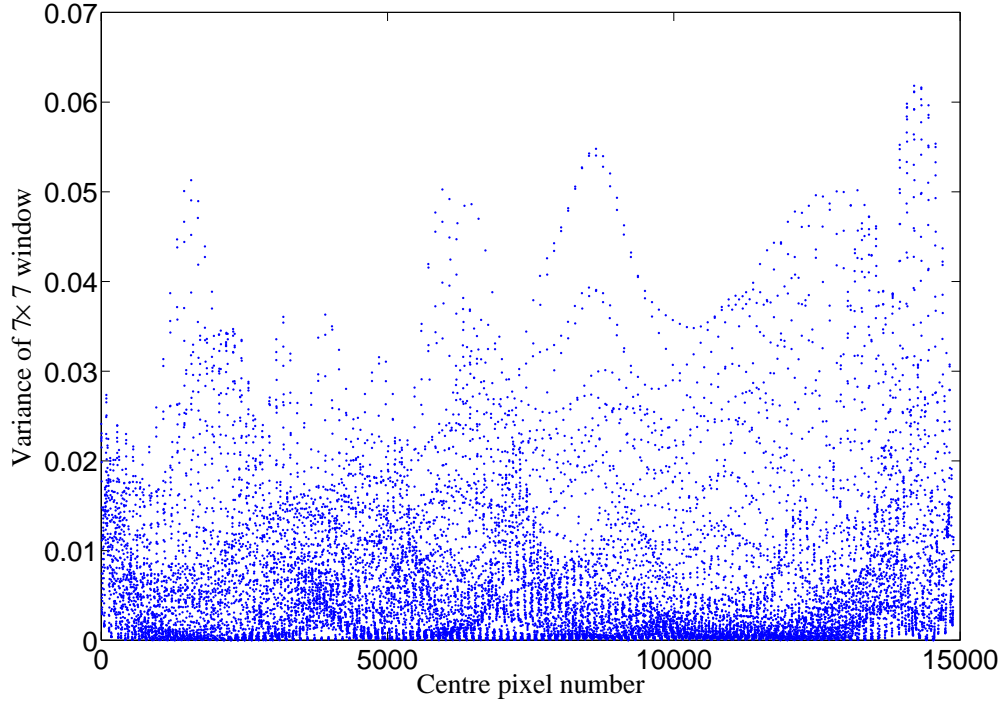


Figure 3.4: Variance plot of *Cameraman* image of size  $128 \times 128$  degraded with motion blur of  $L = 10$ .

Figure 3.4. The boundary pixels are ignored in the plot. It may be observed that maximum number of pixels are having smaller variance. These pixels corresponds to smooth region in the image and contain redundant information, whereas larger variance corresponds to edge regions. Higher variance regions are searched in a blurred image to construct the feature vector. If  $g$  represents the blurred image with blur length  $L$ , then the gray values of the window  $g_w$  is selected whose local variance is larger than a threshold. The feature vector is constructed as  $[g_w(:); L]^T$  where  $g_w(:)$  represents lexicographic ordering of the gray values of the window having larger local variance and  $L$  is the corresponding blur length.

### 3.2.1 Blur Classification

Blur classification is done by constructing a binary decision tree suggested in [81]. The SVM using the binary tree takes the advantage of efficient computation of binary tree architecture and good accuracy of SVM. The binary tree structure for 5-class classification problem is shown in Figure 3.5. A SVM is used in each node

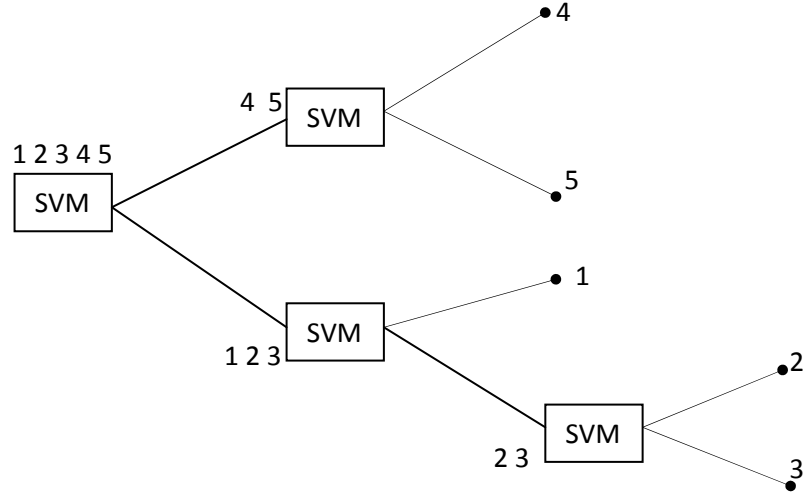


Figure 3.5: Binary decision tree SVM for blur classification.

and takes a decision about the assignment of an input pattern into one of the two possible classes. This is achieved by transferring the input pattern into either left or right sub-tree. This process is continued until each pattern is classified with a unique class.

At first, a training set is constructed using the feature vectors and trains the feature samples with a multi-class SVM. In the testing phase, features of the blurred image with different blur length are collected and fed to the trained SVM model. The output of the SVM is the identified blur length. The sigma parameter of the atmospheric turbulence blur has been identified in the similar manner by constructing a separate multi-class SVM model. Details of the experiments conducted are described in Section 3.3.

### 3.3 Simulation Results and Discussion

To validate the estimation performance of suggested multi-class SVM model, simulation has been performed using standard images including *Cameraman*, *Lena* and *Pepper* etc. The SVM is named after the image which is used to train it e.g. if *Cameraman* is used to train the SVM, the corresponding SVM is called as

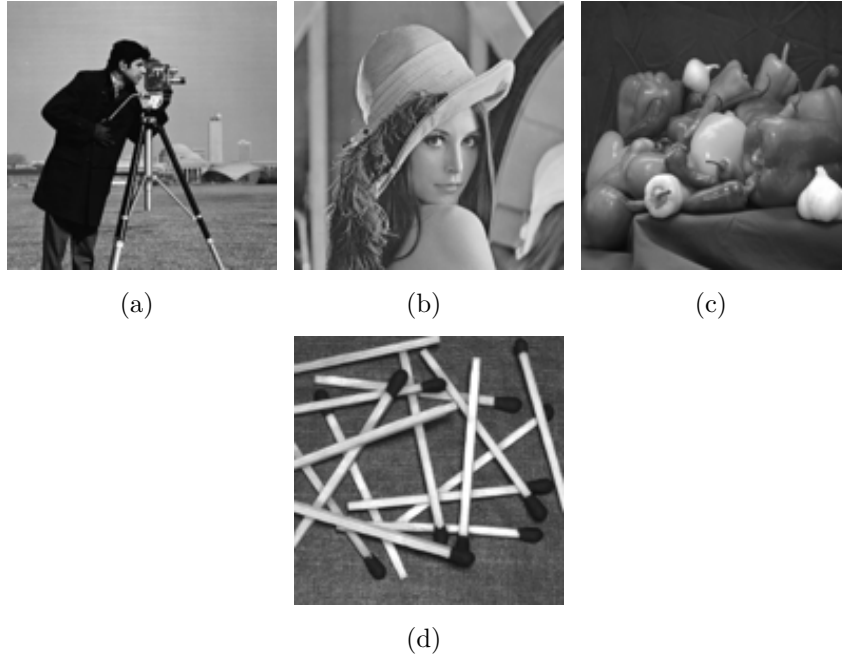


Figure 3.6: Standard images used for testing.

*Cameraman* SVM model. All the images used for training and testing are scaled in the range  $[0, 1]$  in the spatial domain. A window of size  $7 \times 7$  has been used for variance computation. So the size of one sample vector used in the SVM is  $[49 \times 1]$ . The SVM model trained with *Cameraman* image is tested with a different image. Four standard images as shown in Figure 3.6 are taken for simulation to create a test bed. For better understanding the overall simulation is divided in four experiments and described in sequel.

***Experiment 1: Cameraman SVM model for motion blurred image***

The *Cameraman* image is horizontally motion blurred with  $L = 2$  to 30 with a step of 2. The samples from the blurred image are taken by considering the window whose variance is larger than 0.02. The SVM is trained with the collected samples and a *Cameraman* SVM model is created. The SVM model is validated by testing the samples of *Cameraman* image as well as *Lena* image. The vote counts for each of the estimated class for *Cameraman* image and *Lena* image are noted and has been shown in Tables 3.1 & 3.2 respectively.

Values in each column show the number of votes won by different blur lengths. The maximum vote obtained by any row in a particular column is the estimated class which is equal to the blur length. It should be noted that the samples from

Table 3.1: Blur identification performance of *Cameraman* multi-class SVM model for *Cameraman* test image.

	Actual Blur Length				
	Samples→	913	480	276	175
	Blur Length ( $L$ )↓→	5	9	13	17
Estimated Blur Length	5	<b>861</b>	128	29	16
	9	44	<b>321</b>	64	28
	13	3	22	<b>154</b>	41
	17	5	9	30	<b>88</b>

Table 3.2: Blur identification performance of *Cameraman* multi-class SVM model for *Lena* test image.

	Actual Blur Length				
	Samples→	756	515	326	190
	Blur Length ( $L$ )↓→	5	9	13	17
Estimated Blur Length	5	<b>689</b>	117	47	24
	9	38	<b>334</b>	65	27
	13	18	52	<b>197</b>	43
	17	11	12	17	<b>96</b>

*Lena* image are different from the sample used in the training. The proposed multi-class SVM identifies the blur length accurately for the image used during training as well as for any test image. To create the SVM model LIBSVM package [82] is used with the default radial basis kernel function. The value of  $C$  is chosen through experimentation and for the *Cameraman* image SVM model it is taken as 1. Through experimentation, error tolerance and gamma values are chosen to be 0.0001 and 5 respectively. These parameters can also be optimised for different images with different degradations. After the blur parameter is identified, the PSF is constructed and the degraded images are restored. The restoration results for the *Lena* image are shown in Figure 3.7.

### ***Experiment 2: Pepper SVM model for atmospheric blurred image***

In this experiment, the *Pepper* image is degraded with atmospheric turbulence with different  $\sigma$  values in the range  $[0.5 - 5]$ . The training samples of the blurred *Pepper* image are collected using the same procedure as explained in *Experiment*

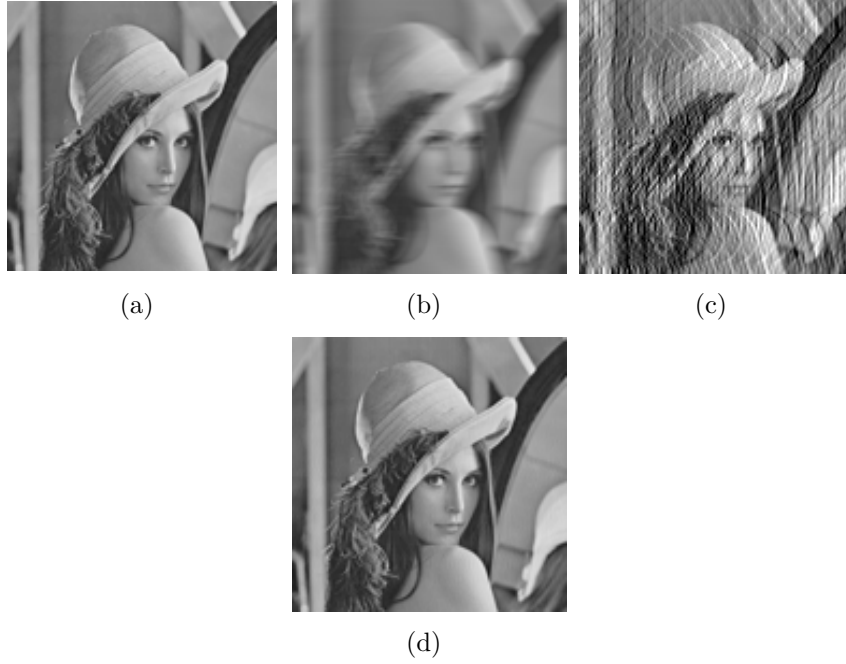


Figure 3.7: Restoration results for *Lena* image: (a) True image. (b) Motion blurred ( $L = 9$ ). (c) Restored after parameter estimation using RBFNN. (d) Restored after parameter estimation using multi-class SVM.

1. Total 1392 samples have been used for training the SVM. To validate the effectiveness of the proposed scheme, testing samples from *Camerman* image also have been used. The performance of the *Pepper* multi-class SVM model for different images are shown in Tables 3.3 and 3.4 respectively. It is inferred from the tabular results that the proposed *Pepper* multi class SVM model also works well for images degraded with atmospheric turbulence blur and it achieves generalisation in terms of different images. Same voting strategy has been followed in this experiment to identify the actual sigma value. The value of  $C$  is taken as 1 and radial basis kernel has been used in the LIBSVM package. Gamma value and error tolerance are chosen to be 4 and 0.0001 respectively through experimentation. The restored image is obtained after constructing the PSF from the identified sigma parameter.

### ***Experiment 3: Comparison with other schemes***

The proposed scheme is compared with the existing blur parameter estimation schemes, like RBFNN blur length estimator described in Chapter 1 and Rekleitis's [64] scheme. To compare the proposed scheme with the RBFNN length

Table 3.3: Blur identification performance of *Pepper* multi-class SVM model for Gaussian blurred *Pepper* test image.

	Actual Sigma					
	Samples→	449	245	224	234	241
	Gaussian Blur ( $\sigma$ )↓→	0.5	1	2	3	4
Estimated Sigma	0.5	<b>444</b>	6	0	0	1
	1	3	<b>222</b>	1	0	0
	2	1	11	<b>128</b>	48	14
	3	0	14	52	<b>103</b>	97
	4	1	2	43	83	<b>129</b>

Table 3.4: Blur identification performance of *Pepper* multi-class SVM model for Gaussian blurred *Camerman* test image.

	Actual Sigma					
	Samples→	348	328	208	165	178
	Gaussian Blur ( $\sigma$ )↓→	0.5	1	2	3	4
Estimated Sigma	0.5	<b>312</b>	8	0	0	0
	1	23	<b>252</b>	1	0	2
	2	12	51	<b>138</b>	38	24
	3	0	14	46	<b>93</b>	67
	4	1	2	23	34	<b>85</b>

estimator, blur angle is estimated with the Gabor filter. Then the images are converted to horizontal blurred image. The estimated values of blur length using different schemes are shown in Table 3.5. It may be observed that SVM method estimates the blur length exactly whereas other methods fail to do so in a noise free environment.

#### ***Experiment 4: Testing with noisy situations***

To test the efficacy of the proposed scheme in presence of noise, another SVM model is created using *Lena* image. It is degraded with horizontal motion blur of different blur lengths similar to *Experiment 1*. Subsequently, Gaussian noise of different strengths SNR (25 dB, 30dB, 40dB) is added to the blurred *Lena* images to create a pool of blurred noisy samples for training. Training patterns are collected using a  $7 \times 7$  window of blurred noisy image having its target value as the blur length ( $L$ ). The trained *Lena* SVM model is tested with some samples from

Table 3.5: Estimated blur length  $L$  in noise free situations

Image	Original $L$	RBFNN	Rekleitis	Estimated $L$ (SVM)
<i>Lena</i> (128x128)	$L = 9$	$L_{est} = 8$	$L_{est} = 6$	$L_{est} = 9$
<i>Cameraman</i> (128x128)	$L = 13$	$L_{est} = 12$	$L_{est} = 11$	$L_{est} = 13$
<i>Pepper</i> (128x128)	$L = 17$	$L_{est} = 18$	$L_{est} = 20$	$L_{est} = 17$
<i>Stik</i> (128x128)	$L = 20$	$L_{est} = 19$	$L_{est} = 17$	$L_{est} = 20$

Table 3.6: Blur length  $L$  estimation in noisy situation

Image	Original $L$	SNR = 30 dB		SNR = 40 dB	
		RBFNN	SVM	RBFNN	SVM
<i>Lena</i> (128x128)	$L = 9$	$L_{est} = 11$	$L_{est} = 14$	$L_{est} = 10$	$L_{est} = 12$
<i>Cameraman</i> (128x128)	$L = 13$	$L_{est} = 12$	$L_{est} = 17$	$L_{est} = 11$	$L_{est} = 16$
<i>Pepper</i> (128x128)	$L = 17$	$L_{est} = 18$	$L_{est} = 20$	$L_{est} = 17$	$L_{est} = 20$
<i>Stik</i> (128x128)	$L = 20$	$L_{est} = 19$	$L_{est} = 17$	$L_{est} = 21$	$L_{est} = 16$

*Lena* image not considered during training and as well as with other blurred noisy images. For comparison, the blur length  $L$  of the same images is also estimated using some well known techniques. The results are summarised in Table 3.6. It is observed that estimated parameters are deviated from the actual blur parameters. The deviation increases with decrease in SNR. The RBFNN has the superior performance as compared to multi-class SVM in presence of noise.

### 3.4 Summary

A novel method has been proposed to identify the blur parameters from the blurred images. The blur identification problem is modelled as a multi-class classification problem and has been solved through multi-class SVM. For a  $k$  class problem  $k - 1$  numbers of hyper planes are constructed. These hyperplanes are used as the decision boundaries which are used to assign a class to an unknown sample. The suggested scheme achieves generalisation in terms of different images and estimates the blur parameters accurately. However, for a different blur type, the SVM needs to be retrained adding that class into the training set. The proposed scheme gives better result than RBFNN in noise free environment but fails to estimate accurately in noisy situation. The proposed scheme is applicable to images

degraded with horizontal motion blur and images degraded with atmospheric turbulence blur. However, this proposed method can be applied to estimate blur parameters only in noise free situation.

# Chapter 4

PSO Based SVR  
for Blind Image Restoration

## Chapter 4

# PSO based SVR for Blind Image Restoration

The blurring in an image can be visualised as lowpass filtering of the image where the edges are lost. It occurs in several situations like camera mis-focus, relative motion between object and camera, atmospheric turbulence and other similar conditions. Images degraded with different blurs look significantly different from each other. However, they look much similar if a block is taken from one image is compared with the same block in the other image. Based on this observation, in this chapter a generalised solution has been proposed to restore an image affected with blur and noise. The suggested method solves a blind deconvolution problem by direct restoration of the true image pixels without estimating the PSF. It utilises the support vector regression (SVR) approach [47] towards blind image restoration with its parameters optimised by particle swarm optimisation (PSO). A direct mapping of the true image from the degraded observation is obtained after SVR parameter optimisation. Different SVR models are created using some known training samples by optimising the parameters. Subsequently, the model is used to restore different blurred images not used during training. The proposed scheme adapts to different blurs with Gaussian noise of various strengths. Exhaustive simulation results support the performance of the proposed scheme.

The rest of the chapter is organised as follows. Section 4.1 gives a brief introduction to the SVR. The role of parameters on the performance of the SVR is discussed in Section 4.2. Section 4.3 presents the proposed parameter optimisation scheme using PSO. Experimental results with discussions are given in Section 4.4.

Finally, a summary of the chapter is presented in Section 4.5.

## 4.1 Support Vector Regression

An introduction to support vector machine has been already provided in the previous chapter. In this chapter, SVM has been utilised to solve a regression problem. Let us consider a training set  $(x_1, y_1), (x_2, y_2), \dots, (x_N, y_N)$  from a vector,  $x_i \in \mathbb{R}^n$  with corresponding targets  $y_i, i = 1, 2, \dots, N$ .  $\varepsilon$ -SVR determines a linear function defined on  $x_i$  as,

$$f(x) = \langle w, x \rangle + b \quad (4.1)$$

where  $w$  is a high-dimensional weight vector and  $b \in \mathbb{R}$  as the bias such that there is at most  $\varepsilon$  distance from the actual data and  $f(X)$  should be flat.  $\langle \rangle$  denotes the dot product. No care is taken as long as errors are less than  $\varepsilon$ . But, any deviation more than  $\varepsilon$  is not accepted. Flatness means the value of  $w$  should be as small as possible. This can be written as convex optimisation problem:

$$\text{Minimise } \frac{1}{2} \|w\|^2 \quad (4.2)$$

$$\text{Subject to } \begin{cases} y_i - \langle w, x_i \rangle - b \leq \varepsilon \\ \langle w, x_i \rangle + b - y_i \leq \varepsilon \end{cases}$$

In this case it is assumed that a function  $f$  exists which approximates the data set  $(x_i, y_i)$  with  $\varepsilon$  precision. Introducing slack variables  $\xi_i, \xi_i^*$ , the problem can be stated as [66],

$$\text{Minimise } \frac{1}{2} \|w\|^2 + C \sum_{i=1}^N (\xi_i + \xi_i^*) \quad (4.3)$$

$$\text{Subject to } \begin{cases} y_i - \langle w, x_i \rangle - b \leq \varepsilon + \xi_i \\ \langle w, x_i \rangle + b - y_i \leq \varepsilon + \xi_i \\ \xi_i, \xi_i^* \geq 0 \end{cases}$$

The parameter  $C$  controls the trade-off between the flatness of  $f$  and tolerance level of error  $\varepsilon$ . This deals with a  $\varepsilon$ -insensitive loss function expressed as,

$$|\xi|_\varepsilon = \begin{cases} 0, & \text{if } |\xi| \leq \varepsilon \\ |\xi| - \varepsilon, & \text{otherwise} \end{cases} \quad (4.4)$$

The graphical representation of the  $\varepsilon$ -insensitive loss function is shown in the Figure 4.1. The optimisation problem defined in (4.3) is easily solved in its dual

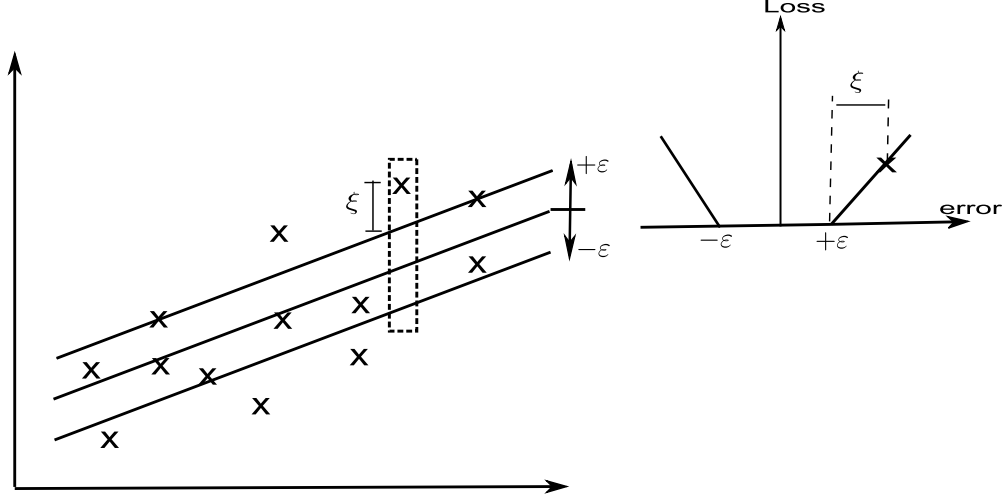


Figure 4.1: The soft margin loss setting for a linear SVM.

formulation. The dual optimisation problem can be written as [83],

$$\begin{aligned} & \text{maximise} \begin{cases} -\frac{1}{2} \sum_{i,j=1}^N (\alpha_i - \alpha_i^*) (\alpha_j - \alpha_j^*) \langle x_i, x_j \rangle \\ -\varepsilon \sum_{i=1}^N (\alpha_i + \alpha_i^*) + \sum_{i=1}^N y_i (\alpha_i - \alpha_i^*) \end{cases} \\ & \text{subject to} \sum_{i=1}^N (\alpha_i - \alpha_i^*) = 0 \text{ and } \alpha_i, \alpha_i^* \in [0, C] \end{aligned} \quad (4.5)$$

where  $\alpha_i, \alpha_i^*$  are Lagrange multipliers.

To address non linear regression problems, the linear SVR is extended to nonlinear SVR by mapping the input space into a high dimensional feature space through a kernel function  $\phi(x)$ . In such case,  $(x, x')$  is replaced by  $k(x, x')$ . Typical kernel functions used in the SVR are RBF, polynomial, linear and defined as,  
*RBF kernel:*

$$k(x_i, x_j) = \exp \left( -\frac{\|x_i - x_j\|^2}{2\sigma^2} \right) \quad (4.6)$$

*Polynomial kernel:*

$$k(x_i, x_j) = (1 + x_i \cdot x_j)^d \quad (4.7)$$

*Linear Kernel:*

$$k(x_i, x_j) = x_i^T x_j \quad (4.8)$$

In the above equations  $x_i$  and  $x_j$  are input vector spaces. The parameter  $\sigma$  in (4.6) represents the spread of Gaussian kernel and the parameter  $d$  in (4.7) is the degree of the polynomial. The operator  $(\cdot)$  between  $x_i$  and  $x_j$  represents the inner product.

## 4.2 Influence of the parameters on the performance of SVR

SVR is characterised by number of parameters and Vapnik [66] addressed that the two most relevant are the kernel parameter  $\sigma$  and the penalty parameter  $C$ . The parameter  $C$  determines the trade-off between the complexity of the model and approximated error. The parameter  $C$  is normally optimised using cross validation. However, for a large volume of data, the cross validation technique increases the complexity of the problem. The kernel parameter  $\sigma$  in SVR plays a major role. If it is overestimated, the exponential will behave almost linearly and it would loose the power of transforming to a higher dimensional plane. On the other hand, if it is underestimated, regularisation will be affected and the decision boundary will be sensitive towards noise in training data.

If SVR is applied for image restoration without parameter optimisation, the parameters are adjusted heuristically for good result. Such parameters donot provide satisfactory results across different images and blur conditions. So an adaptive parameter estimation for given degradation is necessary. In the proposed scheme, different types of blurs have been considered along with Gaussian noise of different strengths which is used in practice.

### 4.3 Proposed PSO based SVR for Blind Image Restoration

In the present work, the SVR approach towards blind image deconvolution is improved by optimising the parameters of SVR. A Genetic algorithm (GA) based approach has been proposed [84] to optimise the SVR parameters. Authors in [85] have also suggested a GA based approach for optimising the SVM parameters. The PSO is utilised in this work for faster convergence and easier implementation. PSO also has fewer adjusting parameters than GA. PSO is discussed in a nutshell followed by detailed procedure of our proposed method in sequel.

#### 4.3.1 Particle Swarm Optimisation

PSO is a stochastic optimisation technique initially developed by Eberhart and Kennedy [86] and subsequently modified to a more generalised form [87, 88]. It is an evolutionary computation technique based on intelligent behaviour of swarm. A swarm consists of particles, which represent the solution. They fly in a multidimensional search space. Each particle changes its position according to its own experience, and experience collected from the neighbouring particles. In this way, the particles move towards the best solution. The performance of each particle is measured using a fitness function which is application dependent. In the present work, root mean square error function (RMSE) is used as the fitness function and is defined as,

$$RMSE = \left\{ \frac{1}{MN} \sum_{i=1}^{MN} (\hat{f} - f)^2 \right\}^{1/2} \quad (4.9)$$

where  $f, \hat{f}$  are true and restored images respectively each of size  $M \times N$ .

The present particle position (*presentx*) and velocity ( $V$ ) of each particle are updated using the following two update equations defined as,

$$\begin{aligned} V &= V + c1 * rand() * (pbest - presentx) \\ &\quad + c2 * rand() * (gbest - presentx) \end{aligned} \quad (4.10)$$

and

$$presentx = presentx + V; \quad (4.11)$$

where,  $rand$  is a random number between 0 and 1 and  $c1$ ,  $c2$  are two weighting constants or accelerating constants. The local best solution of the particle is defined as the  $pbest$  and  $gbest$  is the global best solution of all the particles achieved so far.

### 4.3.2 Parameter Optimisation of SVR

The proposed scheme is similar to the SVR approach [47] in which a pair of blurred and true images are used for training purpose. A mapping from the blurred image to the true image is done. To create a training pattern of pixels, a  $7 \times 7$  window from the blurred image is taken and stacked in a column format results to a pattern of size  $[49 \times 1]$ . The corresponding centre pixel from the true image is used as target value. The window is shifted from top to bottom and the gray values of the true image are recorded. The SVR is trained with a random initial set of parameters. The parameters of the SVR i.e.  $C, \sigma$  are updated till the RMSE between the restored image and true image is below a threshold level (T). In this work, a threshold value of 0.01 is used in the experiments.

The trained SVR with the optimised set of parameters is used to restore images pixel by pixel. The flow chart of the proposed optimisation scheme is shown in Figure 4.2.

## 4.4 Experimental Results

Experiments are carried out in the MATLAB environment to validate the efficacy of the proposed scheme in different blur and noise conditions. The proposed approach is tested on several standard images, including *Lena*, *Cameraman*, *Pepper* with the SVR model. The SVR model trained with particular image is used to restore a different blurred image, e.g., the SVR model created with *Lena* image is used to restore *Pepper* image. The SVR model trained with *Lena* image is denoted as *Lena* SVR and so on. The peak signal to noise ratio (PSNR) is used

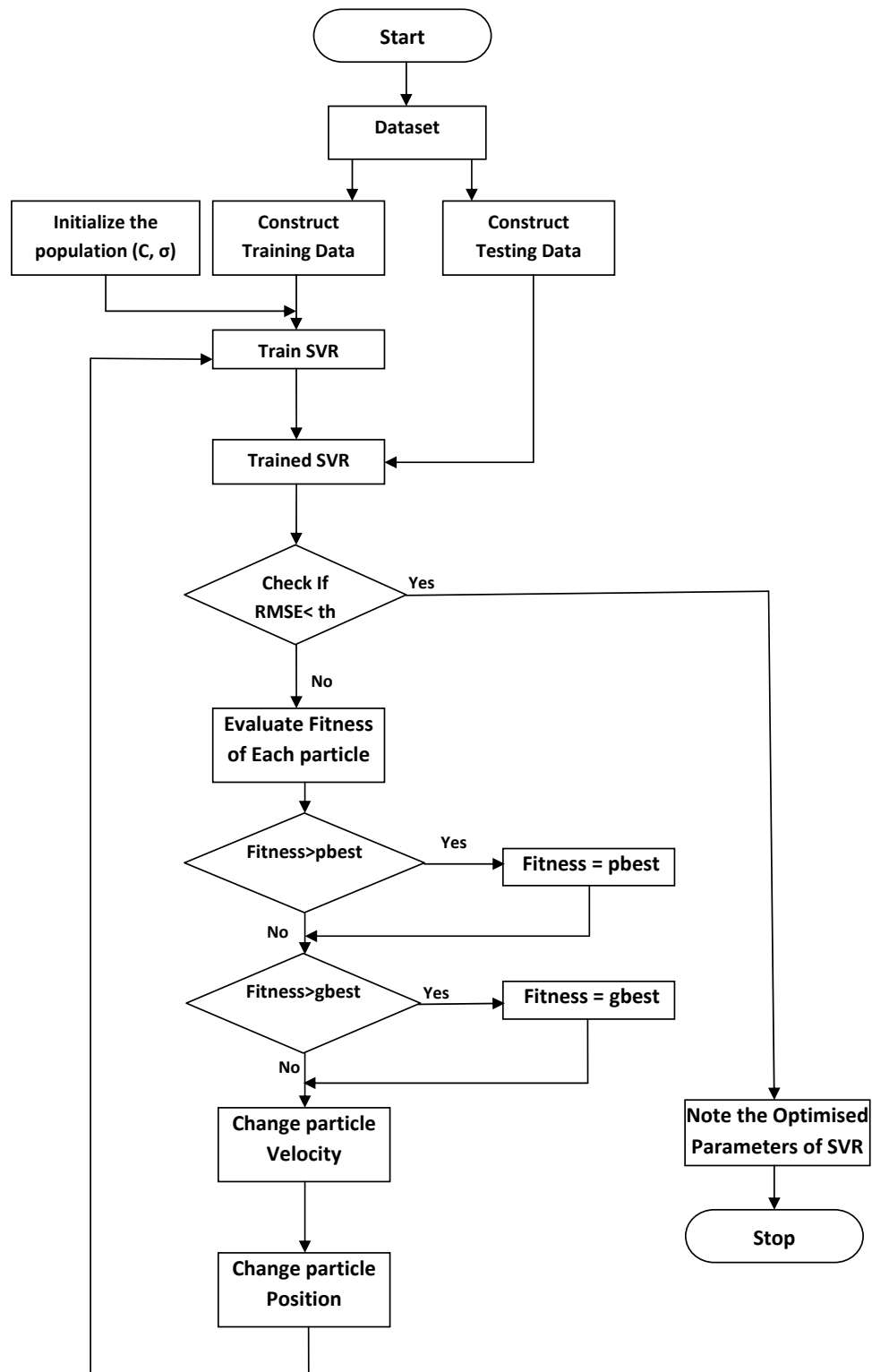


Figure 4.2: Flow chart of the proposed scheme for SVR parameter optimisation.

as the performance measure to compare the results of the proposed scheme with other standard schemes.

The images used in training and testing are normalised in the range  $[0-1]$ . In all experiments, images of size  $128 \times 128$  are considered. The PSO parameters are kept fixed for each experiment. Population size of 30 has been used in the simulation. The value of  $c1$  and  $c2$  were chosen to be 1.4 through experimentation.

Blurred images are created using the standard blur functions including motion blur, out-of-focus blur and Gaussian blur. Then Gaussian noise is added to the blurred images. To compare the results, the images are also restored with PCA [37], SVR [47] and with maximum likelihood estimation technique for blind image deconvolution.

The overall simulation study is divided into three experiments and are discussed in sequel.

#### ***Experiment 1: Restoration of Lena image with motion blur***

Standard *Lena* image is degraded with motion blur parameters  $L = 10$  &  $\theta = 45^\circ$ . Subsequently, Gaussian noise of strength (SNR = 40 dB) is added to the resultant image. Training patterns are accumulated from the  $7 \times 7$  window of blurred noisy image having its target pixel as the corresponding centre pixel in the original image. In *Lena* SVR model, the optimised parameters are found to be  $C = 2.3, \sigma = 0.8$ . The *Lena* SVR model is subjected to restoration of degraded *Lena* image, with different motion blur parameters and Gaussian noise ( $L = 8$  &  $\theta = 30^\circ$ , SNR = 40 dB). Restoration results are shown in Figure 4.3. Further, *Lena* images degraded with defocus aberration and noise ( $R = 3$  & SNR = 30 dB) and Gaussian blur and noise ( $\sigma = 0.9$ , SNR = 40 dB) respectively are also subjected to restoration using the aforesaid *Lena* SVR model. The objective parameter PSNR (dB) is computed and compared with other schemes as shown in Table 4.1.

It may be observed that the proposed optimised SVR shows superior performance. It may also be noted that, optimised SVR on one degradation phenomenon works well for restoring images degraded with similar type of blur as well as for other types of blurs in the same image.

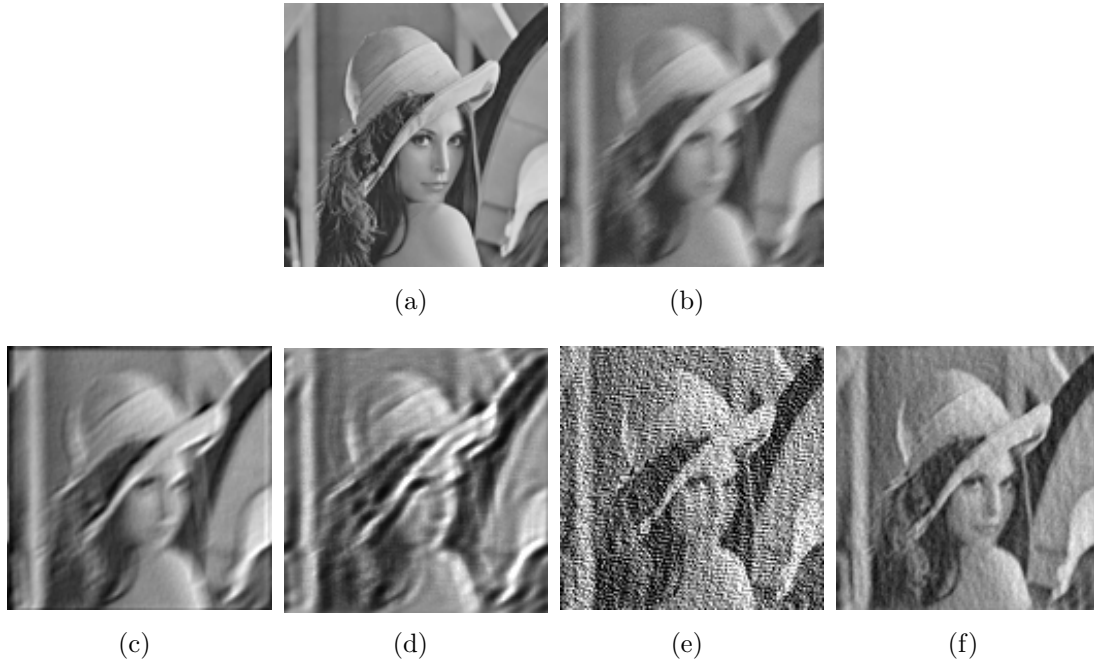


Figure 4.3: Restoration of *Lena* Image: (a) Original *Lena* image. (b) Motion Blurred *Lena* image (SNR = 40 dB). (c) Restored with PCA. (d) Restored with Maximum Likelihood. (e) Restored with SVR without parameter optimisation ( $C = 1$ ). (f) Restored with PSO based *Lena* SVR model

Table 4.1: PSNR (dB) comparison of restored *Lena* image using *Lena* SVR model and other schemes for different degradations.

Blur & Noise	PCA	Maximum likelihood	SVR	PSO based SVR
Motion blur ( $L = 8, \theta = 30^\circ$ , SNR = 40 dB)	18.45	19.21	25.81	30.23
Out-of-focus ( $R = 3$ , SNR = 30 dB)	20.34	22.31	30.41	33.08
Gaussian ( $\sigma = 0.9, 5 \times 5$ , SNR = 40 dB)	19.32	20.37	27.11	29.21

### ***Experiment 2: Restoration of images not considered during training***

To validate the efficacy of the trained *Lena* SVR model (*Experiment 1*), *Pepper* images with various degradations are used for testing. The restored images for out-of-focus blur ( $R = 3$ , SNR = 30 dB) are shown in Figure 4.4 and comparative analysis of PSNR (dB) is shown in Table 4.2. Further, an optimised *Pepper* SVR model is generated in the similar direction and *Cameraman* image is used for testing with different degradations. The performance analysis is shown in

Table 4.3.

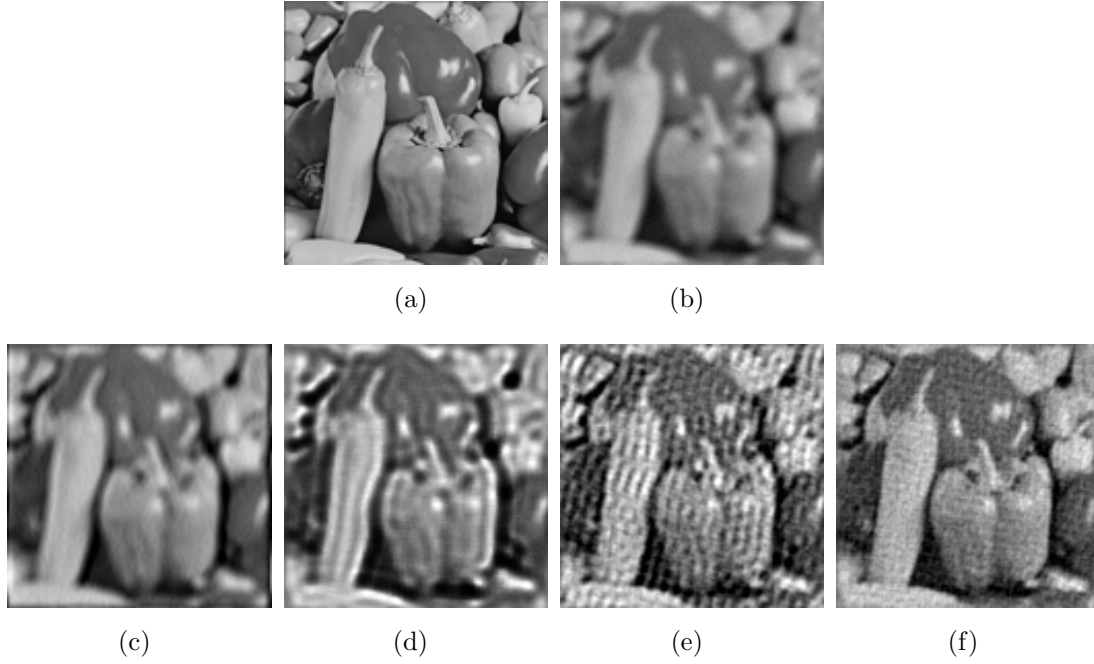


Figure 4.4: Restoration of *Pepper* Image: (a) Original *Pepper* image. (b) Out-of-focus blurred *Pepper* image (SNR = 40 dB). (c) Restored with PCA. (d) Restored with Maximum Likelihood. (e) Restored with SVR without parameter optimisation ( $C = 1$ ). (f) Restored with PSO based *Lena* SVR model.

Table 4.2: PSNR (in dB) comparison of restored *Pepper* image using *Lena* SVR model and other schemes for different degradations.

Blur & Noise	PCA	Maximum likelihood	SVR	PSO based SVR
Motion blur ( $L = 8, \theta = 30^\circ$ , SNR = 40 dB)	16.51	18.14	24.65	29.12
Out-of-focus ( $R = 3$ , SNR = 40 dB)	19.08	21.36	29.04	31.48
Gaussian ( $\sigma = 1, 3 \times 3$ , SNR = 40 dB)	20.28	21.75	29.61	31.54

It is inferred that the optimised SVR trained with one image on a particular degradation is capable of restoring different images with various degradations.

### ***Experiment 3: Restoration of photographic blurred image***

The proposed scheme is subjected to two naturally blurred images captured on a camera (Canon EOS 400D). The *Canon* image has the out-of-focus blur and the *Key* image has the motion blur. The noise on both the images are assumed to be

Table 4.3: PSNR (in dB) comparison of restored *Cameraman* image using *Pepper* SVR model and other schemes for different degradations.

Blur & Noise	PCA	Maximum likelihood	SVR	PSO based SVR
Motion blur ( $L = 12, \theta = 25^\circ$ , SNR = 40 dB)	19.15	20.34	27.98	30.24
Out-of-focus ( $R = 3$ , SNR = 30 dB)	19.14	21.31	30.43	32.16
Gaussian ( $\sigma = 1.5, 3 \times 3$ , SNR = 40 dB)	19.87	21.37	28.41	31.42

Gaussian. The restoration performances on the images using the Lena SVR model along with other schemes are shown in Figures 4.5 & 4.6 respectively. It may be observed that the proposed PSO based SVR scheme has superior performance as compared to the existing schemes.

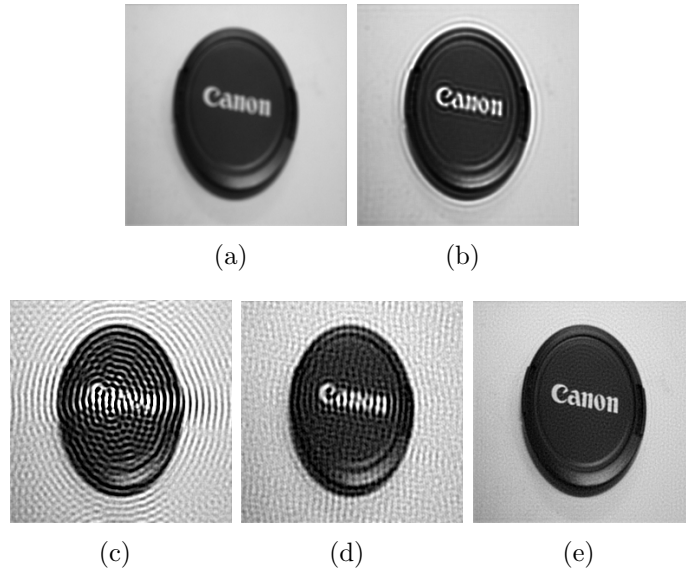


Figure 4.5: Restoration of *Canon* Image: (a) Blurred and noisy *Canon* image. (b) Restored with PCA. (c) Restored with Maximum Likelihood. (d) Restored with SVR without parameter optimisation ( $C = 1$ ). (e) Restored with PSO based *Lena* SVR model.

## 4.5 Summary

In this chapter, a PSO based parameter optimisation scheme for SVR has been proposed which in turn is utilised for blind image restoration. The parameters of

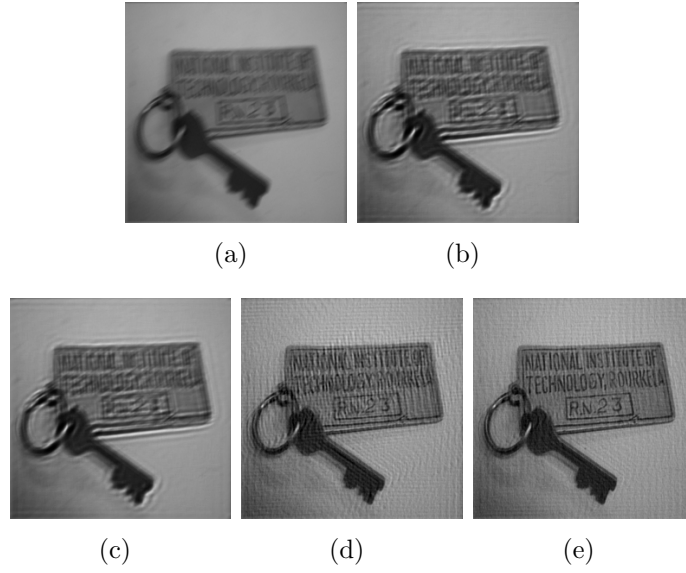


Figure 4.6: Restoration of *key* Image: (a) Blurred and noisy *Key* image. (b) Restored with PCA. (c) Restored with maximum likelihood. (d) Restored with SVR without parameter optimisation ( $C = 1$ ). (e) Restored with PSO based *Lena* SVR model.

SVR play a vital role in restoration performance which in turn becomes the basis of the proposition. SVR parameters are optimised on known training patterns from standard images. The optimised SVR is then subjected to same as well as different images those which are not used for training. Both subjective as well as objective (PSNR in dB) restoration performances are studied and compared with the competent schemes. It is observed that, the proposed scheme outperforms others.

# Chapter 5

## PSO Based Regularisation Parameter Estimation

---

# Chapter 5

## PSO Based Regularisation Parameter Estimation

Because of imperfections in imaging systems and other non-ideal conditions, the observed image often appears to be the degraded versions of the original image [1, 6]. The image degradation during acquisition makes it difficult for further analysis. So, the need of image restoration is mandatory for improving accuracy in different applications. The important issue in image restoration is to remove the blur in presence of noise. This chapter solves this problem by using regularisation by proper estimation of a regularisation parameter. A generalised cross validation (GCV) criterion has been utilised to estimate the optimised value of regularisation parameter. The particle swarm optimisation (PSO) has been used to optimise the GCV criterion.

The rest of the chapter is organised as follows. Image restoration using regularisation technique has been described in Section 5.1. Brief description of GCV is provided in Section 5.2. Section 5.3 describes the proposed regularisation parameter estimation based on PSO. Experimental results have been provided in Section 5.4. Finally Section 5.5 gives the summarisation of the chapter.

### 5.1 Regularised Image Restoration

In most of the applications, it is assumed that the image degradation model is linear and can be modelled as a two-dimensional convolution between the original image  $f(x, y)$  with the point spread function (PSF)  $h(x, y)$ . In matrix form, the

observed image can be written as,

$$g = Hf + \eta \quad (5.1)$$

where  $g$  and  $f$  are lexicographically ordered vectors. Blurring operator  $H$  is a block circulant with circulant block (BCCB) matrix obtained from the PSF  $h(x, y)$ . The problem of deblurring has been addressed in the literature [22, 33, 35]. It has been widely known that image restoration is an ill posed problem, and it can be effectively handled using regularisation method [9, 55]. Very often, the blurring matrices are ill conditioned, which occurs due to the wide magnitude range of the eigen values [89]. Noise amplification occurs at small eigen values.

Regularisation theory assumes that the original image is smooth and therefore, yields satisfactory results by adding a smoothness constraint to the original minimisation function. Image restoration model using regularisation is derived as [6],

$$O(\hat{f}, \alpha) = \|g - H * \hat{f}\|^2 + \alpha \|Q(\hat{f})\|^2 \quad (5.2)$$

where,  $\hat{f}$  represents the estimated image.  $\alpha$  and  $\|Q(\hat{f})\|^2$  are the regularisation parameter and regularisation operator respectively. The solution to the image restoration problem can be obtained by minimising equation ( 5.2) and is derived as [6],

$$\hat{f}(\alpha) = (H^T H + \alpha Q^T Q) H^T g. \quad (5.3)$$

The parameter  $\alpha$  controls the trade-off between the fidelity to the image data and smoothness to the solution and plays an important role during image restoration. An adaptive value of  $\alpha$  is necessary for degraded images with varied noise strength and PSF. For better understanding, *Lena* image is restored from 30 dB noise with PSF (out-of-focus with radius = 5) with different values of  $\alpha$ . The mean square error (MSE) plot for various  $\alpha$  values and different noise strengths are shown in Figure 5.1.

It is clearly observed from the Figure 5.1 that for low SNR condition, the value of  $\alpha$  significantly affects the mean square error. The degree of smoothness is controlled by the parameter  $\alpha$  and is generally dependent on the SNR of the noisy blurred image. If the SNR of the noisy blurred image is high, then a small value

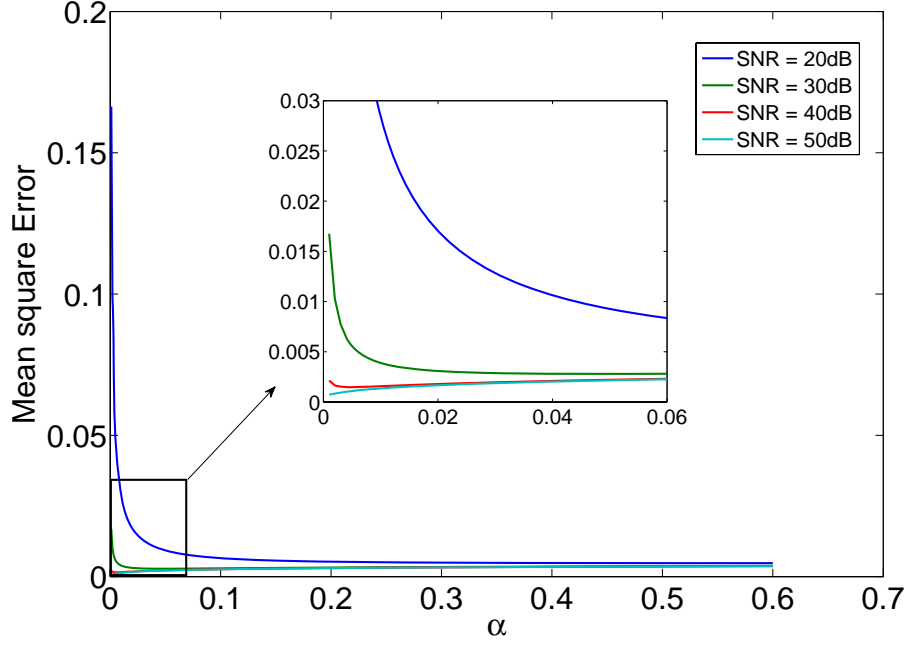


Figure 5.1: Variation of MSE for different  $\alpha$  at different noise conditions for *Lena* image.

of  $\alpha$  gives the desired smoothness, whereas, a high value of  $\alpha$  serves the purpose for degraded images with low SNR.

The effects of  $\alpha$  on restored *Cameraman* and *Lena* images are shown in the Figures 5.2 and 5.3 respectively. If the value of  $\alpha$  is increased, the restored image becomes smooth and noise is removed whereas, if  $\alpha$  is decreased, restored images become sharp but some noise component still appears in the restored image. So the difficulty lies with the choice of regularisation parameter for change in noise situation. The proposed method concentrates on the proper estimation of a regularisation parameter  $\alpha$ .

The second term  $\|Q(\hat{f})\|^2$  in (5.2) is the quadratic norm of the estimated image which is expected to be small. In general it is assumed that images are smooth and there is a high level of correlation among the pixels. The filter  $Q$  is chosen as a high pass filter which emphasises the high frequency content i.e. roughness in the image. Typical choice of  $Q$  is a discrete Laplacian operator

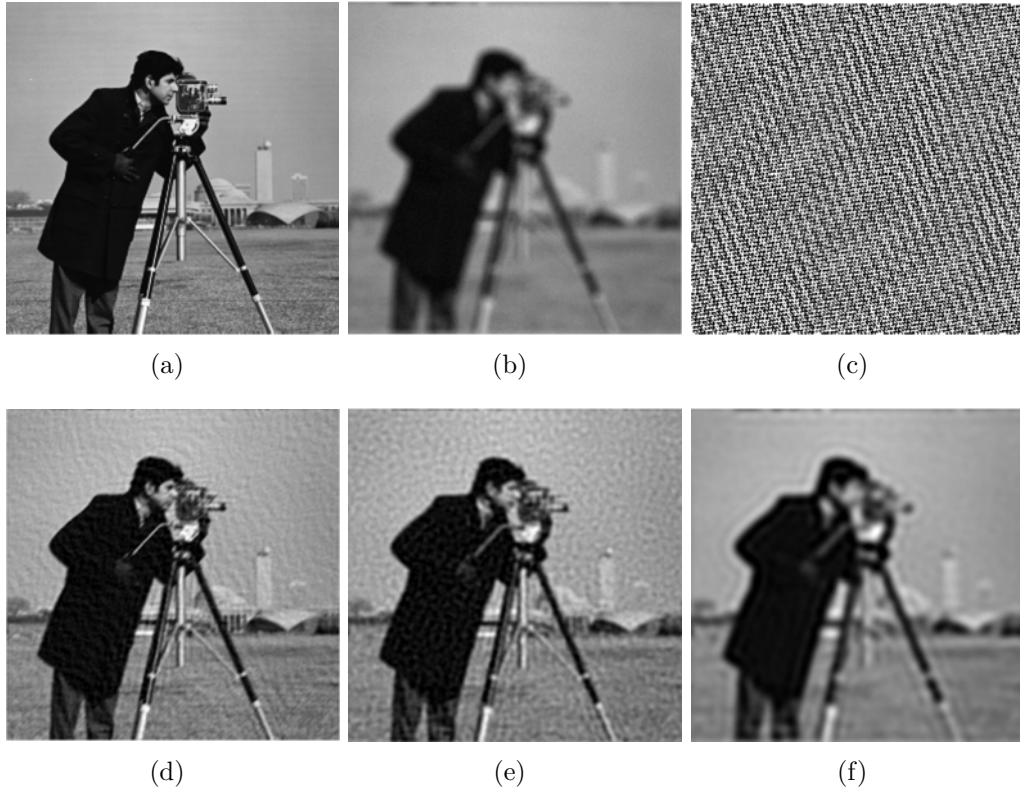


Figure 5.2: Effect of  $\alpha$  in regularised restoration: (a) Original *Cameraman* Image. (b) Noisy blurred *Cameraman* image with SNR = 40 dB. (c) Restored with inverse filter  $\alpha = 0$ . (d) (e) (f) Restored with regularisation filter with ( $\alpha = 0.01$ ), ( $\alpha = 0.004$ ), ( $\alpha = 0.2$ ).

which is given as [1],

$$Q = \begin{bmatrix} 0 & -0.25 & 0 \\ -0.25 & -1 & -0.25 \\ 0 & -0.25 & 0 \end{bmatrix} \quad (5.4)$$

The restored image is obtained by minimising the regularisation model (5.2). This solution is called Tikhonov solution [55]. The image stabilisation is controlled by the function  $Q(\hat{f})$ . Though the selection of regularisation operator affects the restored result, even a Laplacian operator can produce good results. A good result can also be obtained using an impulse as regularisation operator. This is observed from the Figure 5.4.

The minimisation function in discrete Fourier transform (DFT) can be written

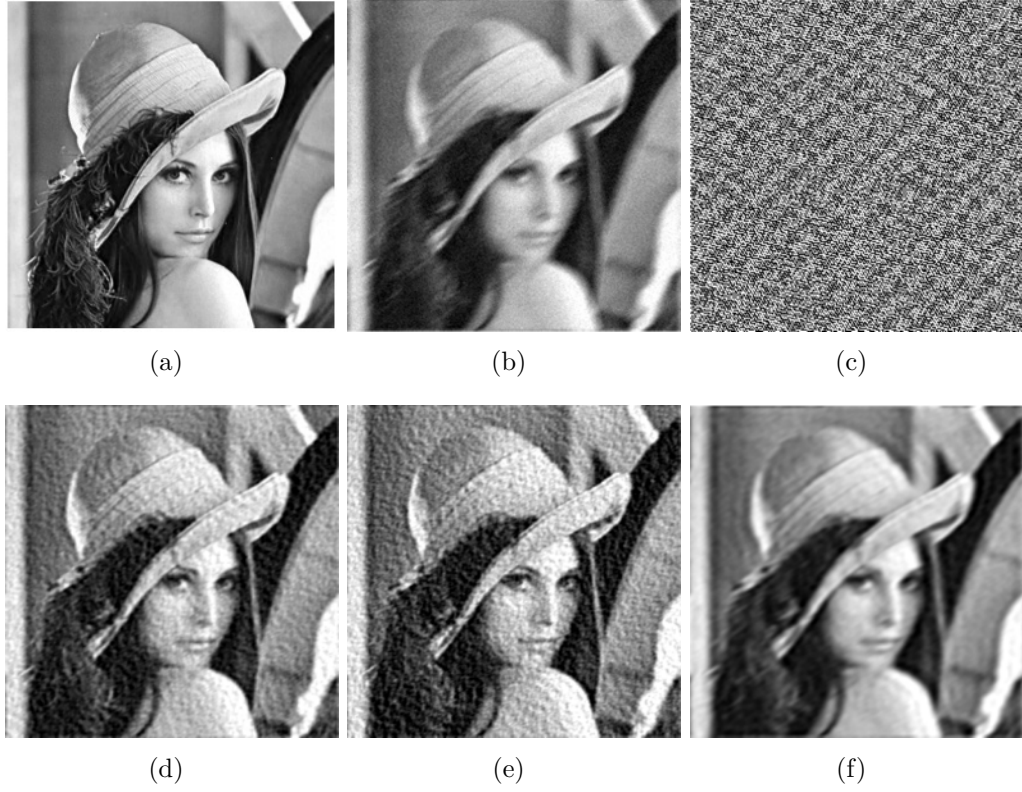


Figure 5.3: Effect of  $\alpha$  in regularised restoration: (a) Original *Lena* Image. (b) Noisy Blurred *Lena* image with SNR = 30 dB. (c) Restored with inverse filter  $\alpha = 0$ , (d) (e) (f) Restored with regularisation filter with ( $\alpha = 0.01$ ), ( $\alpha = 0.004$ ), ( $\alpha = 0.2$ ).

as,

$$\sum_{u,v} \{|G(u,v) - H(u,v)F(u,v)|^2 + \alpha|Q(u,v)F(u,v)|^2\} \quad (5.5)$$

where  $G(u,v)$ ,  $H(u,v)$ ,  $F(u,v)$  are corresponding Discrete Fourier transforms of function  $g(x,y)$ ,  $h(x,y)$  and  $f(x,y)$  respectively.  $Q(u,v)$  is the Fourier transform of the the Laplacian operator after padding zeros. The regularised restored result in DFT domain is obtained by differentiating above equation with respect to the image DFT coefficients and setting the result to zero. The solution to the minimisation of (5.5) is given by

$$\hat{F}(u,v) = \frac{H(u,v)}{|H(u,v)|^2 + \alpha|Q(u,v)|^2} G(u,v) \quad (5.6)$$

The performance of the filter depends on the regularisation term  $\alpha|Q(u,v)|^2$  used in the denominator of (5.6). If this term is zero, the filter reduces to an inverse

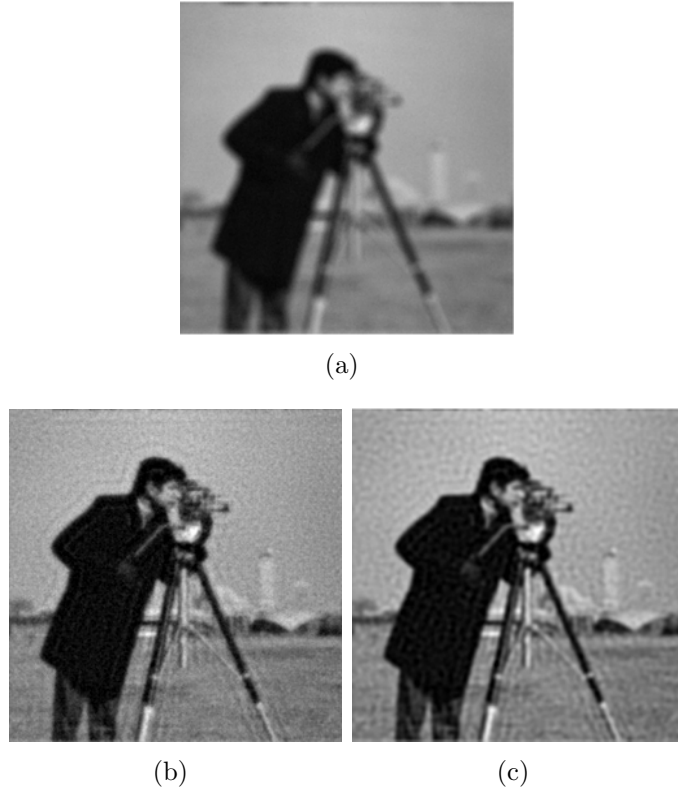


Figure 5.4: Effect of regularisation operator on restoration: (a) Noisy Blurred *Cameraman* image with  $\text{SNR} = 40$  dB. (b) Restored image without any filtering i.e.,  $Q = 1$   $\alpha = 0.01$ . (c) Restored image with  $Q$  as Laplacian operator and  $\alpha = 0.01$ .

filter. For a nonzero regularisation term, the noise amplification is significantly reduced by preventing the division by very small numbers especially at higher frequencies.

To find the optimum value of  $\alpha$  is one of the challenging issues and number of methods have been proposed [59, 60, 90–92]. The discrepancy principle has been applied for choosing the regularisation parameter [89, 93]. Using this principle, regularisation parameter is chosen as a function of error level. L-curve method has been used to select the regularisation parameter which is based upon the plot of the norm of regularised solution and data fitting residual [94, 95]. Reeves [96] proposed a spatial adaptive regularisation technique which uses a multistage estimation procedure to estimate the optimal choice of local regularisation weights. Reeves also suggested a method of generalised cross validation (GCV) [56] which gives a reliable estimate of the regularisation parameter. However, the

minimisation of GCV function plays an important role in the correct estimation of the regularisation parameter. It has been observed that traditional minimisation algorithms fail to achieve the global minimum because GCV function is multimodal in nature. Therefore even the GCV criterion is capable of optimising  $\alpha$  but doesn't produce good results. In order to remove this bottleneck, a PSO based global optimisation scheme has been proposed to minimise the GCV error function. The global minimisation of GCV function gives proper estimate of regularisation parameter. The improved simulation results justify the efficacy of the proposed scheme.

## 5.2 Generalised Cross Validation

Cross validation criterion is used to test the correctness of assumptions about a set of data by dividing the data into two sets. Instead of using the whole set of data in estimation, one portion of data is left out during estimation. The set that have been left out during estimation is used for validation. The sample mean square error is defined as,

$$e(\alpha) = \frac{1}{N} \|g - H\hat{f}(\alpha)\|^2 \quad (5.7)$$

where  $N \times N$  is the total number of pixels in the blurred image. The above equation minimises at  $\alpha = 0$ . So this equation cannot be used to find the optimum value of  $\alpha$ . GCV criterion has been used efficiently for estimating the regularisation parameter [56, 57] for image restoration. To apply GCV, each pixel of the blurred image is considered as one set of data. For a fixed value of regularisation parameter, the restored image is obtained using all the pixels leaving a single pixel. Then the restored image is reblurred again to predict the noisy blurred image pixel that has been left out in the restoration process. Each observation gives a different restored image. The regularisation parameter which minimises the mean square error over all the observations gives the correct value of  $\alpha$ . The cross validation error is written as,

$$E(\alpha) = \frac{1}{N} \sum_{k=1}^N [g_k - Hf^k(\alpha)]^2 \quad (5.8)$$

The optimised value of  $\alpha$  is obtained when  $E(\alpha)$  is minimum. GCV error is little variation of the cross validation error. GCV error is given as [56],

$$E(\alpha) = \frac{1}{N} \sum_{k=1}^N [g_k - H f^k(\alpha)]^2 w_{kk}(\alpha) \quad (5.9)$$

where  $w_{kk}(\alpha)$  are the weights and given by,

$$w_{kk}(\alpha) = \left[ \frac{1 - a_{kk}(\alpha)}{1 - \frac{1}{N} \sum_{j=1}^N a_{jj}(\alpha)} \right]^2 \quad (5.10)$$

$a_{jj}(\alpha)$  is  $(j, j)$ th element of the matrix  $A(\alpha)$  and is given as,

$$A(\alpha) = H (H^T H + \alpha Q^T Q)^{-1} H^T \quad (5.11)$$

The matrix  $A(\alpha)$  is called influence matrix. The GCV error measure using influence matrix is given as [56]

$$GCV(\alpha) = \frac{\frac{1}{N} \|(I - A)g\|^2}{[\frac{1}{N} \text{tr}(I - A)]^2} \quad (5.12)$$

where  $I$  is the identity matrix. The plot of the GCV function for various values of  $\alpha$  for the *Cameraman* and *Lena* image are shown in Figures 5.5 and 5.6 respectively. It is observed that GCV functional has multiple numbers of minimums and the traditional minimisation algorithms are prone to get stuck in the local minima. Though GCV criterion is capable of determining the regularisation parameter, it may not give the optimum value of  $\alpha$  for an unknown noise strength. The global minimisation of GCV function is required to optimise the regularisation parameter. PSO being a global optimisation technique it has been utilised it for optimising  $\alpha$ . The PSO is discussed in a nutshell followed by the proposed algorithm in sequel.

### 5.3 Regularisation Parameter Estimation using PSO

As discussed earlier in Chapter 4, PSO is a stochastic optimisation technique developed in 1995 by Eberhart and Kennedy [86] and works in the principle of bird flocking. PSO has also been successfully applied in blind image deconvolution

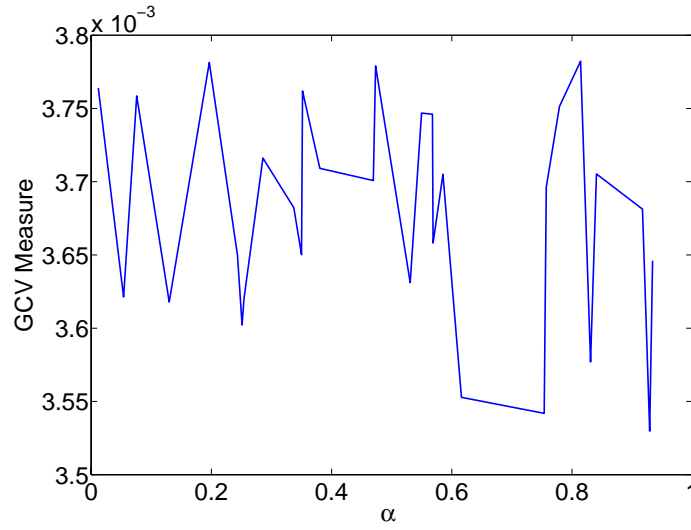


Figure 5.5: GCV function for degraded *Camerman* image.

in [97]. In the proposed scheme, regularisation parameter ( $\alpha$ ) is used as the position of the particle and initialised as a random number between 0 to 1. Velocity of the particles are also chosen randomly between 0 and 1. The position ( $\alpha$ ) and velocity are updated in the successive iterations. The updating equations for position and velocity are defined in equations (5.13) and (5.14) respectively as,

$$V = V + c1 * rand() * (pbest - presentx) + c2 * rand() * (gbest - presentx) \quad (5.13)$$

$$presentx = presentx + V; \quad (5.14)$$

where,  $V$  is the particle velocity and  $presentx$  is the current particle position ( $\alpha$ ) and  $rand$  is the random number between 0 and 1.  $c1$  and  $c2$  are two weighting constants or accelerating constants. GCV error function has been utilised as the objective function. The details formulation of the proposed algorithm is described in *Algorithm 1*. Finally the restored image is obtained using the optimum value of regularisation parameter ( $\alpha$ ) using equation (5.3).

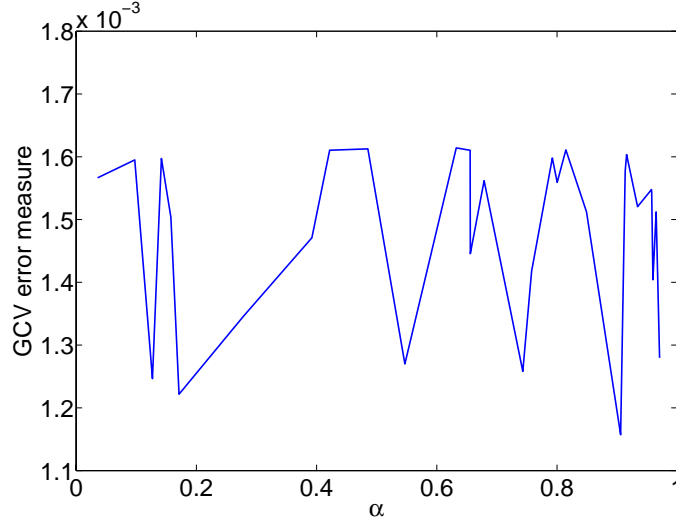


Figure 5.6: GCV function for degraded *Lena* image.

## 5.4 Results and Discussion

To validate the efficacy of the proposed algorithm, simulation has been carried out in MATLAB environment. For comparison, some benchmark approaches have also been simulated under similar environment. PSNR (dB) is used as a performance measure to evaluate the restoration quality. The overall simulation work is described in two different experiments in detail.

### ***Experiment 1: Estimation of $\alpha$ for motion blurred and noisy images***

Standard images including *Cameraman*, *Lena*, *Tree* are degraded with motion blur of different strengths and Gaussian noise is added to each of the images. The GCV function is formulated for each of the blurred images and minimised using PSO to determine the value of  $\alpha$ . Population size of 30 has been used to run the PSO. Experimentally the values of  $c_1$  and  $c_2$  are chosen to be 1.4 each. The values of  $c_1$  and  $c_2$  can also be calculated using cross validation. The images are restored using regularised deconvolution method after obtaining  $\alpha$ . For comparison, same images have also been restored using other methods [56] and [60]. Restoration results for *Cameraman* image is shown in Figure 5.7. PSNR (dB) of restored images have been provided in Table 5.1. The  $\alpha$  value for *Cameraman* is found to be 0.053. It is inferred from the results that the proposed scheme for  $\alpha$  estimation gives better restored results compared to other schemes.

**Algorithm 1** PSO for  $\alpha$  optimisation**Input:** Random Values between 0 to 1**Objective Function:** GCV Error function defined in equation (5.12)**Output:** Optimum  $\alpha$ 

- 1: The observed blurred image is received and converted into lexicographic ordered vector.
- 2: The regularisation parameter ( $\alpha$ ) values are initialised as random numbers between 0 and 1. These values denote the solutions and are called the particles of the PSO algorithm.
- 3: For each value of  $\alpha$ , the GCV error or the objective function is evaluated by taking

$$Q = \begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix}$$

- 4: The  $pbest$  and  $gbest$  are updated in each iteration as

$$pbest(i+1) = \begin{cases} pbest(i) & \text{if } GCV(\alpha_i) < GCV(\alpha_{i+1}) \\ pbest(i+1) & \text{otherwise} \end{cases} \quad (5.15)$$

and

$$gbest(i+1) = \arg \min_{pbest} GCV(pbest_n) \quad 1 < n \leq N \quad (5.16)$$

where  $i$  denotes the iteration number and  $N$  denotes the number of population.

- 5: Then the velocity and position of each particle are updated as per the update equations (5.13) and (5.14) respectively. The position of the particles is the regularisation parameter.
- 6: If number of iterations is less than the maximum, repeat the steps from 3 to 5. Otherwise the algorithm is terminated.
- 7: The  $\alpha$  value is obtained from the final  $gbest$  after the maximum iteration is reached.

**Experiment 2: Estimation of  $\alpha$  for out-of-focus blurred and noisy images**

Same set of images used in Experiment 1 are out-of-focus blurred and Gaussian noise is added to the blurred image. The  $\alpha$  value is determined for each of the blurred images similar to *experiment 1*. The PSO parameters are kept same as in *experiment 1*. The Restoration results for a synthetic image *ABC* and the *Lena* image are shown in Figures 5.8 and 5.9 respectively. PSNR (dB) comparison for restored images using different methods is given in Table 5.2. Simulation results for out-of-focus blurred images validate the effectiveness of the proposed regularisation scheme.

Table 5.1: PSNR (dB) comparison of various regularised deblurring algorithms for images degraded with Motion blur and Gaussian noise.

Image	Blur Extent	SNR (dB)	PSNR (dB) of Restored image by		
			GCV	Arnoldi process	Proposed
<i>Cameraman</i>	$L = 10, \theta = 45$	20	24.321	26.134	28.1106
<i>Cameraman</i>	$L = 15, \theta = 30$	30	25.6309	27.396	29.7838
<i>Lena</i>	$L = 8, \theta = 20$	20	27.701	28.11	30.8462
<i>Lena</i>	$L = 15, \theta = 30$	30	28.316	31.79	32.01
<i>ABC</i>	$L = 8, \theta = 20$	20	28.18	29.74	29.916
<i>ABC</i>	$L = 15, \theta = 30$	30	32.38	36.47	38.292

Table 5.2: PSNR (dB) comparison of various regularised deblurring algorithms for images degraded with out-of-focus blur and Gaussian noise.

Image	Blur Extent	SNR (dB)	PSNR (dB) of Restored image by		
			GCV	Arnoldi Process	Proposed
<i>Cameraman</i>	$R = 5$	20	23.3209	24.6410	25.4781
<i>Cameraman</i>	$R = 7$	30	29.4165	31.1221	32.3731
<i>Lena</i>	$R = 5$	20	24.3150	26.2130	26.6270
<i>Lena</i>	$R = 7$	30	30.1302	32.1756	33.5737
<i>ABC</i>	$R = 5$	30	24.0164	30.7054	32.4097
<i>ABC</i>	$R = 7$	20	31.6835	36.5424	38.5323

## 5.5 Summary

A PSO based GCV error minimisation technique has been proposed for regularisation parameter estimation. The proposed scheme exploits GCV function to determine the regularisation parameter. The global minimisation of GCV function plays an important role in determining the regularisation parameter. The

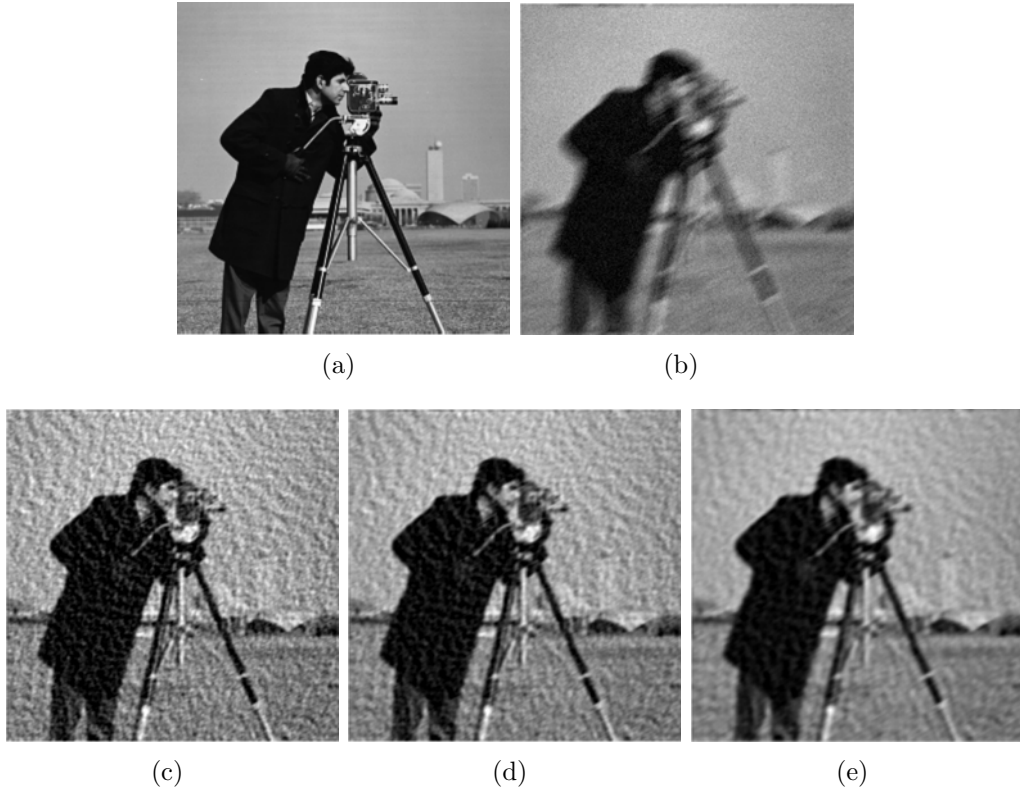


Figure 5.7: Restoration of *Cameraman* Image: (a) Original *Cameraman* Image. (b) Motion blurred and noisy image (SNR = 30 dB). (c) Restored with GCV ( $\alpha = 0.004$ ). (d) Restored after Arnoldi regularisation ( $\alpha = 0.01$ ). (e) Restored after PSO Based GCV minimisation ( $\alpha = 0.053$ ).

suggested scheme works well in moderate noise conditions for the images degraded with motion and out-of-focus blur. The visual quality and PSNR values of the restored images obtained in the proposed scheme are superior to that of competent schemes. The performance is satisfactory both for motion as well out-of-focus blurred images.

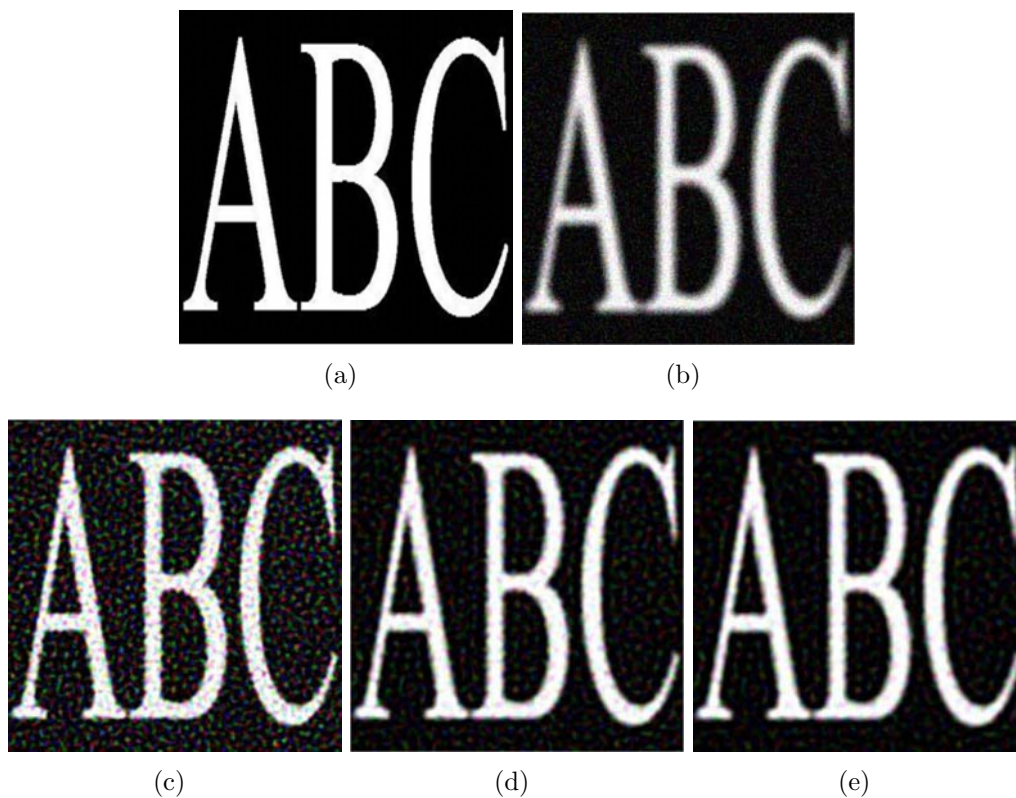


Figure 5.8: Restoration of ABC Image: (a) Original ABC image. (b) out-of-focus Blurred ABC image (SNR = 30 dB). (c) Restored with GCV ( $\alpha = 0.015$ ). (d) Restored after Arnoldi regularisation ( $\alpha = 0.054$ ). (e) Restored after PSO based GCV minimisation ( $\alpha = 0.093$ ).

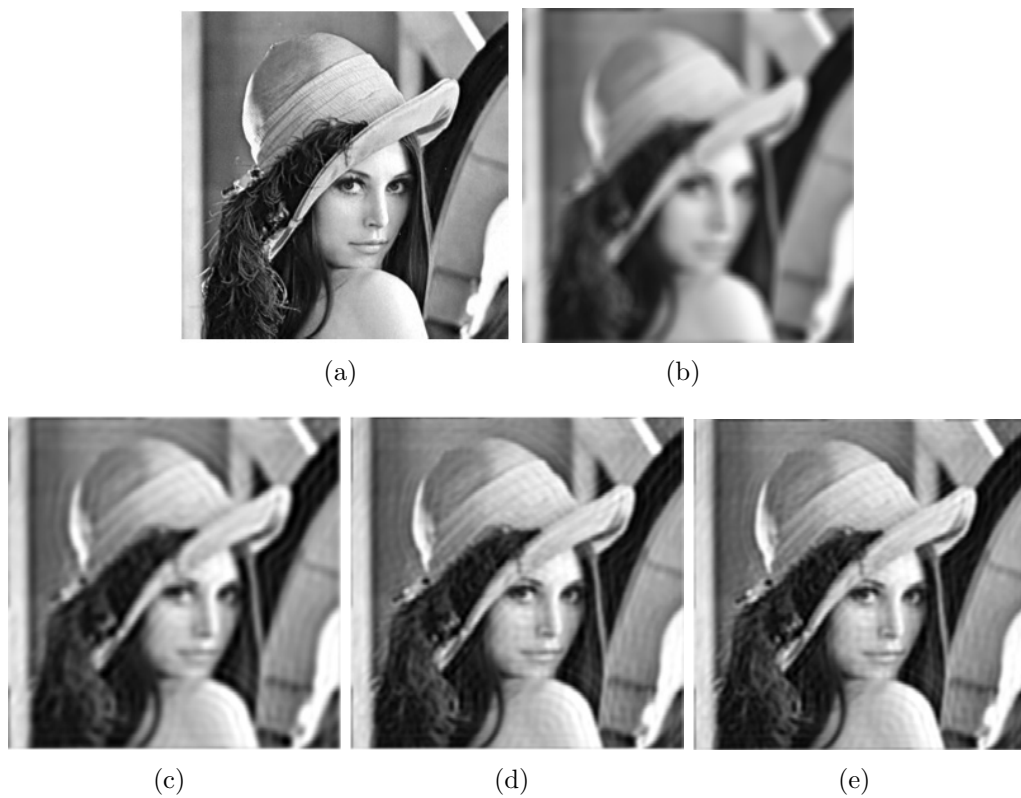


Figure 5.9: Restoration of *Lena* image: (a) Original *Lena* image. (b) Out-of-focus blurred *Lena* image (SNR = 60 dB). (c) Restored with GCV ( $\alpha = 0.048$ ). (d) Restored after Arnoldi regularisation ( $\alpha = 0.006$ ). (e) Restored after PSO based GCV minimisation ( $\alpha = 0.004$ ).

# Chapter 6

Rotational Motion Deblurring  
using Elliptical Modelling

## Chapter 6

# Rotational Motion Deblurring using Elliptical Modelling

Restoration of images that has been degraded due to relative motion between the object and camera is an important aspect of digital image processing [1]. Removing linear motion blurs from images are not of great deal today and can be handled using a variety of approaches. But the task becomes tedious when relative rotary motion occurs between camera and object which causes substantial degradation of image quality. This type of situation appears when the imaging system is mounted in a rotating satellite or a missile. Images are also affected by rotational motion blur when the foreground objects rotate. The high speed rotational motion during the exposure time makes the identification of the object extremely difficult. Rotational motion blur in images also causes undesired effects in important applications, which require high-quality images.

The image degradations can be classified depending on the PSF. If the PSF does not vary with space and time, the resulting blur is called spatially invariant blur, otherwise it is known as spatial variant blur. In spatial invariant degradations if the point spread function is known, the deblurring can be a deconvolution problem with the known convolution kernel. There always exists a difficulty in the deconvolution process because the convolution kernels are most often severely ill posed. Accordingly several known algorithms like Wiener filtering, regularised least squares filtering, Lucy-Richardson deconvolution can be employed to obtain the true image [10]. When PSF is not available, blind deconvolution schemes are employed to reconstruct the original image. Spatial variant deblurring problems

are more difficult because more than one PSF acts on the image and PSF for each pixel in the image are different. For spatial variant degradations, the linear convolution model can't be used because of multiple degradation functions. It is very difficult to get an accurate model for spatial variant blur. However, it can be modelled as a spatial invariant for different portions of the degraded image.

In the present chapter, a motion deblurring scheme has been proposed based on the assumption that images have been blurred due to relative rotational motion between the object and the imaging device. The motion may be either circular or elliptical. The rotational motion between the object and the camera causes a spatial variant blur in an image. Very few algorithms have been proposed in the literature to restore the images corrupted with spatial variant degradation. Hansen [98] modelled the spatial variant problems into several spatial invariant problems. Sawchuk [99] proposed a coordinate transform restoration (CTR) method to restore rotationally blurred images. The geometric transformation requires accurate interpolation of large number of pixels which leads to estimation errors. The CTR method requires transformation from rectangular coordinate to polar coordinate and vice-versa, which increases the computational complexity. In CTR method, some parts of the image still remain blurred. Sawchuk and Peyrovian [99] expressed the blur model equivalent to a radiation model in which the degradation of the pixels increases when they are away from the rotational centre. Hong [100] utilised block circular property of the blur kernel for deconvolution. They used Bresenham's circle drawing algorithm to fetch the pixels along a particular circular path. They use regularised least squares solution for deconvolution to make their algorithm robust to noise. However, these methods can be applied only when the motion is purely circular. A more generalised approach is proposed in this chapter by taking the motion path as elliptical. The elliptical motion model can be used for deblurring of circularly blurred images as well as elliptically blurred images. Hong [100] suggested a cross-correlation method to identify the rotational motion parameters. Rotational motion parameters has been found directly from the blurred images. The rotational motion problem has more than one degradation functions which is categorised as spatial variant problem. A spatial variant degradation model is shown in Figure 6.1. The number

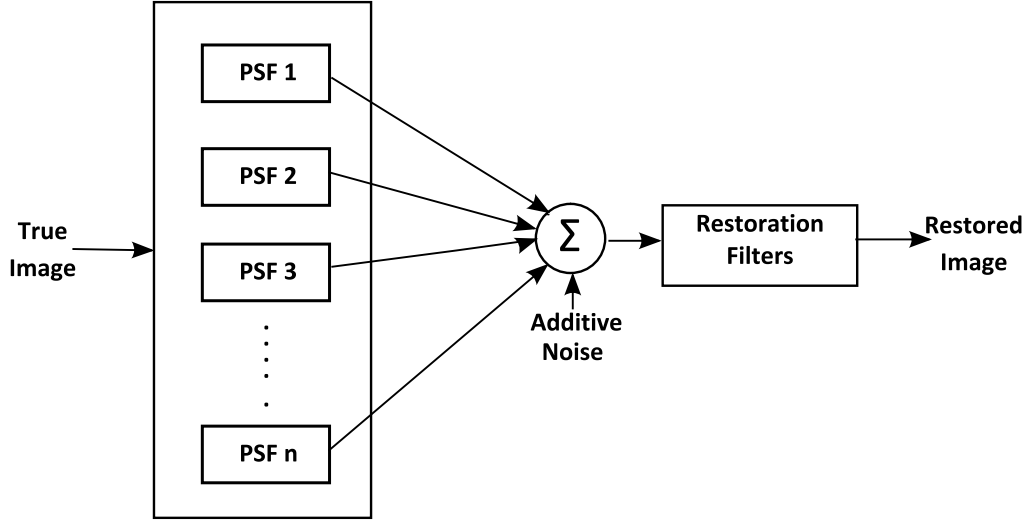


Figure 6.1: Spatial variant blur model.

of PSFs ( $n$ ) vary from image to image and depends on the blurring environment.

This chapter is organised as follows. An analysis of elliptical motion blur has been described in Section 6.1. The proposed deblurring method is presented in Section 6.2. Finally, the simulation results and summary are provided in Sections 6.3 and 6.4 respectively.

## 6.1 Elliptical Motion Blur Analysis

The motion blur problem is considered as an elliptical rotation with constant velocity. For elliptical motion, the blurring paths are taken as elliptical. For each pixel in a elliptical blurring path, PSF is assumed to be same. If  $f(x, y)$  and  $g(x, y)$  are true and blurred images, then  $g(x, y)$  can be expressed as,

$$g(x, y) = \frac{1}{T} \int_0^T f[x - r_x \cos(\omega t), y - r_y \sin(\omega t)] dt \quad (6.1)$$

where  $\omega$  is the angular velocity of the camera and the ellipse equation is given as,

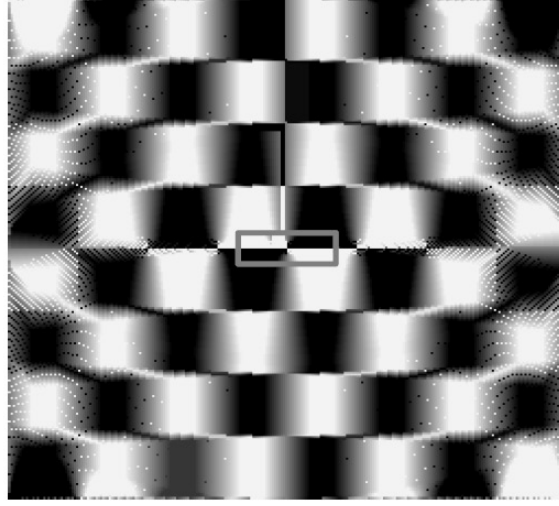
$$\left(\frac{x}{r_x}\right)^2 + \left(\frac{y}{r_y}\right)^2 = 1$$

$r_x$  and  $r_y$  represents major and minor axis respectively.

Therefore the problem reduces to number of one dimensional problems. Each of the elliptical path is deblurred separately using the conventional methods used

for spatial invariant problem. In terms of convolution, blurred image  $g$  for each elliptical path can be written as,

$$g(n) = \frac{1}{L} \sum_{i=0}^L f(n-i) + \xi(n) \quad (6.2)$$



	121	122	123	124	125	126	127	128	129	130	131	132	133	134	135
122	162	243	243	243	243	243	243	137	108	81	54	27	0	0	0
123	167	243	243	243	243	243	243	139	106	76	46	15	0	0	0
124	174	243	243	242	243	243	243	152	104	69	35	0	0	0	0
125	243	243	243	243	243	243	243	243	91	0	0	0	0	0	0
126	213	243	243	243	243	243	243	243	0	31	0	0	0	0	0
127	243	243	243	243	243	243	243	243	0	0	0	0	0	0	0
128	243	243	243	243	243	243	243	243	0	0	0	0	0	0	0
129	0	0	0	0	0	0	0	0	243	243	243	243	243	243	243
130	0	0	0	0	0	0	0	0	243	213	243	243	243	243	243
131	0	0	0	0	0	0	31	0	152	243	243	243	243	243	243
132	0	0	0	0	0	35	69	91	139	174	208	243	243	243	243
133	0	0	0	0	15	46	76	104	137	167	197	228	243	243	243
134	0	0	0	0	27	54	81	106	135	162	189	216	243	243	243

139	104	0	0	0	0	0	243	243	243	243	139	104	69	0	0	0	0	243	243	243	243	243	243
-----	-----	---	---	---	---	---	-----	-----	-----	-----	-----	-----	----	---	---	---	---	-----	-----	-----	-----	-----	-----

Figure 6.2: Elliptical motion blur model.

where  $g(n)$  represents the  $n^{th}$  degraded pixel along an elliptical path.  $\xi(n)$  represents noise associated with it. Length  $L$  represents the length of the blur in each ellipse i.e. number of pixels that has been passed through the scenery points in the exposure time  $T$ .

In the Figure 6.2, it has been clearly shown that how pixels along a particular

elliptical path is obtained and it is taken as an intensity image. If pixels are on different paths, they donot maintain any regularity in the spectrum. In the spectrum, the pattern is that a series of minima distributed uniformly on the frequency axis with constant intervals. The constant interval is same for all the circles in the blurred image spectrum. So the rotary centre identification is based on the constant zero interval in the frequency axis.

## 6.2 Proposed Deblurring Method

The proposed method deconvolves the blurred image after dividing it into number of concentric ellipses. There are two parameters required for constructing the PSF for a particular elliptical path *i.e.* one is angle of the blur and other is the rotary centre. Angle of the blur is calculated according to (6.3). Rotary centre is determined using the method used in [100].

$$\theta = \frac{rpm}{60} \times \pi \quad (6.3)$$

Bresenham's ellipse drawing algorithm has been used to fetch the pixels along a particular elliptical path. For a particular ellipse, blur length  $L$  is given by (6.4). The following steps describes the algorithm for deconvolution.

**Step 1:** Divide the blurred image into number of concentric ellipses.

**Step 2:** Fetch the pixels from each elliptical path and consider them as a rectangular image. The rectangular image is equivalent to a horizontal motion blurred image. (Bresenham's ellipse drawing algorithm is used to fetch the pixels along a elliptical path.)

**Step 3:** Calculate the blur length  $L$  according to (6.4).

**Step 4:** Deblur it by using Lucy–Richardson algorithm or Wiener filter.

**Step 5:** Integrate all the pixels in the same manner it has been fetched.

$$L = \text{int}(N.\theta)/360, \quad \theta = \omega T \quad (6.4)$$

To divide the image into concentric ellipses, and to fetch the pixels along a particular elliptical path Bresenham's ellipse drawing has been used [101]. Bresenham's algorithm accurately calculates the pixels aligning orthogonal into an ellipse. However, when the image is integrated by number of concentric ellipses, some points are not covered by any of the ellipses. The missing pixel problem is shown in the Figure. 6.3.

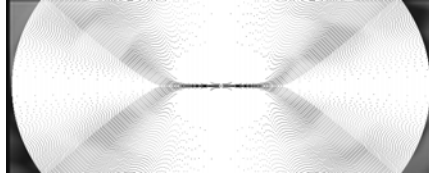


Figure 6.3: Missing pixel problem.

### 6.2.1 Missing Pixel Interpolation

As discussed, pixels along a elliptical path are fetched without any overlap or loss. But during integration of different concentric ellipses into an image some of the missing pixels are not covered by any of the multiple ellipses. The distribution of missing pixels depends on the algorithm rather than image content. A missing pixel is interpolated based on strategy which preserves the texture information of the neighbourhood of missing pixels. A shape recognition method is used to interpolate the pixels which are missed. This method recognises the shape of the neighbourhood and interpolates the missing pixels based on the shape. This method preserves the object edge from the background because shape is the expression of texture information of a neighbourhood. It is observed from the Figure 6.4 that black dots appear in the restored image without interpolation.

## 6.3 Simulation Results and Discussion

To validate the efficacy of the proposed elliptical modelling, an experiment has been carried out on standard images and their results are listed in this section. Subjective as well as objective measurements of the results are made. Peak signal to noise ratio (PSNR) is the performance metric considered for the restoration

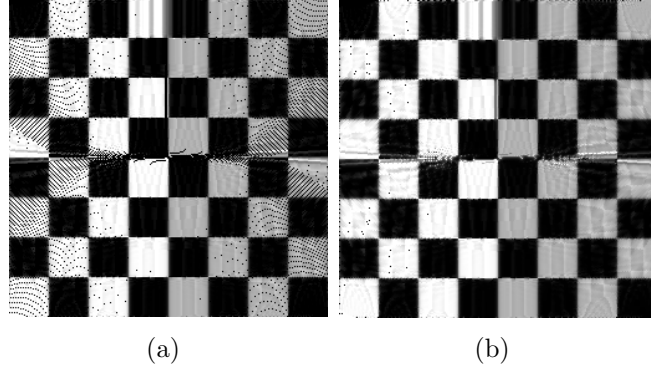


Figure 6.4: Restored image corrupted by missing pixels: (a) Restored image without interpolation. (b) Restored image after interpolation.

quality measurement.

$$\text{PSNR (dB)} = 10 \log_{10} \left( \frac{255^2}{\text{MSE}} \right) \quad (6.5)$$

$$\text{MSE} = \frac{1}{MN} \sum_{x=1}^M \sum_{y=1}^N \left( f(x, y) - \hat{f}(x, y) \right)^2 \quad (6.6)$$

where,  $M \times N$  is the size of the image, and  $f(x, y)$  and  $\hat{f}(x, y)$  represent the pixel values at  $(x, y)^{th}$  location of original and restored image respectively.

The rotational motion has been simulated using Bresenham's midpoint ellipse drawing algorithm. Pixels corresponding to concentric ellipses are extracted by varying the radii of the ellipse. Each ellipse is then stretched to form a one dimensional array of pixels. These arrays are then motion blurred by computing the blur length from the exposure time ( $T$ ) and rotational speed ( $\omega$ ). In the experiment, exposure time and rotational speed in rpm have been kept as 0.01 sec and 120 rpm respectively. The blurred array are then replaced back in their respective positions in the image from where they were extracted. The resultant image thus formed is an elliptically blurred image. Some of the standard images including *Stik*, *Vase*, *Lena*, *Chekerboard* are elliptically blurred in this manner. These images are also blurred circularly in the similar fashion by keeping both the radii of ellipse same.

For restoration, initially, the blurred *Lena* image is divided into a number of ellipses. The pixels in each elliptical path is taken as an horizontal blurred image and restored using Wiener filter. Restoration results for the *Lena* image are shown

in Figure 6.5. Similar steps have been followed for restoring other elliptical as well as circular blurred images. In the experiments, the proposed elliptical model has been used for both circular motion as well as elliptical motion deblurring. The restoration results of the *Stik*, *Vase* and *Chekerboard* images blurred due to circular as well as elliptical motion are depicted in the Figures 6.6, 6.7 and 6.8 respectively. It may be observed that the restored images are free from blur and edges are retrieved. However, there appears a black line at the centre of the restored results. This is due to inaccurate formation of the smallest ellipse at the centre. The minor axis of the smallest ellipse is not long enough to construct a proper ellipse. The PSNR measure for different restored images is shown in Table 6.1. It is also inferred that the proposed method provides acceptable PSNR of the restored image. Nevertheless, there exists a scope for further reduction in computational complexity and improvement in restoration performance.

Table 6.1: PSNR (in dB) of restored images.

Image	Circular rotation	Elliptical rotation
<i>Lena</i>	24.42	23.80
<i>Stik</i>	19.51	19.01
<i>Vase</i>	21.13	20.28
<i>Chekerboard</i>	24.32	22.45

In the later part of the experiment, The exposure time of the camera and rotational speed are varied. The PSNR plot of the restored results were obtained for each exposure time and rotational speed and given in Figure 6.9. It is observed that PSNR decreases with increase in exposure time and rotational speed.

## 6.4 Summary

The problem of rotational motion blur in images has been addressed in this chapter. The performance depends upon the modelling of PSF for spatial variant blur. The approach deals with an elliptical blur model where circular blur is treated as a special case. The spatial variant blur has been divided into number of spatial invariant problems and Wiener filter has been used to restore the images. The effectiveness of the proposed scheme has been verified for various rotational



(a)



(b)



(c)

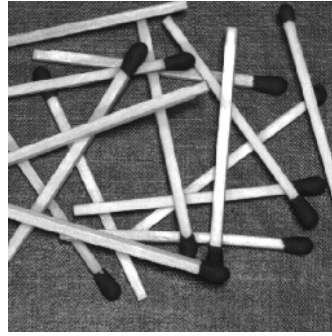


(d)

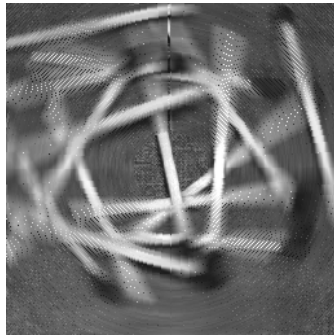


(e)

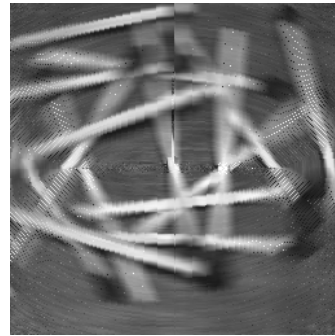
Figure 6.5: Restoration of *Lena* image: (a) True image. (b) Circularly blurred. (c) Elliptically blurred. (d) Restored image of (b). (e) Restored image of (c).



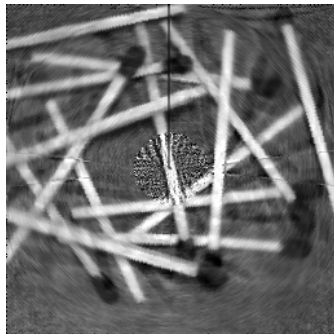
(a)



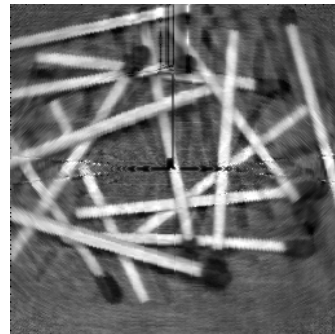
(b)



(c)



(d)

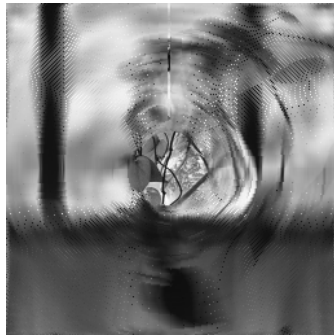


(e)

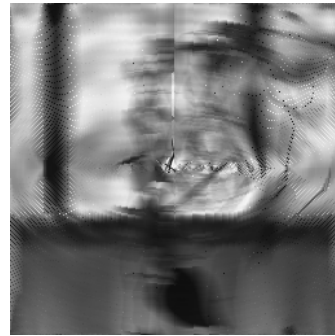
Figure 6.6: Restoration of *Stik* image: (a) True image. (b) Circularly blurred. (c) Elliptically blurred. (d) Restored image of (b). (e) Restored image of (c).



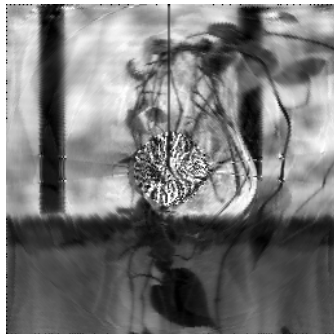
(a)



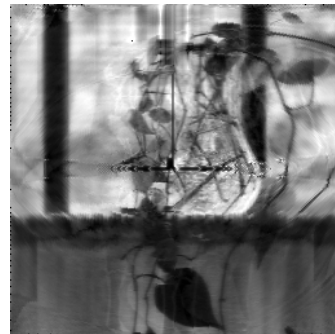
(b)



(c)



(d)



(e)

Figure 6.7: Restoration of *Vase* image: (a) True image. (b) Circularly blurred. (c) Elliptically blurred. (d) Restored image of (b). (e) Restored image of (c).

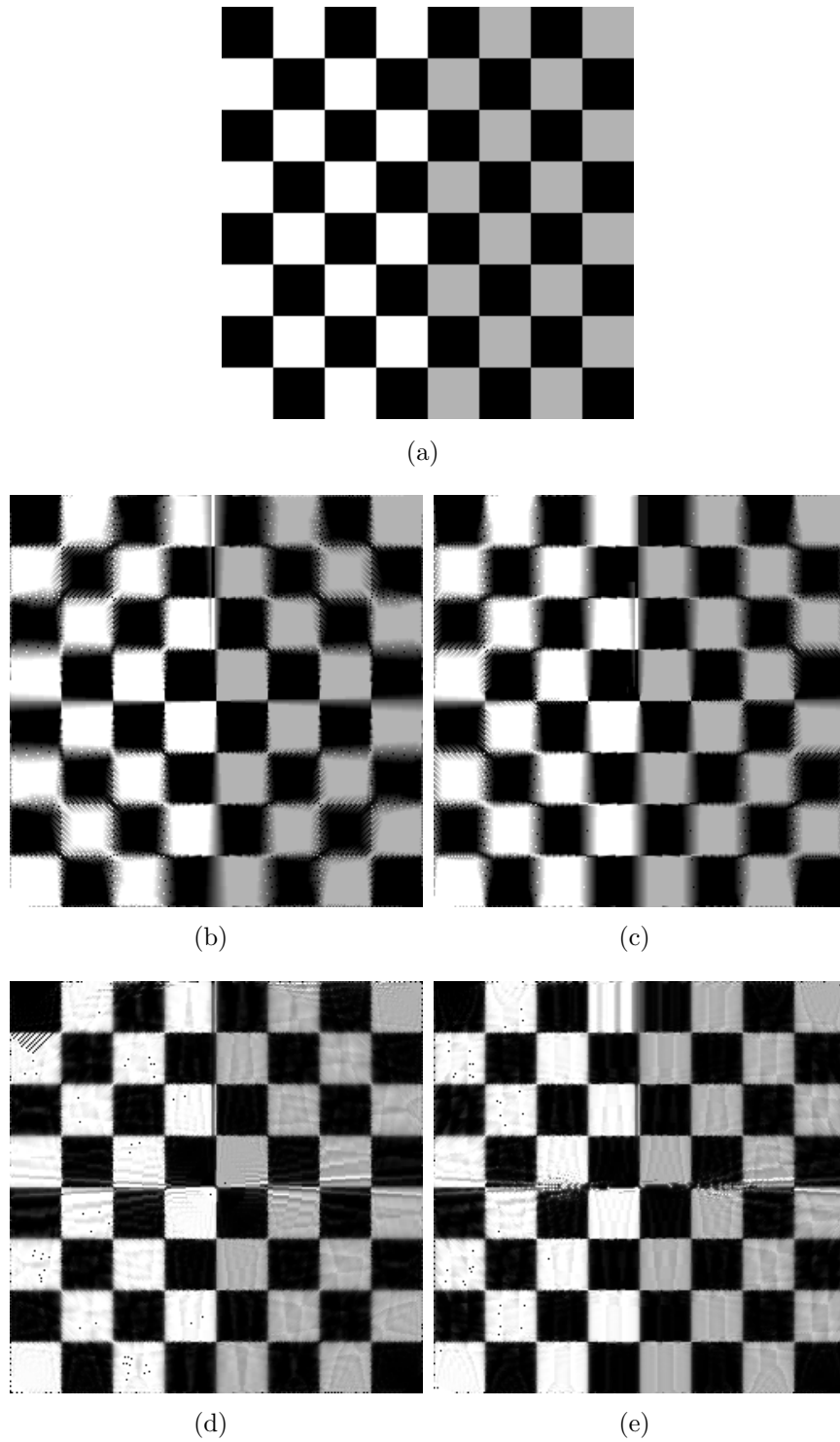
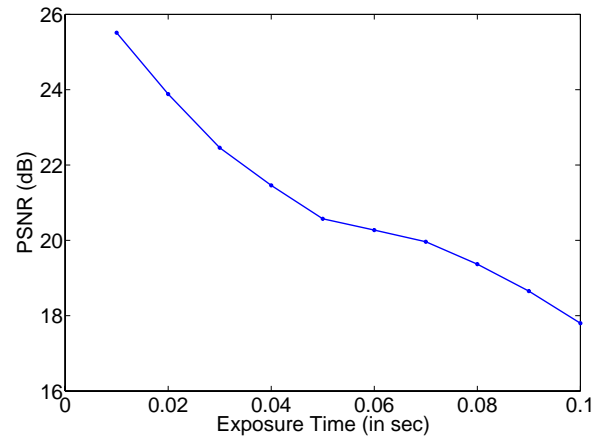
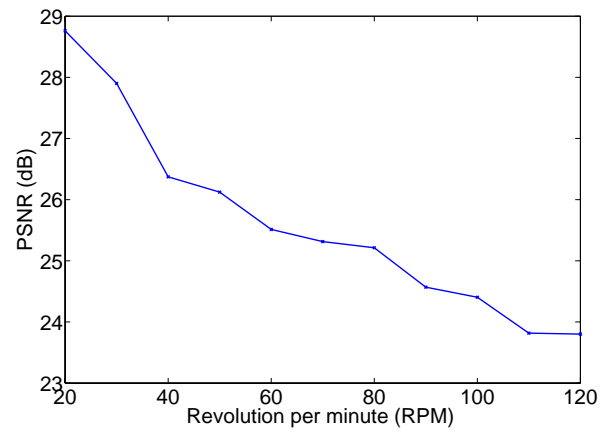


Figure 6.8: Restoration of *Chekerboard* image: (a) True image. (b) Circularly blurred. (c) Elliptically blurred. (d) Restored image of (b). (e) Restored image of (c).



(a)



(b)

Figure 6.9: (a) Variation of PSNR (dB) with respect to exposure time. (b) Variation of PSNR (dB) with respect to rotational speed.

motion blurred images. However, the method is susceptible to noise.

# Chapter 7

## Conclusions and Future Work

# Chapter 7

## Conclusions and Future Work

Image blurring is a common phenomenon that exists in various applications like photography, remote sensing, medical imaging, etc. The degradation may occur due to camera mis-focusing, atmospheric turbulence, relative object-camera motion and various other reasons. For the last few decades, researchers are working in the field of image restoration. The problem is still open due to its ill-posed nature and requires significant research.

In this thesis, attempts have been made to restore images from their degraded observations and evaluations are performed both visually and quantitatively. Chapter 2 deals with the motion blur parameter estimation for subsequent restoration using Wiener filter. In this regard, Gabor filter and radial basis function network have been used to estimate blur angle and blur length respectively. Performance analysis have been made on only blurred images as well as noisy blurred images. The proposed scheme estimates the blur parameters close to the true value. Comparative analysis demonstrates the efficacy of the proposed scheme. However, the suggested scheme produces good restored images when the Gaussian noise of strength more than 25 dB SNR.

Multi-class SVM is used in Chapter 3 to determine the length parameter of motion blur ( $L$ ) and sigma ( $\sigma$ ) parameter of atmospheric turbulence blur. The local variance feature has been used as the criteria to collect the samples from the blurred image for training the multi-class SVM. The output of SVM gives the blur length. Similar strategy has been followed for determining the  $\sigma$  parameter of turbulence blur by creating another SVM model. The experimental results verify

that the proposed multi-class SVM based blur parameter estimation provides accurate results. However, the trained SVM fails to estimate the blur parameters accurately for noisy blurred images. In addition, a SVM trained on a particular blur condition suitable to predict the similar blur parameters only.

In Chapter 4, support vector regression (SVR) is utilised for blind image restoration. The parameters of the SVR, including  $C, \sigma$  are optimised using particle swarm optimisation (PSO) before it is applied to restore images. A mapping of the true image is directly obtained from the noisy blurred image without estimating the PSF. SVR models are created with known training samples after optimising the parameters using PSO. The advantage of this method is that it can restore images affected with different blur with Gaussian noise of unknown strength. The proposed scheme is used in more generalised applications. However, if the blur (or image) characteristic is significantly different from the trained blur (or image) or if SNR is very low, the algorithm fails to approximate.

An approach towards estimating the regularisation parameter is proposed in the Chapter 5. Generalised cross validation (GCV) criterion is employed to obtain an optimum regularisation parameter. The optimised regularisation parameter is achieved by minimising the GCV error criterion using PSO technique. The proposed scheme helps to provide an optimum regularisation parameter depending on varying strengths of Gaussian noise. Subsequently, the optimised regularisation parameter is used to restore the degraded image using regularised restoration technique. The experimental results are shown to validate the efficacy of the proposed scheme. The limitation of this method lies in its computational complexity.

The deblurring of images affected with rotational blur is discussed in Chapter 6. Rotational blurs are considered as spatial variant blur and attempts have been made to model this as a combination of spatial invariant problems. This is done through elliptical modelling. Number of concentric ellipses are drawn on the blurred image and pixels in each elliptical path are considered to be affected with a unique spatial invariant PSF. Pixels are fetched along each elliptical path and deblurred using Wiener filter. Finally, they are integrated in the similar manner in which they have been fetched. The proposed approach effectively restores the

images affected with circular rotational blur as well as elliptical rotational blur. The suggested method is well suited for satellite imaging applications where object and camera motion is elliptical. However, the proposed scheme is susceptible to noise and computationally complex.

The proposed schemes along with a number of reported schemes are simulated on standard and naturally blurred images under different parametric PSFs. The restoration performances like PSNR, visual quality are evaluated for each proposed approach. By and large, the proposed schemes have an upper hand to their respective counterparts.

## **Scope for Further Research**

The research findings made out of this thesis has opened several research directions, which have a scope for further investigations. The proposed schemes mostly deal grayscale images, which can be extended to colour images. The computational complexity of each algorithms can be studied and schemes need to be devised to implement them in parallel for better response time. It is also observed that most of the image restoration problems can be modelled as multi-objective optimisation problem for better approximation of blur parameters.

Restoration of spatially variant blurred images is in its infancy. One of the suggested scheme deals with rotational blur which is variant in nature. Eventhough the scheme generates a comparable result, the scheme takes awesome computation. Hence, direction for reduction of computation will be an interesting direction of research.

## Bibliography

# Bibliography

- [1] R. C. Gonzalez and R. E. Woods. *Digital Image Processing*. Addison Wesley, 2nd edition, 1992.
- [2] Al Bovik. *The Essential Guide to Image Processing*. Academic Press, 2009.
- [3] Jae S. Lim. *Two-Dimensional Signal and Image Processing*. Prentice Hall, 1990.
- [4] R. Cucchiara, M. Piccardi, and A. Prati. Neighbor cache prefetching for multimedia image and video processing. *IEEE Transactions on Multimedia*, 6(4):539 – 552, August 2004.
- [5] S. Iyer and S. V. Gogawale. Image enhancement and restoration techniques in digital image processing. *Computer Society of India Communications*, pages 6–14, June 1996.
- [6] A. K. Jain. *Fundamentals of Digital Image Processing*. Prentice-Hall of India, 1989.
- [7] Abd-Krim Seghouane. A note on image restoration using  $c_p$  and mse. *IEEE Signal Processing Letters*, 15:61–64, July 2008.
- [8] Abd-Krim Seghouane. Model selection criteria for image restoration. *IEEE Transaction on Neural Networks*, 20(8):13571363, August 2009.
- [9] Moon Gi Kang and Aggelos K. Katsaggelos. General Choice of the Regularization Functional in Regularized Image Restoration. *IEEE Transactions on Image Processing*, 4(5):594 – 602, May 1995.
- [10] Mark R. Banham and Aggleos K. Katsaggelos. Digital Image Restoration. *IEEE Signal Processing Magazine*, 14(2):24 – 41, March 1997.
- [11] C. L. Chan, A. K. Katsaggelos, and A. V. Sahakian. Image sequence filtering in quantum-limited noise with applications to low-dose fluoroscopy. *IEEE Transactions on Medical Imaging*, 12(3):610 –621, sep 1993.

- [12] Y-S Han, D. M. Herrington, and W. E. Snyder. Quantitative angiography using mean field annealing. In *Proceedings of Computers in Cardiology*, pages 119–122, 1992.
- [13] C. Charalambous, F. K. Ghaddar, and K. Kouris. Two iterative image restoration algorithms with applications to nuclear medicine. *IEEE Transactions on Medical Imaging*, 11:2–8, March 1992.
- [14] V. A. Oliveira and J. M. Nightingale. Maximum image restoration in nuclear medicine. *IEE Proceedings on Communications, Speech and Vision*, 137(3):163–169, june 1990.
- [15] Hamid Soltanian-Zadeh, Joe P. Windham, and Andrew E. Yagle. A Multidimensional Nonlinear Edge-Preserving Filter for Magnetic Resonance Image Restoration. *IEEE Transactions on Image Processing*, 4(2):147–161, February 1995.
- [16] Xiaobing Lee, Ya-Qin Zhang, and A. Leon-Garcia. Information loss recovery for block-based image coding techniques—a fuzzy logic approach. *IEEE Transactions on Image Processing*, 4(3):259–273, mar 1995.
- [17] T. P. O. Rourke and R. L. Stevenson. Improved image decompression for reduced transform coding artifacts. *IEEE Transactions on Circuits and Systems for Video Technology*, 5(6):490–499, dec 1995.
- [18] T. Ozcelik, J. C. Brailean, and A. K. Katsaggelos. Image and video compression algorithms based on recovery techniques using mean field annealing. *Proceedings of the IEEE*, 83(2):304–316, feb 1995.
- [19] Yongyi Yang, N. P. Galatsanos, and A. K. Katsaggelos. Regularized reconstruction to reduce blocking artifacts of block discrete cosine transform compressed images. *IEEE Transactions on Circuits and Systems for Video Technology*, 3(6):421–432, dec 1993.
- [20] A. Zakhor. Iterative procedures for reduction of blocking effects in transform image coding. *Circuits and Systems for Video Technology, IEEE Transactions on*, 2(1):91–95, mar 1992.
- [21] S. Hein and A. Zakhor. Halftone to continuous-tone conversion of error-diffusion coded images. *IEEE Transactions on Image Processing*, 4(2):208–216, feb 1995.

- [22] Mark R. Banham and Aggleos K. Katsaggelos. Digital Image Restoration. *IEEE Signal Processing Magazine*, 14(2):24 – 41, March 1997.
- [23] A. K. Jain and S. Ranganath. Applications of two dimensional spectral estimation in image restoration. In *Proceedings of IEEE International Conference on Acoustics, Speech, Signal Processing*, pages 1113–1116, 1981.
- [24] B. R. Hunt. The application of constrained least squares estimation to image restoration by digital computer. *IEEE Transactions on Computers*, C-22(9):805 –812, September 1973.
- [25] J. Biemond, R. L. Lagendijk, and R. M. Mersereau. Iterative methods for image deblurring. In *Proceedings of IEEE*, pages 1113–1116, 1990.
- [26] A. O. Aboutalib and L. M. Silverman. Restoration of motion degraded images. *IEEE Transaction on Circuit Systems*, CAS-22:278–286, March 1975.
- [27] Mohsen Ebrahimi Moghaddam and Mansour Jamzad. Motion blur identification in noisy images using mathematical models and statistical measures. *Pattern Recognition*, 40:1946 – 1957, 2007.
- [28] R. Lokhande, K. V. Arya, and P. Gupta. Identification of parameters and restoration of motion blurred images. In *Proceedings of the 2006 ACM Symposium on Applied Computing*, volume 9 of *SAC 2006*, pages 301 – 305, Dijon, France, April 2006.
- [29] Jinlong Lin, Chao Zhang, and Qingyun Shi. Estimating the Amount of Defocus Through a Wavelet Transform Approach. *Pattern Recognition Letters*, 25(4):407 – 411, March 2004.
- [30] Michael Cannon. Blind deconvolution of spatially invariant image blurs with phase. *IEEE*, ASSP-24(1):56 – 63, July 1976.
- [31] Omri Shacham, Oren Haik, and Yitzhak Yitzhaky. Blind restoration of atmospherically degraded images by automatic best step-edge detection. *Pattern Recognition Letters*, 28(15):2094 – 2103, 2007.
- [32] Igor Aizenberg, Dmitriy V. Paliy, Jacek M. Zurada, and Jaakko T. Astola. Blur identification by multilayer neural network based on multivalued neurons. *IEEE Transactions on Neural Networks*, 19(5):883–898, May 2008.

- [33] M. M. Sondhi. Image Restoration: The Removal of Spatially Invariant Degradation. *IEEE*, 60(7):842 – 857, July 1972.
- [34] G. R. Ayers and J. C. Dainty. Iterative blind deconvolution method and its applications. *Optics letters*, 13:547–549, July 1988.
- [35] D. Kundur and D. Hatzinakos. Blind Image Deconvolution. *IEEE Signal Processing Magazine*, pages 43 – 64, May 1996.
- [36] Deepa Kundur and Dimitrios Hatzinakos. A Novel Blind Deconvolution Scheme for Image Restoration using Recursive Filtering. *IEEE Transaction on Signal Processing*, 46(2):375 – 390, February 1998.
- [37] Dalong Li, Russell M. Mersereau, and Steven Simske. Blind image deconvolution through support vector regression. *IEEE Transactions on Neural Networks*, 18(3):931–935, May 2007.
- [38] L. B. Lucy. An iterative technique for the rectification of observed distributions. *Astronomical Journal*, 79(6):745–754, 1974.
- [39] W. H. Richardson. Bayesian-based iterative method of image restoration. *Journal of Optical Society of America*, 62(1):55–59, January 1972.
- [40] William T. Freeman and Edward H. Adelson. The design and use of steerable filters. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(9):891 – 906, September 1991.
- [41] A. Tekalp, H. Kaufman, and J. Woods. Identification of image and blur parameters for the restoration of noncausal blurs. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 34(4):963 – 972, august 1986.
- [42] R. L. Lagendijk, A. M. Tekalp, and J. Biemond. Maximum likelihood image and blur identification: a unifying approach. *Optical Engineering*, 29(5):422–433, May 1990.
- [43] R. L. Lagendijk, J. Biemond, and D. E. Boekee. Identification and restoration of noisy blurred images using the expectation-maximization algorithm. *IEEE Transaction on Acoustics, Speech and Signal Processing*, 38(7), July 1990.
- [44] Stanley J. Reeves and Russell M. Mersereau. Blur Identification by the Method of Generalized Cross-Validation. *IEEE Transactions on Image Processing*, 1(3):301 – 311, July 1992.

- [45] C. M. Cho and H. S. Don. Blur identification and image restoration using a multilayer neural network. In *IEEE Int.Joint Conference on Neural Networks*, pages 2558–2563. IEEE, 1991.
- [46] B. C. McCallum. Blind deconvolution by simulated annealing. *Optics Communications*, 75(2):101–105, February 1990.
- [47] Dalong Li, Russell M. Mersereau, and Steven Simske. Atmospheric turbulence-degraded image restoration using principal components analysis. *IEEE Geoscience and Remote Sensing Letter*, 4(3):340–344, July 2007.
- [48] Abd-Krim Seghouane. A kullbackleibler divergence approach to blind image restoration. *IEEE Transaction on Image Processing*, 20(7):2078–2083, July 2011.
- [49] Qian Du and Ivica Kopriva. Dependent component analysis for blind restoration of images degraded by turbulent atmosphere. *Neurocomputing*, 72(10-12):2682 – 2692, 2009.
- [50] Qi Shan, Jiaya Jia, and Aseem Agarwala. High-quality motion deblurring from a single image. *ACM Transactions on Graphics (SIGGRAPH)*, 27(3):73:1–73:10, August 2008.
- [51] Ignazio Gallo, Elisabetta Binaghi, and Mario Raspanti. Semi-blind image restoration using a local neural approach. *Neurocomputing*, 73(1–3):389 – 396, December 2009.
- [52] Dalong Li and Steven Simske. Atmospheric turbulence degraded-image restoration by kurtosis minimization. *IEEE and Geoscince and Remote Sensing Letters*, 6(2):244–247, April 2009.
- [53] C. Vural and W. A. Sethares. Blind image deconvolution via dispersion minimization. *Digital Signal Processing*, 16(2):137 – 148, 2006.
- [54] Michal Dobes, Libor Machala, and Tomas Frst. Blurred image restoration: A fast method of finding the motion length and angle. *Digital Signal Processing*, 20(6):1677 – 1686, 2010.
- [55] A. N. Tikhonov and V. Y. Arsenin. Solutions of Ill-Posed Problems. Winston, 1977.

- [56] S. J. Reeves and R. M. Mereseru. Optimal Estimation of Regularization parameter and Stabilizing Functional for Regularized Image Restoration. *Optical Engineering*, 29:446 – 454, May 1990.
- [57] Haiyong Liao, Fang Li, and Michael K. Ng1. Selection of regularization parameter in total variation image restoration. *Journal of Optical society of America*, 26(11):2311 – 2320, 2009.
- [58] R. Neelamani, H. Choi, and R. G. Baraniuk. Forward: Fourier-wavelet regularized deconvolution for ill-conditioned systems. *IEEE Transaction on Signal Processing*, 52(2):418 – 433, February 2003.
- [59] Xianjin Wu, Runsheng Wang, and Cheng Wang. Regularised Image Restoration Based on Adaptively Selecting Parameter and Operator. In *International Conference on Pattern Recognition (ICPR'04)*, volume 3, pages 662 – 665, September 2004.
- [60] Xie Kai and Li Tong. Arnoldi process based on optimal estimation of the regularisation parameter. In *IEEE International Workshop on Imaging Systems and Techniques*, pages 340 – 343, may 2009.
- [61] L. Hong, Y. Wan, and A. K. Jain. Fingerprint image enhancement : Algorithm and performance evaluation. *IEEE transaction on Pattern Analysis and Machine Intelligence*, 20(8):777–789, August 1998.
- [62] M. Leena Silvester and V. K. Govindan. Convolutional neural network based segmentation. In *Computer Networks and Intelligent Computing*, volume 157 of *Communications in Computer and Information Science*, pages 190–197. Springer Berlin Heidelberg, 2011.
- [63] Mohsen Ebrahimi Moghaddam and Mansour Jamzad. A comprehensive study to find the effect of different noise removal methods on linear motion blur parameter estimation. In *13th International Conference on Systems Signals and Image Processing (2006)*, Budapest, Hungary, September 2006.
- [64] Ioannis M. Rekleitis. Optical flow recognition from the power spectrum of a single blurred image. In *Proceedings of the International Conference on Image Processing*, page 25, 1996.
- [65] Ioannis. M. Rekleitis. Steerable filters and cepstral analysis for optical flow calculation from a single blurred images. In *Vision Interface (1996)*, pages 159–166, Toronto, Canada, May 1996.

- [66] Vladimir N. Vapnik. The nature of statistical learning theory. *Springer-Verlag*, 1995.
- [67] Thorsten Joachims. Making large-scale svm learning practical. LS8-Report 24, Universität Dortmund, LS VIII-Report, 1998.
- [68] Loo-Nin Teow and Kia-Fock Loe. Robust vision-based features and classification schemes for off-line handwritten digit recognition. *Pattern Recognition*, 35(11):2355 – 2364, 2002.
- [69] C. Choisy and A. Belaid. Handwriting recognition using local methods for normalization and global methods for recognition. In *In Proceedings of Sixth Int. Conference On Document Analysis and Recognition*, pages 23 – 27, 2001.
- [70] Tapan Kumar Bhowmik, Pradip Ghanty, Anandarup Roy, and Swapan K. Parui. Svm-based hierarchical architectures for handwritten bangla character recognition. *International Journal on Document Analysis and Recognition (IJDAR)*, 12(2):97–108, 2009.
- [71] P. Hong, Q. Tian, and T. S. Huang. Incorporate support vector machines to content-based image retrieval with relevance feedback. In *In Proceedings of International Conference on Image Processing*, pages 750 – 753, 2000.
- [72] H. Druker, B. Shahrany, and D. C. Gibbon. Support vector machines: relevance feedback and information retrieval. *Information Processing & Management*, 38:305 – 323, 2002.
- [73] T. Van Gestel, J. A. K. Suykens, D. E. Baestaens, A. Lambrechts, G. Lanckriet, B. De Moor B. Vandaele, and J. Vandewalle. Financial time series prediction using least squares support vector machines within the evidence framework. *IEEE Transaction On Neural Networks*, 12(4):809 – 821, 2001.
- [74] M. H. Yang and B. Moghaddam. Gender classification using support vector machines. In *Proceedings of IEEE Int. Conference on Image Processing*, pages 471 – 474, 2000.
- [75] Y. Zhang, R. Zhao, and Y. Leung. Image classification by support vector machines. In *In Proceedings of IEEE Int. Conference on Intelligent Multimedia, Video and Speech Processing*, pages 360–363, 2001.

- [76] A. Balasubramaniam, Guturu Parthasarathy, and B.N. Chatterji. Knowledge based approach to cluster algorithm selection. *Pattern Recognition Letters*, 11(10):651 – 661, 1990.
- [77] J. Weston and C. Watkins. Multi-class support vector machines. In *Proceedings of ESANN99*, pages 219–224, Brussels, 1999. D. Facto Press.
- [78] Antoine Bordes, Lon Bottou, Patrick Gallinari, and Jason Weston. Solving multiclass support vector machines with larank. In *In 24th International Conference on Machine Learning, Corvallis, USA*, Corvallis, June 2007.
- [79] Vladimir N. Vapnik. Statistical learning theory. *Wiley-Interscience, Newyork*, 1998.
- [80] John C. Platt, Nello Cristianini, and John Shawe-Taylor. Large margin dags for multiclass classification. In *Advances in Neural Information Processing Systems*, volume 12, pages 547 – 553. 2000.
- [81] Gjorgji Madzarov, Dejan Gjorgjevikj, and Ivan Chorbev. A multi-class svm classifier utilizing binary decision tree. *Informatica*, 33(2):233–241, May 2009.
- [82] C. C. Chang and C. J. Lin. *LIBSVM: A library for support vector machines*. <http://www.csie.ntu.edu.tw>.
- [83] A. J. Smola and B. Schlkopf. A tutorial on support vector regression. *Statistical Computing*, 14(3):199–222, 2004.
- [84] Chih-Hung Wua, Gwo-Hshiung Tzeng, and Rong-Ho Lin. A novel hybrid genetic algorithm for kernel function and parameter optimization in support vector regression. *Expert Systems with Applications*, 36:47254735, 2009.
- [85] Cheng-Lung Huang and Chieh-Jen Wang. A ga-based feature selection and parameters optimization for support vector machines. *Expert Systems with Applications*, 31:231–240, 2006.
- [86] J. Kennedy and R. C. Eberhart. Particle Swarm Ooptimisation. In *Proc. IEEE International conf. on neural networks*, volume IV, pages 265 – 270, November 1995.
- [87] J. Kennedy and R. C. Eberhart. A discrete binary version of the particle swarm algorithm. In *IEEE International Conference on Computational Cybernetics and Simulation.*, volume 5, pages 4104 –4108, oct 1997.

- [88] Huiyuan Fan. A modification to particle swarm optimisation algorithm. *Engineering Computations*, 19(8):970–989, 2002.
- [89] H. Engl, M. Hanke, and A. Neubauer. Regularization of inverse problems. Kluwer Academic, 1996.
- [90] A. K. Katsaggelos, J. Biemond, and R. M. Mersereau. A Regularized Iterative Image Restoration Algorithm. *IEEE Transaction on Image Processing*, 39:914 – 929, 1991.
- [91] R. L. Lagendijk, J. Biemond, and D. E. Boekee. Regularized Iterative Image Restoration with Ring Reduction. *IEEE Trans. Acoustics, Speech, And Signal Processing*, 36:1874 – 1887, 1998.
- [92] M. Mignotte. A segmentation-based regularization term for image deconvolution. *IEEE Transaction on Image Processing*, 15(7):1973 – 1984, 2006.
- [93] P. Hansen. *Rank-Deficient and Discrete Ill-Posed Problems*. SIAM, 1998.
- [94] P. Hansen and D. O. Leary. The use of l-curve in the regularization of discrete ill-posed problems. *SIAM J. of Sci-comp*, 14:1487 – 1503, 1993.
- [95] C. Vogel. Computational methods for inverse problems. SIAM, 1998.
- [96] S. J. Reeves. Optimal Space-Varying Restoration in Iterative Image Restoration. *IEEE Transaction on Image Processing*, 3:319 – 324, May 1994.
- [97] Tsung-Ying Sun, Chan-Cheng Liu, Yu-Peng Jheng, Jyun-Hong Jheng, Shang-Jeng Tsai, and Sheng-Ta Hsieh. Blind Image Deconvolution via Particle Swarm Optimisation with Entropy Evaluation. In *Eighth International Conference on Intelligent Systems Design and Applications*, pages 265 – 270, November 2008.
- [98] P. C. Hansen, J. G. Nagy, and D. P. OLeary. Deblurring images: matrices, spectra, and filtering. *SIAM*, 2006.
- [99] A. A. Sawchuk and M. J. Peyrovian. Space-variant image restoration by coordinate transformations. *J. Opt. Soc. Am.*, 64(2):138144, 1976.

- [100] H. Hong and T. Zhang. Fast restoration approach for rotational motion blurred image based on deconvolution along the blurring paths. *Optical Engineering*, 42(12):471 – 3486, 2003.
- [101] Donald. Hearn and M. Pauline Baker. *Computer Graphics*. Prentice Hall of India, New Delhi, India, second edition, 1997.

# Dissemination of Work

## Journals

1. **Ratnakar Dash**, Pankaj Kumar Sa and Banshidhar Majhi. .Particle Swarm Optimisation based Support Vector Regression for Blind Image Restoration *Journal of Computer Science & Technology, Springer Publications*, 2010. Accepted for publication .
2. **Ratnakar Dash**, Pankaj Kumar Sa and Banshidhar Majhi. Blur Parameter Identification using Support Vector Machine. *International journal on Signal and Image processing, ACEEE publisher, USA*, Accepted for publication.
3. **Ratnakar Dash**, Pankaj Kumar Sa and Banshidhar Majhi. Spatial Variant Deblurring of Images. *International Journal of Computational Vision and Robotics, Inderscience Publishers*, 2(1):80–88, 2011.

## Conferences

1. **Ratnakar Dash**, Pankaj Kumar Sa and Banshidhar Majhi. Blur Parameter Identification using Support Vector Machine. In *Proc. of International Conference. on Advances in Computer Science*, . pages 89–92, Noida, India, Dec 2011.

2. **Ratnakar Dash**, Pankaj Kumar Sa and Banshidhar Majhi. RBFN based Motion Blur Parameter Estimation. In *IEEE International Conference on Advanced Computer Control, ICACC'09*, pages 327-331 , Singapore, Jan 2009.
3. **Ratnakar Dash** and Banshidhar Majhi. Particle Swarm Optimization Based Regularization for Image Restoration. In *World Congress on Nature & Biologically Inspired Computing (IEEE Conference)*. pages 1254-1257, India, Dec 2009.

### Communicated

1. **Ratnakar Dash**, and Banshidhar Majhi. *Motion Blur Parameters Estimation for Image Restoration.*, Revised version submitted to International Journal of Image and Graphics, World Scientific Publishing Company, USA

## **Ratnakar Dash**

Assistant Professor  
National Institute of Technology, Rourkela  
Rourkela – 769 008, India.

Ph: +91-661-2462365 (O), +91-9861379338 (M)

e-mail: ratnakar.dash@gmail.com

### **Qualification**

- Ph.D. (Continuing)  
NIT Rourkela
- M.Tech. (Electronics & Communication)  
VSSUT, Burla Orissa [First Division]
- B.E. (ECE)  
National Institute of Science and Technology, Berhampur, Orissa [First division]
- Diploma Engg (ETC)  
State Council of Technical Education and Training, Orissa, [First division, Honours]
- 10th  
Board of Secondary Education, Orissa, [First division]

### **Publications**

- 03 Journal Articles
- 08 Conference Papers

### **Permanent Address**

Ramakrishna Nagar, Lochapada Road, Berhampur – 760 001, Orissa, India.

### **Date of Birth**

February 14, 1978