BAYESIAN APPROACHES FOR IMAGE RESTORATION

WEIMIAO YU

(*M. ENG., NUS AND XJTU*)

A DISSERTATION SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF DOCTOR OF PHILOSOPHY

DEPARTMENT OF MECHANICAL ENGINEERING

NATIONAL UNIVERSITY OF SINGAPORE

SEPT. 2007

To my parents

ACKNOWLEDGMENTS

I wish to express my gratitude and appreciation to my supervisor, A/Prof. Kah Bin LIM for his instructive guidance and constant personal encouragement during every stage of my PhD study. I gratefully acknowledge the financial support provided by the National University of Singapore through Research Scholarship that makes it possible for me to finish this study. My gratitude also goes to Mr. Yee Choon Seng, Mrs. Ooi, Ms. Tshin and Mr. Zhang for their helps on facility support in the laboratory so that my research could be completed smoothly.

It is also a true pleasure for me to meet many nice and wise colleagues in the Control and Mechtronics Laboratory, who made the past four years exciting and the experience worthwhile. I am sincerely grateful for the friendship and companionship from Chu Wei, Duan Kaibo, Feng Kai, Zhang Han, Liu Zheng, Long Bo, Hu Jiayi and Zhou Wei, *etc.*

Finally, I would like to thank my parents, brother and sister for their constant love and endless support through my student life. My gratefulness and appreciation cannot be expressed in words.

TABLE OF CONTENTS

SUMMARY

Recovering signals (1-D/2-D) from their degraded and noisy version is generally called digital signal/image restoration or reconstruction. Some of the signals themselves contain discontinuities, which encode important, crucial and significant information. The removal of the noise and the preservation of the discontinuities are conflicting interests. The Bayesian approaches for discontinuity identification and edge preserved noise removal in 1-D/2-D digital signals are discussed and studied in this dissertation.

Bayesian model selection is robust, however expensive in calculation due to the "curse of dimensionalities". An approach is proposed in this work to reduce the calculation burden in Bayesian model selection. By increasing the model order sequentially according to a confidence level, the model order may be determined and evaluating the posterior distribution directly can be avoided. Some simplifications of the calculation for the normalization constant make it more efficient. Since the model *à priori* distribution is not uniform, the so-called posterior evidence is suggested for the model selection. The simulations proved its robustness and accuracy.

Bayesian Kernel approach based on sparse kernel learning is presented for both 1-D and 2-D discontinuous signal denoising. The image is assumed to be locally piecewise linear. The constructed kernel matrix is a hybrid of linear and quadratic in 2-D lattice. The cost function and hyper parameter tuning are studied in detail. The presented cost function is proven to only have one global minimum; therefore the problem of ill-conditioning is successfully solved. Experimental results show that the proposed method is robust and can preserve the discontinuities in its formulation. The proposed approach also achieved an excellent restoration results compared with other recent methods, some of the errors are much smaller than existing methods when Signal to Noise Ratio (SNR) is low.

The Bayesian approach for blur identification in the spatial domain is proposed. This approach can identify extent of the Point Spread Function (PSF) and its parameter. No assumption is made on the

shape of the PSF, so it can be applied for any shape of PSF. The image is decomposed into smooth region and edge region. Four linear motion blur PSFs are applied to blur the experimental images, thus they are successfully identified by the proposed approach. The assumptions are validated by the experimental results. The results also show it is robust and promising in blur identification.

Finally, the discrete formulation of compact coordinate transformation is presented. Under-sampling phenomena are studied and discussed. The constraints of coordinates transform are relaxed in the discrete formulation. The optimization of the compactness and a cost function is proposed. The experiments and simulations show that the presented method can solve the spatially variant problem of rotational blurred image restoration.

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS

$f(x, y)$/ $f(i, j)$:	Original Continuous Image/Digital Image;
$g(x, y)$/$g(i, j)$:	Observed Continuous/Digital Blurred Image;
$d(x, \alpha, y, \beta)$:	General PSF;
$e_n(x, y)$/ $e_n(i, j)$:	Environment Noise;
$e_m(x, y)$/ $e_m(i, j)$:	Modeling Noise;
Q:	Covariance of Environment Noise;
$\wp$:	Support of the Image;
$\wp_d$:	Support of the PSF;$\mathcal{SP}$
$G(v, u)$:	Fourier Transform of $g(x, y)$;
$D(v, u)$:	Fourier Transform of $d(x, y)$;
$F(v, u)$:	Fourier Transform of $f(x, y)$;
$N(v, u)$:	Fourier Transform of $n(x, y)$;
$\mathcal{S}$:	Set of Discrete Sites in Markov Random Field;
$\mathcal{L}$:	Set of Labels;
$\mathcal{N}$:	Neighborhood System;
$\mathcal{C}$:	Clique of MRF;
$U(f)$:	Emerge Function of MRF;
$V_c(f)$:	Potential Function of MRF;
$\mathcal{P}$:	Probability;
$\mathcal{p}$:	Probability Density Function;
$\mathcal{W}$:	Window for the Sparse Kernel in 2-D Formulation;
$\mathbb{F}$:	Markov Random Field;
S_w:	Window Size of the Bayesian Kernel;
T_{add}:	Confidence Threshold of adding an Edge;
$L_{f'}^{\infty}$:	Infinite Norm Measurement of Step Edge Preservation;
L_S:	Measurement of Step Edge Preservation;
L_R:	Measurement of Step Roof Preservation;

K:	Kernel Matrix;
$\phi(i,i')$:	Kernel Function;
w:	Weights for the Basis Function;
η:	Scaled Parameter for Linear Kernel Function;
α_i:	Hyper-parameter for the Weights;
$bd(i)$:	Sign for the Edge; 1 for edge; 0 for non-edge;
ρ:	Hyper-parameter for the MRF/MC;
σ_w^2:	Variance of the fitting Error with the Window;
m:	Location of the Edge
p:	Model Order of the Piecewise Constant.
d_i:	Magnitude of the Edge.
$L(x,y)$:	Shadow Image;
ℓ_y:	Identification Parameter for the PSF's Extent;
$\hbar(\ell_y)$:	Sub-Space of the Parameter ℓ_y;
P:	Similarity Measurement of the PSF;
r:	Rotation Radius;
θ:	Rotation Angle;
$\mathbb{R}^2$:	Real Domain
$\mathbb{Z}^2$:	Integral Domain
$\mathcal{C}$ / $\mathcal{C}_z$:	Continuous/Discrete Cartesian Domain
$\mathcal{P}$ / $\mathcal{P}_z$:	Continuous/Discrete Polar Domain
$\mathbb{T}_{\mathbb{R}}$ / $\mathbb{T}_{\mathbb{R}}^{-1}$:	Continuous Coordinates Transform and Its Inverse
$\mathbb{T}_{\mathbb{Z}}$ / $\mathbb{T}_{\mathbb{Z}}^{-1}$:	Discrete Coordinates Transform and Its Inverse
Δ_r / Δ_θ:	Polar Resampling Resolution for r and θ;
$S_n(r \mid \Delta_r)$ / $S_n(\theta \mid \Delta_\theta)$:	Sampling Operation in Polar Domain for Given Δ_r and Δ_θ;
$\Gamma(l, k)$:	Image in Discrete Polar Domain;

LIST OF ABBREVIATION

CT:	Computational Tomography;
MRI:	Magnetic Resonance Imaging;
SNR:	Signal to Noise Ratio;
PSF:	Point Spread Function;
i.i.d.:	Identical and Independent Distributed;
ML:	Maximum Likelihood;
AR:	Auto Regressive;
PDF:	Probability Density Functions;
LF:	Likelihood Function;
GCV:	Generalized Cross Validation;
VQ:	Vector Quantizer;
MAP:	Maximum à posteriori;
MRF:	Markov Random Field;
MCMC:	Markov Chain Monte Carlo;
SA:	Simulated Annealing;
db:	Decibels;
ICM:	Iterative Conditional Mode;
PIM:	Piecewise Image Model;
LIM:	Local Image Model;
PICO:	Piecewise Constant;
PILI:	Piecewise Linear;
LOMO:	Local Lonotonicity;
LOCO:	Local Convexity/Concavity;
GSG:	Generalized Stochastic Gradient;
LPA:	Local Polynomial Approximation;
ICI:	Intersecting Confidence Interval;
FFT:	Fast Fourier Transform;
AIC:	Akaike's Information Criterion;

MDL:	Minimum Description Length;
DKE:	Difference Kernel Estimation;
LSE:	Least Square Error;
TV:	Total Variation;
GML:	General Linear Model;
RMS:	Root Mean Square Error;
MSE:	Mean Square Error
MAE:	Mean Absolute Error
MAD:	Maximum Absolute Difference;
PSNR:	Peak Signal to Noise Ratio;
ISNR:	Improved Signal-Noise Ratio;
SVM:	Support Vector Machine
LSNR:	Local Signal-Noise Ratio;
PLSE:	Penalized Least Square Estimation;
ARD:	Auto Relevance Determination;
RVM:	Relevant Vector Machine;
LP:	Locally Piecewise Model;
ROI:	Region of Interest;
PRR:	Polar Resampling Resolution;
SP:	Support Points;
VP:	Virtual Points;

Chapter 1

Introduction and Literature Review

1.1. Introduction

Vision is a crucial sense for human beings. Combined with other senses, such as tactile, acoustical, gustatory, and olfactory, we are able to feel the real world. In the past decades, digital cameras have become cheaper and are widely used in our daily life. Digital cameras are also applied in many systems in scientific and industrial fields, such as astronomy, geophysics, robotics, assembly lines, biomedical application, satellitic investigations and security surveillance. These cameras play a crucial function in those systems and applications, just like the eyes of human.

About 44% of human beings have different problems with eyes, such as myopia, hypermetropia, askance and astigmatism *etc*. The imaging systems are not always perfect too, even though they are carefully designed, tested and calibrated. For example, Computational Tomography (CT) and Magnetic Resonance Imaging (MRI) are widely used in the biomedical research or hospital diagnosis, but both are sensitive to the environmental noise and the subjects in question. Some of the problems for the imaging system itself are too expensive to resolve and some are impossible to be fixed at all, while the images themselves are savable.

Hubble Telescope is a famous example of image restoration in astronomy. It took five years to design and launch it on its 615 km orbit in the sky with cost of about US$1.8 billion for this project at that time. Then scientists found the captured images were not as good as expected. Finally, they found out that the lens had two micron manufacturing error. It took another three years for the engineers to make another set of lens, and then launched a space shuttle to replace

them. During those three years, the telescope took thousands of precious photos and most of them cannot be taken again. The scientists have to rely on the technology of image restoration to save the degraded images.

Another example is the investigation of the crash of Columbia space shuttle. The crash of the space shuttle cost the life of 7 astronauts and billions of US dollars. Without the technology of image restoration, it is almost impossible to "see" that it was only a small piece of sponge that broke the ceramic layer at the shuttle left wing and thus caused the disaster.

Cases of imperfections in images are also found in the industries and daily life. In many industrial applications, moving, swinging and vibration of the cameras are the main reasons for obtaining blurred digital images [1]. Generally, the blurred images are not only degraded by a deterministic blurring process, but are also affected by noise. Film grain, electronic noise, quantization error, the atmosphere, and the recording medium all might introduce noise into the captured image. Noise is a random process, which plays a significant role in image restoration. Image restoration cannot be studied only without mentioning the noise, since nearly all the images are degraded by noise.

Generally, image restoration is defined as the problem of recovering an image from its blurred or/and noisy rendition for the purpose of improving its quality. Therefore, three kinds of image restoration problems are commonly encountered:

(a). Restore image from its blurred version when Signal to Noise Ratio (SNR) is high;

(b). Restore image from it noisy version without blurring;

(c). Restore image from its blurred version with poor SNR;

When the SNR is high, as in case (a), recovering the blurred image is a relatively easy process, which is known as deconvolution. Many methods have been developed to solve this problem. [2]~[5] provide extensive overview of those classical deconvolution techniques. More recent

methods will be reviewed in the following literature review section.

The removal of noise from image is an arduous task, as in case (b). This is because images are actually discontinuous functions in a regular 2-D array. The discontinuities are important since they encode information, such as the boundary of an object. The removal of noise and preservation of the discontinuities are two conflicting interests. Compromises have to be made between smoothing and sharpening. The main strategy is to define an estimate on a quantitative basis (*e.g.* a certain criterion or cost function) and to incorporate *à priori* information and constraints about the actual image, such as blur and noise, into the estimation process[3]. This strategy is known as regularization in the mathematics literature.

For a blurred image with poor SNR, as in case (c), there are basically two approaches to handle it. The first method is to remove the noise and then treat the problem as a case (a) problem. The second method is to balance the noise removal term and blurring removal term within one cost function as suggested by Miller [6].

Image restoration is well known to be a typical example of ill-posed inverse problem. This implies that the solution may not be unique or may not exist at all. From a mathematical point of view, therefore, image restoration is a regularization process in which a well-defined estimate of the actual image is determined from the degraded image using *à priori* information and the constraints. Often, *à priori* information about the image, the blur, and the noise are not readily available. It is thereafter necessary to identify these parameters too. In consequence, the process of restoring an image has two stages: identification and restoration.

Identification is a process that identifies all the necessary information needed in the restoration. This information generally includes the Point Spread Function (PSF) and the variance of the noise. In some industrial applications, parameters of the blurring could be well defined. The important parameters are available to describe the blurring process, such as the motion function of the camera, the distance between the objects and the camera, the frequency of the camera

vibration/motion speed and exposure time of the camera. In this case, the PSF of the blurring process could be determined analytically. However, these parameters are often not available or difficult to obtain in many industrial applications. There are two reasons:

- Some parameters are not stable in the imaging system and it would be difficult to estimate them accurately. Some of the parameters could be measured by adding sophisticated instruments, but it will make the system more complicated and expensive.
- In some situations, the blurred images are not reproducible, and not all the parameters are readily available.

Many scientists and engineers are more interested in identifying the blurring based on the blurred image itself only, which is called blind blur identification. The only available information for the identification task is the blurred image itself.

Strictly speaking, it is impossible to estimate the noise, since it is a random process. What we can estimate are the statistical parameters, such as mean and variance. Noise in the image model is normally assumed to be identical and independent distributed (*i.i.d.*) Gaussian noise, then the variance and mean will be sufficient to describe the noise term. The degraded image can be considered to be the convolution of the original image and PSF in spatial domain, with additive *i.i.d.* Gaussian noise. Thereafter, the restoration procedure is the inverse operation of the above procedure.

The problem of rotational blur is more complicated. This is because in the case of linear motion blur, the problem on identification and restoration is spatially invariant, while it becomes a spatially variant problem in the case of rotational blur. When the problem is spatially variant, using traditional methods to describe the nature of the problem is not feasible. It is impossible to use different PSFs to describe every pixel in the image. Thereafter, a new formulation is needed to solve the problem of rotational blur identification and restoration.

1.2. Objectives

The main objective of this work is to develop an efficient and novel image restoration method to restore blur images. The causes of blur are very different, each of which presents a different challenge, and hence have to be tackled separately. In this dissertation specifically, an approach that is able to restore images blurred due to rotational motion of the camera will be developed. The procedures are summarized in Figure 1.1.

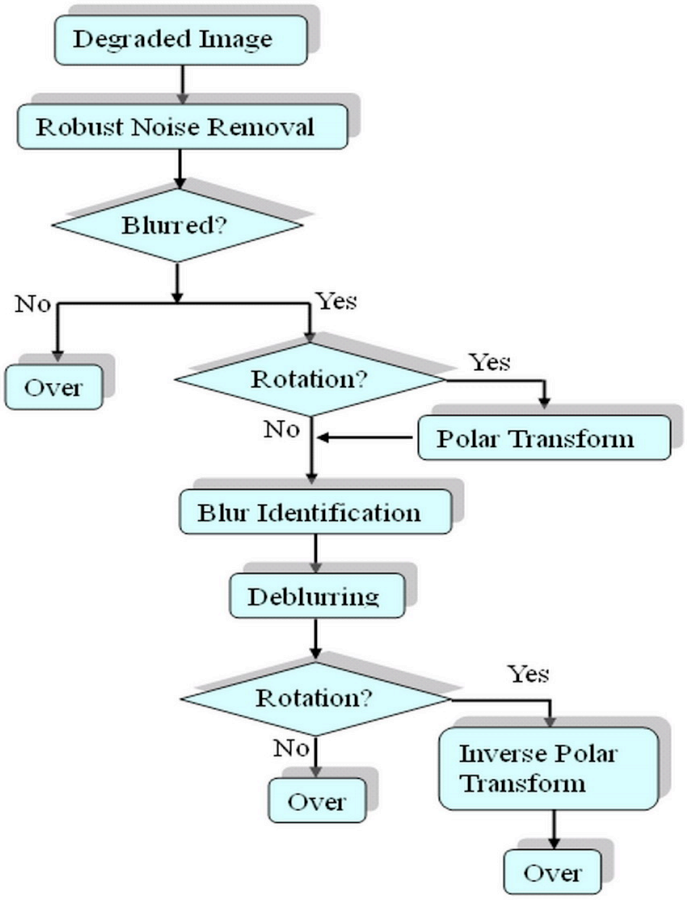


Figure 1.1 The Procedure of Image Restoration

The main steps after a blurred image has been captured are:

1. Noise removal.
2. Checking whether blurred image is caused by rotational motion?

If rotational blur → carry out polar transformation;

Else go to step 3.

3. Blur Identification
4. Deblurring → Deconvolution

   If rotational blur → carry on inverse polar transformation;

   Else go to step 5.
5. Presentation of the final restored image.

As image restoration comprises the above steps, each of which must be treated separately, before putting them together to form a cohesive whole. At this juncture, it would be necessary to spell out the objectives and approaches to be adopted in this work.

Noise Removal

It is well know that de-noising is an important issue in image restoration since it is an ill-conditioned problem. One of the reasons is that the noise will be easily amplified during filtering, such as high-pass filter for edge detection. The usual proposed cost functions for de-noising are non-convex, which is easily affected adversely by small variation in noise properties. This is highly undesirable. In this work, a convex cost function is proposed such that the problem will be solved by looking for a single global minimum. Therefore the problem becomes a well-posed problem.

Rotational Blur

There are cases where an image is blurred due to the rotation of the camera and/or the object in question. In this case, a rotational blurred image will be observed. The handling of rotationally blurred images is a complex spatially variant problem, which is very difficult to solve in practice. This is because the properties of the blur vary across the image, *i.e.* each pixel would exhibit different Point Spread Function (PSF). In this dissertation, a Compact Discrete Polar Transformation and its inverse are developed. With this transformation, the spatially variant

problem can be changed into a spatially invariant one. The restoration processing is done in the new transformed coordinates. The concept of compactness will be proposed. A significant advantage of the proposed novel approach in coordinates transformation is that the image is still uniformly sampled in the new polar coordinates, therefore, the existing image identification and restoration approaches can be easily implemented.

Blur Identification and Image Restoration

Blur identification is a crucial step in image restoration. The main aim is to identify the properties of blur, represented by PSF. Edge or discontinuity identification is useful and helps in blur identification. In this dissertation, both discontinuity identification and discontinuous functions/signals estimation based on Bayesian approach will be studied. An efficient edge identification based on Bayesian inference for the 1-D piecewise constant signal will be presented. Thereafter, a novel image restoration algorithm based on the Bayesian sparse kernel approach is proposed. This formulation guarantees the preservation of both step edges and roof edges in the restored images.

Importance of Noise Removal

Probably, the most basic (fundamental) image restoration problem is de-noising while preserving the image structure [7], such as the edges. The main emphasis of the methodology developed in this work is on the robust noise removal, with which, the blur, represented by PSF, will be identified correctly. If the image is rotationally blurred, it will be transformed to a new Polar coordinates for blur removal.

The basic reasons why edge preserved robust noise removal is so important and main concerns in this work, are:

- Noise is an important issue in image restoration. It will be easily amplified in the process of deconvolution. But, with an image of better quality, the deconvolution

procedure will become much easier.

- Noise removal will make the blur identification easier, since most of the existing blur identification methods are sensitive to noise. The original contents of the image when removing the noise must be preserved as far as possible, which makes the noise removal more challenging.

In the ensuing chapters, the fulfilment of the aforesaid objectives will be discussed in detail.

1.3. Literature Review

Two methodologies are popular in the field of image restoration, known as frequency approaches and stochastic approaches. Both of them have a long history from about 1970, or earlier. The former is based on spatial-frequency analysis. It applies Fourier transform, wavelet and filter design under this framework. The frequency approaches are faster and hence suitable for blurred image with high SNR (case a). However, when the SNR is low, most of them cannot yield satisfactory results. The stochastic approaches apply the correlation analysis, Markov random field and Bayesian inference based on *à priori* knowledge of the distribution of the variables. Stochastic approaches are robust and thus suitable for degraded images containing only noise (case b) or blurred image with low SNR (case c), since the noise is also included in the model. However, these approaches are computationally intensive and relatively slow.

A broad literature review is presented in this section, which will not be limited to the approaches in the image processing community. The related achievements in other fields are also included, such as applied statistics, machine learning, *etc*. Many efforts are trying to make the review as comprehensive as possible, however, due to the relatively large scope, it is limited only to those more relevant literatures. Some reviews of the previous works are also included in the individual chapters for their relevance and for easy comparisons.

1.3.1. Image Identification

Image identification is an indispensable step in image restoration. It provides necessary information for image restoration. Typically, the PSF and the noise variance are estimated at this stage. It is also called as blur identification in some other literatures.

Probably the most straightforward way to obtain the PSF of an image formation system is to make use of an analytic description of the system [1]. Unfortunately, in most of the practical applications, it is not always possible, because all the parameters are not readily known as stated in the last section. Another solution is to record an original image with known pattern, *e.g.* an impulse or an edge, for several times. Then the PSF can be obtained from the observations [5]. In fact, this method can be regarded as the solution of the classical system identification problem, where the response of a linear system is identified given its input and output. However, such an approach is rarely applied in blur identification because of the instability of the system and the associated environment.

A more practical approach assumes that certain patterns in the observed blurred image are *à priori* known [8]. However, since this method does not account for the noise in the observed image, small deviations between the assumed idealized pattern and the actual (unknown) pattern in the original image give rise to large deviations in the PSF identified in this way.

In the past decades, many scientists and engineers concentrated on blind blur identification. A number of advanced methods have been proposed. A successful approach towards image identification is from Stockham *et al*. [9] and Cannon [10]. Their spectral and cepstral methods concentrate on the PSF whose Fourier transform has a regular pattern of zero-crossings. This algorithm is based on inspection of the noisy and blurred image, using the fact that the frequency responses of some blurs have regular zero-crossings that determine the extent of the PSF [3]. Although these methods are still commonly used today for identification of blurs such as uniform motion blur and out-of-focus blur, PSFs which do not have zeros in their Fourier magnitude, such

as atmospheric turbulence [11], cannot be identified by these classical methods. Another deficiency of these methods is that the estimation of the location of the zero-crossings or the peaks is somewhat arbitrary, since it depends on eyes. Furthermore, this method is also sensitive to noise. When the SNR is low, acceptable results cannot always be obtained. This is because the presence of noise will mask the zero-crossing locations in the spectrum. These methods can only identify the extent of the PSF, but cannot determine the parameters of the PSF.

The PSF estimation is selected to provide the best match between the restoration residual power spectrum and its expected value, derived under the assumption that the candidate PSF is equal to the true PSF in [12]. The *à priori* knowledge required is the noise variance and the original image spectrum. A blind blur identification and order determination scheme is presented in [13]. With the blurs satisfying a given condition, the authors establish the existence and uniqueness of the result, which guarantees that single-input/multiple-output blurred images can be restored blindly, though perfectly in the absence of noise, using linear FIR filters. Results of simulations employing the blind order determination, blind blur identification, and blind image restoration algorithms are presented in their paper. In images with low SNR, indirect image restoration performs well while the direct restoration results vary with the delay but improve with larger equalizer orders.

The statistical approaches are robust since it includes the noise in the model. Model-based least square error and maximum likelihood (ML) blur identification methods have been proposed for the identification of the general symmetric PSF of finite extent. Various implementations of the ML estimator have been discussed [11]~[17]. These blur identification methods have the following advantages [11]:

i). They are criterion-based;

ii). They can identify a lager class of blurs;

iii). They can incorporate the presence of observation noise in the estimation.

Although ML identification enjoys the abovementioned theoretical advantages, the methods have not been applied widely in practice due to some shortcomings. Firstly, in most of the ML algorithms, the Auto Regressive (AR) model is assumed to be the image model. If this model does not fit the image, it does not produce reliable results. The Probability Density Functions (PDF) is assumed to be Gaussian distribution in most of the ML algorithms. The Likelihood Function (LF) to be maximized is highly nonlinear with respect to the unknown parameters. It requires the use of numerical optimization [11]. However, since the Likelihood Functions may contain several local maxima, convergence to the global maximum cannot be guaranteed [11]. The assumption of the symmetry of PSF makes this method unsuitable to the problems in which the PSF is non-symmetrical, such as accelerated motion blurs. Some important works of the blur identification, as well as image restoration, include Generalized Cross Validation (GCV)[18], Vector Quantizer (VQ) [19] ~ [20] and regularization approaches[21] ~ [22]. Other methods could be found in [23] ~ [25].

Besides the PSF, noise variance is also an important issue in image identification. Many blur identification methods, which are based on statistical models, can give a good estimation of the noise variance simultaneously. The estimation of the noise variance itself will relatively easier. The most straightforward way is the Wiener filter [26] [27], which possesses adaptive characters. Wavelet approaches are also popular and some newly developed methods are presented in [28] ~ [31].

1.3.2. Image Restoration

Image restoration has been extensively studied for its obvious practical importance as well as its theoretical interest. Literature on the subject is abundant and highly varied since the problem arises in almost every branch of engineering and applied physics. Most existing image restoration methods have a common estimation structure in spite of their apparent variety. Everything could

be summed up in a single word: regularization [3]. Generally, the image restoration methods could be classified into two general categories according to the difference in regularization. The first category is called Criterion-based methods. The central idea of these methods is that the solution is defined to be the image that satisfies a predetermined optimality criterion. Based on different criteria, the methods are classified as [4]: Minimum-norm least squares, Linear Minimum Mean Square (LMSE) error, and Maximum *à posteriori* probability (MAP). The second category is called Constrained-optimization methods. These methods optimize an optimality criterion subject to constraints on the solution. The constraints and the criteria reflect the *à priori* information about the ideal image. According to these constraints, these methods are [4]: Constrained least squares, Miller regulation, Maximum entropy, *etc*. Both Criterion-based methods and Constrained-optimization methods can be implemented in frequency domain and spatial domain. Normally, it was called deconvolution techniques. Many scientists and engineers have been partial to the statistical Bayesian inference in the past decades due to its robustness to noise. It is strongly related to a branch of statistics called function regression. Function regression is a classical statistical problem, which has existed more than 100 years. It provides tools for estimating functions, curves, or surfaces from the degraded and noisy data.

Julian Besag presented an excellent fundamental way to treat image restoration problem statistically in both regular and irregular lattices [32]. The Hammersley-Clifford theory was proved. From then on, the Markov Random Field (MRF) model became popular in image processing. However, due to high computational load, it developed relatively slowly in practice for decades. With the rapid development of computer technology and efficient algorithms, many related papers about the Bayesian Image processing have been published in the past 15 years.

S. Geman and D. Geman contributed their idea of “line process”, also known as the Gibbs sampler, which is a typical treatment of discontinuities in image processing [33]. Thereafter, the Gibbs sampler became popular not only in image processing tasks, but is also applied in many

other fields.

Bouman provided a Maximum *à posteriori* (MAP) model [34], which allows realistic edge modeling while promising stable solutions. Bouman argues that a non-convex cost function will cause the estimation to be ill-conditioned. Therefore in order to obtain a stable estimation, a convex cost function is preferred. Similar study about the convexity of the cost function appeals to other researchers due to its practical significance [35] ~ [37]. The idea of using a non-standard norm function as the cost function is presented in [34], so that the discontinuities in the signal will not be over-penalized and the edge could be preserved. According to the simulation, when the norm is between 1 and 2, say $\|.\|^{1\sim2}$, the MAP estimation is guaranteed to be a continuous function of the input data, while slow convergence of the MAP estimation may occur in practice [34]. A systematic study on the existing cost functions in the Bayesian frameworks was also discussed. Some other researchers presented other different ideas about limiting the penalty of local differences at a prescribed threshold [38] [39].

Stephens introduced a Bayesian retrospective multi change-points identification algorithm [40]. Gibbs sampler is applied to relieve the computational efforts. By applying it on three real data sets, it is shown that this method is able to identify the discontinuous points in the data sequence.

A Bayesian curve fitting to deal with the signals of piecewise polynomials characteristics is proposed by Denison *et al.* in [41]. The reversible jump Markov Chain Monte Carlo (MCMC) method [42] is applied to calculate the joint posterior distribution. In the reversible jump MCMC [42], the Gibbs sampler could jump between the state spaces of different dimensionalities according to preset conditions. Denison's formulation [41] has been successful in giving good estimation for smooth function, say continuous and differentiable, as well as functions which are not differentiable and perhaps even not continuous, at a finite number of points. The detailed pseudo code and procedure of the algorithm are also provided.

Molina *et al.* presented the Bayesian image restoration for blurred images[43]. They follow the

Gemans' line process concept using the Compound Gauss-Markov Random Field. Both stochastic and deterministic relaxations are considered. It also provided the basis of applying the Simulated Annealing (SA) and Iterative Conditional Mode (ICM) in the deblurring problem of image processing. The convergences of the two iterative algorithms are proven theoretically and tested by synthetic and real blurred images.

Two important image models for the problem of image restoration are introduced in [44], which are Piecewise Image Model (PIM) and Local Image Model (LIM). PIM also includes the Piecewise Constant (PICO) and Piecewise Linear (PILI). LIM includes the Local Lonotonicity (LOMO) and Local Convexity/Concavity (LOCO). The corresponding regularization functions for different image models are given. The author provided a cross-validation approach to select the proper regularization parameter to compromise between noise removal and edge sharpening. The approach can be used to select the proper image model. The experimental results gave the comparison of their method and other parameter selection method. Similar problems are discussed in [45] and [46]. Although the author starts from a MRF point of view for the image restoration, the final cost function is very similar to the PIM and LIM in [44]. A Bayesian estimation of the regularization parameter is given and sampling problem from the posterior distribution is analyzed in [45]. In [46], a Generalized Stochastic Gradient (GSG) algorithm with a fast sampling technique is devised aiming to achieve simultaneous Hyper-parameter estimation and pixel restoration. Image restoration performances of Posterior Mean performed during GSG convergence and of Simulated Annealing performed after GSG convergence are compared experimentally.

A novel non-parametric regression method was proposed in [47]. The approach is based on the Local Polynomial Approximation (LPA). The Intersecting Confidence Intervals (ICI) method [48] is applied to define the adaptive varying scales (window size) of LPA estimation. The ICI algorithm will give a varying scale adaptive estimate defining a single best scale for each pixel.

Since the FFT is applied to the convolution, the speed of the method is fast. The LPA-ICI approach could be viewed as the selection of the weights for each pixel according to different window size. This approach can achieve the best performance among the approaches which can be found in all the reviewed literature.

Besides the Bayesian methodology, Thomas introduced an automatic smoothing method for recovering discontinuous regression functions [49]. Three different criteria are derived from three fundamentally different model select methods: Akaike's Information Criterion (AIC) [50], Generalized Cross-Validation (GCV) [51] and the Minimum Description Length (MDL) [52]. A modified genetic algorithm is applied for the practical optimization. This paper is under the framework of spine methodology and the three above criteria are also not based on Bayesian rule, while in the experimental works and simulation, these methods yield sound results. It gives us an overview of those non-Bayesian based methods in discontinuous function regression in the past ten years.

Recovering an image from its noisy version is actually a 2-D discontinuous function regression problem. Function regression for the discontinuous function is relatively new compared to continuous function regression. It was also called jump analysis or change point analysis in some other literatures. Two related topics are included in this field: discontinuity identification and discontinuous function estimation. Discontinuity identification is to identify the positions of discontinuities in the given functions/signals, such as the edge detection and the change point analysis. In those applications, the localities of the discontinuities will furnish sufficient information. Discontinuous function estimation focuses on good estimation for the given discontinuous functions/signals. Two approaches are both active in discontinuous function estimation. Some authors treat the functions in the sub-intervals separated by the identified discontinuous points as a continuous function regression problem; alternatively others remove the discontinuities in the functions first and then consider the entire function as a conventional

continuous function regression problem.

In the statistical community, the similar problem of noise removal attracted many scientists too. Conventional function regression methods work well in the estimation of the continuous function, while those estimators can not deal with the signals/images in the presence of discontinuities. From the statistical point of view, they can not converge to the true discontinuous points. Many real applications in image processing frequently involve those discontinuous points, such as edge detection. The kernel method became popular and active in the past decades in both function regression and classification. It provides an important tool for image restoration. The Nadaraya-Watson kernel was presented by Nadaraya [53] and Watson [54] separately. It was initially designed for the continuous function problem, while it was modified for the discontinuous function problem later. Some similar kernel methods appeared, such as those presented by Priestley and Chao [55], Gasser and Muller [56], Cheng and Lin [57]. Some other algorithms were also proposed, such as Difference Kernel Estimation (DKE) [58] ~ [59], Difference of two one-sided Local linear Kernel estimator [58]. These methods could be categorized as local kernel methods, since they only deal with the local information of the given signal according to the given bandwidth. The bandwidth of the local kernel, *e.g.* the width of the kernel window, has a significant impact on the results. Many researchers suggested different criteria on how to select a proper bandwidth, such as the General Cross-Validation (GCV) [60], Cp criterion[61] and the Least Square Error (LSE) criterion [62] ~ [63].

Since its introduction in a classical paper by Rudin, Osher and Fatemi [64], Total Variation (TV) minimizing models have become one of the most popular and active methods for image restoration. TV based regularization is a typical case of geometry-driven diffusion for image restoration. A reliable and efficient computational algorithm for image restoration is proposed in [65]. A piecewise linear function (a measure of total variation) is minimized subject to a 2-norm inequality constraint (a measure of data fit) for discrete image. The blur is removed first by

finding a feasible point for the inequality constraint. Noise and other artifacts are then removed by subsequent total variation minimization. Y. L. You, *et al*. presented an anisotropic regularization to exploit the piecewise smoothness of the image and the PSF [66]. This method is derived from anisotropic diffusion and adapts both the degree and direction of regularization to spatial activities and orientations of the image and PSF. TV is used as a constraint in a general convex programming framework in [67]. David, *et al.* discuss about how to choose a proper regularization parameter in the unconstraint TV formulation to determine the balance between the goodness of fit to the original data and the amount of regularization to be done to the images [68]. T.F. Chan, *et al*. presented a good overview on the development of the TV method in image restoration in the passed 15 years in [7]. Other algorithms for the TV method in image restoration can be found in [69] ~ [75].

Many other approaches are also proposed in image restoration focusing on different aspects, such as robust noise removal [76] ~ [79], edge preserved denoising [80] ~ [87], and ring artifacts [88] ~ [91], *etc*.

1.4. Contributions and Organization of the Dissertation

In this dissertation, image restoration problem is studied in the framework of stochastic approaches.

Mathematical models of image, blurring and noise will be provided in Chapter 2. In this Chapter, the General Linear Model (GLM) and kernel learning are introduced for the task of image restoration. Markov Chain (MC) and Markov Random Field (MRF) models of 1-D signal and 2-D image are discussed. Bayesian inference of the above model is briefly reviewed. Chapter 2 ends with the definitions on the fidelity criteria of image quality. They will be used to evaluate the results of restored images.

Chapter 3 presents the identification of discontinuities in 1-D piecewise constant signal based on

Markov Chain and Bayesian Inference. The *à priori* distribution of models parameters and regularization strategy are discussed. An efficient edge detection algorithm for 1-D piecewise constant signal is provided. The proposed method can avoid the direct evaluation of marginal likelihood function in a high dimensional space. The posterior evidence is suggested for the model selection. Then the experimental results are presented to verify its robustness and accuracy.

The formulation of the Relevance Vector Machine is presented at the beginning of Chapter 4. The mathematical model of the piecewise linear signal are then presented. Based on this model, the scaled linear kernel is proposed as the kernel function. Thereafter, the given formulation is applied for the 1-D discontinuous function regressions. The discontinuities/edges can be preserved in the restored signals. The identification of the discontinuities is successful and the fitness of the data is excellent according to the experimental results.

The Relevance Vector Machine (RVM) and Sparse Probabilistic Linear Model are generalized to image restoration tasks in Chapter 5. The advantages and limitations of RVM and its solution are discussed. Some important issues, such as the kernel function and the tuning of the hyper-parameter are studied. The cost function of the global optimization is presented. The proposed cost function has only one global minimum, such that the ill-conditioned problem becomes well-posed. The experimental results are compared with other existing methods at the end of Chapter 5 and excellent performance has been achieved.

Chapter 6 presents a formulation of the blind blur identification in spatial domain based on the piecewise constant assumptions. The image is divided into edge region and smooth region. It is actually ML estimation of the PSF. Both the extent and the parameter of the PSF can be thus accurately determined. Since no assumption is made on the shape of the PSF, it can be used to identify the PSF of any shape. The proposed identification algorithm is proved to be robust by the experimental results.

Chapter 7 proposes an innovative Compact Discrete Polar Coordinates Transform for the

identification and restoration of rotational motion blurred image. A relaxed formulation of the transform function and its inverse is presented. Based on the analysis and assumptions, the linear interpolation is applied in the Polar domain to solve the problem of under-sampling. The cost function of the compactness is introduced. Experimental results shows that this transformation can successfully convert the rotational motion blur problem to a spatially invariant problem.

Finally, the conclusion is presented in Chapter 8.

Chapter 2

Mathematical Model for Image, Noise and Blurring Process

In this Chapter, the mathematical representation of image, noise, blurring process and all the associated quantities are presented. These models, which will be used extensively in this dissertation later, will allow us to understand and resolve the intended problems, using available mathematical tools.

2.1. Image and Image Representation

Mathematically, an image is a function of two dimensions with specified characteristics. Some images need to be processed in order to improve their quality for better human or automatic machine interpretation. The process is commonly known as image restoration technology.

In this work, only monochrome scene of the nature will be discussed, which can be expressed by a bivariate function $f(x, y)$ $\{-\infty < (x,y) < \infty\}$, where (x, y) denotes the spatial location in the image. An image is only a part of such function $f(x, y)$, since imaging device is normally spatially limited. For example, the lens has limited zoom and depth, and the CCD matrix cannot be infinite. Thereafter, the image becomes $f(x, y)$ $\{(x, y) \in \wp \subset \mathbb{R}^2\}$, where $\wp = \{0 \le x \le X, 0 \le y \le Y\}$ is the support of the image. The function $f(x, y)$ is often called the image intensity function. For computers to handle an image, ranging from image storage in the hard disk to image processing with some computer software, the image needs to be digitized beforehand in both spatial location and with the associated brightness values. Theoretically, a continuous image could be digitized in the spatial domain by the δ function. This is the functionality of the so-called digitizer device in camera or scanner. Every point, say (i, j), is a pixel. The digitized value of the function f at pixel

(i, j) is also known as the gray level. This digital image of size N by M is then denoted by $f(i, j)$ $\{(i\ ,j)\in\ \wp\subset\mathbb{Z}^2\}$, where $\wp=\{0\le i\le N,\ 0\le j\le M\}$. The number of pixels of a digital photo is commonly referred to as image resolution. For example, the above image is of size $N\times M$. Another common definition of the resolution of a digital image is the Dots Per Inch (DPI). More details of sampling theory and digitization of image can be found in most image processing textbooks, such as [1][2][5]. In this dissertation, unless otherwise specified, all images mentioned refer to monochrome digital images.

A noisy image can be considered to consist of the original image $f(i, j)$ and the noise $e_n(i, j)$. Expressed mathematically, the noisy and degraded version $g(i, j)$ of the original image $f(i, j)$ is given by:

$$g(i,j)=f(i,j)+e_n(i,j)\quad \{1\le i\le N,1\le j\le M\} \tag{2.1}$$

Eq. (2.1) is a typical noisy image model. The major goal of image restoration is to estimate $f(i, j)$ based on the observed data $g(i, j)$.

When quick motion is involved, the image regions are illuminated only in a partial integral of time interval in the blurring process. The intensity (or gray level value) is not only decided by the region being sampled, but it is also influenced by the adjacent regions based on the given PSF. Furthermore, the noise is inevitable. In this way, the noisy blurred image in the continuous domain could be modeled by:

$$g(x,y)=\iint_{\wp_d} f(\alpha,\beta)\,d(x,\alpha,y,\beta)\,d\alpha\,d\beta+e_n(x,y)\quad \{(x,y)\in\wp\} \tag{2.2}$$

where $g(x, y)$ is the observed noisy blurred image;

$f(x, y)$is the original image;

$d(x, \alpha, y, \beta)$ is the PSF;

$\wp_d$ is the support of the PSF, which is often considerably smaller than $\wp$;

e_n is the noise term, which is often independent from the image. The noise is assumed to be a zero-mean homogeneous Gaussian distributed white noise process with zero mean and unknown variance σ_n^2.

When the PSF is spatial invariant, Eq. (2.2) could be simplified as:

$$g(x,y) = \iint_{\wp_d} f(\alpha,\beta)\, d(x-\alpha, y-\beta)\, d\alpha\, d\beta + e_n(x,y) \quad \{(x,y) \in \wp\} \tag{2.3}$$

In discrete formulation, Eq. (2.3) could be written as:

$$g(i,j) = \sum_{(k,l)\in\wp_d} d(k,l) f(i-k, j-l) + e_n(i,j) \quad \{(i,j) \in \wp\} \tag{2.4}$$

The observed blurred image in Eq. (2.4) is actually the convolution of the PSF and the original image with the additive *i.i.d.* noise. In image processing literature, the above expression could be represented in a concise way:

$$g = d \otimes f + e_n \tag{2.5}$$

where $\otimes$ denotes the convolution operator. Since the convolution operation in the spatial domain will be the multiplication operation in the frequency domain, some researchers prefer to express the model in frequency domain:

$$G(v,u) = D(v,u) \times F(v,u) + E_n(v,u) \tag{2.6}$$

where $G(v, u)$, $D(v, u)$, $F(v, u)$ and $E_n(v, u)$ are the Fast Fourier Transform (FFT) of $g(i, j)$, $d(i, j)$, $f(i, j)$ and $e_n(i, j)$, respectively. The technique of determining the inverse of the above blurring procedure is known as deconvolution.

The above mathematical expressions shown in Eq. (2.1) ~ Eq. (2.6) of image are simple, but not

all the 2-D functions are eligible to represent an meaningful image. Most of the images are not continuous and have piecewise characteristics. In image processing, positions, at which the image intensity function *f* has "jumps", are called step edges and positions at which the first-order derivatives of *f* have "jumps" are called roof edge. They will be discussed in Chapter 3 in detail. Due to the existence of such discontinuities, the image restoration task is more challenging. The removal of the noise (smoothing) and preservation of the discontinuities (sharpening) are conflicting interests. Compromise between the sharpening and smoothing must be achieved to keep the discontinuities. This is also known as "Edge Preserved Noise Removal".

At this juncture, the support of the image and the PSF should be defined. The support of a function/image is the region of interest within a coordinate system. The region outside of the support is either undefined or unsampled. The coordinate system and the support of the image are shown in Figure 2.1. Different coordinate systems could be used for sampling and digitization. In this dissertation, both the image and the PSF are in Cartesian coordinates with uniform sampling interval in 2-D array. It is also known as the regular lattice. Some types of blurring PSF could be described by a vector, such as the linear motion PSF.

Figure 2.1 The Coordinates System and the Support of the Image

In some applications, other types of noise could be more appropriate, such as Laplace noise for electrical noise. In this work, it is assumed that the noise is Gaussian distributed since it is the most commonly used noise model in image processing. Theoretically, it is impossible to completely and accurately describe such random process. However it could be represented by its statistical parameters, such as the mean and the variance. Due to the *i.i.d.* assumption defined above, the noise will have covariance matrix $Q = \sigma^2 I$, where I is an $NM \times NM$ identical matrix with all the diagonal elements equal to 1.

2.2. General Linear Model and Kernel Trick

In this section, an extremely important model used in image representation will be discussed due to its wide applications. This model will be used extensively in Chapter 4 and Chapter 5. Any data which may be described in terms of a linear combination of basis functions with an additive Gaussian noise component satisfies the General Linear Model (GLM), also known as the General Additive Model. Hastie and Tibshirani [92] gave an excellent summary on this topic. One of the most successful GLM is the Support Vector Machine (SVM). This model is also well defined and discussed in the literature of neural networks and machine learning [93] ~ [96]. Due to its success in many fields, such as the gene study, classification and function regression, SVM is well developed in the past 20 years. Some Bayesian treatments for the SVM are proposed recently, such as [97].

Generally, the GLM is described by a linear combination of a set of basis/kernel functions $\phi(k', k)$. In image processing, the image f could be modeled by:

$$f(k; \vec{w}) = \sum_{k=1}^{NM} \phi(k', k) w_k \tag{2.7}$$

where w_k is the weight for different basis functions. Thereafter the observed image g will be:

$$g(k;\vec{w}) = \sum_{k=1}^{NM} \phi(k',k) w_k + e_n(k) \tag{2.8}$$

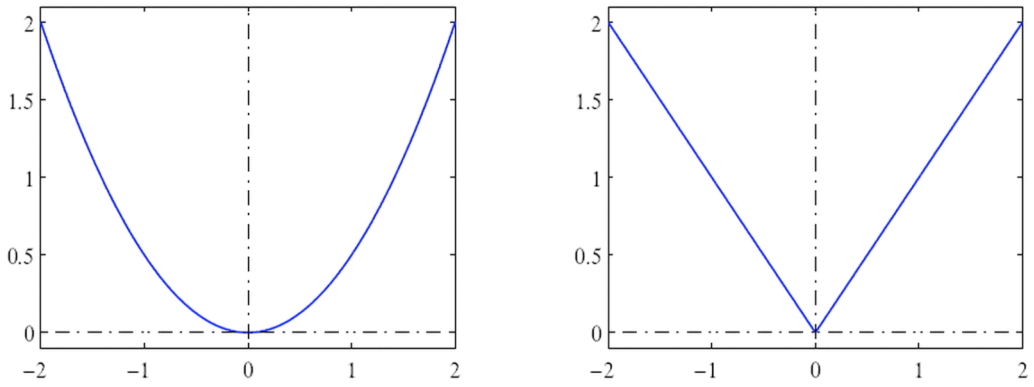
The image is re-arranged into a column vector by line scanning scheme. $\phi(k',k)$ is known the basis function, and also called as the kernel function in kernel machine learning. More concisely, the above expression can be written as:

$$\vec{f} = \boldsymbol{K}\vec{w} \tag{2.9}$$

and

$$\vec{g} = \boldsymbol{K}\vec{w} + \vec{e} \tag{2.10}$$

where $\boldsymbol{K}$ is the Kernel matrix, with elements $\phi(k',k)$.



(a). Quadratic Kernel Function (b). Linear Kernel Function

Figure 2.2 Quadratic and Linear Kernel Function

GLM provides many theoretical conveniences in both theoretical inference and practical numerical calculation. The discontinuities in the image introduce the non-linearity into the image model. The expressions in Eq. (2.9) and (2.10) are linear; however the non-linearity could be encoded into the model by the non-linear kernel function. The nonlinear kernel function maps the variables into a new hyperspace. In this hyperspace, the model could have a linear expression. This is known as the *kernel trick* to treat the nonlinearity by a linear expression. Basis functions need to satisfy the following three conditions:

1. Symmetry : $\phi(k',k)=\phi(k,k')\ \forall k,k'$

2. Non-negativity: $\phi(k',k)\geq 0 \quad \forall k$

3. Cauchy-Schwarz Inequality: $[\phi(k',k)]^2 \leq \phi(k,k)\phi(k',k') \quad \forall k,k'$

Many functions satisfy the above conditions, such as the Inhomogeneous Polynomial kernel function, Homogeneous Polynomial kernel function, Gaussian Kernel function, Sigmoid Kernel function, *etc.* The Quadratic Kernel and the Linear Kernel are shown in Figure 2.2.

2.3. Definition of Discontinuity

In many image processing works, such as enhancement, recognition, restoration, *etc*. discontinuities play an important role in providing necessary key information for the task. In this work, we shall begin the formulation of 1-D signal for ease and clarity in presentation. The model presented here shall be further discussed in Chapter 3~5.

Many mathematicians prefer to define the continuity of a function via a so-called epsilon-delta definition, while a simpler description is given here. The given function is f: $X \rightarrow Y$ (X and $Y \subset \mathbb{R}$), where the pre-image of every open set in Y is open in X. More concretely, a function $f(x)$ of a single variable is said to be continuous at point x_o if the following relation holds:

$$\lim_{x\to x_0^+} f(x) = \lim_{x\to x_0^-} f(x) = f(x_0) \tag{2.11}$$

where $f(x_0) < \infty$. $f(x_0)$ is defined in the domain of f. If $\forall\, x_o \in X$ satisfy Eq. (2.11), function $f(x)$ is continuous in its entire defined space X. If $f(x)$ is infinitely differentiable, it could be written according to the Taylor expansion:

$$f(x) = f(x_o) + f'(x_o)(x-x_o) + f''(x_o)(x-x_o)^2 .... \frac{f^{(n)}(x_o)}{n!}(x-x_o)^n + ... \tag{2.12}$$

In many practical tasks, only the lower order derivatives in Eq. (2.12) are significant and

important. When the function is continuous at point x_o, it means that function $f(x)$ is at least second order differentiable at point x_0. A function $f(x)$ is continuous at point x_0 if it satisfies Eq. (2.11) and its second order derivative is also differentiable at the entire space in question. Removal of noise from such function is not difficult, the SVM approaches and other local kernel methods can successfully handle this task. In this work, some of the points of function $f(x)$ will not satisfy Eq. (2.11), because at these points, the first order derivative of the function does not exist, and are hence not continuous at those points. Since the discontinuities in the higher order derivatives in Eq. (2.12) will not result in visual discontinuities, only the discontinuities of the first two terms in Eq. (2.12) will be considered.

The discontinuity of the first term represents step edge in an image; it is also referred to as a jump in some literatures. It is defined as:

$$\begin{cases} \lim_{x \to x_0^+} f(x) = c_1 \\ \lim_{x \to x_0^-} f(x) = c_2 \\ c_1 \neq c_2 \end{cases} \tag{2.13}$$

where c_i (i=1,2) are finite constants. The left-limit and right-limit of $f(x)$ at x_0 both exist, but are not equal, *i.e.* not continuous. The difference between c_1 and c_2, $d = c_2 - c_1$, is also known as the magnitude of the step edge.

The discontinuity of the second term in Eq. (2.12) is called a roof edge, which means the first order derivative of $f(x)$ is not continuous. Expressing it mathematically,

$$\begin{cases} \lim_{x \to x_0^+} f(x) = \lim_{x \to x_0^-} f(x) = c_0 \\ \lim_{x \to x_0^+} f'(x) = c_1 \\ \lim_{x \to x_0^-} f'(x) = c_2 \\ c_1 \neq c_2 \end{cases} \tag{2.14}$$

where c_i (i=0,1,2) are finite constants, $f'(x)$ is the first order derivative of the function $f(x)$. Eq.

(2.14) indicates that the function $f(x)$ is continuous at x_0 but its first order derivative is not.

Based on the above definitions, and to handle discontinuities which are important in dealing with edges, the function is assumed to be piecewise linear/constant. The discontinuous function $f(x)$ can be written as:

$$f(x) = a_0 + b_0 x + \sum_{j=1}^{p} (a_j + b_j x) I(x_j) \qquad x \in X \tag{2.15}$$

where a_j and b_j are the constant coefficients for the piecewise linear function. The definition of function I is given by:

$$I(x_j) = \begin{cases} 0 & x \le x_j \\ 1 & x > x_j \end{cases} \qquad j = \{1, 2 \ldots p\} \text{ and } x \in X \tag{2.16}$$

where p is the number of discontinuous points and $\{x_j, j = 1, 2, \ldots p\}$ are the discontinuous points in function $f(x)$.

According to Eq. (2.15), there are three possible types of discontinuities in the given function, as shown in Figure 2.3:

1. Step edge: $a_j \neq 0$, but $b_j = 0$ $(1 \le j \le p)$;
2. Roof edge: $a_j = 0$, but $b_j \neq 0$ $(1 \le j \le p)$;
3. Hybrid of Step and Roof: $a_j \neq 0$ and $b_j \neq 0$ $(1 \le j \le p)$.

The above discussions are valid in the continuous domain $\mathbb{R}$. In image processing using computer, the image in question must be digitized. The process is known as sampling or digitization. The sampling system, is either a camera or a scanner, the image resolution is usually well defined to suit the task at hand. Sampling methods vary with applications. In some cases, data could be obtained at any given point and the sampling interval may not be uniformly distributed. In this work, images are captured by CCD camera with uniformly distributed

rectangular sensor array. The sampling interval is thus fixed and sampled points are uniformly distributed. Any points between the two sampled points is considered as "unobservable".

(a). Step Edge

(b). Roof Edge

(c). Hybrid of Step and Roof Edge

Figure 2.3 Different Discontinuities in the 1-D Piecewise Linear Function

The existence of discontinuities in the image function demands a high sampling frequency (as per Nyquist Sampling Theory). The sampling interval should be small enough to make these discontinuous points observable. After sampling, the function will be in discrete domain represented by Eq. (2.17):

$$f(i) = a_0 + b_0 i + \sum_{j=1}^{p} (a_j + b_j i) I(m_j) \tag{2.17}$$

where $i=\{1,2,3...N\}$ is the index of sampled points and $j=\{1,2...p\}$ is that of the discontinuous point. N is the total sampling points and p is the number of discontinuous point.

The definition of function I is:

$$I(m_j)=\begin{cases}0 & k \le m_j \\ 1 & k > m_j\end{cases} \quad j=\{1,2,..p\} \text{ and } k=\{1,2,..N\} \tag{2.18}$$

where $m_j \in \{1,2,3\cdots N\}$ is the positions of discontinuous points in the signal.

The signal given in Eq. (2.17) is actually a Markov Chain(MC). In Chapter 3, the details of how to generate such a signal will provided.

2.4. Markov Random Field and Bayesian Inference

Many computer vision problems can be posed as labeling problems, the solution of which is a set of labels assigned to image pixels. Image Processing is also a natural representation for the study of Markov Random Field (MRF) [32]. Generally, MC is for 1-D signal and MRF is for 2-D images. MC will be discussed and studied in details in Chapter 3 and Chapter 4.

Labeling problem is specified in terms of a set of *sites* and a set of *labels*. Let $\mathcal{S} = \{1,2,\ldots NM\}$ be the discrete set of $N \times M$ *sites*, which are actually points in the Euclidean space/Cartesian lattice, such as image pixels. *Sites* on a lattice in this dissertation are considered as spatially regular. A *label* is an event that may happen at a site. Let $\mathcal{L}$ be the set of *labels*. The *label* set may be categorized as continuous or discrete. In this dissertation, $\mathcal{L}$ is an interval of $\mathbb{R}$, say $\mathcal{L} = [L_{min}, L_{max}] \subset \mathbb{R}$. For example, in image processing tasks, $\mathcal{L}$ is between [0,255], which is the commonly used 256 gray levels for a monochrome image.

The labeling problem is to assign a *label* from the label set $\mathcal{L}$ to each of the *sites* in $\mathcal{S}$. Edge detection in image processing, for example, is to assign a label from the set {edge, non-edge} to sites in $\mathcal{S}$. The set $f=\{f_1=L_1, f_2=L_2,\ldots f_{NM}=L_{NM}\}$ is called labels of sites in $\mathcal{S}$ in terms of the labels in $\mathcal{L}$. When each site is assigned a unique label, f can be regarded as a function with domain $\mathcal{S}$ and image $\mathcal{L}$. Because the support of the function is the whole domain $\mathcal{S}$, it is a mapping from $\mathcal{S}$ to $\mathcal{L}$,

which is

$$f : \mathcal{S} \to \mathcal{L} \tag{2.19}$$

The definition of $\mathcal{S}$ is actually equivalent to $\wp$ for digital image, the support of the image.

In the terminology of random fields, a labeling is called a *configuration*. A configuration can be corresponding to an image, an edge map, and so on. When all the sites have the same label set $\mathcal{L}$, the set of all possible labelings, which is the configuration space $\mathbb{F}$, follows the Cartesian produce:

$$\mathbb{F} = \mathcal{L} \times \mathcal{L} \times ... \mathcal{L} = \mathcal{L}^{N \times M} \tag{2.20}$$

where $N \times M$ is the total number of the pixels in the image. In image restoration, for example, when the image size is 256×256 and the gray level is 256, then the total number of possible configuration in $\mathbb{F}$ is $(256)^{256 \times 256}$. Examination of all configurations is infeasible. In some literatures, this is called as the "curse of dimensionalities". Some constraints need to be imposed. In fact, only few configurations in the configuration space are meaningful in most applications. The constraints are also called the *à priori* knowledge in Bayesian inference.

The *sites* in $\mathcal{S}$ are related to one another via a neighborhood system $\mathcal{N}$. A neighborhood system for $\mathcal{S}$ is defined as:

$$\mathcal{N} = \{\mathcal{N}_i \mid \forall i \in \mathcal{S}\} \tag{2.21}$$

where $\mathcal{N}_i$ is the set of *sites* neighboring *i*. The neighborhood relationship has the following properties:

1. A given *site* is not neighboring to itself: $i \notin \mathcal{N}_i$;
2. Neighboring relationship is mutual: $i \in \mathcal{N}_{i'} \Leftrightarrow i' \in \mathcal{N}_i$.

For a regular lattice, the neighborhood of i is defined as the set of *sites* within a radius of r from i:

$$\mathcal{N}_i = \{i' \in \mathcal{S} \mid (i-i')^2 \le r^2, i' \ne i\} \tag{2.22}$$

The commonly used 4-neighbor and 8-neighbor system in image processing literature is defined by $r = 1$ and $r = \sqrt{2}$, respectively. In the MRF, there is another concept called cliques, which is all the possible combination of the *sites* within the neighborhood.

$\mathbb{F}$ is said to be a Markov Random Field on $\mathcal{S}$ with respect to a neighborhood system $\mathcal{N}$ if and only if the following two conditions are satisfied:

1. $\mathcal{P}(f) > 0 \quad \forall f \in \mathbb{F}$ (2.23)

2. $\mathcal{P}(f_i \mid f_{\mathcal{S}-\{i\}}) = \mathcal{P}(f_i \mid f_{\mathcal{N}_i})$ (2.24)

In this work, capital $\mathcal{P}$ stands for the probability and p means the probability density function. $\mathcal{S} - \{i\}$ is the set difference. The first condition is called Positivity and the second is the Markovity. The Positivity is assumed for technical reasons, since the probability of a configuration to occur must be equal or greater than zero. It is usually satisfied in practice. When the Positivity condition is satisfied, the joint probability $\mathcal{P}(f)$ of any random field is uniquely determined by its local condition probabilities [32]. In MRF, only neighboring *labels* have direct interaction with each other. If the largest neighborhood is selected, in which the neighbors of any *sites* include all other *sites*, then every random field is a MRF with respect to such a neighborhood system [98]. MRF has other properties, such as homogeneity and isotropy, while common images do not have these properties.

There are two ways to specify an MRF, one of which is in terms of the conditional probabilities $\mathcal{P}(f_i \mid f_{\mathcal{N}_i})$ and the other one is the joint probability $\mathcal{P}(f)$. In [32], Besag preferred the joint probabilities in view of the disadvantages of the conditional probabilities approach. Firstly, no

obvious method is available for deducing the joint probability from the associated conditional probabilities. Secondly, the conditional probabilities themselves are subject to some non-obvious and highly restrictive consistency conditions. Thirdly, the natural specification of the equilibrium of statistical process is in terms of the joint probabilities instead of the conditional probabilities. Fortunately, a theoretical result on the equivalence between the MRF and the Gibbs distribution provides a mathematically tractable means of the joint probabilities of MRF, known as the Hammersley and Cllifford theorem[32][99].

A set of random variables is said to be a Gibbs Random Field (GRF) on $\mathcal{S}$ with respect to $\mathcal{N}$ if and only if its configurations obey a Gibbs distribution. A Gibbs distribution takes the following form:

$$\mathcal{P}(f) = Z^{-1} \times e^{-\frac{1}{T}U(f)} \tag{2.25}$$

where

$$Z = \sum_{f \in \mathbb{F}} e^{-\frac{1}{T}U(f)} \tag{2.26}$$

Z is a normalizing constant called the partition function. T is a constant called the Temperature factor. $U(f)$ is the energy function, which is:

$$U(f) = \sum_{c \in \mathcal{C}} V_c(f) \tag{2.27}$$

$U(f)$ is a sum of clique potential $V_c(f)$ over all possible cliques $\mathcal{C}$. The value of $V_c(f)$ depends on the local configuration. Note that Gaussian distribution is a special member of this Gibbs distribution family, which is generally called the Gaussian MRF.

Bayes statistics is a theory of fundamental importance in estimation and decision making. According to this theory, when both the prior distribution and the likelihood function of a pattern

are known or assumed, the best estimation could be achieved by following the Bayes formulation based on the sources of the knowledge. The Maximum *à posterior* (MAP) is sought in many vision works since it was advocated by Geman and Geman[33] and many others. Although the MRF is a straightforward representation of the image both in theory and practice, it is not sufficient to handle the ill-conditioned problem of image restoration. Generally, the MRF cooperates with Bayesian theory to form so-called MRF-MAP estimation.

In Bayesian estimation, a risk is minimized to obtain the optimal estimation. The Bayes risk of an estimate $\hat{f}$ based on f is defined as:

$$R(\hat{f}) = \int_{f \in \mathbb{F}} C(\hat{f}, f) \wp(f \mid g) df \tag{2.28}$$

where $C(\hat{f}, f)$ is a cost function and $\wp(f \mid g)$ is the posterior distribution of estimated f given observed g. The posterior distribution can be calculated from *à priori* and the likelihood. Following the Bayes rule, the posterior probability can be computed by:

$$\wp(f \mid g) = \frac{\mathcal{P}(g \mid f) \wp(f)}{\mathcal{P}(g)} \tag{2.29}$$

where $\wp(f)$ is the prior probability of f, $\mathcal{P}(g \mid f)$ is the conditional probability of the observations, also known as the likelihood function (LF) of f when g is given, and $\mathcal{P}(g)$ is the probability of g to occur, which is a normalization constant.

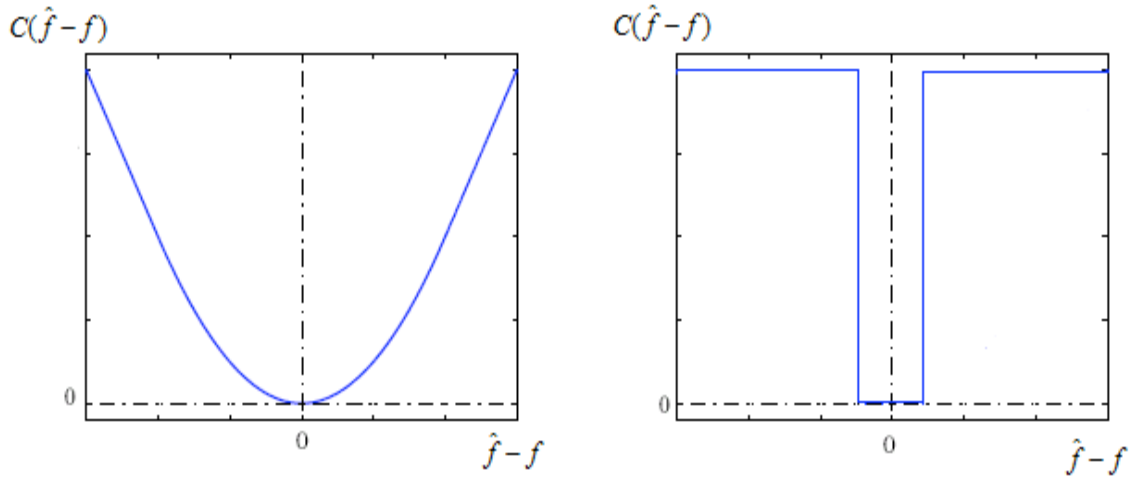
The cost function $C(\hat{f}, f)$ determines the cost of the estimate $\hat{f}$ when f is true. It could be defined according to different preferences and applications. Two popular choices are the quadratic cost function:

$$C(\hat{f}, f) = \left\| \hat{f} - f \right\|^2 \tag{2.30}$$

And the δ(0-1) cost function:

$$C(\hat{f}, f) = \begin{cases} 0 & \text{if } \left\| \hat{f} - f \right\| < \delta \\ 1 & \text{otherwise} \end{cases} \tag{2.31}$$

These two common used cost functions are shown in Figure 2.4.



(a). Quadratic Cost Function (b). δ Cost Function

Figure 2.4 Quadratic Cost Function and δ Cost Function

Different cost functions will yield different estimates of f, for example using quadratic cost function, the minimal variance estimation can be obtained:

$$\hat{f} = \int_{f \in \mathbb{F}} f p(f \mid g) df \tag{2.32}$$

which is actually the mean of the posterior distribution. With the δ(0-1) cost function, the minimum risk estimate is:

$$\hat{f} = \arg \max_{f \in \mathbb{F}} (p(f \mid g)) = \arg \max_{f \in \mathbb{F}} (p(g \mid f) p(f)) \tag{2.33}$$

It is known as the MAP estimation. Obviously, when the prior distribution $p(f)$ is flat, the MAP will be equivalent to the ML estimation.

2.5. Fidelity Criteria of Image Quality

Removal of the noise generally causes a loss of real or quantitative visual information. Because information of interest may be lost during the improvement of image quality, repeatable or reproducible measurement of quantifying the nature and the extent of image loss is highly desirable. Two general classes of criteria are used as the basis for such assessment: objective criteria and subjective criteria. When the observed image *g* is degraded by noise:

$$g(i,j) = f(i,j) + e_n(i,j) \quad \{1 \le i \le N, 1 \le j \le M\} \tag{2.34}$$

If the information loss can be expressed as a function of the original image *f* and the estimated image $\hat{f}$, it is said to be based on an objective criterion. Because of the redundant of image, the definitions of such criteria are somewhat difficult and diverse. The image quality itself usually means the visual effects of the image, *e.g.* which image is better and clearer by examination using our naked eyes, which is highly subjective. A single mathematically defined objective criterion often is not sufficient to describe the image quality. Some criteria have been proposed and have satisfied two basic imperatives for image quality criterion. If a criterion for the image quality is ℓ, then the imperatives can be stated as:

1. When $\hat{f} = f$, then $\ell = 0$;
2. If $\ell_1 < \ell_2$, then $\hat{f}_1$ should have better visual observation than $\hat{f}_2$.

Such criteria commonly used to evaluate the quality of the image in image processing and computer vision community, include the Root Mean Square Error (RMS), Mean Square Error (MSE), Mean Absolute Error (MAE), and Maximum Absolute Difference (MAD). The definitions of the above criteria are shown in Eq. (2.35) ~ (2.38).

$$RMS = \sqrt{\frac{1}{NM}\sum_{i=1}^{N}\sum_{j=1}^{M}[\hat{f}(i,j) - f(i,j)]^2} \tag{2.35}$$

$$MSE = \frac{1}{NM}\sum_{i=1}^{N}\sum_{j=1}^{M}[\hat{f}(i,j) - f(i,j)]^2 \tag{2.36}$$

$$MAE = \frac{1}{NM}\sum_{i=1}^{N}\sum_{j=1}^{M}[|\hat{f}(i,j) - f(i,j)|] \tag{2.37}$$

$$MAD = \max_{i,j}[|\hat{f}(i,j) - f(i,j)|] \tag{2.38}$$

These criteria are actually measurements of the difference of the original image and the estimated image. It is clear that these four criteria satisfy the first imperative as stated above. However, eliminating the error/noise completely is impossible and difficult in most of the applications. Generally the second imperative is more important in the evaluations of the image quality. Fortunately, it is proven that these criteria also generally satisfy the second imperative. These four criteria are based on different norm function, say, RMS is root of 2-norm, MSE is 2-norm squared, MAE is 1-norm and MAD is infinite-norm.

Besides the criteria of the image quality, some descriptions of the noise are also needed, more accurately, a mathematical description of the relationship between the noise, the original image and the estimated image. The commonly used quantitative measurements of the noise are the Signal-Noise Ratio (SNR), Improved Signal-Noise Ratio (ISNR) and the Peak Signal-Noise Ratio (PSNR). The formulations of the three measurements are:

$$SNR = 10log_{10}\left(\frac{\sum_{i=1}^{N}\sum_{j=1}^{M} f(i,j)^2}{\sum_{i=1}^{N}\sum_{j=1}^{M}(e_n(i,j))^2}\right) \tag{2.39}$$

$$ISNR = 10log_{10}\left(\frac{\sum_{i=1}^{N}\sum_{j=1}^{M}(f(i,j) - g(i,j))^2}{\sum_{i=1}^{N}\sum_{j=1}^{M}(f(i,j) - \hat{f}(i,j))^2}\right) \tag{2.40}$$

$$PSNR = 20log_{10}(\frac{\max_{i,j}(f(i,j))}{RMS}) \tag{2.41}$$

All these criteria and quantitative measurements are called the objective fidelities of the image quality. Note that these measurements of the noise and the errors are all related. It is actually very hard to judge whether an image is better than another one based only on one or two such criteria. Obviously, it is difficult to evaluate the image quality, since it is common that one image might have a low noise measurement and smaller error, and yet is worse in visual observation than another image which has higher noise measurement and bigger error. In this case, all the criteria from Eq. (2.35) to (2.41) will be used as the measurements of the image quality in the following Chapters and use them to present an objective comparison of the experimental results.

Chapter 3

Bayesian Inference of Edge Identification in 1-D Piecewise Constant Signal

Edge identification and detection are important processes in image processing works. Edges are the discontinuities in an image as they help in demarcating regions of interests and of different properties. In this chapter, Bayesian Inference approach will be developed for edge identification in 1-D piecewise constant signal with Markovity. In Chapter 4, Bayesian kernel method for edge detection and the reconstruction of the 1-D piecewise linear signal will be proposed.

3.1. Markov 1-D Piecewise Signal with Step Edge

In this Chapter, the signal is 1-D piecewise constant, which is commonly encountered in image processing applications. Different terminologies are used to represent image/signal. In statistical analysis, the term "function" is often used; whilst in image and signal processing, "image/signal' is used. In this dissertation, "function" is used to denote image/signal in continuous domain, *i.e.* before sampling; and "image/signal" represents its discrete version.

As mentioned several times above, discontinuities, such as the outline of the objects, provide key information to help in various image processing procedures; in this work, image restoration. Discontinuities useful to observation are the line discontinuities, or commonly known as edges. The mathematical representation of edges has been presented in details in Section 2.3, based on which the ensuing discussion would follow.

Because the step edge is more common in images and its location is very important in some processing, such as the edge detection, the identification of the step edge will be studied in this Chapter, *e.g.* $b_j = 0$ in Eq. (2.17) in the last chapter. In this regard, the function could be written

as:

$$f(i) = \mu_0 + \sum_{j=1}^{p} d_j I(m_j) \tag{3.1}$$

where μ_0 is the initial value of the function and d_j is the magnitude of the edge. The function I is defined in Eq. (2.18).

Supposing that for a given sampled point, the probability of the point to be a step edge point is:

$$\mathcal{P}(bd(i) = 1) = \rho \tag{3.2}$$

where ρ is a constant between (0, 1). bd is a logical variable to determine whether the given points is a edge point ($bd(i)$ = 1 for edge point and $bd(i)$ = 0 for non-edge point). Then the probability of the point is not an edge is $\mathcal{P}(bd(i) = 0) = 1 - \rho$. Note that ρ is also called the transition kernel in some literatures on Monte Carlo simulation. It indicates how likely it will change the status (edge/non-edge) in the next step. For a given point $f(i\text{-}1)$, the next state will be determined by:

$$f(i) = \begin{cases} f(i-1) & \text{if } bd(i)=0 \\ h & \text{if } bd(i)=1 \end{cases} \tag{3.3}$$

where h is another *label* sampled from $\mathcal{L}$ which is not equal to $f(i\text{-}1)$. The following procedure is applied to generate the piecewise constant signal which has the characteristics of Markovity:

1. Initialize the parameter $0 < \rho < 1$;
2. Generate a random *label* from $U(0,255)$, then give this value to $f(1)$;
3. Generate a random value a from $U(0,1)$, then compare it with ρ;
   a). If $a > \rho$, $bd(i)=0$ and $f(i)=f(i\text{-}1)$
   b). If $a < \rho$, $bd(i)=1$, generate another random *label* from $U(0,255)$ and give it to $f(i)$.
4. Increase i, if $i \leqslant N$, go back to 3, otherwise, terminate.

where $U(0,255)$ is a uniform distribution between 0 and 255. It is assumed that the *label* only can take integer value, as the common gray level values.

Markov chain is time/spatially invariant in this work, *i.e.*, the value of ρ should be a constant or could be approximately treated as a constant with respect to time/space. With the increase in ρ, there will be more edge points within a given length of signal, and vice versa. The number of the edge points p is called the model order in this work. It will be determined by the summation of the variable *bd*, which is:

$$p = \sum_{i=1}^{N} bd(i) \tag{3.4}$$

Note that the number of edges p is also a random variable, which has a *Binomial* distribution according to a given probability ρ. As noise is inevitable in the observed data or sampled image, instead of dealing with the original clean signal represented by Eq. (3.1), only its degraded version is observable, which is represented by Eq. (3.5).:

$$g(i) = f(i) + e(i) \tag{3.5}$$

where e is the noise term, which is assumed to be an *i.i.d* Gaussian noise with zero mean and unknown variance σ^2,say, $e \sim N(0, \sigma^2)$.

Bayesian theory is to use *à priori* knowledge, learnt from the given signal and then use it to update the posterior distribution. Note that unsuitable *à priori* knowledge will yield wrong solutions or limit the applicability of the model. In this case, the reasonable *à priori* knowledge includes:

1. f is piecewise constant. There are only countable discontinuous points in the given signal.
2. The number of edge points is finite, but how many edges are in the signal and where they are unknown.

3. *e* is an *i.i.d* Gaussian noise with zero mean and unknown variance σ^2.

4. The magnitude between any two consecutive pieces should not be known.

3.2. Step Edge Detection Based on Posterior Evidence in 1-D

3.2.1. Estimation of the Edge Points with Known Edge Number

If the number of the edge points *p* is known in the case of multi-edge points, the estimate of the edge can be done with either ML or MAP method. The analytical solutions will be identical. The parameters needed to estimate are given in Eq. (3.6) and (3.7):

$$\vec{\mu} = \{\mu_0, \mu_1, ...\mu_p\} \tag{3.6}$$

$$\vec{m} = \{m_1, m_2 ... m_p\} \tag{3.7}$$

where $\mu_i = \mu_0 + \sum_{j=1}^{i} d_j$, $m_j \le i$ for $i=\{1,2,\ldots N\}$ and $j=\{1,2,\ldots p\}$. If the noise variance is included into the model, the parameter vector is $\vec{\theta} = \{\vec{\mu} = (\mu_0, \mu_1 ... \mu_p),\ \vec{m} = (m_1, m_2 ... m_p), \sigma^2\}$, which give a total of 2(*p*+1) elements. Two special edge points are defined, which are m_0 =1 and m_{p+1}=*N*, for the convenience of inference. In fact, they can be considered as edge points with known positions.

For the problem of identifying edge points given by Eq. (3.1), (3.6) and (3.7), the likelihood function is given by the joint probability of the noise samples:

$$\mathcal{P}(\vec{g} \mid \vec{\theta}, p) = \prod_{i=1}^{N} \mathcal{P}(e(i)) \tag{3.8}$$

Since:

$$\mathcal{P}(e(i)) = \frac{1}{\sigma\sqrt{2\pi}} \exp(-\frac{(e(i))^2}{2\sigma^2}) \tag{3.9}$$

Substitute Eq. (3.8) into Eq. (3.9):

$$\mathcal{P}(g \mid \theta, p) = (2\pi\sigma^2)^{-N/2} \exp(-\frac{1}{2\sigma^2}(\sum_{i=1}^{N} e(i)^2)) \tag{3.10}$$

The summation in Eq. (3.10) could be decomposed into:

$$\sum_{i=1}^{N}(e(i))^2 = \sum_{i=1}^{m_1}(g(i)-\mu_0)^2 + \sum_{i=m_1+1}^{m_2}(g(i)-\mu_1)^2 \ldots \sum_{i=m_p+1}^{N}(g(i)-\mu_p)^2 \tag{3.11}$$

where p is the number of the edge points, is also called as the model order. According to Bayesian rule, the posterior distribution of the parameters will be:

$$\mathcal{P}(\{\vec{\mu},\vec{m},\sigma\} \mid \vec{g}, p) = \frac{\mathcal{P}(g \mid \{\vec{\mu},\vec{m},\sigma\}, p)\mathcal{P}(\{\vec{\mu},\vec{m},\sigma\} \mid p)}{\mathcal{P}(\vec{g} \mid p)} \tag{3.12}$$

In order to deduce the posterior distribution, the prior distribution of the parameter vector needs to be defined. The parameters, $\vec{\mu}$, $\vec{m}$, and σ, are assumed to be independent of each other. Then the prior distribution follows:

$$\mathcal{P}(\{\vec{\mu},\vec{m},\sigma\} \mid p) = \mathcal{P}(\vec{\mu} \mid p)\mathcal{P}(\vec{m} \mid p)\mathcal{P}(\sigma \mid p) \tag{3.13}$$

The first term in Eq. (3.13) is the prior distribution function of the gray levels for the given signal (digitized image). Two conjoint pieces must have different gray levels, however, if they are not conjoint, they are allowed to have the same gray level. All the elements in $\vec{\mu}$ are assumed to be independent and identically and uniformly distributed [100][101]:

$$\mathcal{P}(\mu_i \mid p) = U(0,255) \qquad (i = 0,1,2,\ldots p) \tag{3.14}$$

Another argument then arises on the validation of this assumption of independence. If the property of *i.i.d* represented by Eq. (3.14) holds, it cannot guarantee that any two given conjoint pieces would have "obvious" different values. For example, the probability of μ_i=100 is 1/256. μ_{i+1} will also be sampled according to Eq. (3.14) if it is a step edge. Then the probability that μ_{i+1}=100 is 1/65536. Although the value of the probability is small, it is still not equal to zero. So

if all the conjoint pieces have "enough" distinct value, then all μ_i's are not absolutely independent. Fortunately, the dependence between two conjoint pieces can usually be ignored. In practice, when the signal is generated, the edge magnitude is not regulated. Contrarily, if μ_i and μ_{i+1} have the same value, then the model p-1 is true, instead of there are p edge points. Furthermore, here it should be highlighted that the gray level difference between any two consecutive pieces should be notably higher than the noise level. For example, when the noise level is σ=10, but the difference between two consecutive pieces are 10^{-2}, *i.e.* $|\mu_i - \mu_{i-1}| < 10^{-2}$, then we should not expect the algorithm could identify such a tiny edge point.

The second term is the prior distribution of the edges location. Since no information is available initially, it is natural to use some non-informative prior distribution, which is [100][101]:

$$\not{p}(m_j \mid p) = \frac{p}{N} \quad (j = 1,2,...p) \tag{3.15}$$

where N is the length of the signal. This indicates that the prior probability of every point in the signal to be qualified as an edge point is equal to p/N. Similarly, no element in $\vec{m}$ should be equal to each other. For example, if there are two elements in $\vec{m}$ that are equal, the p-1 is true, instead of there are p edge points.

The last term on the right hand-side of Eq. (3.13) is the prior distribution of the noise. Here the non-informative Jeffrey *à priori* form is used, which is [100] ~ [101]:

$$\not{p}(\log\sigma \mid p) = c \tag{3.16}$$

where c is a constant. Therefore, the following is obtained [100] ~ [101]:

$$\not{p}(\sigma \mid p) = \frac{c}{\sigma} \tag{3.17}$$

Note that the uniform distribution in Eq. (3.14), Eq. (3.15) and Eq. (3.17) are not proper. The denominator in Eq. (3.12) is only a constant. Based on the assumption that the model p is true

(there are p edge points within the given signal) and the determined prior distribution, then we obtain:

$$\wp(\{\vec{\mu},\vec{m},\sigma\}\mid\vec{g},p)\propto\frac{1}{\sigma}\mathcal{P}(\vec{g}\mid\{\vec{\mu},\vec{m},\sigma\},p) \tag{3.18}$$

In order to deduce the location of the edge, the marginal distribution $\wp(\vec{m}\mid\vec{g},p)$ is desired. Integrate it with respect to $\vec{\mu}$ and σ, giving us:

$$\wp(\vec{m}\mid\vec{g},p)=\iint\frac{1}{\sigma}\mathcal{P}(\vec{g}\mid\{\vec{\mu},\vec{m},\sigma\},p)d\vec{\mu}d\sigma$$

$$\propto\frac{1}{\sqrt{(m_1-m_0)\times(m_2-m_1)...(m_{p+1}-m_p)}}\exp[\sum_{i=1}^{N}(g(i))^2-\sum_{j=1}^{p}\frac{1}{m_j-m_{j-1}}S_j^2]^{-\frac{N-2}{2}} \tag{3.19}$$

where:

$$S_j=\sum_{i=m_{j-1}+1}^{m_j}g(i)\quad (j=1,2,\ldots p+1) \tag{3.20}$$

Note that $m_0=1$ and $m_{p+1}=N$ are defined at the beginning of this section.

In addition to the location of the edge points, the magnitude of the edge could be deduced, which could be determined by:

$$\ln\mathcal{P}(d_j,\sigma,m_{j-1},m_j,m_{j+1}\mid\vec{g})\propto$$

$$\ln C_1+C_2[\frac{(m_{j+1}-m_{j-1}-1)(m_j-m_{j-1})-(m_j-m_{j-1})^2}{(m_{j+1}-m_{j-1}-1)}d_j^2+$$

$$2(\frac{(m_{j+1}-m_{j-1}-1)-(m_j-m_{j-1})}{(m_j-m_{j-1})}S_j^T-S_{j-1}^T)d_j+S_j^{T^2}-\frac{1}{(m_{j+1}-m_{j-1}-1)}(S_j^T)^2] \tag{3.21}$$

where C_1 and C_2 are unrelated constants, and:

$$S_j^T = \sum_{i=m_{j-1}}^{m_{j+1}-1} g(i) \tag{3.22}$$

$$S_j^{T^2} = \sum_{i=m_{j-1}}^{m_{j+1}-1} g^2(i) \tag{3.23}$$

Then based on Eq. (3.21), the inference gives us the estimation of the magnitude of the edges d_j:

$$\hat{d}_j = \frac{(m_j - m_{j-1})S_j^T - (m_{j+1} - m_{j-1} - 1)S_j}{(m_{j+1} - m_j - 1)(m_j - m_{j-1})} \tag{3.24}$$

Noise is also an important concern. Estimate of the unknown variance of the Gaussian noise is given by:

$$\frac{1}{N}[\sum_{i=1}^{N}(g(i))^2 - \sum_{j=1}^{p}\frac{1}{m_j - m_{j-1}}S_j^2] \sim N(\sigma^2, \frac{\sigma^2}{N} + \sum_{j=1}^{p}\frac{2\mu_j\sigma^2}{N}) \tag{3.25}$$

According to the central limit theorem, the above term will converge to the real value of the variance when N is a large number. The error of the estimation of the noise variance is proven to be bounded.

Evaluation of the marginal distribution in Eq. (3.19) of low dimensions is feasible, while with the increase of the dimensionalities, it will fall into the "curse of dimensionalities". For example, if p=1, the marginal distribution can be evaluated for N points, while if p =K, then Eq. (3.19) needs to be evaluated for $(N)^K$ points. Even for small values of N and K, say N=256, K=5, calculation of Eq. (3.19) is $(256)^5$=1.1000×10^{12} points, which is very computational intensive to implement in practical.

3.2.2. Posterior Evidence Based on Sequent Model Select

The inference of the above section is for the multi edge points in a given signal when the number of the edge points is already known. Even though analytical solution can be deduced, it is difficult to evaluate the posterior distribution when the model order is high. Since the uniform distribution

is used as the prior and the normalization constant is ignored, the solutions in Eq. (3.19) and (3.24) are the same with the ML solution. Using the ML method to infer the value of the parameters would have serious limitation when the model order is unknown. Firstly, the likelihood does not use information other than the data itself to infer the value of the parameters before the data is obtained. In some cases, a physical understanding will suggest that some values of the parameters are not possible. There are many cases that ML can return parameter estimations outside the sensible range of the parameter in question. Secondly, the likelihood on its own does not limit the number of parameters in a model used to fit the data. In general, the more complex the model, the better fit would be to the data.

ML method will easily over-parameterize the model. The major difference of the ML and the MAP is that the latter (Bayesian) uses *à priori* knowledge to learn from the obtained data. Generally, in practice, the number of edges is not known in the given signal *à priori*. Thus, this given problem has two levels of inference:

1. Assume that there is a number of edges in the signal, say p, is true; find the parameters from the given data based on *à priori* knowledge. This is the first level of the problem, called model fitting.

2. For the given set of assumed model (p=1,2…Q), find the best model p for the given data, also called model selection.

The last Section 3.2.1, only provides the solution for the first level of the inference when p is known. Furthermore, as shown in the same section, it is basically impossible to calculate the marginal posterior distribution in Eq. (3.19) in a high dimensional space when p is large. Since no information is readily available about where the edges are located at the beginning, all the points in the signal could be the candidates of the edges. The number of distinct possibilities for the position of the edge points is $N!/[(N-k)!k!]$. It is not feasible to examine all the possible edge points over the high dimensional space of $(N)^k$. Special treatment is needed in practical

computation.

The Markov Chain Monte Carlo (MCMC) introduced in Chapter 1 is popular in the past years. It is well known to be insensitive to the dimensionality. This makes it a powerful tool in Bayesian framework. For the given problem in Eq. (3.19), the maximum in $\wp(\vec{m} \mid \vec{g}, p)$ needs to be evaluated, but cannot do so analytically. If samples $\{\vec{m}_t, t = 1, 2, ...\}$ are drawn from the posterior distribution, then the expected/approximated posterior distribution will be:

$$E(p(\vec{m} \mid \vec{g}, \vec{\theta}, \sigma)) \approx \frac{1}{n} \sum_{t=1}^{n} \vec{m}_t \tag{3.26}$$

Here a Bayesian approach of the multi-edge model selection is presented. The choice is made due to the advantage offered by the Bayesian formulation. It can solve the two levels of inference mentioned above, *i.e.*, the model fitting and the model selection. According to Bayesian rule, the model fitting is given by:

$$\wp(\vec{\theta} \mid \vec{g}, p) = \frac{\mathcal{P}(\vec{g} \mid \vec{\theta}, p)\wp(\vec{\theta} \mid p)}{\mathcal{P}(\vec{g} \mid p)} \tag{3.27}$$

The model selection is described by:

$$\wp(p \mid \vec{g}) = \frac{\mathcal{P}(\vec{g} \mid p)\wp(p)}{\mathcal{P}(\vec{g})} \tag{3.28}$$

In [102], it is suggested that the evidence $\mathcal{P}(\vec{g} \mid p)$ is sufficient for model select task when the *prior* of the model $\wp(p)$ is uniformly distributed and *a prior* distributions for all the parameters in Eq. (3.27) are proper. Unfortunately, some of the prior distributions in Section 3.2.1 are not proper, therefore, $\mathcal{P}(\vec{g} \mid p)$ cannot be used to estimate the model order in the inference. In order to alleviate this problem, it is assumed that $\wp(p)$ is not a uniform distribution in this work. Fortunately, the sampling systems are normally designed with pre-determined resolution; the captured signal can usually provide us with enough information for the high level processing

tasks.

Prior distribution for the model in Eq. (3.28) should not be uniformly distributed. The model prior distribution will be determined by the hyper-parameter ρ defined in Eq. (3.2). It describes the complexity of the model. In fact, the prior distribution of the model is a *Binomial* distributed in discrete form. *Gamma* distribution can be used to approximate it in the continuous domain. To determined the model order, the "posterior evidence" $\wp(p \mid \bar{g})$ which is in Eq. (3.28) is evaluated, instead of the evidence $\mathcal{P}(\bar{g} \mid p)$. Note that ρ is also a variable which is not known *à priori*. How to estimate its value will be discussed later.

As stated above, the likelihood function can always be increased by increasing the model order. Of course, adding an edge point in the given signal can increase the likelihood when the confidence is low, while "posterior evidence" will be penalized by $\wp(p)$. This phenomenon is known as the Occam's razor [103] ~ [104]**.** It states that if several models are compatible with a set of observations/data, the least complex model is preferred. This can be simply explained by the fact that a model with fewer parameters would usually be less complex and more likely to occur.

In order to calculate $\wp(p \mid \bar{g})$, the evidence $\mathcal{P}(\bar{g} \mid p)$ should be calculated first, which is the normalization constant in Eq. (3.27):

$$\mathcal{P}(\bar{g} \mid p) = \int_{\bar{\theta}} \mathcal{P}(\bar{g} \mid \bar{\theta}, p) \wp(\bar{\theta} \mid p) d\bar{\theta} \tag{3.29}$$

Calculating the normalization constant in Eq. (3.29) is computational intensive, as the state space for the integration is in the order of $(N)^k$, where N is the length of the signal and k is the dimension of the parameter $\bar{\theta}$. Mackay [102] suggested to use the peak value of $\wp(\bar{\theta} \mid \bar{g}, p)$ multiplied by its width, or say variance, to approximate $\mathcal{P}(\bar{g} \mid p)$. Similar idea is used here. Since if there is an edge point in the given model, the probability of the edge point in its right location

will dominant the range of defined space. Based on this idea, Eq. (3.29) can be calculated by:

$$\mathcal{P}(\vec{g} \mid p) \approx \int \mathcal{P}(\vec{g} \mid \vec{\theta}_{ML}, p) \mathcal{P}(\vec{\theta}_{ML} \mid p) d\vec{g} \tag{3.30}$$

In the above equation, only the probability of locating the most likely edge points are integrated to estimate the evidence. By doing so, the evidence is evaluated in the state space of $1 \times N$. Some error could, however appear in this calculation. The normalizing constant could be different for different models due to the nature of the non-proper prior distributions.

Here, the location of the edges can be determined step by step. First assume that there is no edge point in the given signal. By evaluating Eq. (3.27) (which is actually Eq. (3.19)), decide whether the assumption is right. The confidence level of adding a new edge point is 75%. If it is lower than the given confidence level, calculation can be terminated. If a higher confidence is obtained that an edge point exists in the given signal, an edge point is added into a list and find the best value of the edge location based on Eq. (3.19). At the same time, the estimation of $\mathcal{P}(\vec{g} \mid p)$ is updated. Since all the edge points are assumed to be independent of each other, we can now continue to search for another edge point. Note that the calculation of evidence $\mathcal{P}(\vec{g} \mid p)$ is according to Eq. (3.30). Step by step increasing the model order (the number of edge points determined thus far) one by one and then we determine the properties of edge points (location and magnitude) in sequence. The procedure is summarized as follows:

1. Assume p=0;
2. Calculate the posterior distribution according to Eq. (3.19);
3. Determine whether update the estimated hyper $\hat{\rho}$ based on a given confidence level,
    (a). If have higher confidence(>75%), insert an edge point and update $\hat{\rho}$, determined it position according to the Maximum, update the value of evidence $\mathcal{P}(\vec{g} \mid p)$ according to Eq. (3.30).

(b). If have higher confidence(>75%), insert an edge point without update $\hat{\rho}$, update evidence $\mathcal{P}(\vec{g} \mid p)$.

4. Increase p go back to Step 2, until all model orders $p=\{1, 2, \ldots K\}$ are examined.

5. Get the *a prior* of $\mathcal{P}(p)$ according to the estimated $\hat{\rho}$, then determine the model order according to $\mathcal{P}(p \mid \vec{g})$

At the end of the above procedures, the most proper model is selected according to the "posterior evidence". $\mathcal{P}(p)$ is actually controlled by the hyper-parameters ρ. The confidence level of adding an edge point in the model is T_{add}. In the experiments, T_{add} is 75% to demonstrate a high level of confidence that there is one more unidentified edge point in the given signal. If there is a peak in Eq. (3.19) which is higher than T_{add}, an edge points is added into the model and its location is thus determined. The estimation of the hyper-parameter ρ is given by:

$$\hat{\rho} = \frac{k_{T_{add}}}{N} \tag{3.31}$$

where $k_{T_{add}}$ is the number of the identified edge points based on the confidence threshold T_{add}. At last, Eq. (3.28) is evaluated to select the most possible model and simultaneously the estimation of parameter vector $\vec{\theta}$ could be made.

3.3. Experimental Results

3.3.1 Signal Generation and *prior* Distribution of the Models Orders

The piecewise constant signals for the simulation are generated according to the procedure in section 3.1. Piecewise signals with different value of ρ are shown in Figure3.1. With the increase of ρ, the signal will have more edge points for a given length of signal. Figure3.2 shows an example of piecewise constant signal (ρ=0.01) with different noise level. With the increase of the noise level, it is more difficult to identify the small step edges, as indicated by the circle in

Figure3.2.

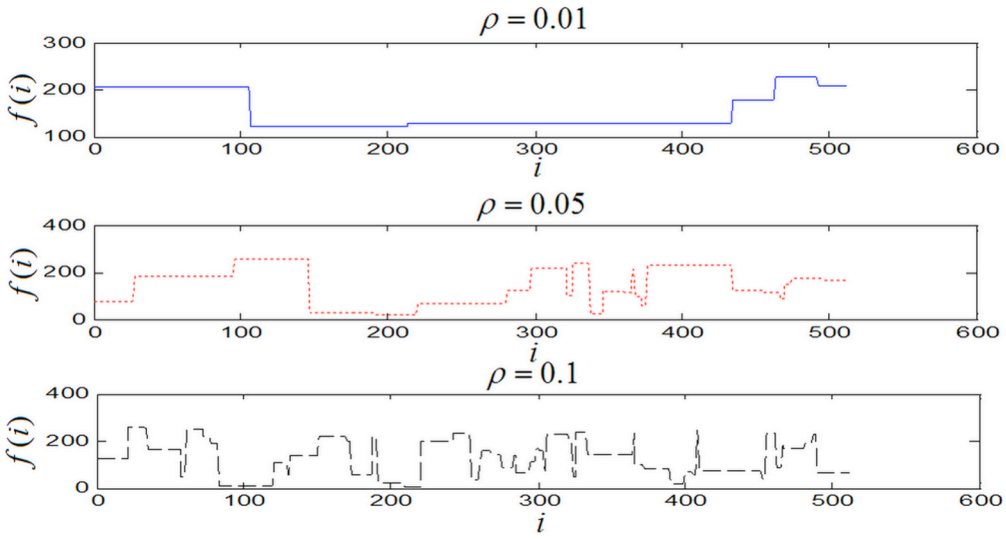


Figure 3.1 Piecewise Constant Signal with Different Values of ρ (No noise)

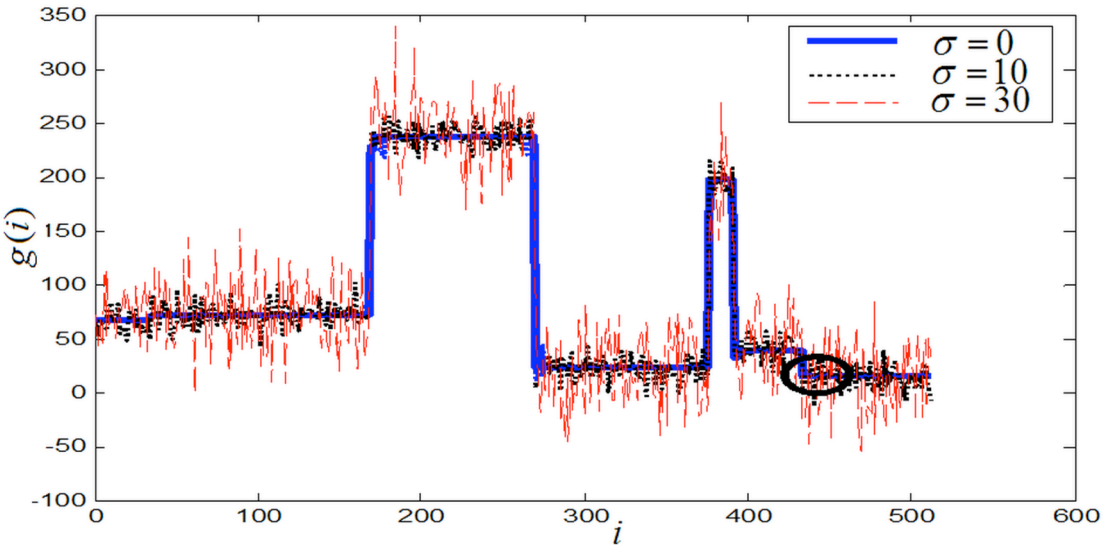


Figure 3.2 Piecewise Constant Signal with Different Noise Variance(σ =0,10,30)

Here the model order is actually the number of edge points of the signal and a pre-determined model set contains all possible numbers of edge points. Traditionally, it is assumed that all the models in a given model set are equally likely to happen. As stated in the last section, the prior distribution of the different models are not the same. The *priori* distribution of the edge point number obeys the Binomial distribution in the discrete space. It could be fitted by Gamma distribution in continuous domain. In order to know the *prior* distribution of the edge point

numbers, the simulation is carried on under an estimated $\hat{\rho}$ based on predetermined confidence threshold according to Eq. (3.31).

10,000 artificial signals with N=256 are generated and the numbers of the edges in the signal is determined according to the procedure in Section 3.1. Then Gamma distribution is used to fit the obtained data. Two explanations about why the Gamma distribution is preferred as the *à priori* of the edge numbers:

1. The support of the gamma distribution is from zero to positive infinity. The probabilities of too low value and too high value are both low. In addition, some moderate value of probabilities will have maximum value, as shown in Figure 3.3 and 3.4. It fits the purpose and assumptions.

2. The gamma distribution is compatible to the Gaussian distribution. Both of them are in exponential form. They are more flexible and smoother. This property also facilitates calculation.

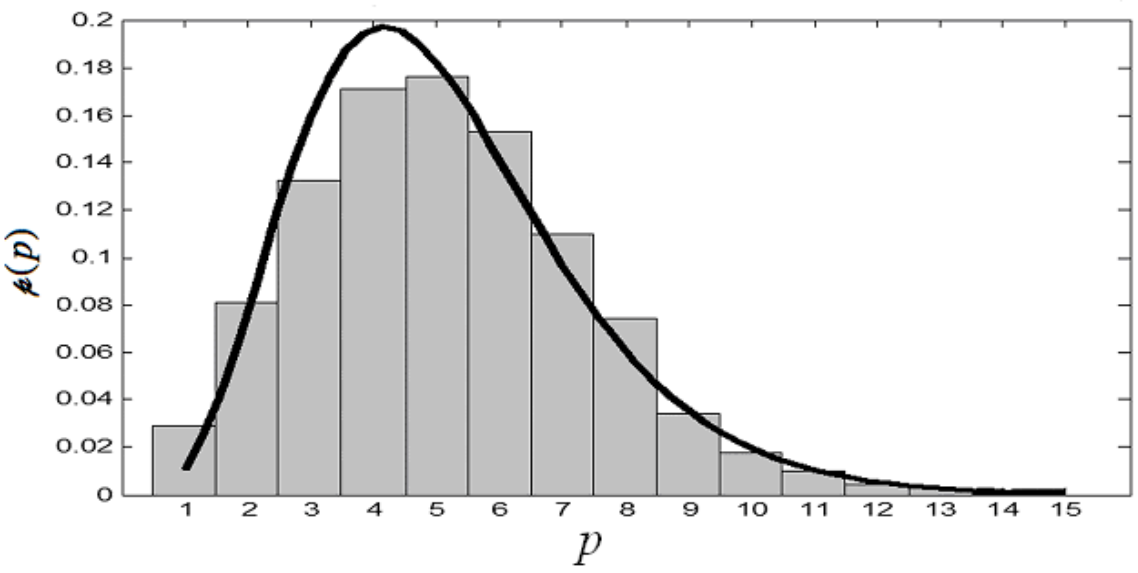


Figure 3.3 Prior Distribution of the Model Order Given ρ = 0.02

One of the simulation results is shown in Figure 3.3 with ρ=0.02. Gamma parameter estimations

are estimated by the ML method. The simulation results of ρ=0.1 is shown in Figure 3.4. The data of simulations fit the assumption well too. Based on the simulation, the parameter estimations of the Gamma *prior* distribution is given in Table 3.1. From this table, the estimate of ρ, which is $\hat{\rho}$ in the table 3.1, is accurate.

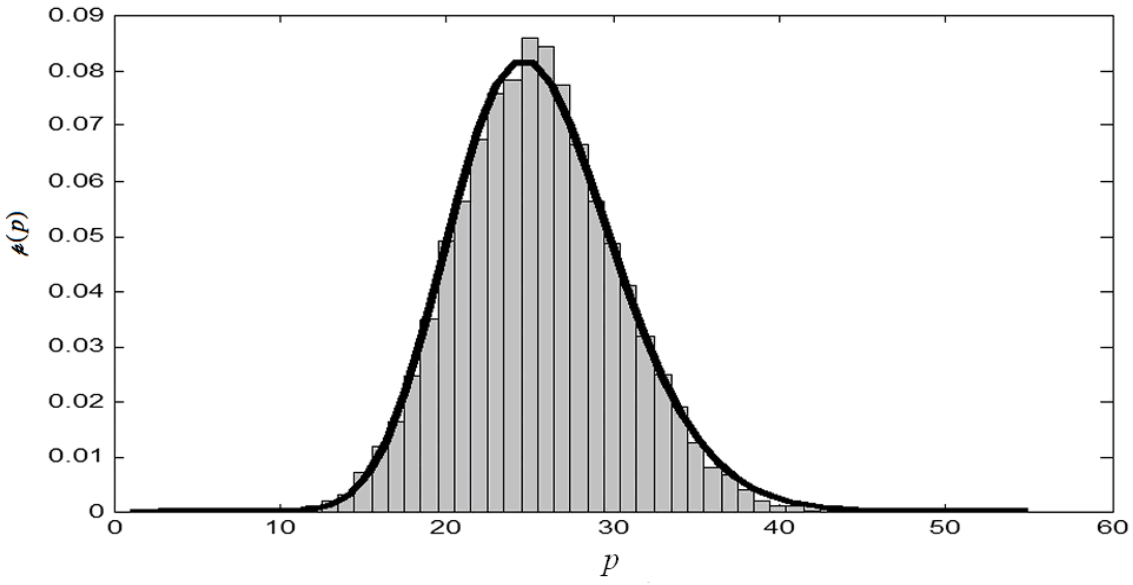


Figure 3.4 Prior Distribution of the Model Order for Given ρ = 0.1

TABLE 3-1 PARAMETER ESTIMATION OF MODEL ORDER PRIOR DISTRIBUTION FOR DIFFERENT ρ

ρ	*a*	*b*	$\hat{\mu}$	$\hat{\sigma}$	$\hat{\rho}$	$\lvert \rho - \hat{\rho} \rvert$
0.01	2.40	1.03	2.56	1.60	0.01	2.34E-06
0.02	5.24	0.98	5.08	2.24	0.02	1.42E-04
0.04	10.33	0.99	10.15	3.09	0.04	3.59E-04
0.08	22.77	0.90	20.40	4.33	0.08	3.02E-04
0.1	25.37	1.01	25.56	4.81	0.01	1.49E-04
0.2	63.25	0.81	51.04	6.40	0.20	6.26E-04
0.5	235.00	0.54	127.63	8.15	0.50	1.43E-03

3.3.2 Experiments for the Single Edge Point Model

In this experiment, the length of the signal is 256 sampling points. One of the signals and its confident limit are shown in Figure 3.5. In this figure, it is easy to identify the location of the edge and the confident limit of the edge location is near 100%. This is because the edge

magnitude is more than 200 and the noise variance is relatively small.

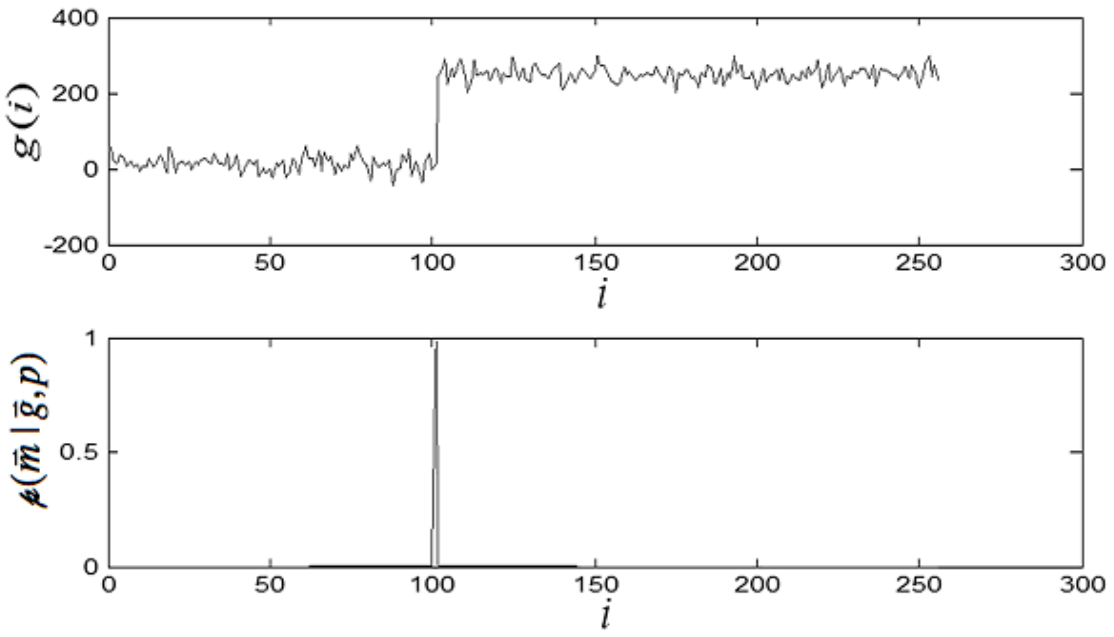


Figure 3.5 Signal with Single Edge Point and the Confidence of Edge Location (Edge=102 and σ=20)

The traditional SNR uses a single global value to describe the signal and noise, while in this work, SNR is not sufficient to describe the local details. Here the Local Signal-Noise Ratio (**LSNR**) is introduced. The definition of the LSNR for an edge is:

$$LSNR = 10 \times \log_{10}[\frac{(d_j)^2}{\sigma^2}] \quad j = \{1,2,...k\} \tag{3.32}$$

where d_j is the magnitude of the edge and σ^2 is the noise variance. According to Eq. (3.32), each edge will have a different LSNR at the same noise level according to its magnitude. This will give us more details about the signal and noise.

Do note that even the marginal likelihood in Eq. (3.19) is not related to the noise variance, while obviously, with the increase in the noise level, it will be more difficult to make a right decision on where the edge is, which is shown in Figure 3.6. The edge magnitude and the noise level will both affect the decision, in this connection, to recognize an edge, and its magnitude should be obviously distinctive amongst the noise level.

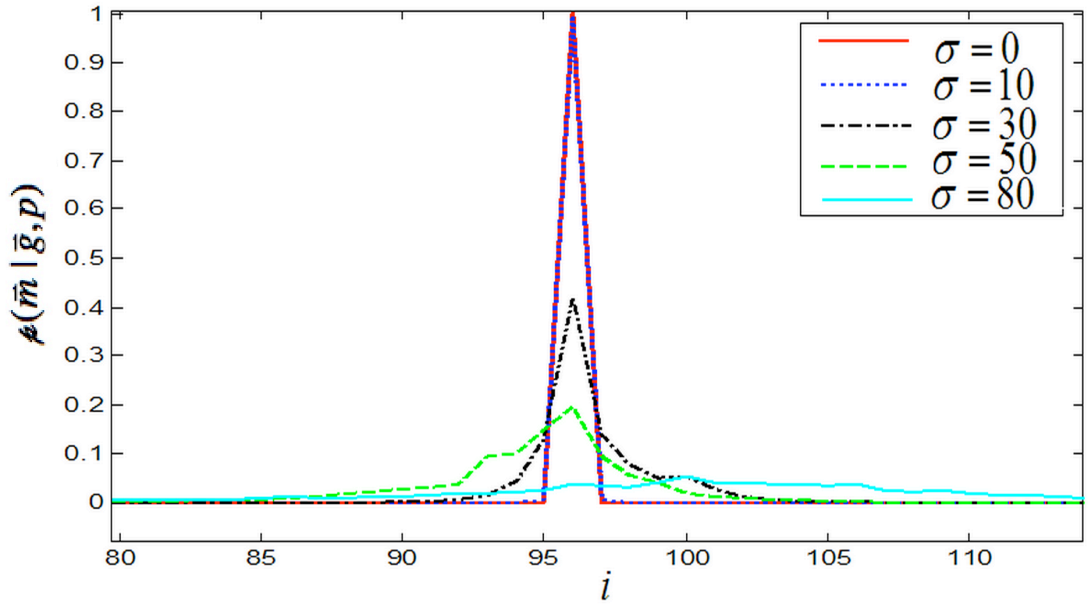


Figure 3.6 Confidence of Edge Location at Different Noise Level (Edge=130)

This leads us to another important issue; how large should the edge magnitude be such that the method can successfully identify the edge and its location? Or, when would the proposed approach fail to identify the edge with the increase in the noise level?

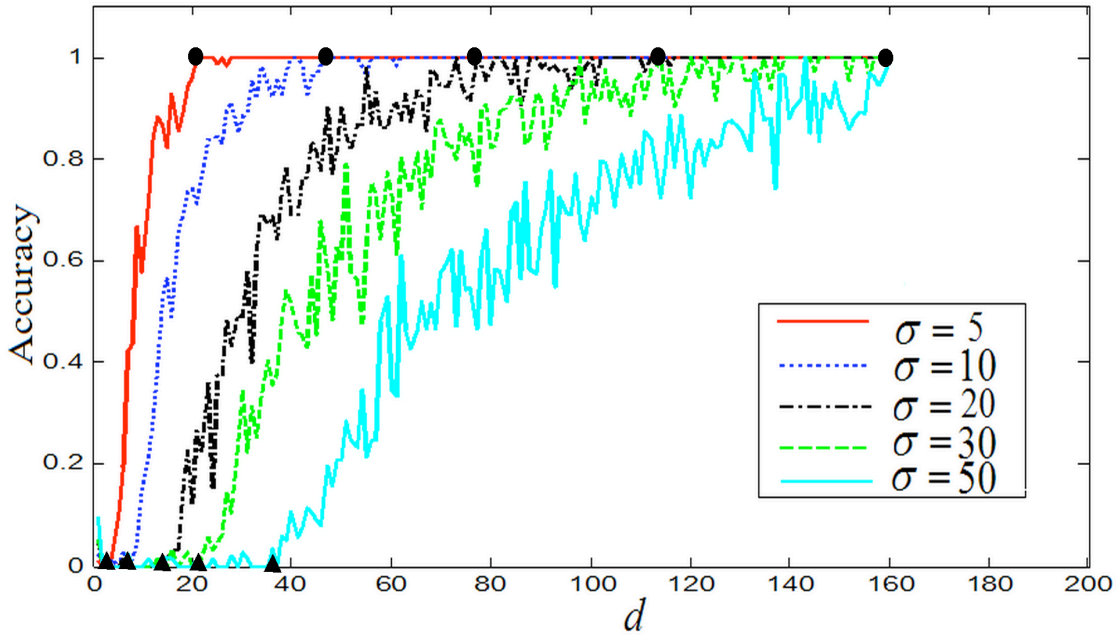


Figure 3.7 Accuracy and the Magnitude of the Edges under Difference Noise Levels

The LSNR will provide an indication on whether the proposed method is sensitive to noise. Figure 3.7 shows the results of the simulation under 5 different noise levels. This experiment is carried out in the following procedure:

1. Select a noise level;
2. Generate 10000 different signals with single edge point and record the edge magnitudes;
3. Based on the single edge point inference, find the location of the edge;
4. Record how many times a right decision are made on the edge location at different edge magnitude;
5. Go back to 1, select another noise level.
6. The procedure goes on until all the intended noise levels have been exhausted.

From Figure 3.7, at a given noise level, the accuracy of identification will increase with the increase in the edge magnitudes. It is also obvious that when the edge magnitude is near zero, the accuracy is also zero. This is the reason for which the edge should have a relatively "obvious" magnitude compared with the noise level so that an accurate decision can be made. Two questions could be answered by looking at Figure 3.7.

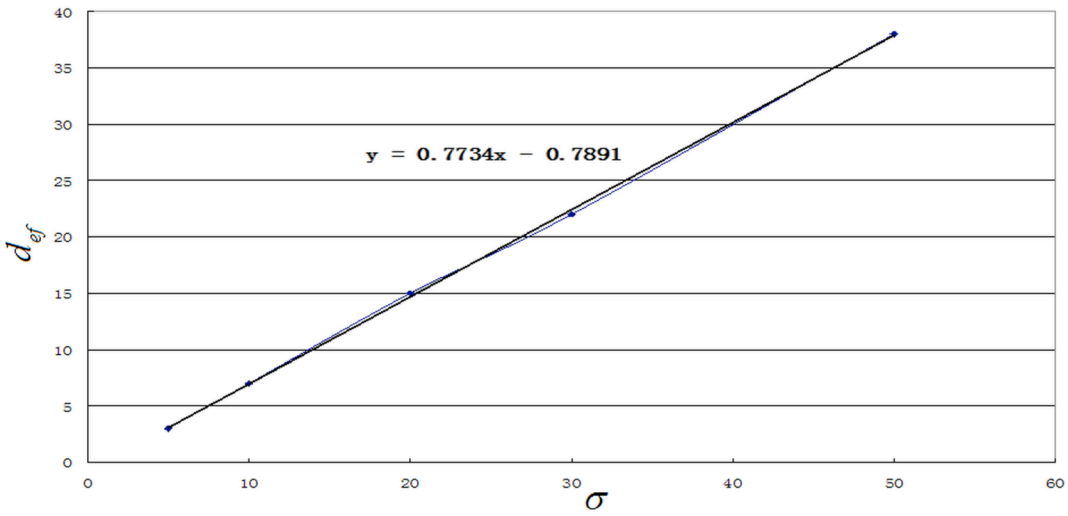


Figure 3.8 Minimum Magnitude of the Edges for Identification under Difference Noise Levels

Firstly, when will the Bayesian method start to be effective? As we know, to reach a decision on the correct identification of an edge, the edge magnitudes should not be too small. But, how "small" should this magnitude be? The points indicated by *black triangle dots* shown in Figure 3.7 is selected, which means when the proposed approach starts to be effective to the given signal. They are shown in Figure 3.8, from which the relationship between the minimum edge magnitude and the noise level is shown to be approximately linear. The slope in Figure 3.8 is a good indication of the obvious edge at different noise levels. According to the simulation, the edge magnitude in gray level must be larger than:

$$d_{ef} > 0.77\sigma + 0.79 \tag{3.33}$$

Eq. (3.33) gives the value of the edge magnitude, at a given noise level, beyond which the proposed formulation can correctly detect an edge point. If LSNR is used, then:

$$LSNR_{ef} = 10 \times \log_{10}(\frac{1}{0.77^2}) \approx 2.2db \tag{3.34}$$

Eq. (3.34) indicates that if the LSNR is smaller than 2.2 decibels (***db***), the proposed approach cannot detect the edge point.

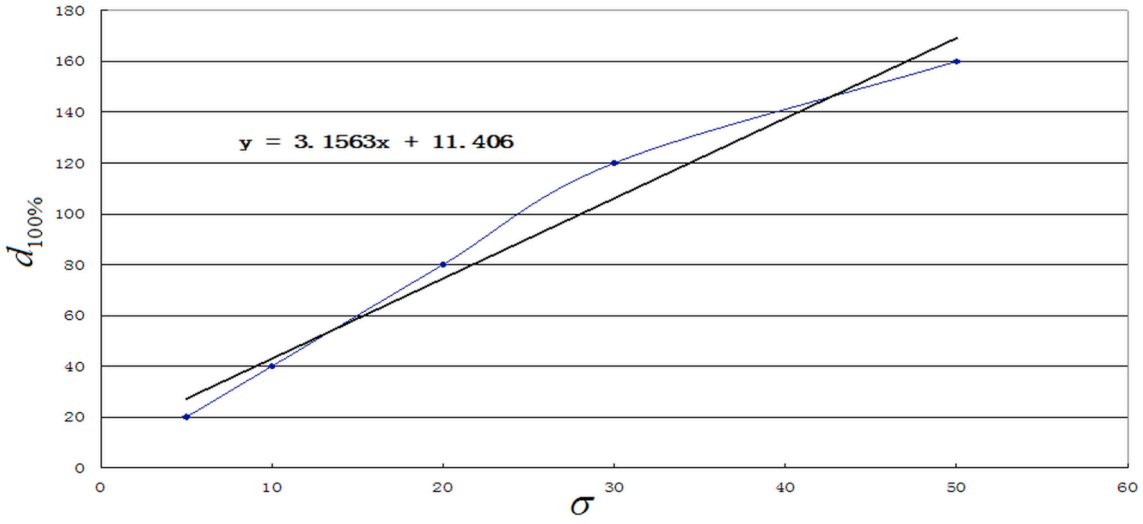


Figure 3.9 Magnitude of the Edges Identification with 100% Confidence at Difference Noise Levels

Secondly, when will the Bayesian method perfectly (at a confident level of 100%) identify the edge locations? Or, how large should the edge magnitude be such that the proposed method could recognize an edge point with high confidence? The first points with 100% confidence in Figure 3.7 at different noise levels are selected, which are indicated by *black round dot*. The results are shown in Figure 3.9, in which they are plotted by the edge magnitude *vs.* the noise level represented by σ. The relationship between identified edge with 100% confidence and the noise variance is also about linear. The slope in Figure 3.9 is a good estimation of "clear" edge at different noise levels. According to the simulation, the edge magnitude must be larger than:

$$d_{100\%} > 3.16\sigma + 11.4 \tag{3.35}$$

The results can be explained as the LSNR limitation of the proposed method. According to Eq. (3.35), the smallest LSNR at which the proposed method could identify the single edge point with about 100% confidence in the given single edge model (used in Figure 3.7) is:

$$LSNR_{100\%} > 10 \times \log_{10}(\frac{3.16}{1})^2 \approx 10db \tag{3.36}$$

Hence, based on the results of simulation on signal with a single edge point, LSNR must be at least 10*db*, or the edge magnitude must be at least 3.16 times larger than the noise variance σ, such that the single edge location can be identified with 100% confidence. On the other hand, if the edge magnitude is smaller than 0.77 times of the noise variance σ, identification of the edge points would be impossible.

3.3.3. Experiments for the Multi Edge Point Model

As stated in the previous sections, the evaluation of the posterior marginal likelihood of the Bayesian inference is computational intensive for cases with high dimensionality. Hence, direct calculation is generally not infeasible. For the example shown in Figure 3.10, p=5 and the length of the signal is 512 sampling points. There are five step edge points in the given signal with a noise variance σ = 30.

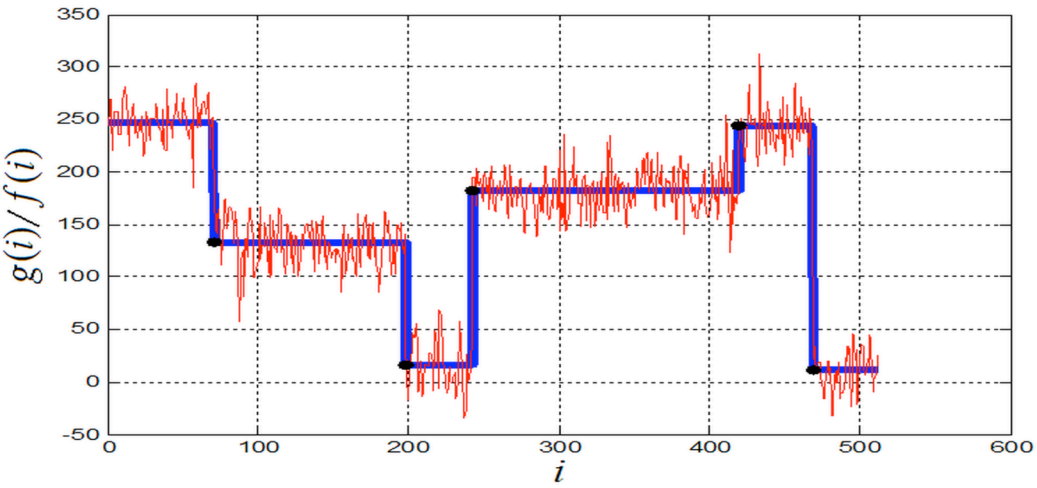


Figure 3.10 Piecewise Constant signal and the Edge Points at (71,199,243,420,469)

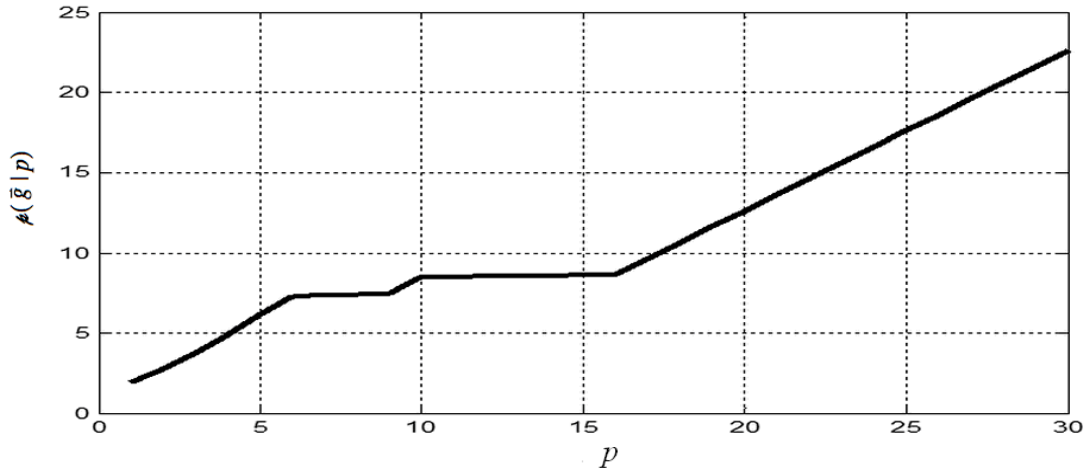


Figure 3.11 Likelihood Function of the Signal Given Difference Model Order

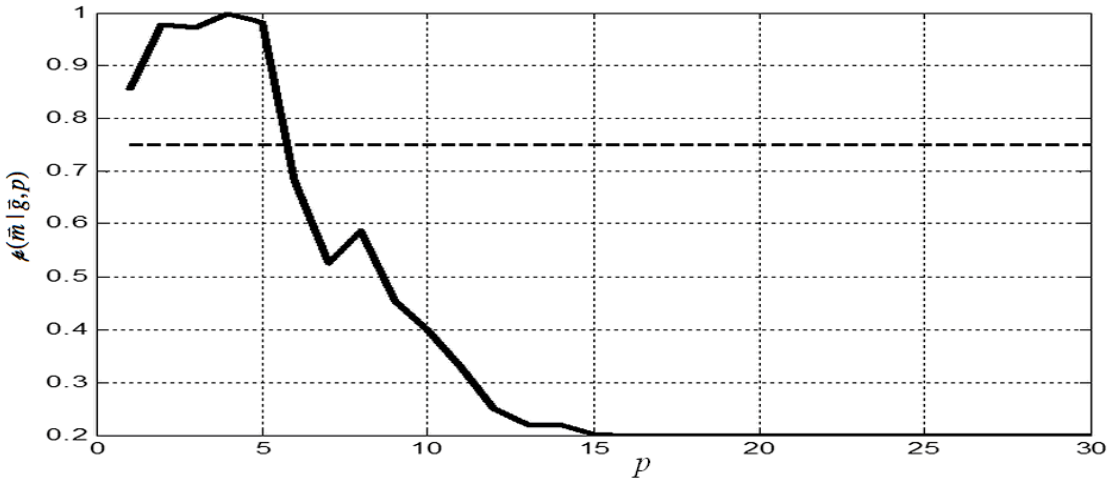


Figure 3.12 Confidence of Increasing Model Order and the Threshold of 75%

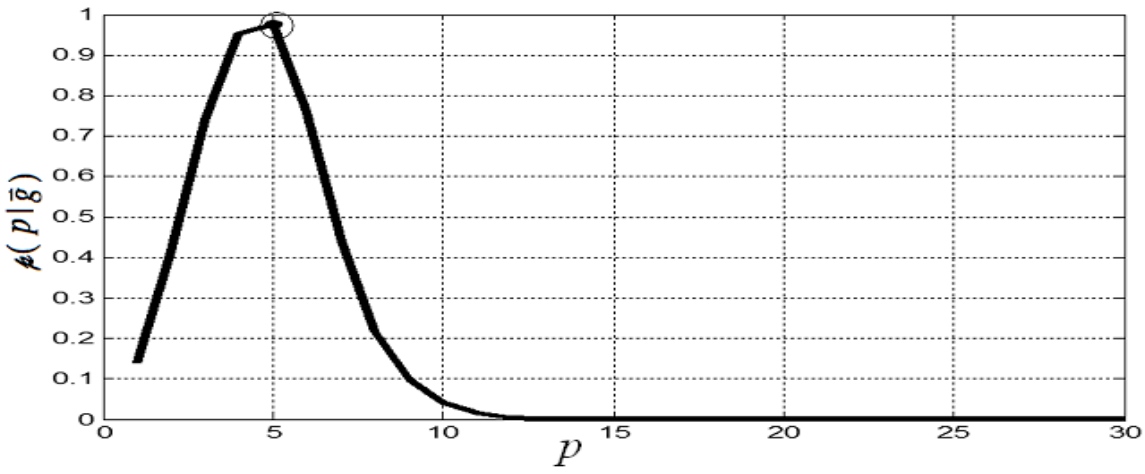


Figure 3.13 Posterior Evidence of the Given Signal with Corresponding Identified Edges at (71,199,243,420,469)

The likelihood function given model orders is shown in Figure 3.11. Figure 3.12 gives the confidence of increasing the model order at different model order p. The threshold confidence of adding an edge into the model is 75%, which is indicated by a *horizontal dash line* in Figure 3.12. The Likelihood with different model orders in Figure 3.11 will increase with the model order as analyzed previously. If the Likelihood function is used as the cost function, it will easily cause the over-fitting.

The *round point* in Figure 3.13 indicates the most likely model order and the numbers at the bottom, (71,199,243,420,469), show the corresponding estimated locations of edge points. As mentioned above, the total likelihood in Eq. (3.28), which is the first term in the numerator, will continuously increase with the increase in the model order, while this increase will be penalized by prior distribution of the model according to the estimated model prior based on the confidence threshold, which is the second term in Eq. (3.28).

In order to know the accuracy of the proposed method for the multi-edge model, a benchmark is provided according to the simulation results. Two definitions are required to evaluate the results:

- Missing Edge: there is an edge point in the signal, while the proposed approach fails to identify it.

- Wrong Edge: there is no such edge point in the signal, while the proposed approach wrongly identifies it as an edge point.

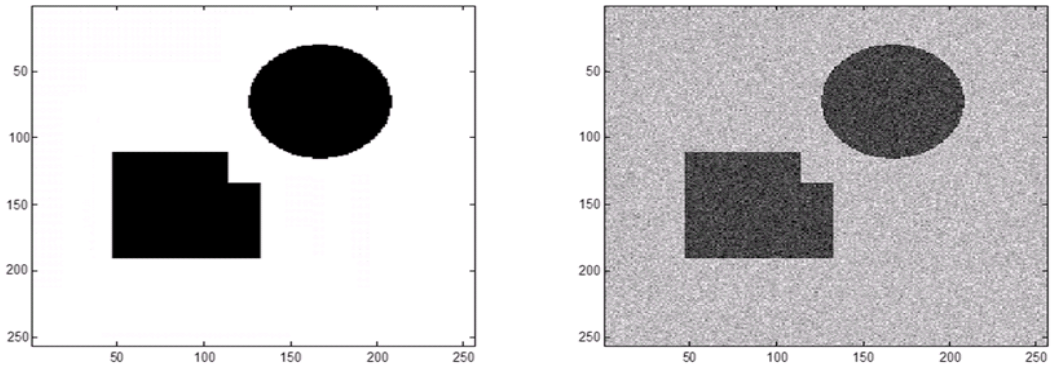
A simulation is carried on for 5000 generated signals of 512 points, at six different noise levels. Table 3-2 shows the results of the simulation.

According to Table 3-2, the proposed method is robust and not-sensitive to noise. One of the reasons for the missing edge points is that the magnitude of edge is not regulated. It is possible that some conjoint pieces have very similar gray level, or the edge magnitude is not obvious, *e.g.* $d < 0.77\sigma$. According to the experimental result on the LSNR, if the magnitude of the edge is too small, it is possible that the proposed method might not identify some of those small edges. However, according to the result in Table 3-2, an excellent accuracy has been achieved.

TABLE 3-2 RESULTS OF THE ACCURACY FOR THE GIVEN SIGNAL AT DIFFERENT NOISE LEVELS

Noise Level σ	5	10	20	30	40	50
Missing Edge (%)	1.50	1.50	1.60	2.10	2.18	5.00
Wrong Edge (%)	1.90	1.70	2.60	3.00	2.67	3.98
Accuracy (%)	98.50	98.50	98.40	97.90	97.82	95.00

3.3.4. Application of Edge Detection for 2-D Image

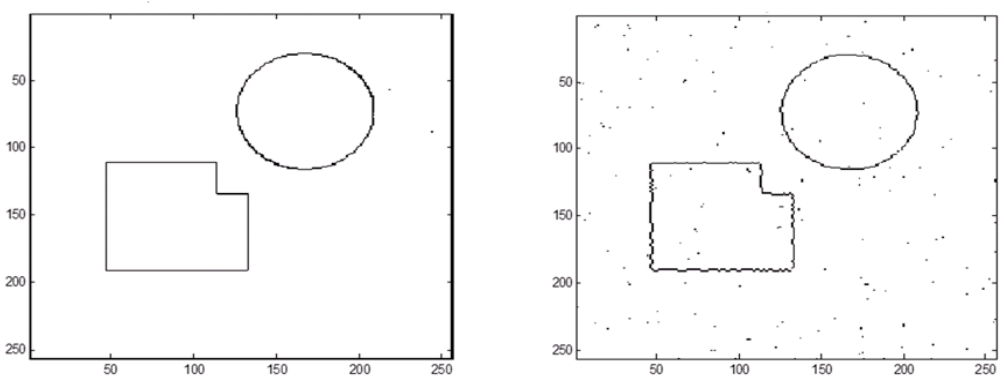


(a). Original Image without Noise (b). Degraded Image with Noise $\sigma = 40$

Figure 3.14 2-D Image for the Edge Detect

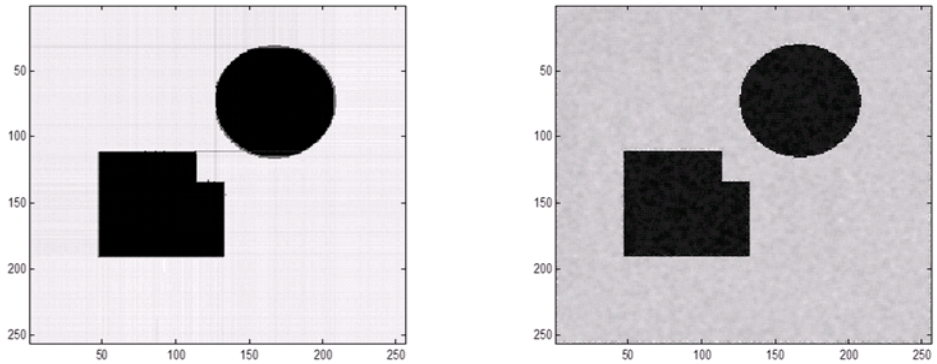
The presented algorithm is applied to the 2-D image edge detection by line scanning scheme. Most of the object recognition algorithms are based on the edge, boundary or shape. If the LSNR is low, most of the traditional edge detection method will encounter problems with salt and pepper noise spots in the image in question.

The test image is shown in Figure 3.14(a). The degraded and noisy observation is shown in Figure 3.14(b) with the noise variance σ equals to 40. The edge magnitude is 255. The LSNR is as low as 16 *db*. The detected edge by the proposed algorithm and the Sobel edge detector are both shown in Figure 3.15(a) and (b), respectively. The proposed method is more robust compared with Sobel edge detection algorithm. In Figure 3.15(b), there are more salt and peppers noise.



(a). Edge Detected by Proposed Method (b). Edge Detected by Sobel Detector

Figure 3.15 Comparison of Edge Detect by Proposed Method and Sobel Detector



(a) Restored Image by Proposed Method (b) Restored Image by Wiener Filter

Figure 3.16 Restored Images by Proposed Method and Adaptive Wiener Filter

The restored images are shown in Figure 3.16. Figure 3.16(a) is the result of the proposed method and Figure 3.16(b) is obtained through the use of the adaptive Wiener filter. In Figure 3.16(a), there are some line artifacts. They are caused by the line scanning scheme, which ignores the vertical/horizontal correlations of the signal. It is not a surprise that our results are better, since the proper prior knowledge is applied and the noise is already included in the model. According to the results, the proposed approach works well and is robust.

3.4. Summary

A novel Bayesian model select method for the piecewise constant signal is presented to identify the discontinuities in the signals based on the posterior evidence. The model select approach is based on the posterior distribution function. Avoiding direct evaluation of the posterior distribution in the state space of high dimensionalities, the model order is increased one by one according to the confidence threshold. By evaluating the so-called posterior evidence, the most likely model order is thus selected for a given signal. The proposed method is different from the algorithm in [42], in which Gibbs sampler could jump between the parameter subspaces of different dimensionalities. The proposed algorithm is robust as shown by the results of the simulations and experiments. The concept of LSNR is proposed in order to describe the local details of the signal and the noise. The presented method works well at low LSNR. The error rate is not equal to zero when LSNR is high, because the edge magnitude has no constraints when the piecewise constant signal are generated. Some edges possibly have very small magnitudes.

A fast computational scheme of the normalization constant is proposed. It is an approximation of the real value of the constant, while it is very efficient. The simulation also showed that this formulation gives accurate results. Although the inference is based on the assumption that signal is piecewise constant, the proposed formulation should be easily generalized to suit other more general and common signals.

Chapter 4

Sparse Probabilistic Linear Model and Relevance Vector Machine for Piecewise Linear 1-D Signal

In Chapter 3, the identification of discontinuities for 1-D piecewise constant signals is presented. The identification is based on posterior evidence, instead of the usual methods, which are based on assumption that all the numbers of edges are equally possible. It is therefore more robust and computationally efficient. In this chapter, theoretical formulation will be developed to deal with 1-D piecewise linear signals, without the restriction on them being constant.

4.1. Relevance Vector Machine

In the past decades, kernel methods inspired many innovative ideas in regression and classification. In this work, the kernel method is used to reconstruct the 1-D piecewise linear signal and identify the discontinuities based on the value of hyper-parameter as well. In this section, the kernel formulation and the kernel function is briefly reviewed. For a given noisy function with N uniform samples at points $\{i\}_{i=1}^{N}$ and targets $\{g(i)\}_{i=1}^{N}$, we wish to acquire a model/function $f(i)$ and estimate the noise component simultaneously. For this problem, the kernel method yields the following solution:

$$f(i;\vec{w}) = \sum_{i=1}^{N} \phi(i',i) w_i \tag{4.1}$$

where $\phi(i',i)$ is a kernel function, effectively defining one basis function for each point in the given function $f(i)$; i and i' are different sampling points in f; w_i is called the weight. For every basis function, there is a corresponding weight. The function f then is represented by a linear

combination of the multiplication of the basis functions and the corresponding weights. This is actually the GLM discussed in Chapter 2.

Selecting a proper kernel function is important in all kernel methods. Normally, the kernel function should satisfy Mercer's condition. Many kernel functions satisfy this condition and they are widely used in practice. Following the piecewise linear assumption stated in Chapter 3, the scaled linear kernel should be used. It is defined in Eq. (4.2):

$$\phi(i',i) = \sqrt{\eta}\sqrt{(i'-i)^2} \quad \{1,2,3...N\} \tag{4.2}$$

where η is the scale parameter, i is the sampling points and N is the length of the signal.

The question of estimating $f(i)$ in Eq. (4.1) could be changed to the Wolfe Dual problem according to the KKT condition [97]. Obviously, the N weights are not equally important. In an extreme situation, the given function $g(i)$ can be simulated without any error by using all the N basis functions and N weights. Of course, different $\vec{w}$ for the same given signal with different noise realization will be found, however it is highly undesired. One of the solutions could be given by using the Least Square Error (LSE) estimation:

$$\hat{\vec{w}} = (K^T K)^{-1} K\vec{g} \tag{4.3}$$

where K is the kernel matrix in Eq. (2.9). And the estimation of the original signal will be given by:

$$\hat{\vec{f}} = K\hat{\vec{w}} \tag{4.4}$$

The likelihood on its own does not limit the number of parameters in a model used to fit the data. In general, the more complex a model, the better is the fit to the data. This is called over-fitting as mentioned in the last Chapter. There is then a Penalized Least Square Estimation (PLSE) presented to overcome this problem. Since the PLSE adds a regularization constant before

calculation the inverse matrix in Eq. (4.3), say ($K^TK+\varepsilon$), where ε is a small positive constant and keeps the same for all basis function. It is obvious that not all the basis functions are equally important based on the given base function to describe the data. Some of them could be eliminated from the model. In statistics, the idea of Auto Relevance Determination (ARD) is not very new. But until recently, Michaela Tipping integrates the ARD in the kernel inference and it was then called the Relevant Vector Machine[105][106][107]. It can identify the "important basis" to estimate the given noisy discontinuous functions.

4.1.1. Likelihood Function and *à priori* Probability

The task of the discontinuous signal/function regression is to estimate the real signal/function *f* and preserve the discontinuities based on the given noisy observation *g*. Since how many edges within the given signal and where they are unknown, any point in the signal could possibly be a discontinuity. Eq. (4.1) describes a function *f*, then the likelihood function for the observation g is given by:

$$\mathcal{P}(\vec{g} \mid \vec{w}, \sigma^2) = (2\pi\sigma^2)^{-N/2} \exp(-\frac{1}{2\sigma^2} \| \vec{g} - K\vec{w} \|) \tag{4.5}$$

where $\vec{g} = (g(1), g(2), ...g(N))^T$ is an $N \times 1$ vector, $\vec{w} = (w_1, ...w_N)^T$. K is an $N \times N$ kernel matrix and every element is determined by $\phi(i', i)$ according to Eq. (4.2). The Maximum Likelihood (ML) estimate of the parameters actually will yield the LSE solution and often causes severe over-fitting as discussed above. The weight coefficients are assumed to be an explicit prior probability distribution in this work. By evaluating the hyper parameters, which control the prior probability distribution, kernels, which are more important and relevant for the given data, are selected.

The popular choice for the distribution over the weight coefficient is the independent zero mean Gaussian distribution [102] [106]:

$$p(\vec{w}\,|\,\vec{\alpha}) = \prod_{i=1}^{N} N(w_i\,|\,0,\alpha_i^{-1}) \tag{4.6}$$

$\vec{\alpha} = \{\alpha_1, \alpha_2, ...\alpha_N\}$ is a vector of N hyper-parameters. Importantly, there is an individual hyper-parameter associated independently with every weight. These hyper-parameters will control the complexity of the model.

The hyperpriors over $\vec{\alpha}$ need to be defined, as well as the variance of the noise. Berger suggested that these parameters are scale parameters, which obey the Gamma distributions in[106][107]. The hyperprior should also be non-informative, since no knowledge is given about where the discontinuous points are and which kernels are important. $\vec{\alpha}^{-1}$ is initialized such that the $\vec{w}$ will have a wide range, or large variance according to Eq. (4.6). Then one of the straightforward prior for the hyper prior is that:

$$p(\alpha_i) = \frac{1}{N} \tag{4.7}$$

More generally, a Gamma distribution can be used for the hyper-parameters, which is given by [106]:

$$p(\vec{\alpha}) = \prod_{i=0}^{N} \Gamma(\alpha_i\,|\,a,b) \tag{4.8}$$

$$p(\beta) = \Gamma(\beta\,|\,c,d) \tag{4.9}$$

where $\beta = \sigma^{-2}$. In order to use the Gamma distribution to approximate a uniform distribution, small values for the parameters in the Gamma distribution should be used, say $a = b = c = d = 10^{-6}$.

There are two reasons that the Gamma distribution is preferred:

a). The support of the Gamma distribution is from zero to positive infinity. This is convenient to simulate events involving only positive values.

b). The Gamma distribution is compatible to the Gaussian distribution. Both of them are in exponential form.

The prior distribution of the hyper-parameters $\vec{\alpha}$ is shown in Figure 4.1.

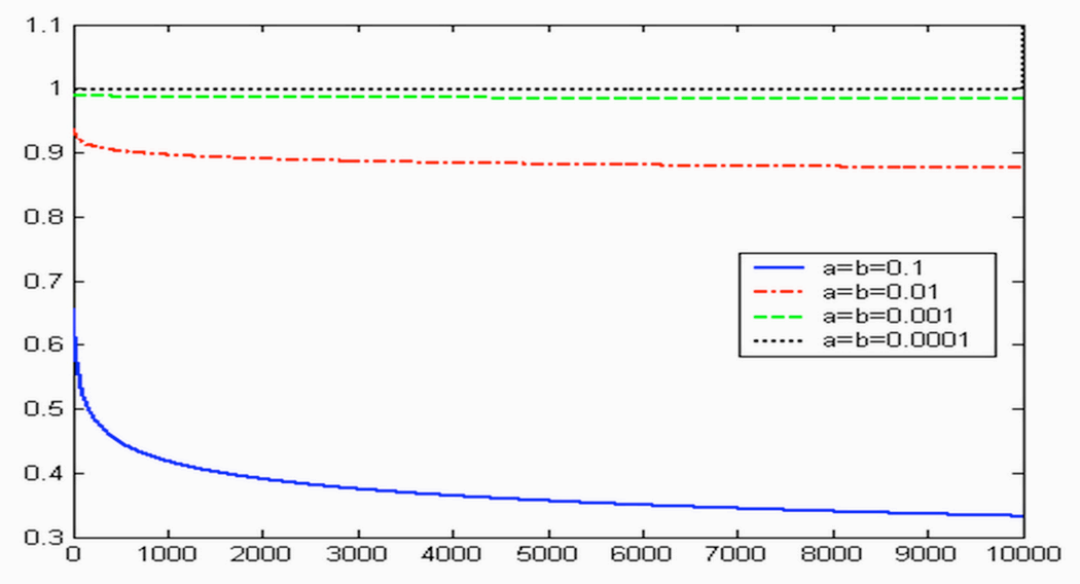


Figure 4.1 Distribution of α with Different Parameters

Every hyper-parameter would typically control one corresponding weight, which associates with each point in the given signal. If it is evident that the data supports such a prior distribution hypothesis, using a broad prior over the hyper-parameters allows the posterior probability mass to concentrate at very large values for some of these hyper-parameters. In consequence, the posterior probability of the associated weights will be concentrated at zero, thus effectively "switching off"' the corresponding inputs, and so deeming them to be "irrelevant". The remaining weights are the so-called "Relevant Vector"[107]. By learning from the noise data, those points, which are more important, can be found and used to reconstructed the original signal. Actually, those points are the discontinuous points.

4.1.2. Posterior Distribution and Evidence

After having defined the prior, Bayesian inference proceeds by computing. In order to estimate the original signal $\vec{f}$, the weights $\vec{w}$ should be determined. According to Bayesian rule, the

posterior distribution of weights is:

$$\wp(\vec{w},\vec{\alpha},\sigma^2 \mid \vec{g}) = \frac{\mathcal{P}(\vec{g} \mid \vec{w},\vec{\alpha},\sigma^2)\wp(\vec{w},\vec{\alpha},\sigma^2)}{\mathcal{P}(\vec{g})} \tag{4.10}$$

where $\vec{w}$ is the weights, $\vec{\alpha}$ is the hyper-parameter which controls the distribution of the weight which is defined in Eq. (4.6) and Eq. (4.8), σ^2 is the variance of the noise and $\vec{g}$ is the observed degraded signal. However, the posterior $\wp(\vec{w},\vec{\alpha},\sigma^2 \mid \vec{g})$ in Eq. (4.10) cannot be computed directly since the normalizing integral on the right-hand-side cannot be determined. Fortunately, Eq. (4.10) could be decomposed as [106]:

$$\wp(\vec{w},\vec{\alpha},\sigma^2 \mid \vec{g}) = \wp(\vec{w} \mid \vec{g},\vec{\alpha},\sigma^2)\wp(\vec{\alpha},\sigma^2 \mid \vec{g}) \tag{4.11}$$

The first term in the right hand side of Eq. (4.11) is the posterior distribution of weights $\vec{w}$, which can be done analytically. Substituting Eq. (4.5) and (4.6) into Eq. (4.12):

$$\wp(\vec{w} \mid \vec{g},\vec{\alpha},\sigma^2) = \frac{\mathcal{P}(\vec{g} \mid \vec{w},\sigma^2)\wp(\vec{w} \mid \vec{\alpha})}{\mathcal{P}(\vec{g} \mid \vec{\alpha},\sigma^2)}$$

$$= (2\pi)^{-(N+1)/2}\left|\Sigma\right|^{-1/2}\exp\{-\frac{1}{2}(w-\mu)^T\Sigma^{-1}(w-\mu)\} \tag{4.12}$$

where the posterior covariance and mean are [106]:

$$\Sigma = (\sigma^{-2}K^TK + A)^{-1} \tag{4.13}$$

$$\mu = \sigma^{-2}\Sigma K^T\vec{g} \tag{4.14}$$

where $\boldsymbol{A}$=*diag*($\alpha_1,\alpha_2,...\alpha_N$). The penalized parameter is determined by $\boldsymbol{A}$, however, now different w_i may have different regularization, which is different from the penalized LSE estimation.

Then the problem of learning becomes to find a hyper-parameter posterior to maximize $\wp(\vec{\alpha},\sigma^2 \mid \vec{g}) \propto \mathcal{P}(\vec{g} \mid \vec{\alpha},\sigma^2)\wp(\vec{\alpha})\wp(\sigma^2)$. As hyperprior is uniformly distributed, the only need is to

maximize the term $\mathcal{P}(\vec{g}\,|\,\vec{\alpha},\sigma^2)$, which is computable and given by [106]:

$$\mathcal{P}(\vec{g}\,|\,\vec{\alpha},\sigma^2)=\int \mathcal{P}(\vec{g}\,|\,\vec{w},\sigma^2)\,p(\vec{w}\,|\,\vec{\alpha})\,d\vec{w}$$

$$=(2\pi)^{-N/2}\left|\sigma^2 I+KA^{-1}K^T\right|^{-1/2}\exp\{-\frac{1}{2}\vec{g}^T(\sigma^2 I+KA^{-1}K)^{-1}\vec{g}\} \quad (4.15)$$

This quantity is known as the marginal likelihood, also referred to as the "evidence of hyper-parameters" by Mackay [102].

The value of α_i and σ^2 which can maximize Eq. (4.15) is wanted, but it cannot be obtained in closed form. The procedure follows Type-II maximum likelihood method. It follows that [105]:

$$L=\log\mathcal{P}(\vec{g}\,|\,\log\vec{\alpha},\log\beta)+\sum_{i=0}^{N}\log p(\log\alpha_i)+\log p(\log\beta) \quad (4.16)$$

Obtaining the first derivatives of this value with respect to α and σ^2 and set them to zero, the re-estimated values are then given by [106][107]:

$$\alpha_i^{new}=\frac{1-\alpha_i\Sigma_{ii}}{\mu_i^2} \quad (4.17)$$

$$(\sigma^2)^{new}=\frac{\left\|\vec{g}-K\mu\right\|}{N-\Sigma_i(1-\alpha_i\Sigma_{ii})} \quad (4.18)$$

The learning algorithm thus proceeds by repeated application of (4.17) and (4.18), concurrently with the updating of the posterior statistics Σ and μ from (4.13) and (4.14), until the maximum of the updates of all the α_i is smaller than some given value, say 10^{-6}. Since $\mathcal{P}(\vec{g}\,|\,\alpha,\sigma^2)$ is in the form of Gaussian distribution, this results will guarantee a unique maximum for $\vec{g}$ [106], which is desirable for most of the tasks. With such a scheme, the ill-conditioned problem becomes well-posed, but the Hessian matrix in Eq. (4.13) may not exist if the hyper-parameter is not properly selected. The selection of the hyper-parameter η is an important issue here. The optimal value of

η will achieve the compromise between the sharpening and smoothing. In order to find the best value of η, the prior knowledge of the noise should be used. Some of the existing methods can provide the estimation of noise variance, such as [28]~[31]. Those methods can be applied to estimate the noise variance, and then to obtain $\hat{\sigma}^2$. In order to achieve a compromise between the sharpening and smoothing, $R = |\hat{\sigma}^2 - \sigma^2|$ is applied as the cost function to find the best η, where $\hat{\sigma}^2$ is the estimate of the noise variance and σ^2 is the noise variance in Eq. (4.10). Further discussion on the optimization of η in 2D applications will be presented in Chapter 5.

4.2. Occam's Razor and Automatic Relevance Determination

There is a general philosophical principle known as **Occam's razor** [103] ~ [104] for model selection, which is highly influential when applied to various scientific theories. Occam's razor is a logical principle attributed to the mediaeval philosopher William Occam, which is originally stated in Latin in 14th century:

> ***Pluralitas non est ponenda sine necessitate.***
>
> ***Entities should not be multiplied unnecessarily.***
>
> *-"Occam's razor" principle attributed by W. Occam.*

This principle states that one should not make more assumptions than the minimum needed. This principle is often called the principle of parsimony. It underlies all scientific modeling and theory building. It admonishes us to choose from a set of otherwise equivalent models of a given phenomenon the simplest one. In any given model, Occam's razor helps us to "shave off" those concepts, variables or constructs that are not really needed to explain the phenomenon. By doing that, developing the model will become much easier, and there is less chance of introducing inconsistencies, ambiguities and redundancies.

In light of Occam's razor, unnecessarily complex models should not be preferred to simpler ones.

When there are two competing models, which make exactly the same predictions, the one that is simpler is the better. The Bayesian kernel inference in the last section can achieve a sparse solution based on the discontinuous points in the function. It is similar to a situation when a person observes a piecewise linear signal, those significant points to represent the signal will be automatically determined.

The Occam's razor is significant to achieve the sound solution. If several models are compatible with a set of observations/data, Occam's razor advises us to select the least complex explanation. This can be simply explained that the probability of a too complex model, which has more free parameters, is generally lower than the probability of a simple model.

The given problem in this Chapter is actually a data modeling problem. Two levels of inference can often be distinguished in the task of data modeling as stated in Section 3.2.2. At the first level of inference, a particular model is assumed to be true, and the model can be fitted to the data. Typically a model includes some free parameters; fitting the model to the data involves determining what values those parameters should probably take based on the given data. The results of this inference are often summarized by the most probable parameter values. This analysis is repeated for each model. The second level of inference is the task of model comparison or model selection. Here the models will be compared in light of the data, and assign some sort of preference or ranking to the alternatives. Bayesian methods, including RVM, are able to consistently and quantitatively solve both the inference tasks. The explanation is below:

- Model fitting: By looking at Eq. (4.12), it is actually posterior = (likelihood $\times$ prior) /evidence. For a given hyper-parameters, find the best value of the hyper-parameters for the given problem.
- Model comparison: At the second level of inference, find out which model is most plausible for the given data. The posterior probability of each model is given as $\mathcal{P}(\alpha,\sigma^2|\bar{g}) \propto$

$\mathcal{P}(\vec{g}|\alpha,\sigma^2)\mathcal{P}(\alpha)\mathcal{P}(\sigma^2)$. Notice that here $\mathcal{P}(\sigma^2)$ and $\mathcal{P}(\alpha)$ are the prior distributions of the hyper-parameters, which are assumed to be of uniform distribution, so their effects can be ignored. The term $\mathcal{P}(\vec{g}|\alpha,\sigma^2)$ is important, which is evidence in Eq. (4.12).

The evidence is the normalizing constant given by:

$$\mathcal{P}(\vec{g}|\alpha,\sigma^2) = \int \mathcal{P}(\vec{g}|\vec{w},\sigma^2)\mathcal{P}(\vec{w}|\alpha)d\vec{w} \tag{4.19}$$

The posterior $\mathcal{P}(w|\vec{g},\alpha,\sigma^2) \propto \mathcal{P}(\vec{g}|w,\sigma^2)\mathcal{P}(w|\alpha)$ will have a strong peak at the most probable value of w_{MP}. The evidence is the accuracy of the model for the given data multiplied by the Occam factor, which is to describe the complexity of the model. The maximization of the evidence is actually to achieve a compromise between the accuracy and the complexity of the model. If the model is too complex, *i.e*. including all kernels, it will be penalized by the Occam factor. The ARD introduces the hyper-parameter to control the complexity of the model such that the Bayesian kernel can make most of the α_i tend to infinite value and then "switch off" those corresponding kernels.

The prior knowledge of the function is integrated into the model when the scaled linear function is selected as the kernel function. The model is assumed to be the most complicated one first, *i.e.*, every point is assumed to be equally possible to be an edge point in the function. By applying the ARD analysis and RVM inference, most of the irrelevant kernels can be removed while keeping the relevant ones, which are actually the discontinuous points or discontinuities. The value of the hyper-parameter α_i will also indicate where the discontinuous points exist in the function.

4.3. Experiments of Sparse Bayesian Kernel for 1-D Piecewise Linear Signals

In this section, the Bayesian kernel formulation is applied to the given piecewise linear signal as described in the last section. Before proceeding, some criteria are needed to evaluate the effects

of discontinuity preservation and data fitting.

One criterion is commonly used to evaluate the fidelity of estimation, *i.e.* the infinity norm, which is defined by:

$$L_{f'}^{\infty} = \max | f'(i) - \hat{f}'(i) | \quad i=1,2,3...N-1 \tag{4.20}$$

where f' is the first order differentiation of f. Two new criteria are defined to evaluate whether the discontinuous points are well preserved. They are given as follows. For a given point, which is a step edge, we define:

$$L_S = \left| \frac{f'(i) - \sum_{j=i-M}^{i+M} \hat{f}'(j)}{f'(i)} \right| \quad \text{when } bd(i)=1 \text{ for step} \tag{4.21}$$

Actually, for each point i, a value of L_S can be calculated. The index (i) is omitted here, since this value is only important at the edge points. In Eq. (4.21), the change in gradient within the region of $\pm M$ sample points is measured. If $f(i)$ is a step edge, $f'(i)$ will have a higher gradient. This large gradient should be preserved in $\hat{f}(i)$. If an edge is preserved by the proposed algorithm, L_S will be near to zero. For a roof edge, it is similar and is given below:

$$L_R = \left| \frac{f''(i) - \sum_{j=i-M}^{i+M} \hat{f}''(j)}{f''(i)} \right| \quad \text{when } bd(i)=1 \text{ for roof} \tag{4.22}$$

Similarly, index (i) is also omitted here. For the non-edge points, *e.g.* $bd(i)=0$, L_S and L_R might be infinite, since $f'(i)$ and $f''(i)$ could be zero. So L_S and L_R will not be calculated for $bd(i)=0$. The value of L_S and L_R will indicate the percentage of a step/roof edge within the region of $\pm M$, which is not preserved by the proposed algorithm, when compared with the original signal. M is equal to 2 in the simulations.

In the following experimental section, the original signal $f(i)$ is shown by bold red dash line, the noisy signal $g(i)$ is shown by light black line, and finally the estimated signal $\hat{f}(i)$ is shown by bold blue line.

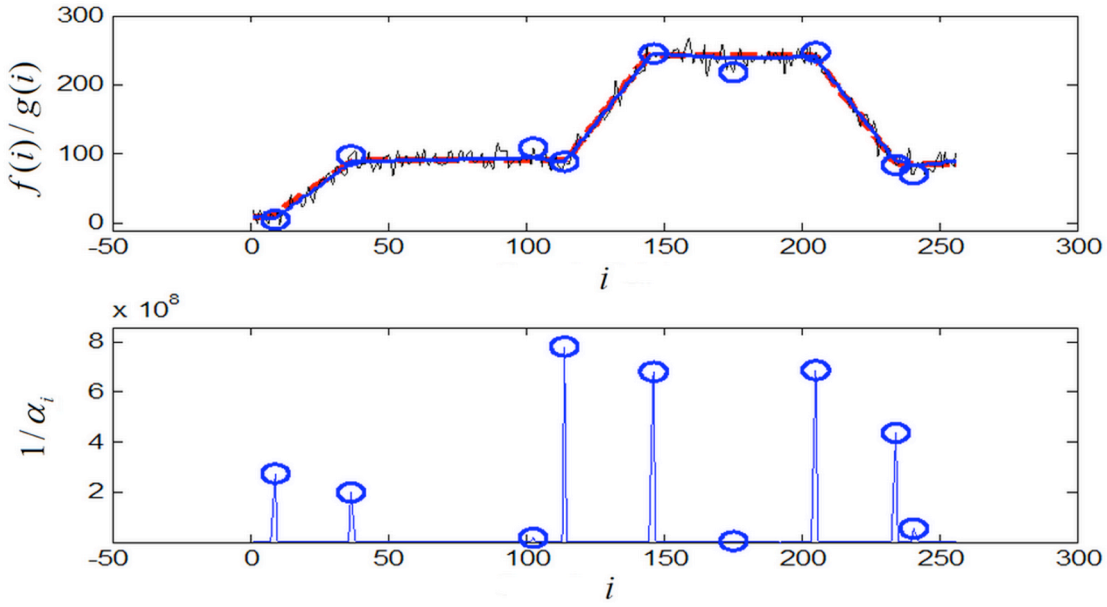


Figure 4.2 Roof Edge Experimental Result: Noise σ = 10. Total 6 Discontinuous Points in the Signal; Number of Used Kernels is 9; Estimated Noise Level $\hat{\sigma}$ = 9.63. RMS of the Fitting is 2.64; $L_{f'}^{\infty}$=6.97. Average of L_R at Those 6 Points is 10%.

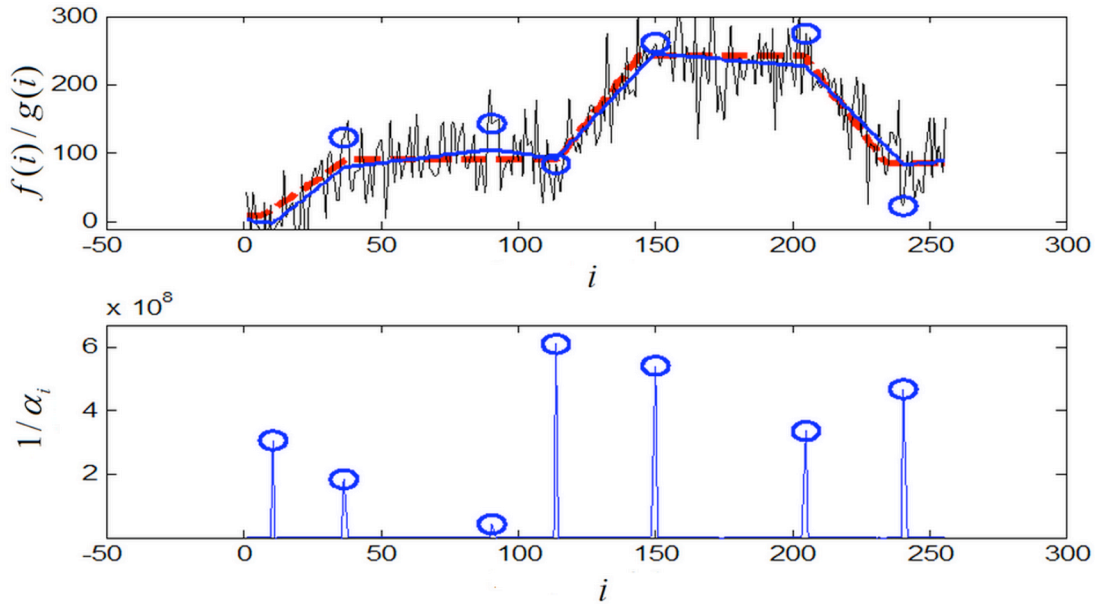


Figure 4.3 Roof Edge Experimental Result: Noise σ=40. Total 6 Discontinuous Points in the Signal; Number of Used Kernel is 7, Estimated Noise Level is $\hat{\sigma}$= 39.10. RMS of the Fitting is 39.26; $L_{f'}^{\infty}$=23.32. Average of L_R at Those 6 Points is 17%

The first experiment is a signal with only six roof edges. The estimated signal at different noise levels is shown in Figure 4.2 and Figure 4.3. Tables 4-1 and 4-2 show the edge preserving effect of the Bayesian kernel approach. The noise is removed and the sharp roof edges are preserved as shown in the two figures. L_R of both the simulation results are low, averaging, about 90% and 83% of "discontinuities" in Figure 4.2 and 4.3 are preserved for a total of 6 discontinuous points, respectively.

TABLE 4-1 DISCONTINUTY PRESERVATION FOR ROOF EGDE WITH NOISE σ=10 (FIGURE 4.2)

Edge Index	$\sum_{j=i-M}^{i+M} \hat{f}''(j)$	$f''(j)$	L_R
1	3.30	2.70	18%
2	-2.81	-2.70	4%
3	5.57	5.04	10%
4	-5.21	-5.04	3%
5	-5.23	-5.21	0.4%
6	4.13	5.21	26%

TABLE 4-2. DISCONTINUTY PRESERVATION FOR ROOF EGDE WITH NOISE σ=40 (FIGURE 4.3)

Edge Index	$\sum_{j=i-M}^{i+M} \hat{f}''(j)$	$f''(j)$	L_R
1	3.47	2.70	22%
2	-2.63	-2.70	3%
3	4.88	5.04	3%
4	-4.62	-5.04	9%
5	-3.62	-5.21	44%
6	4.	5.21	21%

It is interesting to note that L_R of edge points 1 and 6 in both Table 4-1 and Table 4-2 are large. This observation simply indicates that more gradient/discontinuity at those points is lost. The explanation is that those points are close to the ends of the signal, so there is not enough information to determine a better weight for those points. Second possible explanation for losing the discontinuities is that even though the discontinuities are preserved, however they are shifted, and they are not within the defined neighborhood of $\pm M$. The bottom-figures in Figure 4.2 and Figure 4.3 are $1/\alpha_i$. The clear sharp peaks (points circled) indicate that those points are the

discontinuous points in the given function. This proposed method can therefore be effectively applied for the discontinuity identification tasks too. When the noise level increases, both the RMS and $L_{f'}^{\infty}$ both become larger, making it more difficult to achieve a better estimation. At the same time, experimental results show that the Bayesian kernel also gives an accurate estimation of the noise variance σ, as stated in the title of each figure.

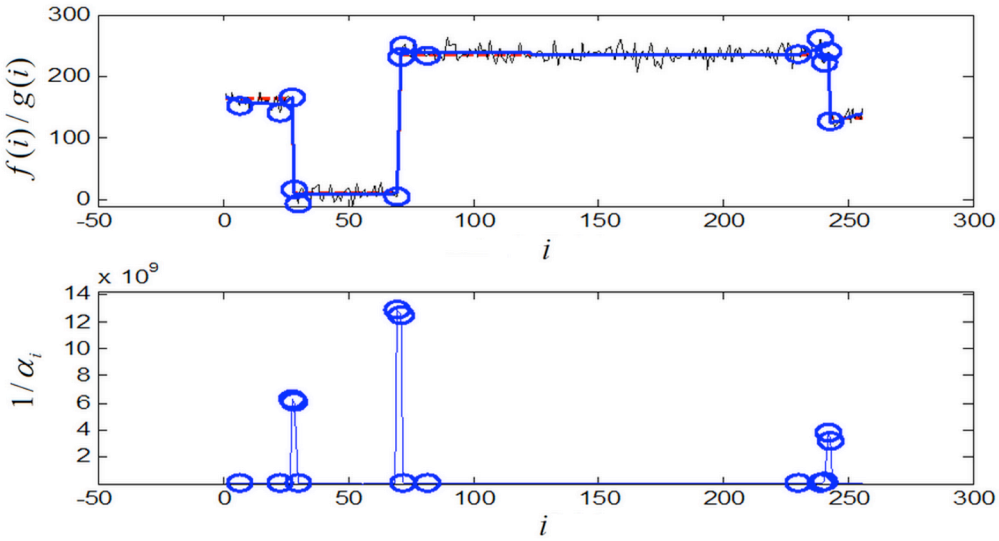


Figure 4.4 Step Edge Experimental Result: Noise σ=10. Total 3 Discontinuous Points in the Signal; Number of Used Kernel is 14; Estimated Noise Level $\hat{\sigma}$= 9.67. RMS of the Fitting is 2.63; $L_{f'}^{\infty}$ = 12.14. Average of L_S at Those 3 Points is 3%.

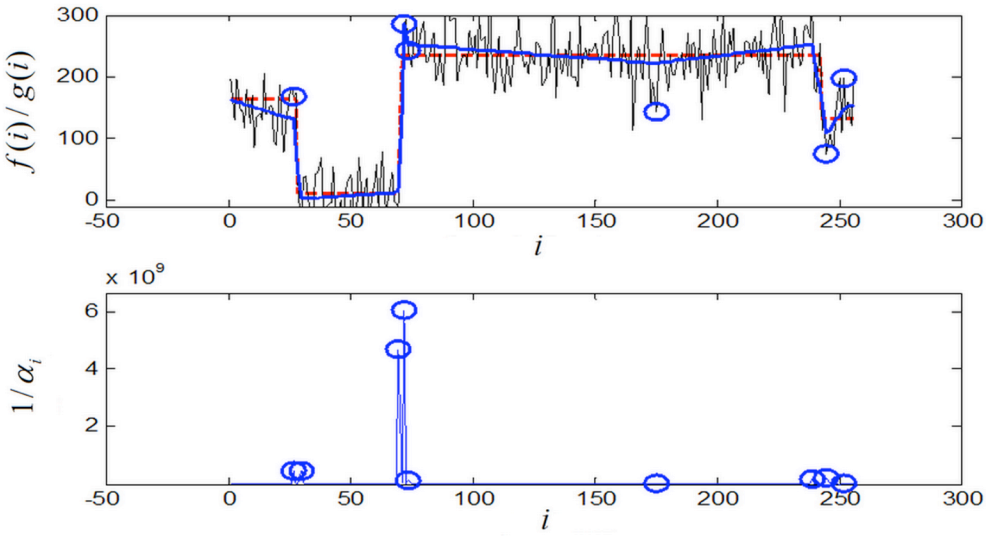


Figure 4.5 Step Edge Experimental Result: Noise σ=40; Total 3 Discontinuous Points in the Signal; Number of Used Kernel is 9; Estimated Noise Level is $\hat{\sigma}$= 39.19. RMS of the Fitting is 13.84; $L_{f'}^{\infty}$ = 87.069. Average of L_S at Those 3 Points is 22.5%.

The results of the Bayesian kernel method for the step edges with different noise levels are shown in Figure 4.4 and 4.5. The analysis is similar to that of the roof edge, amidst some notable differences. Firstly, the step discontinuities are more obvious and the gradients of those points are much larger than those found in the roof-edges shown in Table 4-1 and 4-2. $L_{f'}^{\infty}$ of step edge are also larger, since the singularity of those points are more severe, while L_s in Table 4-3 and Table 4-4 remains comparable to L_R in Table 4-1 and Table 4-2.

TABLE 4-3 DISCONTINUTY PRESERVATION FOR STEP EGDE WITH NOISE σ=10 (FIGURE 4.4)

Edge Index	$\sum_{j=i-M}^{i+M} \hat{f}'(j)$	$f'(j)$	L_S
1	-156.24	-151.97	3%
2	227.18	225.72	0.6%
3	-110.72	-105.19	5%

TABLE 4-4 DISCONTINUTY PRESERVATION FOR STEP EGDE WITH NOISE σ=40 (FIGURE 4.5)

Edge Index	$\sum_{j=i-M}^{i+M} \hat{f}'(j)$	$f'(j)$	L_S
1	-128.88	-151.97	15%
2	272.5	225.72	21%
3	-71.70	-105.19	32%

Some discontinuities in the reconstructed signal $\hat{f}(i)$ is lost. Note that in Figure 4.5, the $1/\alpha_i$ does not have clear peaks at edge points 1 and 3, while those peaks are more obvious in Figure 4.4. The peak of edge point 2 in Figure 4.5 is also smaller. They are caused by the increase in the noise level; higher noise level would make it more difficult to identify the discontinuities.

The last simulation is carried out with a signal containing both the steps and roof edges. The results are shown in Figure 4.6 and Figure 4.7. The results are also encouraging at different noise levels. From the $1/\alpha_i$ plots in both cases, only the step edge have clear peaks. This is due to the fact that the step edges involve more abrupt changes than that of the roof edges, so that they will dominate the hyper-parameter space. Hence, in a signal containing both kinds of edges, it would

be more difficult to detect the roof edges just by looking at the values of $1/\alpha_i$.

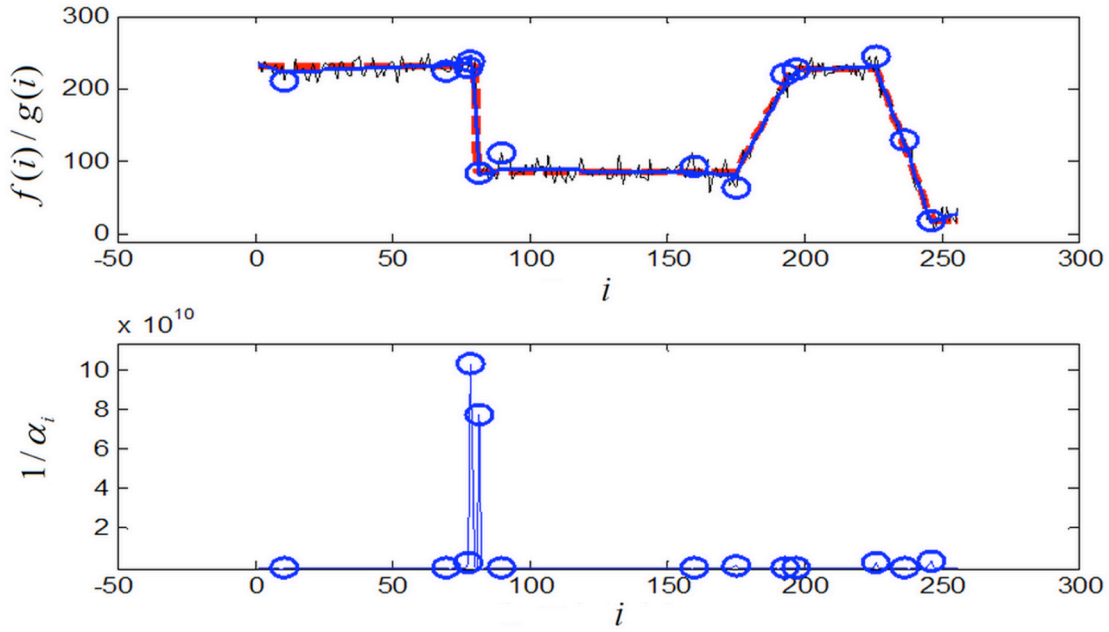


Figure 4.6 Two Types of Edges Experimental Result: Noise σ=10. Total 1 Step Edge and 3 Roof Edges in the Signal; Number of Used Kernel is 13; Estimated Noise Level $\hat{\sigma}$= 10.67. RMS of the Fitting is 5.03; $L_{f'}^{\infty}$ = 32.78. Average of L_S and L_R at Those 4 points is 16%.

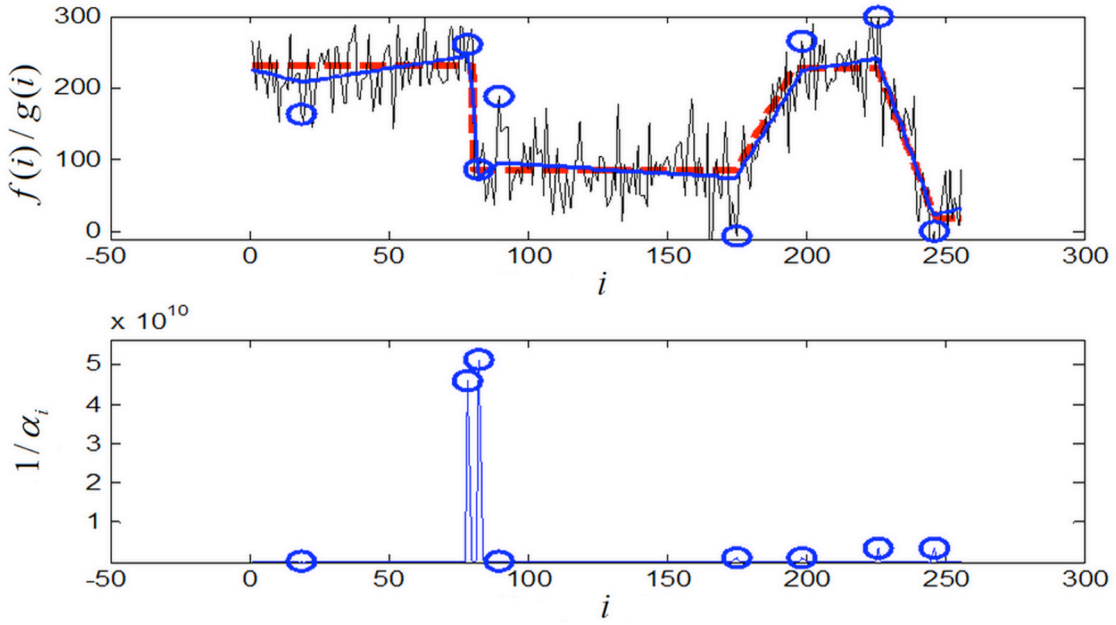


Figure 4.7 Two Types of Edges Experimental Result: Noise σ=40. Total 1 step Edge and 3 Roof Edges in the Signal. Number of Used Kernel is 8; Estimated Noise Level $\hat{\sigma}$= 39.53. RMS of the Fitting is 11.12; $L_{f'}^{\infty}$= 73.68. Average of L_S and L_R at Those 3 Points is 13%.

Comparing the value of L_R for edge point 3 in Table 4-5 and Table 4-6, more discontinuities are lost in Table 4.5. This result is mainly caused by the shift of the discontinuous point. The abrupt changes still exist based on visual observation, while it is not within the neighborhood $\pm M$.

TABLE 4-5 DISCONTINUTY PRESERVATION FOR STEP AND ROOF EGDE WITH NOISE σ=10 (FIGURE 4.6)

Edge Index	$\sum_{j=i-M}^{i+M}\hat{f}''(j)$ or $\sum_{j=i-M}^{i+M}\hat{f}'(j)$	$f'(j)$ or $f''(j)$	L_S or L_R
1	-163.96	-143.4	14%
2	6.97	7.19	3%
3	-6.97	-4.85	44%
4	-10.34	-10.03	3%

TABLE 4-6 DISCONTINUTY PRESERVATION FOR STEP AND ROOF EGDE WITH NOISE σ=40 (FIGURE 4.6)

Edge Index	$\sum_{j=i-M}^{i+M}\hat{f}''(j)$ or $\sum_{j=i-M}^{i+M}\hat{f}'(j)$	$f'(j)$ or $f''(j)$	L_S or L_R
1	-126.05	-143.40	12%
2	6.51	7.19	9%
3	-5.68	-4.85	17%
4	-11.47	-10.03	14%

4.4 Summary

The Bayesian kernel method for discontinuous signal reconstruction based on the piecewise linear assumption is presented. The mathematical model of the discontinuous signals is given based on the assumption that lower order derivatives of the signal are more significant. The proposed Bayesian kernel is able to determine the model order and fit the data simultaneously according to the given noisy data. The global fitting errors $L_{f'}^{\infty}$ are small. Estimated noise σ is accurate. Two new criteria to evaluate the edge preserving effects are also defined, *e.g.* L_S and L_R, which can describe the percentages loss of the discontinuities, *i.e.* discontinuities not detected, using the proposed approach, using a pre-defined neighborhood *M*.

Based on the experimental results, the Bayesian kernel is proven to be able to successfully smooth the piecewise linear signals and at the same time preserve most of the discontinuities. It helps to resolve the two conflicting interests in image processing, i.e., Noise Removal (smoothing) and Edge Detection (discontinuity preservation). On the average, about 80% of the discontinuities could be preserved in the estimated discontinuous signals at different noise levels. The value of $1/\alpha_i$ could be used to determine the location of the discontinuous points in the signals as well. It is promising for real applications.

The learning procedure of the Bayesian kernel is slower than the traditional smoothing procedure, as it is a problem of the order of $O(N^3)$. The inverse of the Hessian could be the restriction to the proposed algorithm. The training samples cannot be too many. In practice, when the number of sample points is more than 2000, Hessian matrix will easily become ill-conditioned. This is also reported in by [106] and [108]. The fast algorithm [108] can relieve this difficulty. In the fast training algorithm, Hessian will generally be smaller than $N \times N$, while training sample points still cannot exceed 4000 [108].

Chapter 5

Sparse Kernel Image Noise Removal with Edge Preservation

Thus far, the theoretical basis for 1-D piecewise linear signals is developed in the previous chapters. Images are 2-D signals. In this regards, theory will be extended on the above development to treat the additional dimension. With the additional dimension, the increase in data size will be a prime consideration. "Travelling window" will be developed to work on Regions of Interest, which are subsets of the whole image, one by one. Noise filtering, especially for blurred images, is an ill-conditioned problem. To resolve this issue, and to improve the image quality, a novel cost function will be proposed, which has only one global minimum. This is made possible by the use of the "travelling windows", the details of which will be presented in this Chapter.

5.1. Relevance Vector Machine in Image Restoration

The two previous chapters presented the approaches to identify and reconstruct the discontinuous signals in 1-D format. From the experimental results presented in Chapter 4, the RVM/Sparse Bayesian Kernel can reconstruct the 1-D discontinuous signal after smoothing (or noise removal) and at the same time preserve the edges. However, there are still some open issues in image restoration tasks as images are in fact 2-D signals. Based on the knowledge gained from working with 1-D signals, a novel image restoration method based on the Sparse Bayesian Kernel in spatial domain will be presented in this Chapter. The technical details will be discussed and then the restoration results will be presented in the experimental section.

5.1.1. Window and Local Piecewise Linear Assumptions

As mentioned in Chapter 2, not all the 2-D signals are eligible to be qualified to be an image.

"Eligible" here means that the 2-D signal must be able to be interpreted and understood by human beings. From the Markov Random Field (MRF) point of view, it has the similar implication. Although there are many configurations of random field, only a small portion of them is meaningful to human beings. An important topic in MRF image restoration is the study of the potential function V in Eq. (2.27). A proper potential function will give satisfactory restoration result. However, the potential function is only applicable within a given neighborhood system, which is generally small in size; 4- and 8- neighborhood systems are commonly used in the MRF restoration. Theoretically, bigger neighborhood system could be applied. However, since the combinations of the clique increase dramatically with the increase in neighborhood size, the calculation will be very intensive, sometimes; it could become infeasible if the size becomes too large.

Model based image processing has its advantages and is popular in the past decades. Among them, the piecewise constant and piecewise linear assumptions include Piecewise Image Model (PIM) and Local Image Model (LIM) in [44]. PIM also includes the Piecewise Constant (PICO) and Piecewise Linear (PILI). LIM includes the Local Lonotonicity (LOMO) and Local Convexity/Concavity (LOCO). Common images are generally object-based. For example, an image normally contains several objects, which have the same or similar gray levels. For example, there are several major objects in the camera man image shown in Figure 5.1 (a), such as the camera man, a camera with tripod and some buildings in the background. For the camera man, the face, his coat and pants are three objects, which have similar gray levels.

In the literature review presented in Section 1.3.2, other noise removal methods have been discussed. Methods based on Fourier Transform are difficult to integrate with the local assumptions, since Fourier Transform does not have resolution in time/spatial domain simultaneously. The explanations are.

- Fourier transform can only identify some specified frequency that exists in the frequency

domain; however, it cannot determine when/where an edge happens.

- At the edge, an abrupt change in spatial domain causes changes over a wide range in frequency. This makes edge preservation in frequency domain difficult. The common aftermath of frequency domain methods, such as the low pass filter or band pass filter, is the "ring" artifacts in the restored image, which is highly undesirable.

Wavelet is another useful tool in image restoration task. It has resolution in both spatial and frequency domain, however, some pertinent questions are still without good answers, such as the selection of the mother wave and the threshold value of the wavelet coefficients for noise removal.

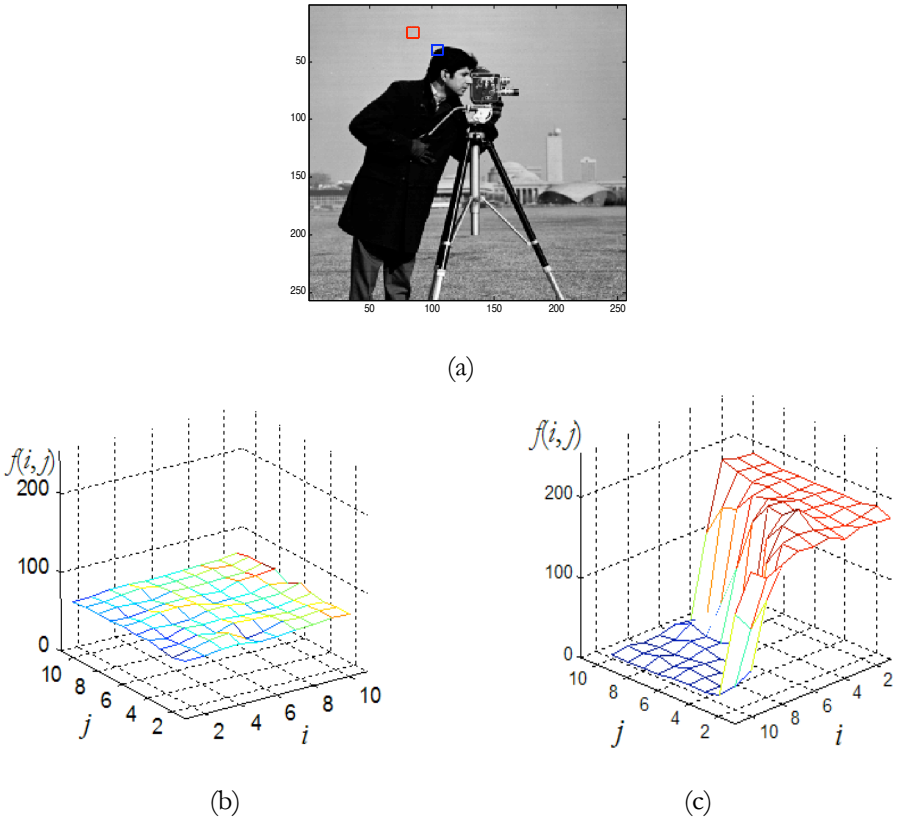


Figure 5.1 Local Piecewise Constant and Piecewise Linear Assumptions for the Cameraman. (a). The Rectangle Indicates the Window Size. (b) Shows the Red Rectangle Portion in 3D. (c) Shows the Blue Rectangle Portion in 3D. (Window Size=10)

Based on the above review and analysis of existing image model, the Locally Piecewise model (LP) for the images is presented in this work. Piecewise here includes both piecewise constant

and piecewise linear. It is true that image itself may not globally fit the piecewise assumption in spatial domain, however, within a small Region Of Interest (ROI); it could be approximated by the LP model according to Taylor theorem [109]~[110]. Figure 5.1 and 5.2 show two such small regions. This small region is called ***window*** indicated by the red and blue rectangle in Figure 5.1 and 5.2. ***Window*** is denoted by $\mathcal{W}$ in this work. From the Figure 5.1(b), 5.1(c) 5.2(b) and 5.2(c), the image within the window are actually piecewise constant or piecewise linear. The local piecewise assumption is important for the edge preservation. Therefore it is assumed that the images are locally piecewise linear or piecewise constant within the window $\mathcal{W}$.

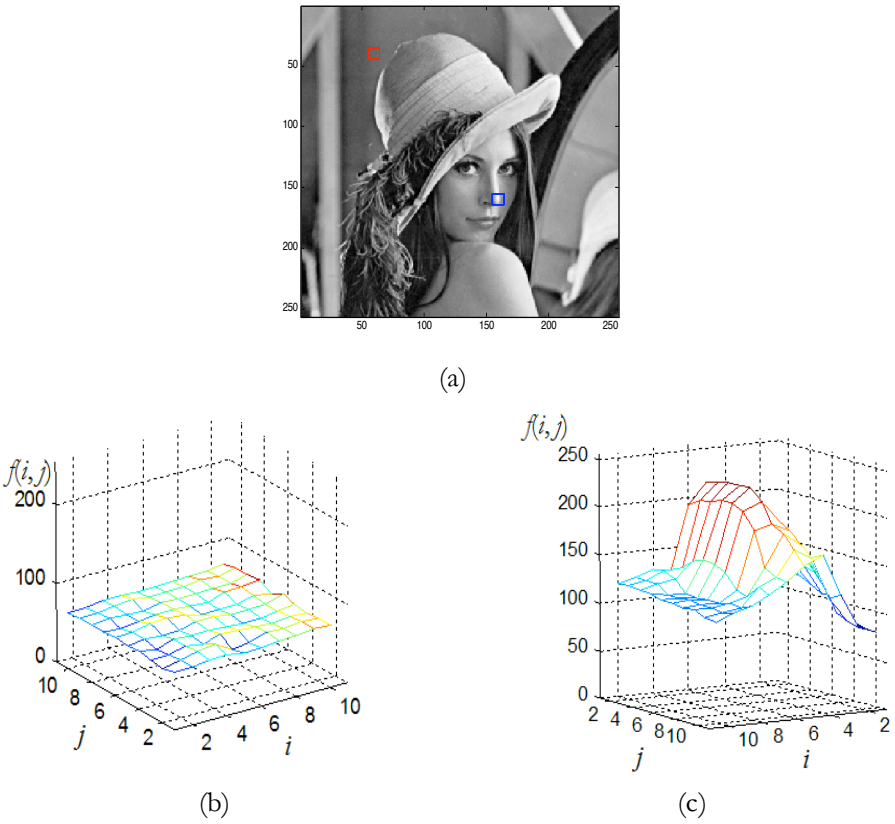


Figure 5.2. Local Piecewise Constant and Piecewise Linear Assumptions for the Lena. (a). The Rectangle Indicates the Window Size. (b) Shows the Red Rectangle Portion in 3D. (c) Shows the Blue Rectangle Portion in 3D. (Window Size=10)

5.1.2. Inverse of the Hessian Matrix

The LP model simplifies the image and process it within the small window $\mathcal{W}$. Higher order terms in Eq. (3.2) could be ignored within the window. This brings us the convenience and efficiency in

the subsequent calculation. However, the Sparse Bayesian Kernel cannot be applied for the whole image due to the following reasons:

- The image may be locally piecewise, but not necessary globally as stated above. The assumption of global piecewise is dangerous and questionable. Furthermore it could cause the loss of some detailed information. The locally piecewise constant and piecewise linear assumptions also indicate that a reasonable simple model could be used to fit the given noisy image within the window only. This local simplification integrates the Occam's razor in the proposed approach. Image, which is too complex, would be considered as noisy image.
- In Eq. (4.13), the inverse of the Hessian matrix needs to be calculated. If the restored image is estimated by using the whole image, the calculation is too intensive, and sometimes can be infeasible since the calculation of the inverse of a square matrix is in the order of $O(N^3)$. For example, for a 256×256 image as shown in Figure 5.1(a), we will need to calculate the inverse of the Hessian matrix of 65536×65536. The calculation is very burdensome. Furthermore, such a large matrix is normally ill-conditioned.

The inverse of the large matrix is a topic in Applied Mathematics, and it will not be discussed in details here. Since the presented piecewise assumption is only made within a given window, the operation of finding the inverse of a large Hessian matrix can be avoided. Instead, the large Hessian matrix is decomposed into many small matrices using multiple windows. This scheme is highly efficient and fast.

5.2. Local Regularization and Global Cost Function

5.2.1. Selecting Kernel Function

Based on the LP assumption, a proper kernel function should be selected. The kernel trick can map the signal to the hyperspace as mentioned in Section 2.2. In the last Chapter, the kernel

function in Eq. (4.2) is defined based on the piecewise assumption in 1-D. Here a similar definition in 2-D is given in Eq. (5.1). For a pixel at coordinates (i, j) within the determined $l \times l$ window, the kernel function is define by:

$$\phi[(i,j),(i',j')] = \sqrt{\eta}\sqrt{(i-i')^2 + (j-j')^2} \quad (i,j) \in \{1,2,3...l\}^2 \tag{5.1}$$

where (i, j) and (i', j') are two different pixels in the window and η is a non-zero scale factor. η is an important parameter, which is essential in achieving satisfactory restoration effects. Its proper value would be a good compromise between the noise and the edge preservation. The optimization of its value will be discussed later.

Note that the authors in [34] concluded that using a non-standard norm function for the MAP solution would be helpful in edge preservation, however, the reason on why this was so remains unclear. The kernel function in Eq. (5.1) is actually a Norm function since it satisfies the three requirements of Norm function, viz. non-negativity, triangular inequality and normality. Observing Eq. (5.1), the defined kernel function is actually a Norm function with values between 1 and 2. If one term in the square root equals to zero, then it becomes a 1-Norm function, otherwise, it is a hybrid Norm function between 1-Norm and 2-Norm. Based on the piecewise assumption and the use of window, the given kernel function will smooth the undesired noise while keeping the edges preserved within the window by selecting a proper η.

5.2.2. Window Size and Kernel Matrix

The size of the window will depend on the image itself and how it fits the piecewise linear assumption. Generally, if the image is complicated, smaller window is preferred. If the image is simple, larger window could be employed. However, the computational issue of the inverse Hessian Matrix needs to be considered. When the window is too big, the computation of the inverse of the Hessian Matrix will be slower. A common image is normally not isotropic, while the window applied for the image restoration is a square region, say $l \times l$.

After determined the window size, the kernel matrix K is:

$$K = \begin{bmatrix} \phi[(1,1),(1,1)] & \phi[(1,1),(1,2)] & \dots & \phi[(1,1),(l,l)] \\ \phi[(1,2),(1,1)] & \phi[(1,2),(1,2)] & \dots & \phi[(1,2),(l,l)] \\ \vdots & \vdots & \vdots & \vdots \\ \phi[(l,l),(1,1)] & \phi[(l,l),(1,2)] & \dots & \phi[(l,l),(l,l)] \end{bmatrix} \tag{5.2}$$

where the kernel function ϕ is defined in Eq. (5.1). The kernel matrix K is an $l^2 \times l^2$ matrix. Kernel matrix when $\eta = 1$ (Eq. (5.1)) is shown in Figure5.3.

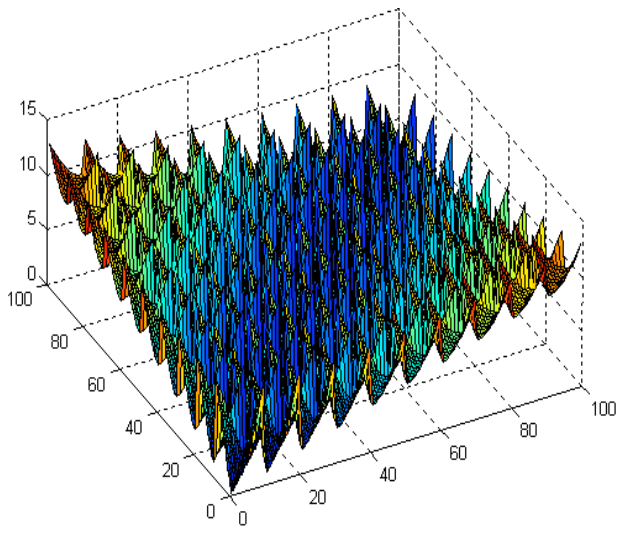


Figure 5.3 The Kernel Matrix Applied in Learning Process (η=1.0)

K is a symmetric matrix. Furthermore, the defined kernel function makes the kernel matrix a hybrid of linear and quadratic one. In other word, the kernel trick projects the sub-image within the window into a hybrid of linear and quadratic hyperspace depending on the locations of the pixels.

5.2.3. Hyper-parameter Tuning

The kernel learning is applied to the sub-image within the window. Every pixel (i, j) will have an associated weight $w(i, j)$ and a corresponding hyper-parameter $\alpha(i, j)$ to determine its distribution. Thereafter the learning is to find the best weights to describe the given image $\hat{f}$, which is:

$$\begin{bmatrix} \hat{f}(1,1) \\ \hat{f}(1,2) \\ \vdots \\ \hat{f}(1,l) \\ \vdots \\ \hat{f}(l,l) \end{bmatrix} = \begin{bmatrix} \phi[(1,1),(1,1)] & \phi[(1,1),(1,2)] & \cdots & \phi[(1,1),(1,l)] & \cdots & \phi[(1,1),(l,l)] \\ \phi[(1,2),(1,1)] & \phi[(1,2),(1,2)] & \cdots & \phi[(1,2),(1,l)] & \cdots & \phi[(1,2),(l,l)] \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \phi[(1,l),(1,1)] & \phi[(1,l),(1,2)] & \cdots & \phi[(1,l),(1,l)] & \cdots & \phi[(1,l),(l,l)] \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \phi[(l,l),(1,1)] & \phi[(l,l),(1,2)] & \cdots & \phi[(l,l),(1,l)] & \cdots & \phi[(l,l),(1,l)] \end{bmatrix} \begin{bmatrix} w(1,1) \\ w(1,2) \\ \vdots \\ w(1,l) \\ \vdots \\ w(l,l) \end{bmatrix} \quad (5.3)$$

For simplicity, the above equation could be written as:

$$\vec{\hat{f}} = K\vec{w} \quad (5.4)$$

The task of edge preserving robust noise removal procedure is to maximize the *posterior* distribution within the window $\mathcal{W}$ according to Bayesian rule:

$$\underset{\mathcal{W}}{p}(\vec{w},\vec{\alpha},\sigma^2 \mid \vec{g}) = \frac{\underset{\mathcal{W}}{P}(\vec{g} \mid \vec{w},\vec{\alpha},\sigma^2)\,\underset{\mathcal{W}}{p}(\vec{w},\vec{\alpha},\sigma^2)}{\underset{\mathcal{W}}{P}(\vec{g})} \quad (5.5)$$

For clarity, the notation $\mathcal{W}$ is omitted. In addition, the Kernel Learning described in this section concerns the Region of Interest in a window only. The hyper prior distribution of the hyper-parameter is given by:

$$p(\alpha_{i,j}) = \frac{1}{l^2} \quad (5.6)$$

where l is the size of the window. Finally, such inference will give us the estimation of the given weights within the window as:

$$p(\vec{w} \mid \vec{g},\vec{\alpha},\sigma^2) = \frac{P(\vec{g} \mid \vec{w},\sigma^2)\,p(\vec{w} \mid \vec{\alpha})}{P(\vec{g} \mid \vec{\alpha},\sigma^2)}$$

$$= (2\pi)^{-(N+1)/2} \left|\Sigma\right|^{-1/2} \exp\{-\frac{1}{2}(w-\mu)^T \Sigma^{-1}(w-\mu)\} \quad (5.7)$$

where:

$$\Sigma = (\sigma^{-2} K^T K + A)^{-1} \tag{5.8}$$

$$\mu = \sigma^{-2} \Sigma K^T \bar{g} \tag{5.9}$$

A=diag[$\alpha_{11}, \alpha_{12}, .. \alpha_{1l}, \alpha_{21}, \alpha_{22}, \alpha_{2l} ... \alpha_{ll}$]. Some of the weights on the right-hand side of Eq. (5.4) will be eliminated according to the given scale factor η after the learning procedure. The sub-image within the window will be reconstructed by only using the non-zero weights according to Eq. (5.4). In other word, the undesired non-smoothness will be removed according the scale factor η. The optimization of Eq. (5.5) is only local regularization. The scale factor η is a significant parameter. If η is too large, the noise will not be sufficiently removed from the image. It is also called over fitting. However, if η is too small, then the image will be over-smoothed, also known as under fitting. The original content of the image will be damaged. The globally optimized η must achieve the compromise between the noise removal and the edge preservation aims.

The prior knowledge tells us that the undesired noise is *i.i.d* Gaussian distributed. The noise term in Eq. (2.1) is of zero mean with unknown variance. However, in order to remove this undesired noise in the image, how severe the noise is, *e.g.* the variance, should be known such that it is possible to determine whether the non-smoothness is caused by the image content or the noise. In order to achieve a global optimization on η, the variance of the noise needs to be estimated. Since the added noise is *i.i.d.* Gaussian distributed noise, the estimation of the noise variance is not difficult. Some of the references provided different approaches, such as the Wiener filter [26] ~ [27], the Wavelet methods[29] and fast estimation of the noise variance [28], [30] and [31].

The strategy of edge preserved noise removal for image restoration is that when the discontinuities are smoothed, the smooth operator will smooth those discontinuities caused by the noise only, not the discontinuities that existed in the original image contents, *e.g.* edges, such that the edge can be preserved and noise is removed. The applied window will give the estimation of a

portion $\hat{f}$. Eq. (2.1) for the given sub-image can be written in a different way:

$$e(i,j) = g(i,j) - f(i,j) \quad \{1 \le i \le l, 1 \le j \le l\} \tag{5.10}$$

However the real non-degraded image $f(i, j)$ is unknown. Instead of using f, its estimation $\hat{f}$ is used. Then Eq. (5.10) becomes:

$$e(i,j) = g(i,j) - \hat{f}(i,j) \quad \{1 \le i \le l, 1 \le j \le l\} \tag{5.11}$$

where g is the noisy image and $\hat{f}$ is the estimation of f in Eq. (5.4). The estimation $\hat{f}$ is based on the LP assumption according to the kernel matrix given in Eq. (5.2). If $\hat{f}$ is a good estimation of f, then the variance σ_w^2 of $e(i, j)$ within the given window should be close to the variance σ_n^2 of the global noise term e_n in Eq. (2.1). Here again, the variance σ_n^2 of the global noise is also unknown, while it can be estimated, say $\hat{\sigma}_n^2$.

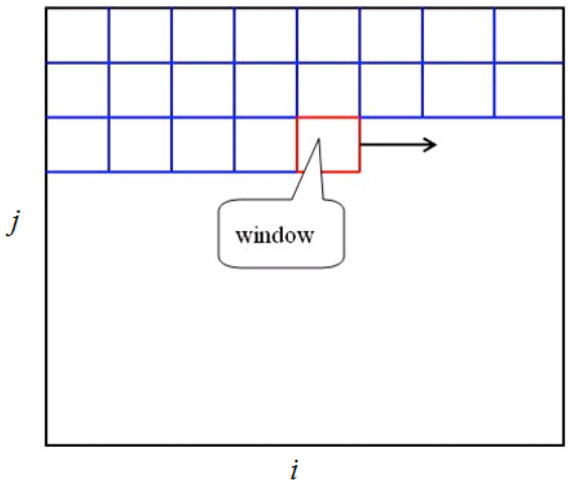


Figure 5.4 The Non-overlap Window Scanning for the Estimation of the Local Variance

According to the prior knowledge, the variance σ_w^2 of $e(i, j)$ within every window can be calculated. Here $\sigma_{w,k}^2$ is simplified as σ_w^2, where k denote the k^{th} windows. Different methods could be used to estimate the local variance. In this work, a line-scanning scheme is applied, as

show in Figure 5.4. The advantage of this scheme is that the entire image will contribute to the estimation of noise variance. Other approaches can also be employed such as the random window selection.

The window travels across the whole image. There is no gaps and overlaps between different windows, as shown in Figure 5.4. For every window, a noise variance for the local fit can be calculated . The calculation of the local variance is:

$$\begin{aligned}\sigma^2_{w,\mathrm{k}} &= \frac{1}{l^2}\sum_{\substack{i=1\\j=1}}^{\substack{i=l\\j=l}}(e(i,j)-\mu)^2 = \frac{1}{l^2}\sum_{\substack{i=1\\j=1}}^{\substack{i=l\\j=l}}(e(i,j))^2 \\ &= \frac{1}{l^2}\sum_{\substack{i=1\\j=1}}^{\substack{i=l\\j=l}}(g(i,j)-\hat{f}(i,j))^2 \quad \{1\le i\le l, 1\le j\le l\}\end{aligned} \tag{5.12}$$

where μ is the local mean of the noise term. According to the prior knowledge, it is zero. Please note that the local variance is actually a function of η, since $\hat{f}$ is a function of η. Based on the following theory, deduce the distribution function of $\sigma^2_{w,\mathrm{k}}$ can be deduced.

Theorem [111]**:** *If the random variable e follows the standard normal distribution, i.e. the Gaussian distribution with zero mean and unit variance, one can draw a sample* $e_1, e_2, \ldots e_{l^2}$ *of size* l^2 *from this distribution and form the sum of squares:*

$$\chi^2 = e_1^2 + e_2^2 + \ldots e_{l^2}^2 \tag{5.13}$$

The random variable χ^2 *(chi-square) follows the probability density of the* χ^2 *distribution with* l^2 *degrees of freedom:*

$$f(\chi^2) = \frac{(1/2)^{l^2/2}}{\Gamma(l^2/2)}\chi^{l^2/2-1}e^{-\chi^2/2} \tag{5.14}$$

And the mean and the variance of χ^2 *is:*

$$E(\chi^2) = l^2 \tag{5.15}$$

$$E(\chi^2 - E(\chi^2))^2 = 2l^2 \tag{5.16}$$

Based on the above theorem, the distribution function of $\sigma^2_{w,\mathrm{k}}$ can be deduced. Suppose that there are K windows to estimate the local fitting error and the noise term $e \sim N(0, \hat{\sigma}^2)$, then we get:

$$E(\sigma^2_w) = \frac{1}{K}\sum_{k=1}^{K}\sigma^2_{w,\mathrm{k}} = \hat{\sigma}^2 \tag{5.17}$$

$$E(\sigma^2_w - E(\sigma^2_w)) = \frac{2\hat{\sigma}^4}{l^2} \tag{5.18}$$

After the window has been made to travel through the image, K number of variances of the local fitting error can be obtained. The number of the given variance will be:

$$K = floor(N/l) \times floor(M/l) \tag{5.19}$$

According to the central limit theory, when K is big enough, the distribution of $\sigma^2_{w,\mathrm{k}}$ could be approximated by Gaussian distribution. For example, when the image size is 256×256 and the window size is 10×10, K will be 625. In order to achieve the best restoration result, an optimized value of the hyper-parameter η is needed through a global cost function, which will be presented shortly.

As $\hat{f}$ is a function of η, the value of η will determine the fitting result. The global optimization will be based on the estimated global variance $\hat{\sigma}_n^2$. After K local variances is obtained, its mean and variance can be calculated according to Eq. (5.17) and (5.18). If a proper η is applied for the

noise removal, then it should remove the undesired noise and keep only the discontinuities in the original image. In other words, the removed part should be a Gaussian distributed noise term. For the $\sigma^2_{w,\mathrm{k}}$, it should belong to $N(\hat{\sigma}^2, 2\sigma^4/l^2)$. The cost function for the global optimization will be:

$$R(\bar{f} \mid \eta, \bar{g}) = \arg\ \min(\mid E(\sigma^2_{w,k}) - \hat{\sigma}^2_n \mid + \sqrt{\mid E(\sigma^2_{w,k} - E(\sigma^2_{w,k})) - \frac{2\hat{\sigma}^4}{l^2} \mid}) \quad (5.20)$$

Eq. (5.20) indicates that the optimized estimation of f is to select such a η, which can make the mean and the variance of variances for all local fitting as close as to the global variance possible. If the fitting noise term satisfied Eq. (5.20), it is only a necessary condition $\hat{f}$ is a good estimate of g. However, since both the image model and the fitting model are all locally piecewise, then Eq.(5.20) becomes a necessary and sufficient condition for $\hat{f}$ is a good estimate of g, *e.g.* the removed discontinuities are the undesired noise. The estimation of the global variance is essential to the fitting results. Fortunately, the existing methods [28] ~ [31] could provide us robust and accurate estimation of such parameter. Since $\sigma^2_{w,\mathrm{k}} \sim N(\hat{\sigma}^2, 2\sigma^4/l^2)$, there is only one maximum in such a Gaussian distribution function and there is only one global minimum in the cost function in Eq. (5.20). This property is useful, the solution is stable and the ill-conditioned problem could be solved for the image restoration.

Eq. (5.20) uses the absolute value of the difference between $E(\sigma^2_w)$ and σ^2_n, however, the magnitude of σ^2_n will have different consequences, which is illustrated in Figure 5.5. Suppose that the estimated global variance is 30, as shown by the red line in Figure 5.5. When η is too large, then $E(\sigma^2_w)$ is smaller than the estimated global variance, which is shown by the blue line with the highest peak. This is called over fitting. The image is over parameterized and the noise is not fully removed from the image. Conversely, when η is too small, and then $E(\sigma^2_w)$ is larger

than the estimated global variance, which is shown by the blue line with the lowest peak. This is known as under fitting. Some of the image contents will be lost. Hence, the proper value of η must be obtained. The optimization of Eq. (5.20) is not difficult, normal gradient approach, linear and nonlinear approach can successfully fulfill the task.

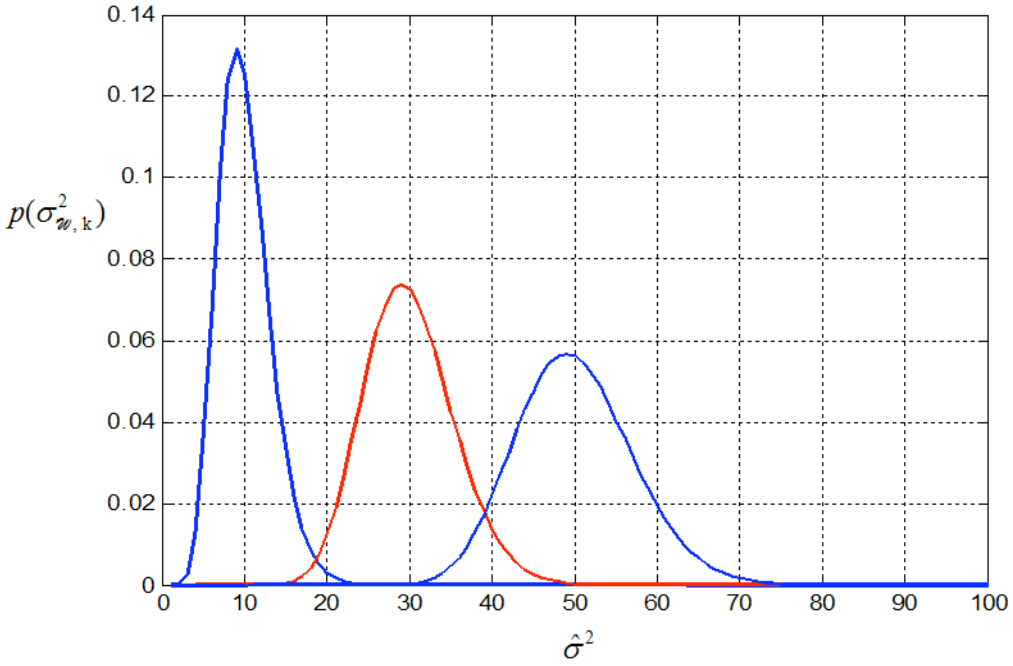


Figure 5.5 Under Fitting and Over Fitting of the Image

According to the above procedure, the hyper-parameter is tuned according to the local fitting variances by using a sliding window and global estimated noise variance. As the window is made to travel across the image and it is applied on small regions of the image, one at a time, the boundary effects will exist. These effects are common in image processing using approaches based on window. In order to overcome the boundary effects, overlaps are needed when a window is moved from one region to the next one when performing noise removal. Note that this is different when the hyper-parameter η is tuned. The amount of overlap could be adjusted depending on needs. For example, if the window moves one pixel at each step, vertical and horizontal directions will have 90% of overlap and the diagonal will have 81% overlap.

5.4. Experimental Results

In this section, the Bayesian kernel method for 2-D images is applied for the task of noise removal. In order to validate the proposed algorithm, four images are selected for the experiments. They are the "camera man", "Lena", "Lamp" and "Egg". The experiments are applied under 4 different noise levels, σ = 5, 10, 15 and 20. Since a too big value of the hyper-parameter will not truly fit the data, the value of η is from 0.1 to a bigger value and make sure the optimal value is in this region. The tuning range of the hyper-parameter η is from 0.1 to 30. The tuning procedure is described in the last section. Since it has been proven that the cost function will only have one minimum in Section 5.3.2, the steepest gradient method is applied for the optimization of η. At the same time, the results of other adaptive methods are presented for comparison, such as the Wiener Filter and Local Polynomial Approximation and Intersecting Confidence Intervals (LAP-ICI) in [47].

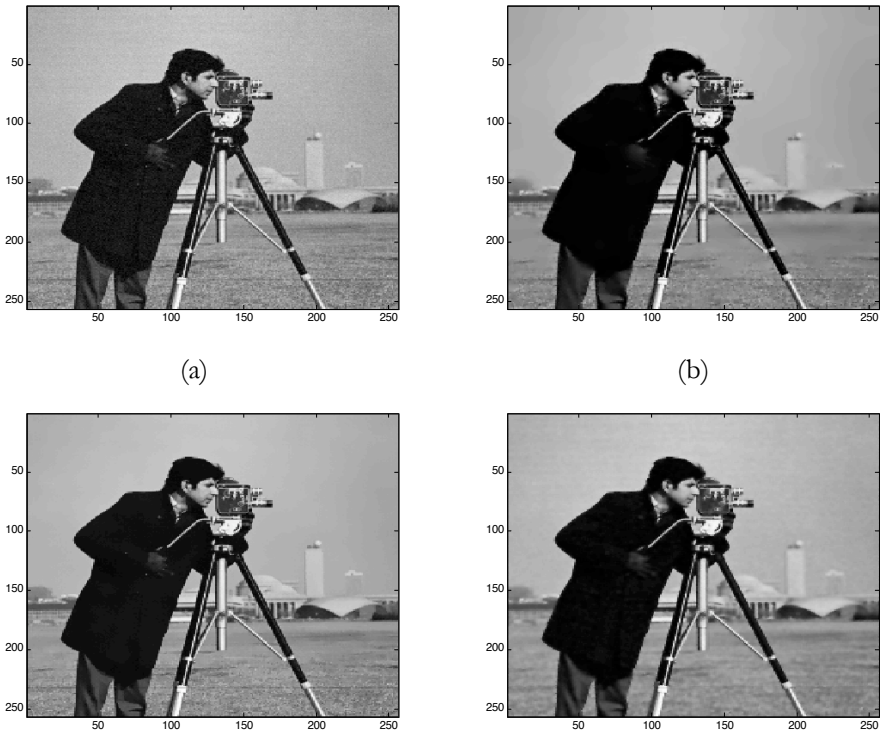
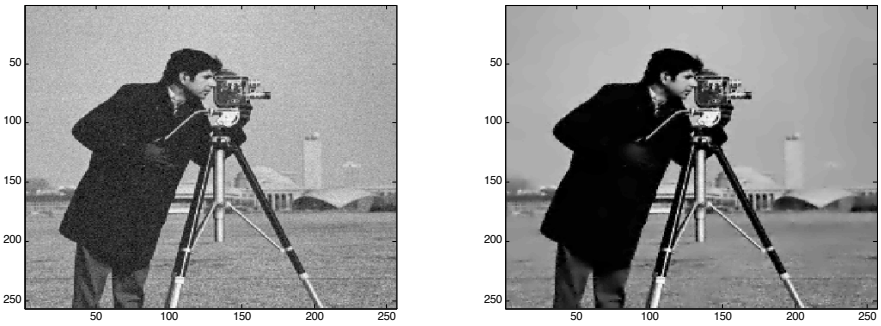


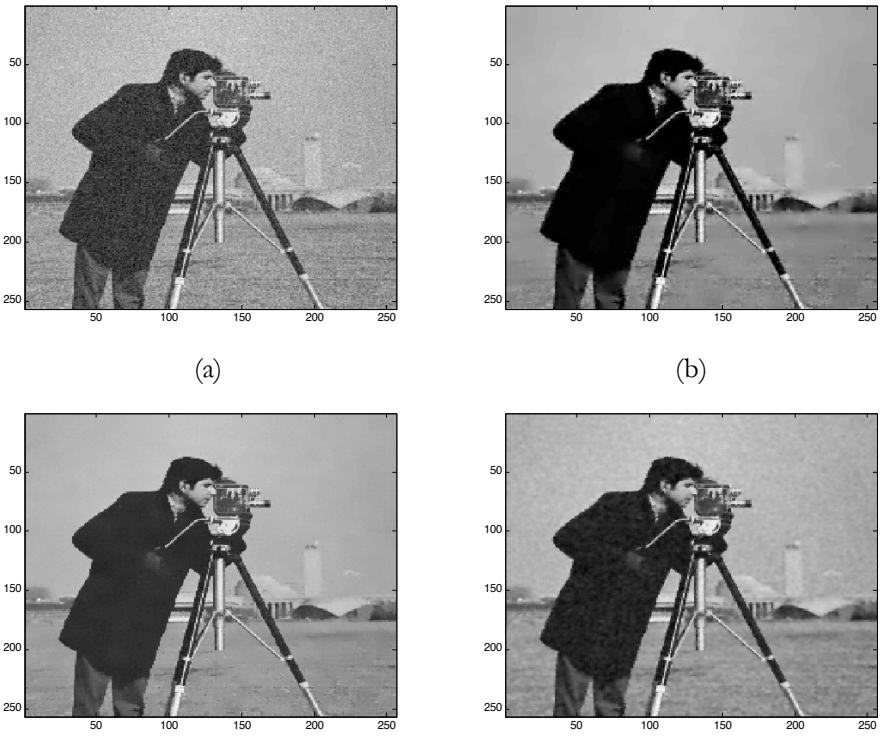
Figure 5.6 The Experimental Results of Cameraman σ =5



(a) (b)

(c) (d)

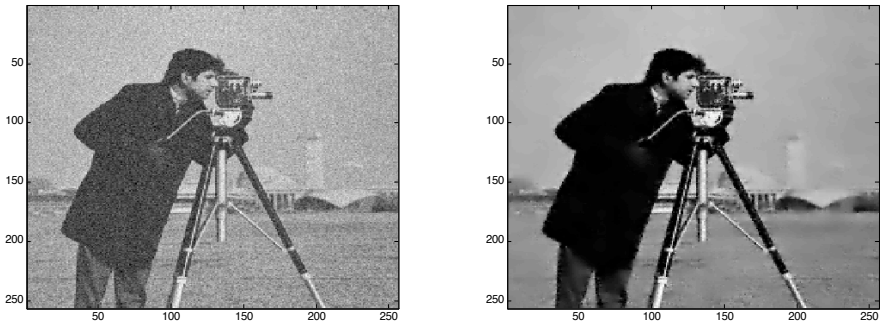
Figure 5.7 The Experimental Results of Cameraman σ =10



(a) (b)

(c) (d)

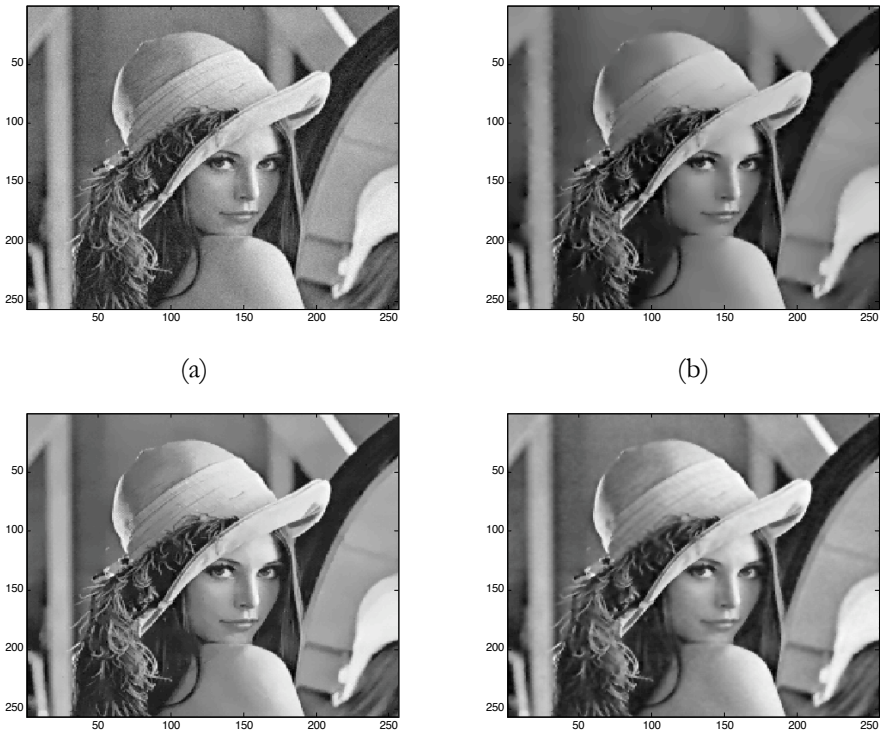
Figure 5.8 The Experimental Results of Cameraman σ =15



(a) (b)

(c) (d)

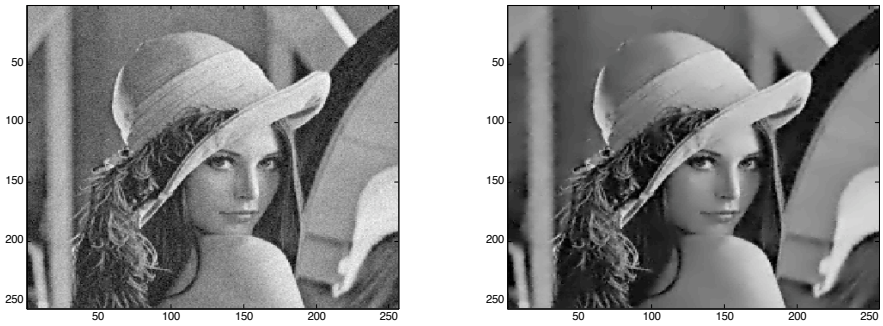
Figure 5.9 The Experimental Results of Cameraman σ =20



(a) (b)

(c) (d)

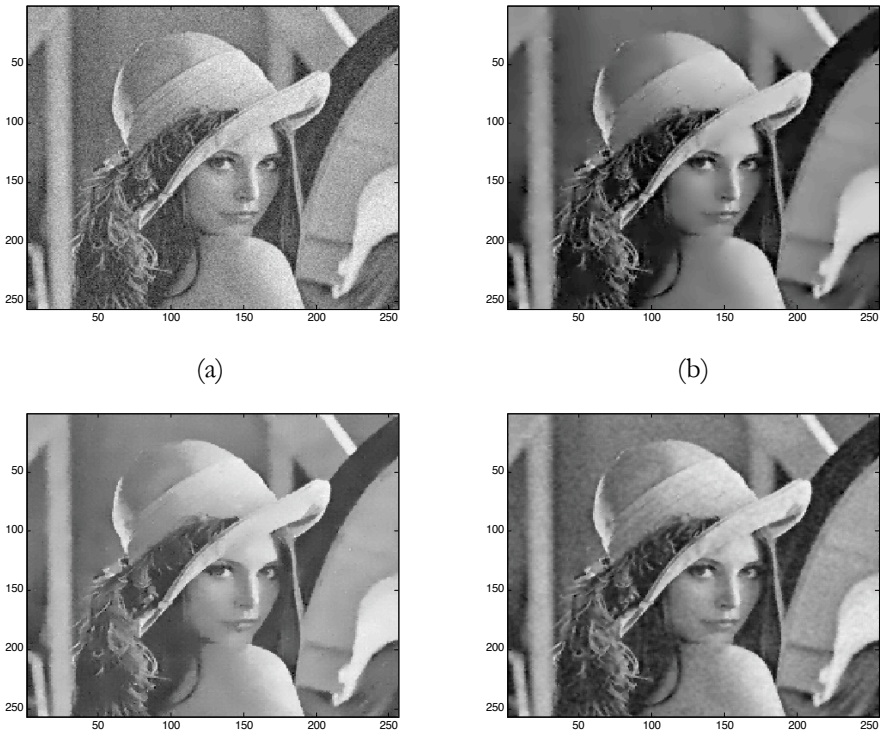
Figure 5.10 The Experimental Results of Lena σ =5



(a) (b)

(c) (d)

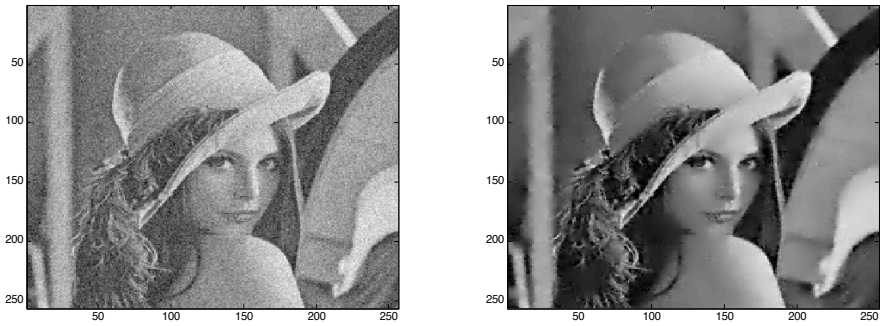
Figure 5.11 The Experimental Results of Lena σ =10



(a) (b)

(c) (d)

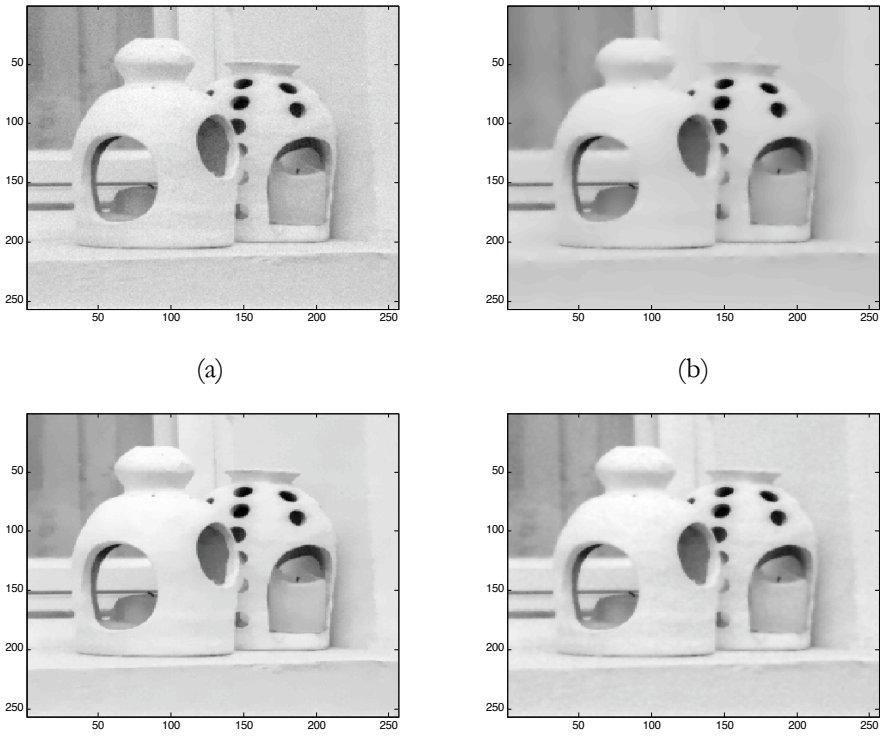
Figure 5.12 The Experimental Results of Lena σ =15



(a) (b)

(c) (d)

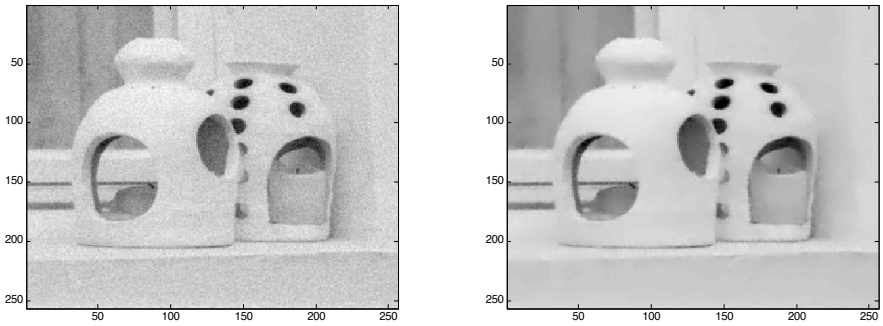
Figure 5.13 The Experimental Results of Lena $\sigma = 20$



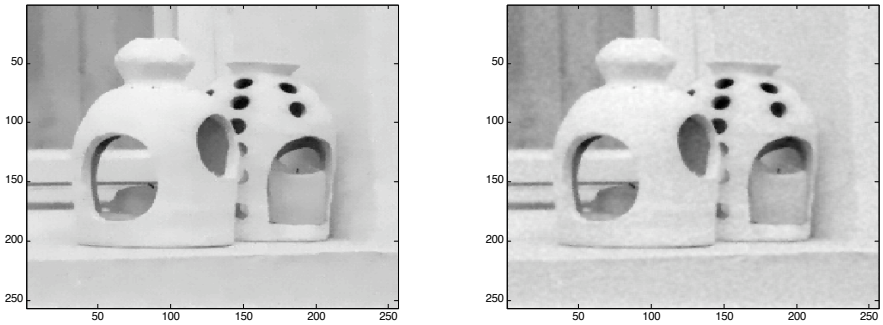
(a) (b)

(c) (d)

Figure 5.14 The Experimental Results of Lamp $\sigma = 5$

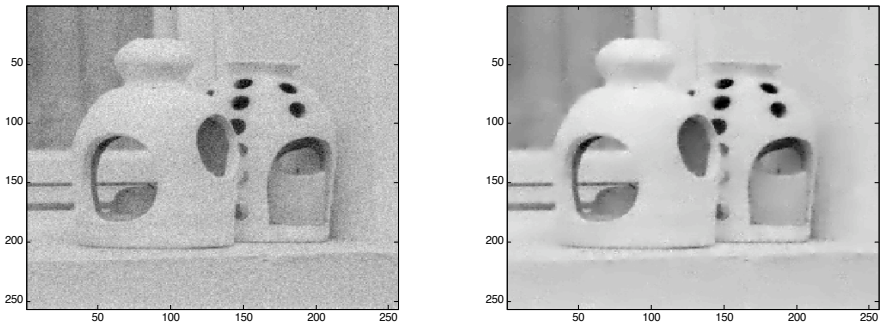


(a) (b)

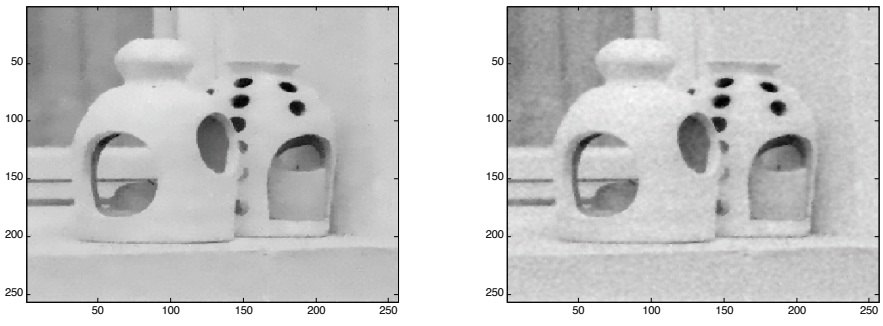


(c) (d)

Figure 5.15 The Experimental Results of Lamp σ =10

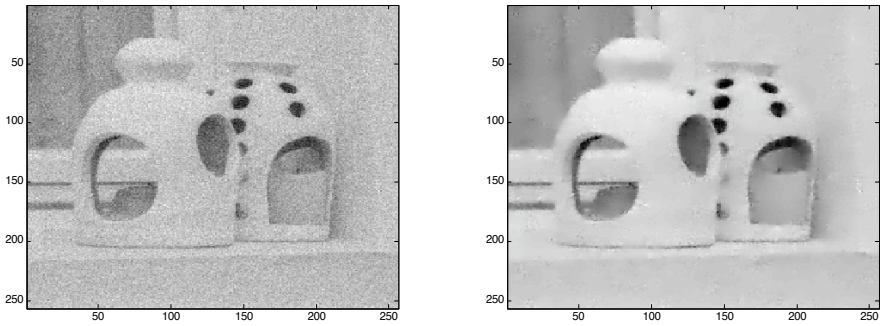


(a) (b)

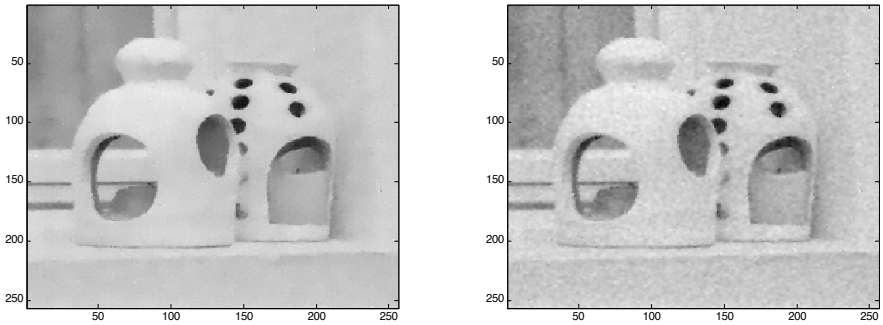


(c) (d)

Figure 5.16 The Experimental Results of Lamp σ = 15

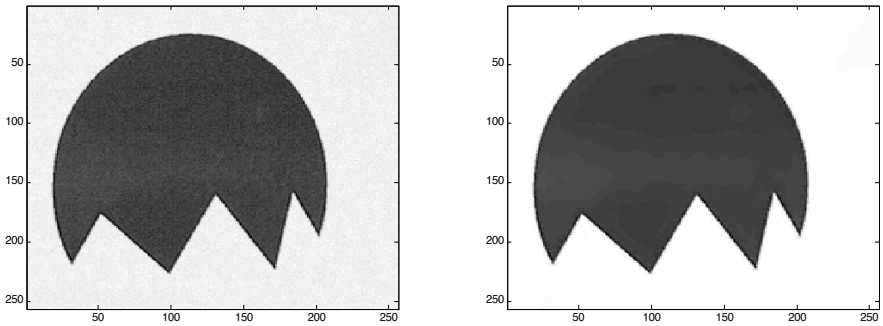


(a) (b)

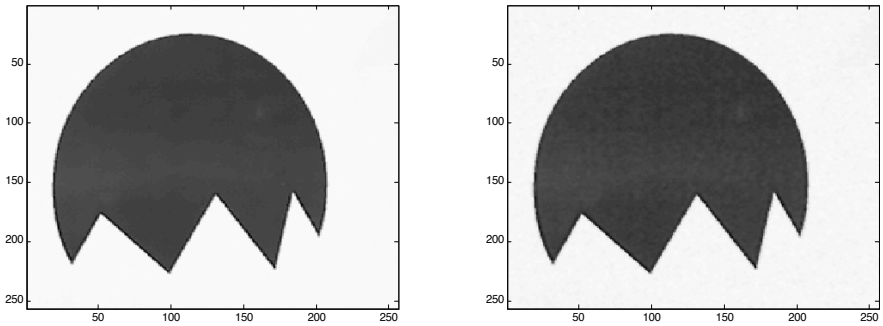


(c) (d)

Figure 5.17 The Experimental Results of Lamp σ = 20

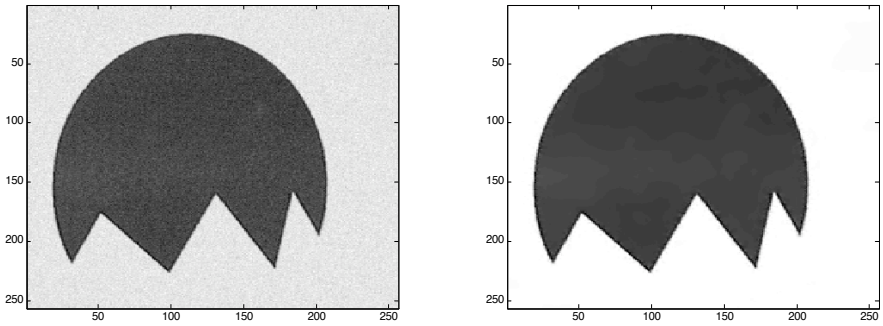


(a) (b)

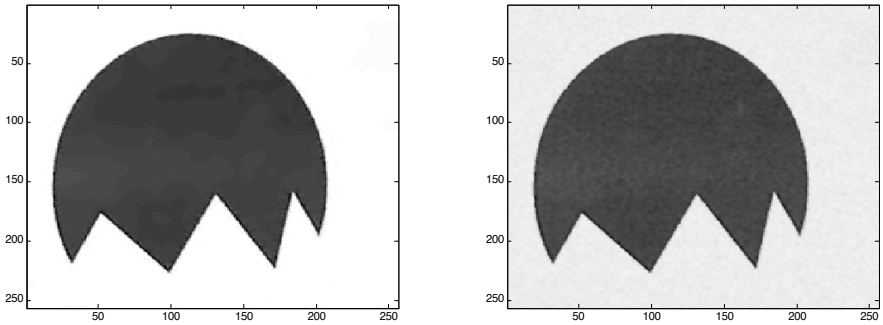


(c) (d)

Figure 5.18 The Experimental Results of Egg σ =5

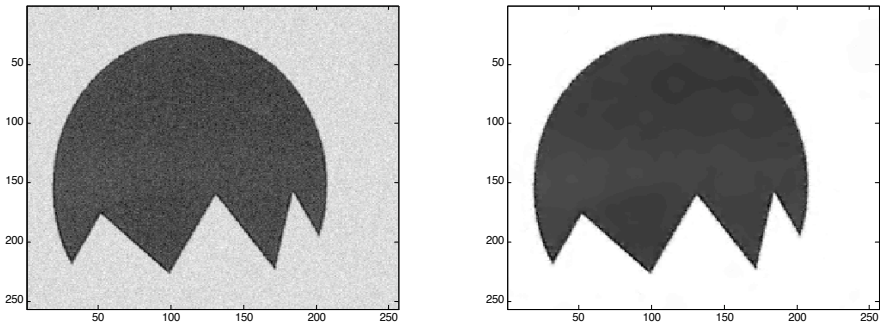


(a) (b)

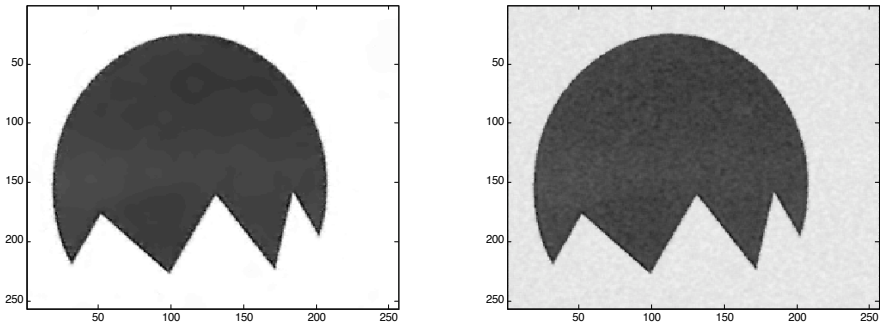


(c) (d)

Figure 5.19 The Experimental Results of Egg σ =10

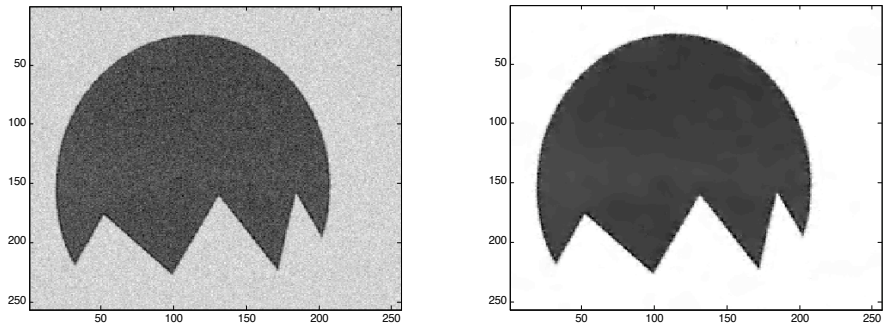


(a) (b)

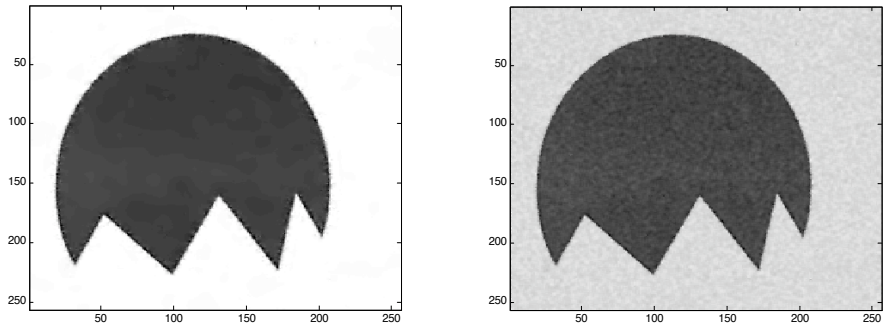


(c) (d)

Figure 5.20 The Experimental Results of Egg σ =15



(a) (b)



(c) (d)

Figure 5.21 The Experimental Results of Egg σ =20

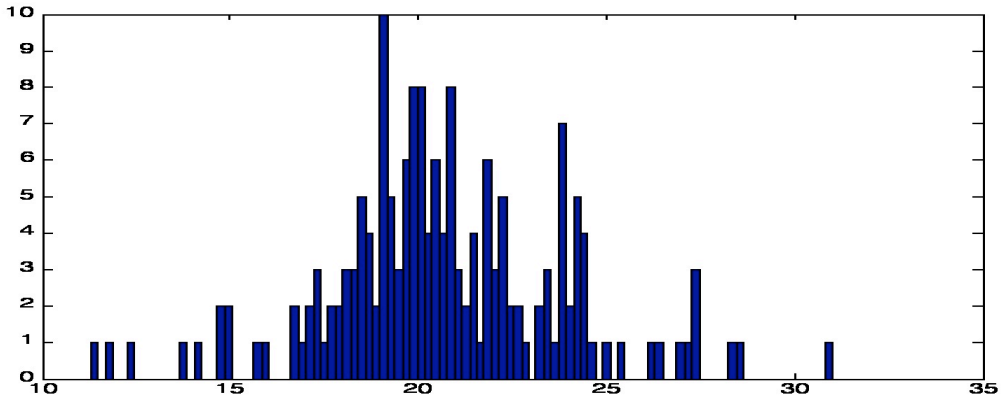
In order to visually compare the results, the experimental results are shown in the above figures (Figure 5.6 to Figure 5.21). In these figures, Figure 5.xx(a) are the noisy images; Figure 5.xx(b) are the result of the proposed approach; Figure 5.xx(c) are the results of the LPA-ICI method and Figure 5.xx(d) are the results of the Wiener filter. From these figures, there are the following observations:

(i) The proposed approach achieves comparable visual effect as the LPA-ICI method. With increasing noise levels, both the proposed approach and the LPA-ICI method clearly out-perform the classical Wiener filter,

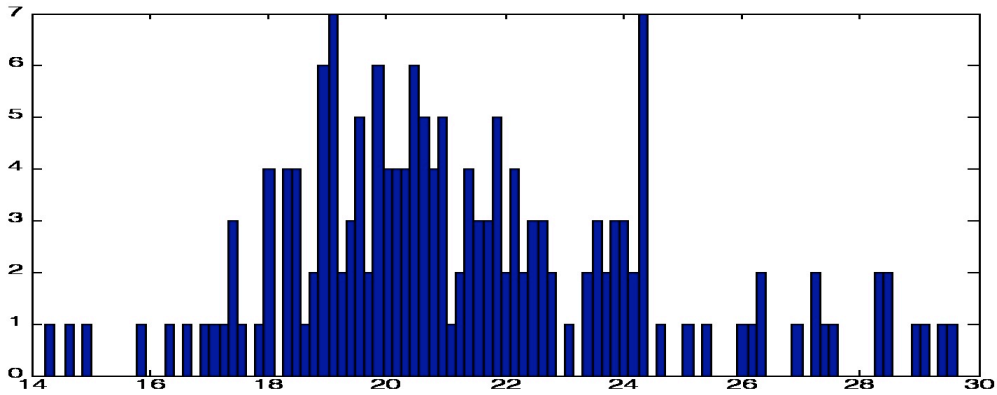
(ii) No significant difference between the restored image of the proposed approach is observed.

(iii) The proposed method is as robust as the LPA-ICI approach. When the noise level increases, it could still achieve good visual results. Generally, when the noise level is high,

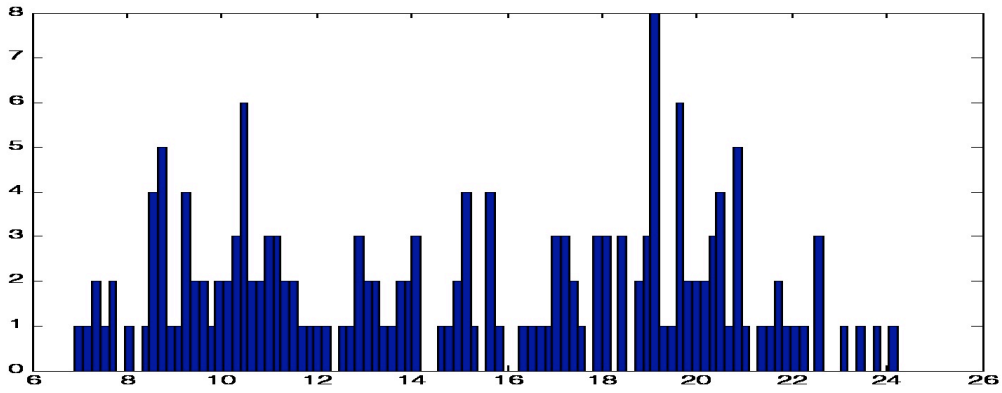
or when SNR is low, removal of the noise commonly will cause the over-smoothing at the edges in the image. However, the presented Bayesian kernel, as well as the LPA-ICI, can preserve the edges and remove the noise.



(a) Optimized Hyper-parameter η = 3.40



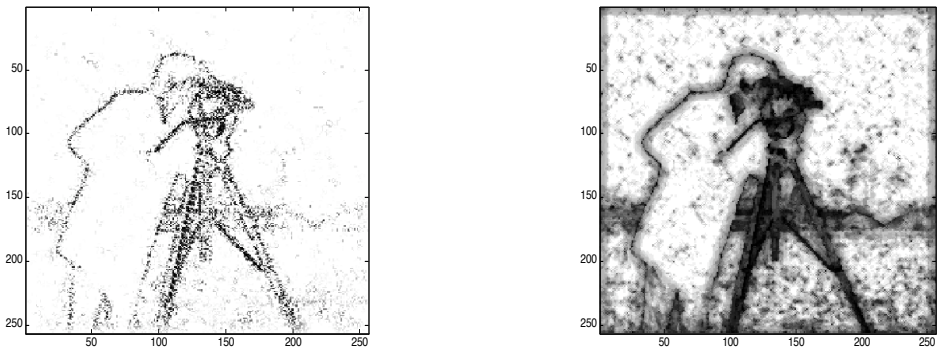
(b) Over-smoothed Hyper-parameter η = 3.00



(c) Under-smoothed Hyper-parameter η = 4.95

Figure 5.22 Hyper-parameter tuning of the Cameraman with σ =20

The tuning of the hyper-parameter is an important step in this approach. A proper value of the scale factor η will give a good estimation of the original image. The hyper tuning results of the "camera-man" image is used to explain the under-smooth and the over smooth consequences. The noisy cameraman image with σ =20 is shown in Figure 5.9. Some results of the hyper tuning are shown in Figure 5.22. Figure 5.22(a) shows the optimized η. This η minimizes the cost function for the tuning procedure shown in Eq. (5.21), however it is not strictly equal to zero. The plot in Figure 5.22(b) shows that η is too small. In this situation, some information of the original image will be damaged and lost, *i.e.* over-smoothing. Figure 5.22(c) shows η is too high. In such cases, noise will not be satisfactorily removed, *i.e.* some discontinuities caused by the noise will still be remained in the restored image.



(a) (b)

Figure 5.23 Edge Preservation of Proposed Approach and LPA-ICI

The significant step in this approach is to identify the "important" pixels, which are the edge points or discontinuities in the image. During the smoothing process for noise removal, these points will be preserved. These points are shown in Figure 5.23(a). The identified edge/discontinuous points using the LPA-ICI approach are shown in Figure 5.23(b). Both methods recognized the edges, although the underlying principles are different.

Computation speed is one important consideration of the algorithm evaluation. The experiments and simulations in this work are all carried on the P4 2.0 with RAM 1.0G machine. The

computation time of the different approaches with noise σ = 10 is shown in Table 5-1. The calculation time for those four images with other three noise levels is similar. Form Table 5-1, the proposed approach is slower than LPA-ICI and Wiener filter. LPA-ICI is about 60 times slower than Wiener Filter and the proposed approach is about 10 time slower that LPA-ICI. One of the reasons is that LPA-ICI and Wiener filter use the Fast Fourier Transform (FFT) to calculate the convolution, which is a famous efficient fast algorithm. However, the proposed approach is purely based on the spatial Local Piecewise assumption. Local information is processed point by point. However, computing speed is not the only issue. If image quality can be further improved by spending more time on calculation, it is still desired.

TABLE 5-1 COMPUTATION TIME OF DIFFERENT APPROACHES

$\sigma = 10$	CAMARAMAN	LENA	LAMP	EGG
Bayesian Kernel approach	15min 22sec	14min 03sec	16min 21sec	11min 35sec
LPA-ICI	1min 17sec	1min 22sec	1min 32sec	56sec
Wiener Filter	0.89sec	0.78sec	0.87sec	0.87sec

As discussed in Section 2.4, all criteria which were defined from Eq. (2.35) ~ (2.41) in Chapter 2 are used to evaluate the experimental results. The data are shown in Table 5-2 to 5-4.

TABLE 5-2 EXPERIMENTAL RESULTS OF THE BAYESIAN KERNEL APPROACH

	CAMERAMAN				LENA				LAMP				EGG			
σ	5	10	15	20	5	10	15	20	5	10	15	20	5	10	15	20
$\hat{\sigma}$	6.22	11.24	15.91	20.68	6.29	11.01	15.89	20.70	5.07	10.09	14.98	20.17	5.20	10.14	15.23	20.19
η	7.06	5.59	4.34	3.40	7.36	6.85	4.78	3.82	7.71	7.53	5.36	4.16	5.65	5.79	3.60	3.05
R	1.84	4.22	6.04	7.78	3.86	4.11	5.98	7.97	4.13	3.83	5.62	7.64	1.77	3.81	5.73	7.65
SNR	28.52	26.19	24.12	22.60	25.54	24.70	22.47	20.93	33.35	32.93	30.58	28.92	37.79	36.08	33.38	31.42
ISNR	0.31	4.02	5.50	6.45	1.17	4.01	5.27	6.27	2.13	7.72	8.95	9.76	6.03	10.35	11.11	11.69
PSNR	34.48	32.15	30.08	28.57	32.97	32.12	29.90	28.36	36.32	35.90	33.55	31.88	40.17	38.47	35.76	33.80
MSE	23.15	39.59	63.86	90.46	32.83	39.83	66.60	94.88	15.18	16.71	28.70	42.13	6.25	9.26	17.26	27.11
RMS	4.81	6.29	7.99	9.51	5.73	6.31	8.16	9.74	3.90	4.09	5.36	6.49	2.50	3.04	4.15	5.21
MAE	3.30	4.17	5.13	5.97	4.35	4.67	5.88	6.90	2.75	2.82	3.63	4.34	1.40	1,69	2.12	2.60
MAX	45.21	49.05	56.45	91.30	45.17	42.53	49.93	67.82	35.30	33.52	48.71	66.32	26.42	35.31	53.40	68.66

TABLE 5-3 EXPERIMENTAL RESULTS OF LPA-ICI APPROACH

	CAMERAMAN				LENA				LAMP				EGG			
σ	5	10	15	20	5	10	15	20	5	10	15	20	5	10	15	20
SNR	28.52	26.19	24.12	22.60	25.54	24.70	22.47	20.93	33.35	32.93	30.58	28.92	37.79	36.08	33.38	31.42
ISNR	3.17	4.87	6.01	6.97	2.56	4.59	5.87	6.90	6.29	8.30	9.64	10.53	8.86	10.67	11.62	12.07
PSNR	37.34	33.01	30.59	29.08	36.71	32.72	30.49	28.98	40.48	36.47	34.24	32.64	43.00	38.80	36.27	34.18
MSE	12.00	32.51	56.81	80.36	13.86	34.79	58.14	82.16	5.83	14.66	24.44	35.35	3.26	8.57	15.32	24.84
RMS	3.46	5.70	7.54	8.96	3.72	5.90	7.63	9.06	2.41	3.83	4.94	5.95	1.81	2.93	3.91	3.03
MAE	2.49	3.80	4.93	5.82	2.83	4.31	5.43	6.37	1.76	2.68	3.42	4.08	1.17	1.81	2.37	3.03
MAX	24.53	45.09	71.78	103.28	21.97	40.58	56.36	89.41	21.99	38.81	48.80	75.81	17.88	41.20	50.58	91.36

TABLE 5-4 EXPERIMENTAL RESULTS OF WIENER FILTER APPROACH

	CAMERAMAN				LENA				LAMP				EGG			
σ	5	10	15	20	5	10	15	20	5	10	15	20	5	10	15	20
SNR	28.52	26.19	24.12	22.60	25.54	24.70	22.47	20.93	33.35	32.93	30.58	28.92	37.79	36.08	33.38	31.42
ISNR	2.27	2.48	4.51	5.52	0.36	3.76	5.33	6.16	4.93	7.04	7.70	7.71	5.05	7.14	7.75	7.85
PSNR	31.90	30.62	29.09	27.63	33.79	31.88	29.95	28.25	39.11	35.21	32.31	29.84	39.17	35.26	32.41	29.96
MSE	41.96	56.32	80.15	112.10	27.19	42.22	65.77	97.30	7.97	19.59	38.24	67.51	7.88	19.35	37.34	65.64
RMS	6.48	7.50	8.95	10.59	5.21	6.50	8.11	9.86	2.82	4.42	6.18	8.22	2.81	4.40	6.11	8.10
MAE	4.15	5.21	6.47	7.78	3.69	4.77	6.04	7.42	2.01	3.30	4.68	6.21	1.87	3.20	4.53	6.05
MAX	47.61	51.50	63.94	85.20	34.43	45.38	51.33	73.72	34.82	36.97	48.04	61.01	32.13	34.87	49.19	63.79

From the experimental results in Table 5-2 to 5-4 and the error analysis:

1. The estimated noise level is accurate, however is still biased. Theoretically, the cost function of the hyper tuning in the proposed approach should be zero, however, because of the bias and the limitation of the iteration process during optimization, the cost function is not zero. Therefore, optimized results are only an approximation.

2. The RMS and the MSE of LPA-ICI is slightly smaller than the proposed approach. One of the RMS error of three different approaches under different noise level is shown in Figure 5.24. The MAE of LPA-ICI is very close to the proposed approach.

3. From the data, the proposed approach and the LPA-ICI method are obviously better than the traditional Wiener filter. The ISNR and the PSNR of the LPA-ICI method is slightly high than the proposed approach, as shown in Figure 5.25. The error criteria of both methods are also close and all increase with the increase of the noise variance.

4. Since this work is focused in the edge preservation, it is important to compare the proposed method with other approaches at the edge. When the noise level is high, *e.g.* variance equals to 15 and 20, the MAX errors, which is the infinite norm, of all four tested images of the proposed approach are all significantly smaller than the LPA-ICI approach as shown in Figure 5.26 and Table 5-2. We know that the maximum of the fitting error normally occurs at the edge points. This indicates the proposed approach performance better than LPA-ICI approach at the edge point, *e.g.* the edge is better preserved by the proposed approach. The cost is more computing time is needed than LPA-ICI.

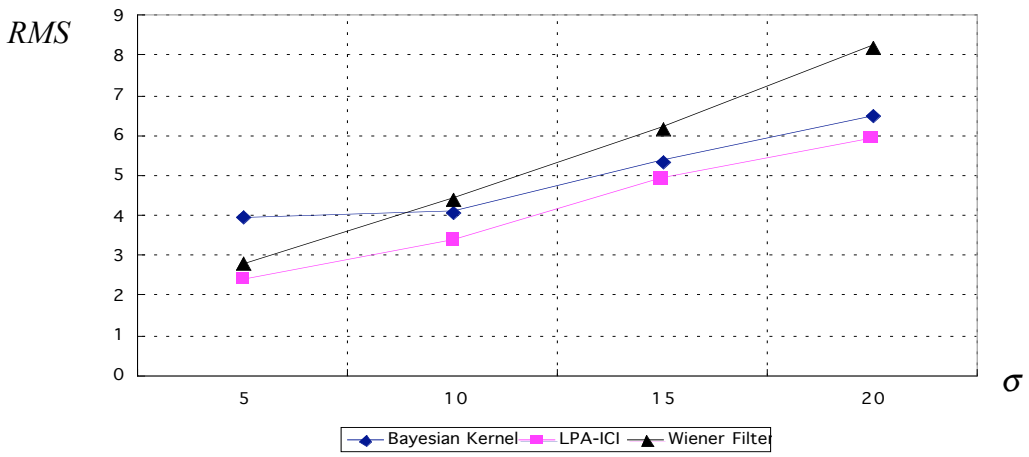


Figure 5.24 RMS Error of Bayesian Kernel, LPA-ICI and Wiener Filter (Image of "Lamp")

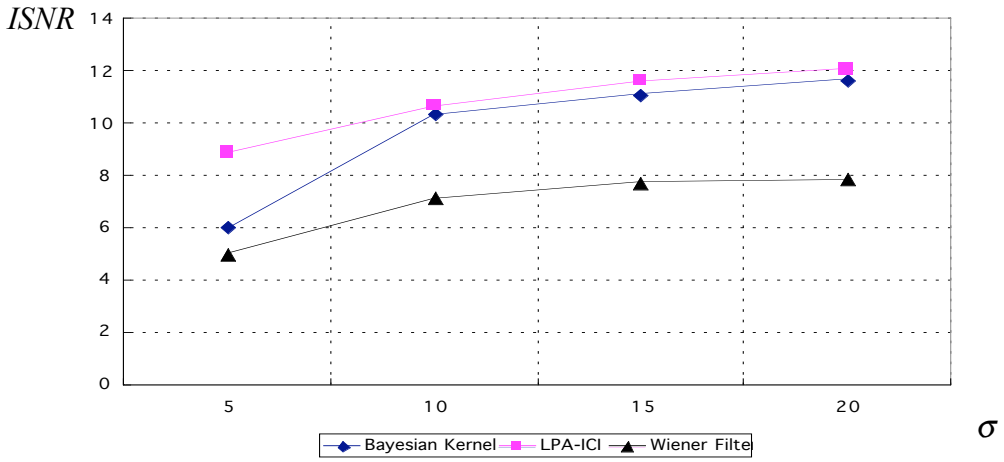


Figure 5.25 ISNR of Bayesian Kernel, LPA-ICI and Wiener Filter (Image of "Egg")

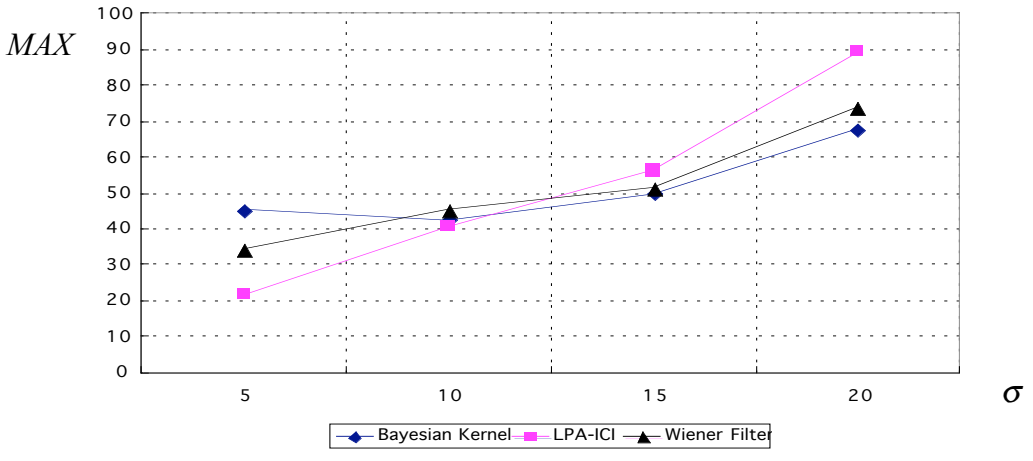


Figure 5.26 MAX Error of Bayesian Kernel, LPA-ICI and Wiener Filter (Image of "Lena")

Wiener filter's MAX error is smaller than the proposed approach, however, due to other errors are much bigger that both LPA-ICI and the proposed approach, its restored images are competitive by visual investigation, as shown in Figure 5.6 ~5.21. The proposed approach achieved good effects and better preserved edges among all those three methods.

5.5. Summary

From the experimental results, the proposed Bayesian kernel approach with the Local Piecewise assumption can successfully remove the undesired noise and preserve the edges very well. The speed of the calculation will be determined by the window size (for the ROI) and the image size. Since the LPA-ICI method uses the Fast Fourier transform (FFT) to perform the convolution, algorithm presented in this work is slower than the LPA-ICI method. For example, the time of the LPA-ICI method needs more than 60 seconds to process a 256×256 image (depends on the number of the and the size of the convolution kernel), while the proposed method will need less than 20 min to process the same image including the hyper tuning procedure. The LPA-ICI generally uses more than one scales, however the proposed approach only uses one scale.

The results of the proposed method are comparable with the LPA-ICI method. From the SNR point of view, LPA-ICI is slightly better than the presented approach due to the bias of global noise estimation. There is no significant visual difference between them. MAX errors of the proposed approach is significantly smaller than the LPA-ICI when SNR is low. Since the edge encoded important information of an image, it is meaningful to compare the error at the edges. One obvious advantage of the proposed approach is that it performs better than LPA-ICI in the meaning of MAX error, which means the error at the edge points is smaller or the edges are better preserved than the LPA-ICI.

There is one important reason that affects the results of the proposed method. The estimations of the noise variance globally and locally are both biased. This will cause the cost function in Eq.

(5.21) to be only an approximation of the minimum. From Table 5-2, even the optimized value of the cost function finally is small, but it is not equal to zero. Better performance of the proposed method can be expected by faster and better tuning approach for the hyperparameter.

The approach in this work is the first Bayesian kernel approach for edge preserved image restoration method. So far there is no such attempt amongst the reviewed literature on this topic. The significance of the proposed inference and formulation is that the method itself guarantees the stability of the estimation since the given cost function only has one global minimum. The ill-condition of the image restoration is successfully solved. The proposed approach achieves comparable image quality both in visual inspections and quantitative criteria compared with the LPA-ICI, which is the best approach reported in the image restoration literature so far.

It is clear that the proposed method performs almost equally well as the IPA-ICI, however it is better for edge preservation according to MAX error. It does not require the convolution and de-convolution process in image restoration, while the LPA-ICI depends heavily on convolution process and the use of FFT to speed up the computation. The approach mentioned here is first such attempt, and with further refinements and computation algorithm, the presented approach here would possibly surpass the LPA-ICI, both in computational efficiency and accuracy.

Chapter 6

Statistical Approach for Motion Blur Identification

The next step after noise removal would be the identification of blur, if any, as mentioned in Chapter 1 (Figure 1.1). This step is to determine of the nature of the blurring process. In this Chapter, a statistical approach to determine the characteristics of the blur – Point Spread Function (PSF) will be developed. From the literature, problems concerning blurring of image due to motion in a single direction are more difficult to handle than those due to bi-directional motions such as out-of-focus blurring since they are more ill-conditioned. We therefore, shall present our approach to handle cases involving blurring due to single directional motion.

6.1. Introduction

After the noise is successfully removed and the contents/edges of the image are preserved, the information on the blurring should be identified which is needed for the deconvolution task, *e.g.* the PSF and the noise. In the procedure of deblurring, both *à priori* knowledge of the noise and the blurring process are needed. As mentioned in Chapter 2, some existing methods could give a good estimation of the noise variance, effort in this Chapter will be focused in the identification of the PSF.

The blurred image is actually the convolution of the original image and the PSF. It is difficult to separate the two convolved signals without any *à priori* information about either one of them. Therefore, certain assumptions regarding the original signal must be made in a blur identification algorithm, and they usually also include, the PSF, and/or the noise characteristics. Some algorithms assume a particular model for the PSF while others assume the characteristics of the

signals themselves. Furthermore, the algorithms may incorporate partial information about the signal and/or the PSF. The more *à priori* information that is incorporated into the algorithm, the better will be the fidelity of the estimate of the original signal to the actual one [22]. A list of assumptions, models, and partial knowledge that are frequently used to aid in identifying and separating two unknown convolved signals, the image and the PSF [22]:

- Image or/and PSF is/are non-negative.
- Spatial extent of image is much greater than that of the PSF.
- PSF conserves mean value of the original signal.
- Form of blur is known (e.g. motion blur).
- Power spectrum estimate of image is known.
- The image and/or PSF have/has finite extent.

The assumptions that are made on the data as well as the *à priori* information required by the algorithm will determine the applicability of a particular technique to a given set of data. Clearly, if the required partial information for the data is not available, then the technique that requires it will not be useful. Similarly, if the assumptions made about the data do not conform well to the physical situation, the technique is less likely to be successful in the deconvolution of the two signals. Thus, it is desirable to have techniques, which require fewer assumptions about the data and little partial information about the original signal and/or the PSF [22]. Such techniques will find broader applicability to real-life blur identification problems.

6.2. Analysis of the Blur Effect in the Derivative Image

6.2.1. The Derivative of Blurred Image

In this section, the Derivative Image of the blur image will be studied and analyzed. Support is the region of a function in which it is defined. In fact, the support of the PSF is usually finite, *i.e.,*

the PSF of the blurring process usually only has a limited support. The support of PSF $d(x, y)$ was defined and denoted as $\wp_d$ in Chapter 2, which is $(x,y) \in \wp_d = ([-b,b],[-b,b])$, as shown in Figure 6.1.

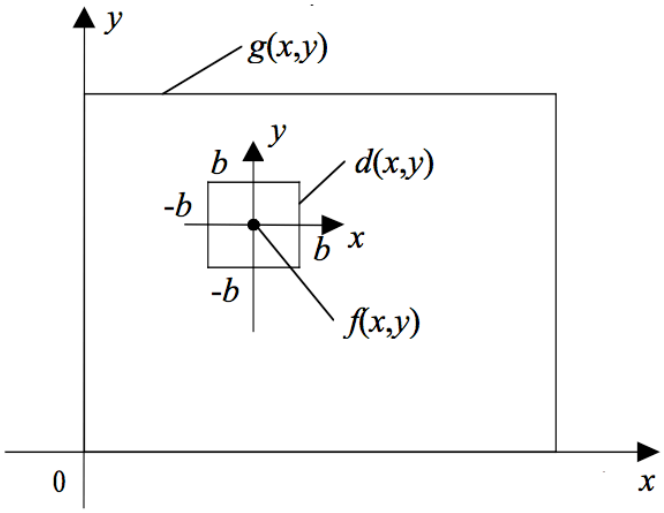


Figure 6.1 The Definition of PSF and the Blurred Image

Suppose that there is a point $f(x,y)$ in the original image plane x-y, and then the corresponding point of the blurred image is $g(x,y)$, as shown in Figure 6.1. This point is given by:

$$g(x,y) = \iint_{\wp_d} f(\alpha,\beta)\, d(x-\alpha, y-\beta)\, d\alpha\, d\beta + e_n(x,y) \quad \{(x,y) \in \wp\}$$

$$= \int_{x-b}^{x+b} \int_{y-b}^{y+b} f(\alpha,\beta)\, d(x-\alpha, y-\beta)\, d\alpha\, d\beta + e_n(x,y) \tag{6.1}$$

e_n is the additive noise term. Suppose that the blur is caused by the motion in the y-direction only. The formulation is similar for the motion blur alone x-direction. The 2-D PSF could be simplified as a 1-D PSF in this particular case:

$$d(x,y) = \begin{cases} d(y) & x = 0 \quad and \quad y \in [-b,b] \\ 0 & x \neq 0 \end{cases} \tag{6.2}$$

Furthermore some necessary constraints of the PSF are needed:

$$d(-b) = d(b) \tag{6.3}$$

$$\int d(y)dy = 1 \tag{6.4}$$

Eq. (6.3) indicates that the PSF is symmetrical at the boundary of its support, and PSF satisfying Eq. (6.4) fulfils the energy conservation condition.

After the simplification, the previous Eq. (6.1) can be expressed as:

$$g(x,y) = \int_{y-b}^{y+b} f(x,\beta)\ d(y-\beta)\ d\alpha + e_n(x,y) \tag{6.5}$$

For simplicity in representation, $g(x,y)$ is denoted as $g(y)$, $f(x,\beta)$ as $f(\beta)$, and $en(x,y)$ as $en(y)$. The derivative of $g(y)$ with respect to y is given as:

$$\frac{\partial g(y)}{\partial y} = \frac{\partial}{\partial y}\left(\int_{y-b}^{y+b} f(\beta)\ d(y-\beta)\ d\beta + e_n(y)\right)$$

$$= \int_{y-b}^{y+b} f(\beta)\ d'(y-\beta)\ d\beta + f(y+b)\ d(-b) - f(y-b)\ d(b) + \frac{\partial e_n(y)}{\partial y} \tag{6.6}$$

Here $\partial g(y)/\partial y$ is the derivative of the blurred image. The derivative of the blurred image will provide the information on the PSF, such as the extent and parameter of the PSF; however, the derivative of the image should be considered separately within two different regions: edge region and smooth region.

6.2.2. The Edge and Smooth Regions in the Derivative Image

Basically, there are two kinds of possible regions in an image. First is the edge-region, in which the gray level increases or decreases abruptly over a small distance. Second is the smooth-region, in which the gray level changes very slowly. The definition of the two regions is given following. Let us assume that there is a sharp edge in the original image $f(y)$ located at y'. In addition, $f(y)$, y

$\neq y$' changes very slowly such that the gray levels values could be regarded as a constant in a small region $[y'-a, y'+a]$ in the neighborhood of y', where a>0. The edge region could be modeled by [112]:

$$\begin{cases} E[f(y)] = A; \;\; E[f(y) - A]^2 = \sigma_n^2; \;\; (y' \le y \le y'+a) \\ E[f(y)] = B; \;\; E[f(y) - B]^2 = \sigma_n^2; \;\; (y'-a \le y < y') \\ A > B; a > b; \end{cases} \tag{6.7}$$

where $E[\bullet]$ is the expectation; $f(y)$ is the original image in Eq. (6.1); A and B are two different gray levels of the image pixels. A>B means the edge is ascending. $[y'-a, y'+a]$ is the edge-region support (Note that [-b,b]is the support of the PSF in Eq. (6.2)). σ_n^2 is the estimated variance of the noise. Only the ascending sharp edge will be analyzed, which is modeled by Eq. (6.7). The analysis of a descending sharp edge is similar. In this edge region, two things need to be noted. Firstly, when $a > b$, the image $f(y)$ could be treated as a constant in the support of the PSF. Secondly, the edge of the practical image may not be so sharp that it can jump to a higher gray level in an infinitesimal small region. The model presented in this dissertation is only an ideal model. The differences between this ideal model and the practical situation will be discussed in the section on experimental results.

After setting the model of an edge-region, a smooth-region is defined. In a small image region $[y-a, y+a]$, if $f(y)$ satisfies:

$$\begin{cases} E[f(y)] = C; \;\; E[f(y) - C] = \sigma_n^2; \;\; (y'-a \le y \le y'+a) \\ a > b; \end{cases} \tag{6.8}$$

Then this region is defined as a smooth-region with $[y-a, y+a]$ as its support. Again, in this case $a > b$, Eq. (6.8) implies that in the small region $y-a < y < y+a$, the gray level values of the pixels could be treated as a constant statistically. The changes of the gray level in the region

are only caused by the noise. Here the supports of the edge-region and the smooth-region are much smaller than the image size. The overlap area of the two regions is negligible.

Next, let's analyze the effect of blurring process on the two different regions. First, we look at the effect of the blurring process on the edge-region. Substitute Eq. (6.7) into Eq. (2.3) and then determine the expectation of Eq. (6.7) in the region of [-*a*,*a*][112]:

$$E\left(\frac{\partial g(y)}{\partial y}\right)=\begin{cases}0 & y\in[y'-a,y'-b)\\ (A-B)\ d(y-y') & y\in[y'-b,y'+b]\\ 0 & y\in(y'+b,y+a]\end{cases} \tag{6.9}$$

Here, it is obvious that the expectation of the derivative of the image along the direction of motion could reflect the characteristics of the PSF. The support of the PSF could be determined by the extension of the non-zero region of $E(\partial g(y)/\partial y)$. Furthermore, the PSF is only multiplied by a constant (*A*-*B*). Considering the second constraint of PSF, stated in Eq. (6.4), the influence of the constant could be eliminated.

Secondly, the behavior of the smooth-region in blurring process is different. Substitute Eq. (6.8) into Eq. (2.3) and considering together with Eq. (6.7):

$$E(\frac{\partial g(y)}{\partial y})=0 \tag{6.10}$$

Furthermore, it is easy to prove that if the PSF does not satisfy Eq. (6.3), the expectation of the derivative in the smooth-region will only be shifted from zero to a new constant, which is dictated by the boundary value of $d(y)$.

From the above analysis, the conclusion is that the blurring process will not change the derivative of the image along the direction, of which the blurring occurs in a smooth-region. Near a region of a sharp edge, the derivative will reflect the characteristics of the blurring process, *i.e.*, the extent and the parameters of PSF.

6.3. Blur Identification in Spatial Domain

6.3.1. Extraction of the PSF Extent

After having obtained the expectation of the derivative of the blurred image, a shadow image $L(x,y)$ is defined[112]:

$$L(x,y)=\begin{cases} 1 & \dfrac{\partial g(y)}{\partial y} > T \\ 0 & -T < \dfrac{\partial g(y)}{\partial y} < T \\ -1 & \dfrac{\partial g(y)}{\partial y} < -T \end{cases} \qquad -\infty < x,y < +\infty \tag{6.11}$$

where T is a positive threshold. For simplicity and based on the assumption that the blurring is due to the motion in the y-direction, $L(x,y)$ can be simplified as $L(y)$. In fact, the shadow image is a continuous black and white image. The magnitude of the edges as a random variable, which is assumed to be a zero-mean homogeneous Gaussian distributed random variable. Furthermore, the *à priori* probability of an ascending sharp edge to be approximately equal to the *à priori* probability of descending sharp edge.

The threshold is applied to extract the useful information and reduce the influence of the noise. From Eq. (6.11), more information can be extracted from the derivative image with lower threshold. However, to reduce the noise effect, T should not be equal to zero. Selection of a good value for the threshold will be discussed in the next section.

The identification parameter ℓ_y therefore can be defined. If a region of the shadow image $L(x,y)$ satisfies:

$$\begin{cases} L(x,[y_1 \quad y_2]) \neq 0 \\ L(x, \lim\limits_{\Delta y \to 0}(y_1 - \Delta y)) = 0 \\ L(x, \lim\limits_{\Delta y \to 0}(y_2 + \Delta y)) = 0 \end{cases} \tag{6.12}$$

The parameter ℓ_y is defined by:

$$\ell_y = sign\ \{L(x, y \in (y_1 \quad y_2))\} \bullet \left| \ y_2 - y_1 \ \right| \tag{6.13}$$

The function $sign(\bullet)$ is to obtain the sign of the terms within the enclosing curly brackets. All the regions which satisfy Eq. (6.12), is defined as the ***sub-space*** with the parameter ℓ_y:

$$L\{x,[y_1 \quad y_2]\} \in S(x, y \mid \ell_y) \tag{6.14}$$

The sub-space is part of the shadow image. The shadow image is divided according to the different values of ℓ_y. As analyzed in the last section, the derivative of the blurred image along the direction of motion will characterize the PSF. According to Eq. (6.9) and (6.10), the area of sub-spaces with the parameter $\pm\ell_y$ which is equal to the extent of the PSF is expected to dominate other sub-spaces. Here the sign of the parameter can be disregarded since it only indicates the ascent or descent edges. The area of the subspace with the parameter $\left|\ell_y\right|$ is calculated. It is defined by an integral function shown in Eq. (6.15) as follow [112]:

$$\hbar(\left| \ \lambda_y \ \right|) = \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty}\left|S(x, y \mid \left| \ \lambda_y \ \right|)\right| dxdy \qquad \lambda_y \in (-\infty, +\infty) \tag{6.15}$$

The area of the sub-space $\hbar(\ell_y)$ is assumed a zero-mean Gaussian random variable. Then the probability density function (PDF) of this random variable $\hbar(\ell_y)$ is given by [22]:

$$p(\ell_y) = \frac{\hbar(\ell_y)}{\int_{-\infty}^{+\infty} \hbar(\ell_y)\, d\ell_y} \tag{6.16}$$

Furthermore, the PDF of $\hbar(\ell_y)$ is treated as a zero-mean Gaussian distribution. The integral function implies the absolute area of the sub-space with respect to the parameter ℓ_y.

It is well known that most of the images are degraded by noise. If all the edges in the image are sharp, in other word, they satisfy Eq. (6.7), the probability density function in Eq. (6.16) is assumed to be a zero-mean Gaussian distribution; otherwise this assumption may not be valid. If there is an image without blur, and substituting $|\ell_y|$ into Eq. (6.16), $p(|\ell_y|)$ is expected to be monotonically decreasing. As discussed above, the blurring process will increase the probability at ℓ_y, which is equal to the extent of the PSF. In other words, the blurring process will damage the monotonically decreasing characteristics of $p(|\ell_y|)$. The blurring extent could be identified by locating the exclusive peak of $p(|\ell_y|)$, which is located at $\ell = 2b$, theoretically.

Some images do not strictly satisfy Eq. (6.7) becasue:

(a) Some edges in the images may not be so sharp that the gray level of the pixels at the edge would jump to a higher level in an infinitesimal small region.

(b) The smooth region near the edges may be smaller than the support of the PSF.

The performance of the algorithms of these two situations will be discussed in the section on experimental results.

6.3.2. Parameter Identification

As shown by the analysis in the previous section, the blurring extent can be identified by locating the exclusive peak in $p(|\ell_y|)$. In some applications, prior knowledge about the PSF is not available. In this situation, not only we want to know the extent of the PSF, but we are also interested in what is the type of the PSF. During the identification process, the threshold of the shadow image is an important parameter. When the blur is caused by linear motion, a small difference in the value of the thresholds will not influence the identification result. But, this would not be the case when the PSF has a shape similar to a parabola. If the value of the threshold is too small, too much noise would be included in the shadow image, which would

make the identification difficult. On the other hand, the value of threshold should not be too high, as it will truncate the extent of the PSF.

In practice, the threshold is selected dynamically. To determine the threshold, a meaningful range within which the desired threshold may be found is given. The range is divided into equally spaced values. For each value, the distribution function $p(\lambda_y)$ given in Eq. (6.16) is calculated. The peak value is then identified. When $p(\lambda_y)$ is determined for all values of the threshold within the said range, the minimum values among these highest peaks are identified. The value of threshold at which the peak corresponds to is chosen to be the final threshold for the identification of the extent of the PSF. The effect of the threshold will be analyzed in the section 6.4.2 of the experimental results.

After identified the blur extent of the PSF by locating the exclusive peak in $p(|\lambda_y|)$ at $\lambda_y = 2\hat{d}$, the sub-space with parameter $\lambda_y = 2\hat{d}$ is denoted as $S(x, y \mid 2\hat{d})$. In fact, in all of the sub-spaces with parameter λ_y, $S(x, y \mid 2\hat{d})$ contains the most information on the PSF. The shape of the *PSF* is given by [112]:

$$\hat{d}'(y) = \frac{\iint_{(x,y)\in S(x,y|2\hat{d})} \left|\frac{\partial g(x,y)}{\partial y}\right| dx\, dy}{\int_{(x,y)\in S(x,y|2\hat{d})} |L(x,y)|\, dx} \tag{6.17}$$

After having obtained the initial estimation $\hat{d}'(y)$, it is normalized to obtain the final estimation of the PSF $\hat{d}$. The normalization is only to make the PSF satisfy Eq. (6.4). As no assumption is made about the form of the PSF, thus this method could be applied to the estimation of the PSF of any shape.

In practice, once a blurred image is obtained, the blur direction is known which is commonly along one of the coordinate axes.

1. Under this assumption, calculate the derivative of the blurred image according to Eq. (6.6).

2. Based on the earlier discussion on thresholding presented earlier, determine the shadow image according to Eq. (6.11).

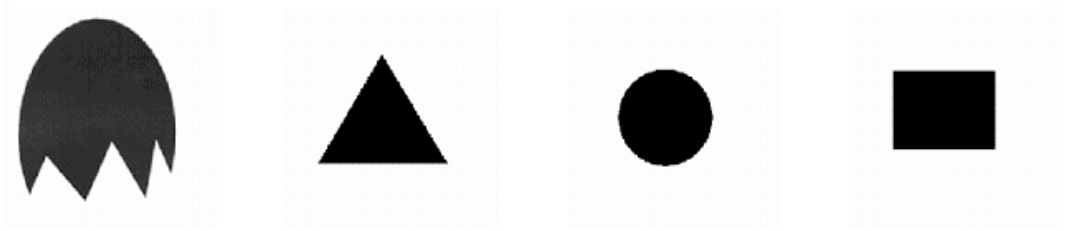
3. $p(\lambda_y)$ is then calculated to identify the extent of the PSF.

4. Finally, apply Eq. (6.17) to identify the parameter of the PSF.

6.4. Experimental Results

6.4.1. Experimental Images and PSFs

(a) Lena (b). Tools (c). Cameraman (d)Non-Smoking



(e) EggShell (f). Triangle (g). Circle (h)Rectangle

Figure 6.2 Images in the Experiments

In order to test the performance of the algorithm proposed in this chapter, it is applied to some blurred images in this section. The images for the experimental section are shown in Figure 6.2.

Some of the test images satisfy the assumption that the edges are sharp, which is indicated by the edge model of Eq. (6.7). It requires the edges to be ascending or descending within one or two pixel interval. The $p(|\ell_y|)$'s of these images are monotonically decreasing, such as "Lena", "Tools" and "Camera-man" as shown in Figure 6.3.

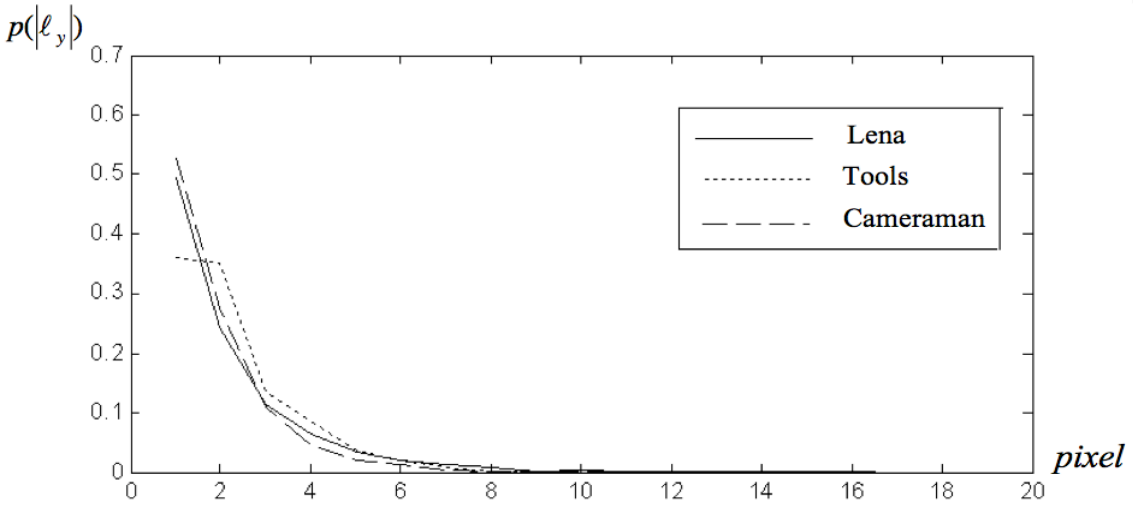


Figure 6.3 Monotonically Decreasing $p|\ell_y|$ of the Test Images (SNR>60)

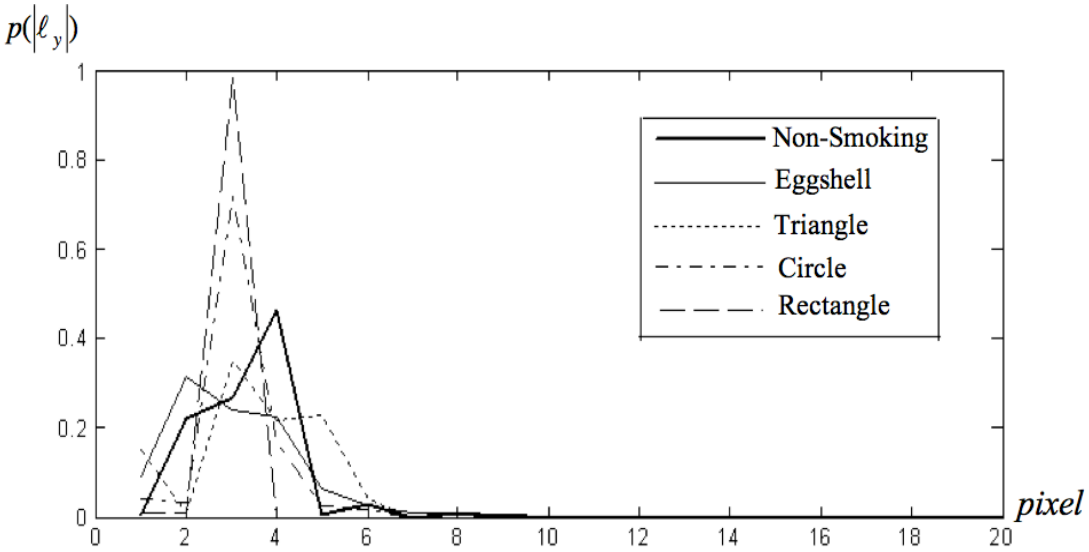


Figure 6.4 Non-monotonically Decreasing $p|\ell_y|$ of the Testing Images (SNR>60)

Some test images do not strictly satisfy Eq. (6.7). The non-monotonic $p(|\ell_y|)$'s of the five images are shown in Figure 6.4. They are Non-Smoking sign, Eggshell, Triangle Circle and Rectangle. It is obvious that there are peaks within peaks over 2- 4 pixels width along the ℓ_y axis. This is because the width of the edge is not small, *i.e.*, the edge is not sharp enough.

As this method is based on statistical principles, $p(|\ell_y|)$ should be monotonically decreasing at the lower SNR, Those images are Lena, Tools and Cameraman, which are show in Figure 6.5.

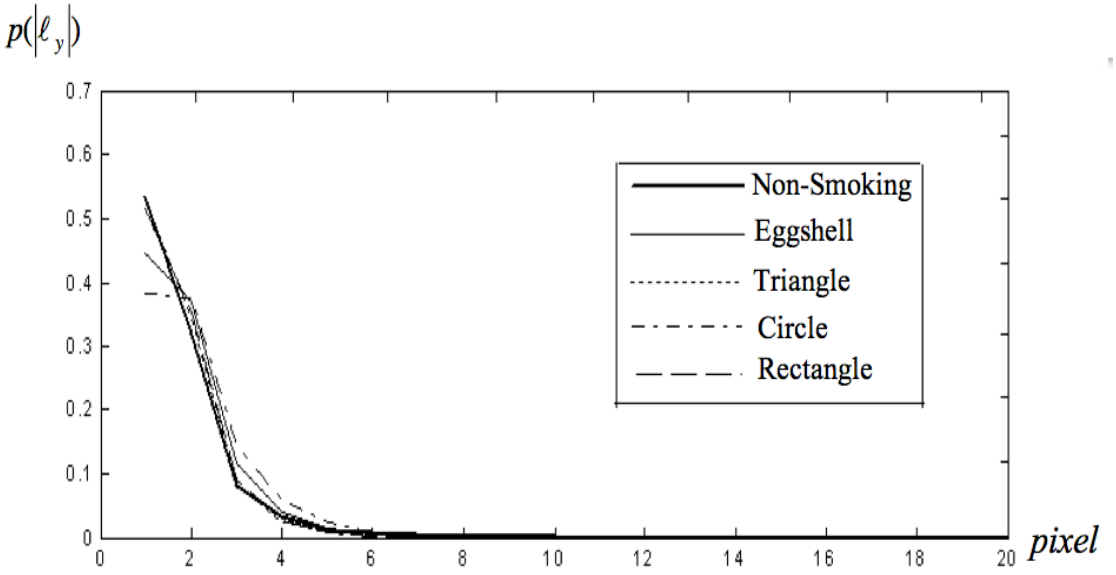
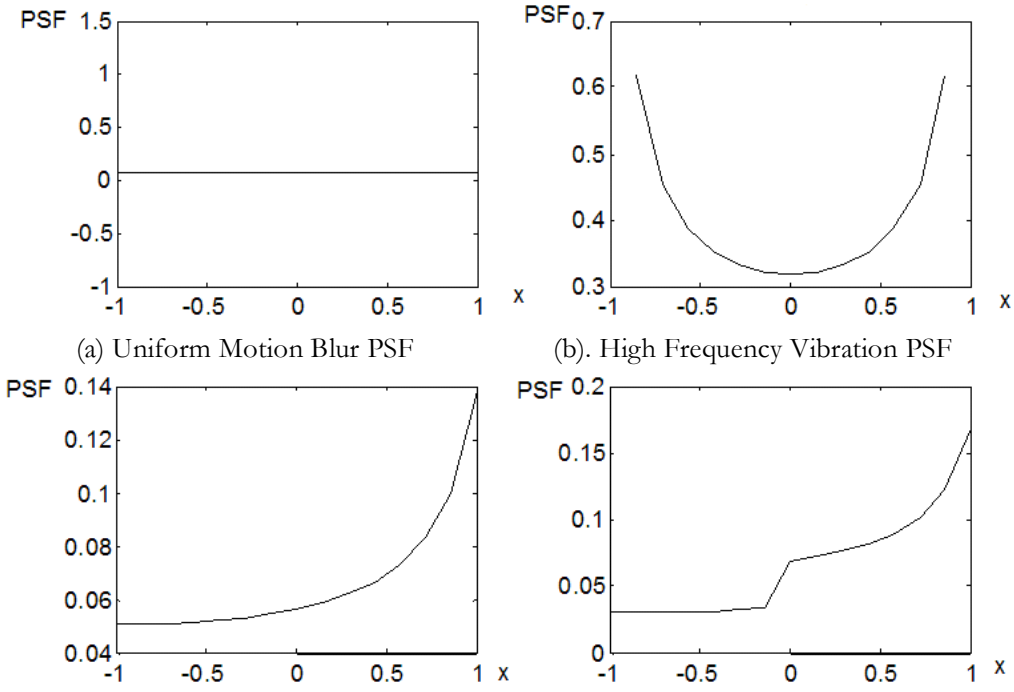


Figure 6.5 Monotonically Decreasing $p(|\ell_y|)$ of the Testing Images (SNR≥25)

6.4.2. Identification of the PSF Extent

In carrying out the experiment, the original test images are blurred by four pre-selected PSFs and the algorithm of PSF extent identification is applied as described in this chapter in Section 6.3. The four PSFs are shown in Figure 6.6, which are common in the industrial applications. They are the Uniform Motion Blur PSF, High Frequency Vibration PSF, Continuous Low Frequency Vibration PSF and Discontinuous Low Frequency Vibration PSF.



(a) Uniform Motion Blur PSF (b). High Frequency Vibration PSF

(c) Continuous Low Frequency Vibration PSF (d) Continuous Low Frequency Vibration PSF

Figure 6.6 PSFs in the Experiments.

In order to analyze the effect of the threshold T in the identification process, a 3-D plot of $p(|\ell_y|)$, threshold T and ℓ_y, commonly known as waterfalls, is shown in Figure 6.7. This waterfall is the result of the identification of the test image, Lena shown in Figure 6.2. The SNR value is 50, and the PSF used to blur the image is the discontinuous low frequency PSF (Figure 6.6 (d)), and the extent of this PSF is 15 pixels. Different values of thresholds T are chosen, from 0 to 40%. From Figure 6.7, it is obvious that when the threshold T increases, the exclusive local peaks shift to the left. This is because the high thresholds will truncate the extent of the PSF. When the threshold T is higher than 25% this method is not valid for this image any more.

The waterfalls of some of the blurred images are analyzed and most of the optimized thresholds T of the shadow image of those images fall in the region from 10% to 25% of the maximum absolute value of derivative images, in Eq. (6.6). In fact, every blur identification could produce a waterfall. But the calculation of the waterfall is computational intensive. Thus, instead of

calculating the waterfall of the blurred image across the range of threshold from 0% to 40%, we look for the threshold T in a narrower range from 10% to 25% as described in the last section.

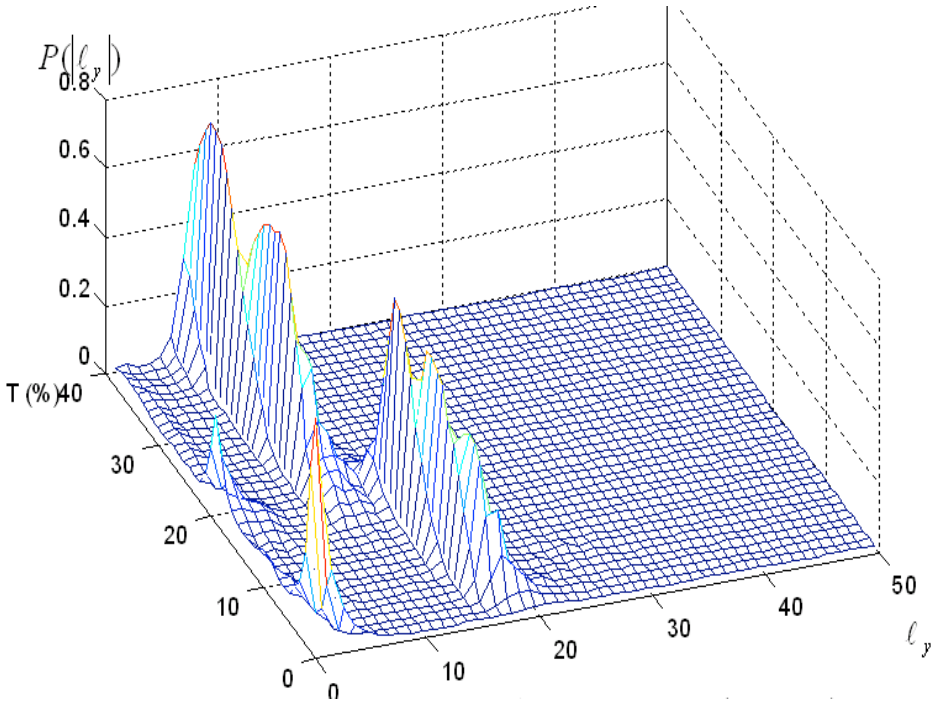


Figure 6.7 Waterfall of Identification Result of PSF Extent (SNR=50).

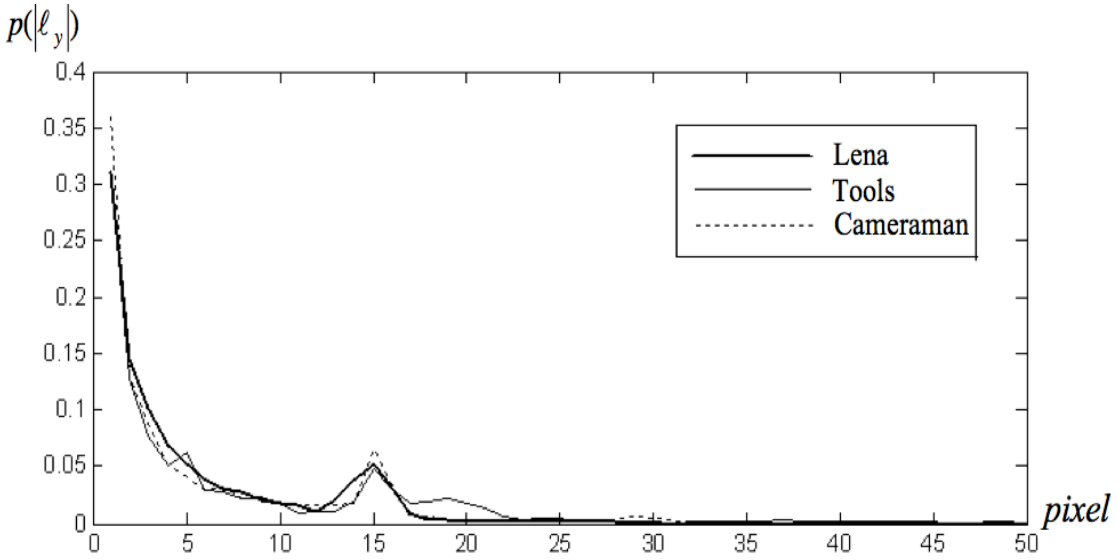


Figure 6.8 PSF Extent Identification Results of Blurred Image from $p|\ell_y|$ (SNR=40, T for Different Images is Indicated in the Corresponding Tables. PSF is the Uniform Linear Motion Blur. $p|\ell_y|$s of the Original Images are Monotonically Decreasing, as Shown in Figure 6.3.)

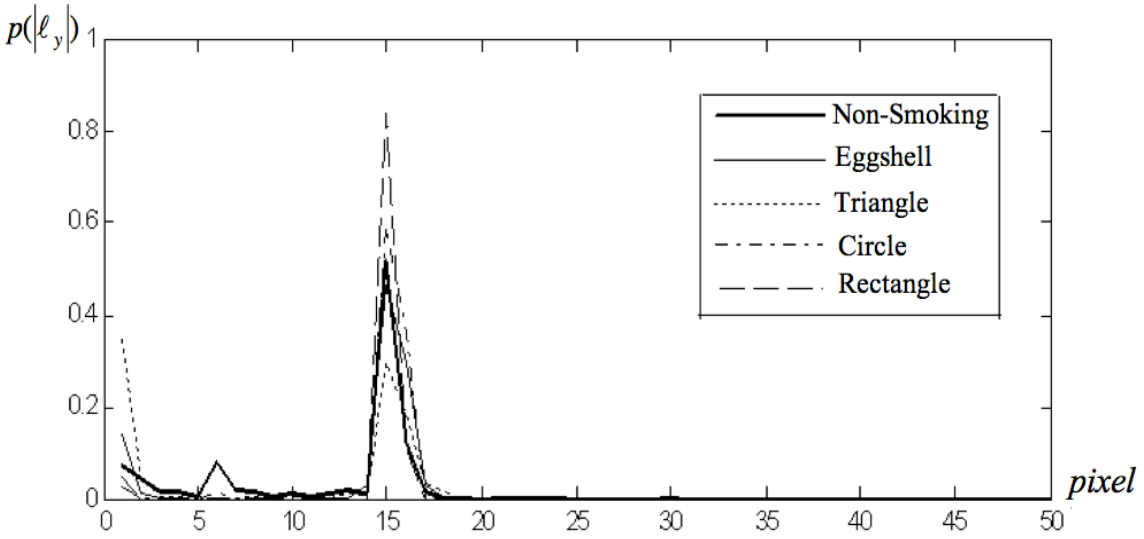


Figure 6.9 PSF Extent Identification Results of Blurred Image from $p|\ell_y|$ (SNR=40, T for Different Images is Indicated in the Corresponding Tables. PSF is the Uniform Linear Motion Blur. $p|\ell_y|$s of the Original Images are Non-monotonically Decreasing, as shown in Figure 6.4.)

Some identification results of blur extents (SNR = 40) are shown in Figure 6.8 and Figure 6.9. From Figure 6.8, the blurring process will change the shape of $p(|\ell_y|)$ if compared with Figure 6.3. As analyzed in the last section, locating the exclusive peaks in $p(|\ell_y|)$ could identify the blurring extent, which is $|\ell_y|=15$. From Figure 6.9, there are still some very small peaks at which the value of $|\ell_y|$ is low, but they can be ignored when compared with the high peaks, which indicate the extent of the PSF.

The effects of noise on the identification of the extent of blur should be considered. The most important aspect of this study is to determine at what value of SNR this method cannot work anymore. The results are listed in Table 6-1 to 6-4. The value in bracket indicates the dynamically selected threshold value.

From Tables 6-1 to 6-4, the proposed method could identify the extent of the PSF without any error for the given eight images when SNR$\geq$30. When the SNR is 20db, the error is about 2-3 pixels. The proposed approach has good tolerance to noise and accurate in the identification of PSFs' extent when the noise level is high.

TABLE 6-1 BLUR EXTENT IDENTIFICATION OF BLURRED IMAGES IN SPATIAL DOMAIN (UNIFORM MOTION BLURRING - FIGURE 5.6(A))

SNR	Lena	Tools	Cameraman	Nonsmoking	Egg	Triangle	Circle	Rectangle
60 *db*	15(0.22)	15(0.12)	15(0.12)	15(0.12)	15(0.13)	15(0.10)	15(0.09)	15(0.10)
50 *db*	15(0.23)	15(0.11)	15(0.14)	15(0.11)	15(0.14)	15(0.11)	15(0.11)	15(0.09)
40 *db*	15(0.23)	15(0.13)	15(0.22)	15(0.14)	15(0.13)	15(0.14)	15(0.14)	15(0.12)
30 *db*	15(0.34)	-	-	15(0.20)	14(0.20)	13(0.18)	13(0.18)	14(0.22)
20 *db*	15(0.20)	-	-	-	-	-	-	-

TABLE 6-2 BLUR EXTENT IDENTIFICATION OF BLURRED IMAGES IN SPATIAL DOMAIN (HIGH FREQUENCY BLURRING - FIGURE 5.6(B))

SNR	Lena	Tools	Cameraman	Nonsmoking	Egg	Triangle	Circle	Rectangle
60 *db*	15(0.20)	15(0.18)	15(0.20)	15(0.15)	15(0.12)	15(0.12)	15(0.16)	15(0.18)
50 *db*	15(0.25)	15(0.18)	15(0.20)	15(0.14)	15(0.15)	15(0.12)	15(0.15)	15(0.17)
40 *db*	15(0.26)	15(0.19)	15(0.22)	15(0.16)	15(0.16)	15(0.13)	15(0.18)	15(0.19)
30 *db*	15(0.28)	15(0.20)	-	-	-	-	14(0.22)	-
20 *db*	-	-	-	-	-	-	-	-

TABLE 6-3 BLUR EXTENT IDENTIFICATION OF BLURRED IMAGES IN SPATIAL DOMAIN (CONTINUOUS LOW FREQUENCY BLURRING - FIGURE 5.6(C))

SNR	Lena	Tools	Cameraman	Nonsmoking	Egg	Triangle	Circle	Rectangle
60 *db*	15(0.15)	15(0.18)	15(0.20)	15(0.12)	15(0.16)	15(0.16)	15(0.16)	15(0.14)
50 *db*	15(0.16)	15(0.20)	15(0.22)	15(0.10)	15(0.17)	15(0.15)	15(0.15)	15(0.12)
40 *db*	15(0.18)	15(0.18)	15(0.25)	15(0.14)	15(0.20)	15(0.17)	15(0.17)	15(0.16)
30 *db*	15(0.20)	-	13(0.22)	15(0.20)	11(0.22)	-	12(0.20)	16(0.17)
20 *db*	-	-	-	-	-	-	-	-

TABLE 6-4 BLUR EXTENT IDENTIFICATION OF BLURRED IMAGES IN SPATIAL DOMAIN (DISCONTINUOUS LOW FREQUENCY BLURRING - FIGURE 5.6(D))

SNR	Lena	Tools	Cameraman	Nonsmoking	Egg	Triangle	Circle	Rectangle
60 *db*	15(0.10)	-	15(0.08)	15(0.08)	15(0.14)	15(0.14)	15(0.15)	15(0.14)
50 *db*	15(0.11)	-	15(0.09)	15(0.08)	15(0.12)	15(0.15)	15(0.15)	15(0.15)
40 *db*	9(0.15)	-	15(0.25)	15(0.14)	15(0.20)	15(0.17)	15(0.17)	15(0.16)
30 *db*	11(0.16)	-	16(0.10)	-	17(0.17)	13(0.20)	15(0.18)	15(0.20)
20 *db*	-	-	-	-	-	-	-	-

-: means cannot find any exclusive peaks.

6.4.3. Parameter Extraction of the Experimental Images

As discussed in the preceding sections, after the extent of the PFS is identified, the parameters of the PSF would be extracted. Here a criterion P is defined to describe the accuracy of the

parameter extraction:

$$P = \sqrt{\frac{\sum_{1}^{N} [\hat{d}(i) - d(i)]^2}{E}} \tag{6.18}$$

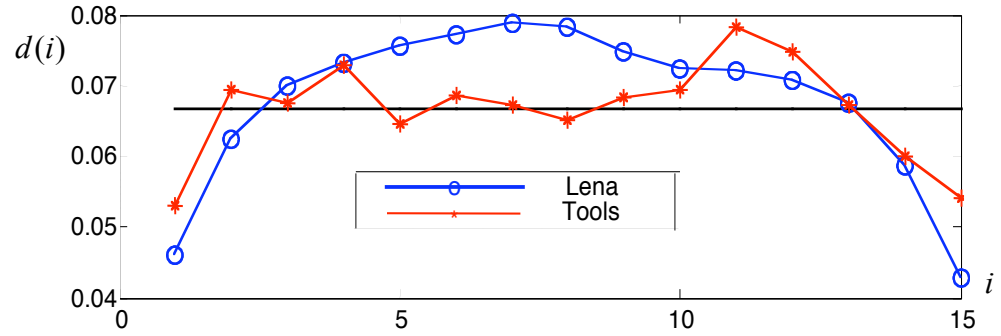
where $d(i)$ is the parameter of the known PSF; $\hat{d}(i)$ is the parameter of estimated PSF; and E is the extent the PSF (in pixel). In effect, P is actually a similarity measurement of d and $\hat{d}$.

The mean of the criterion P is the Root Mean Square Error (RMSE). If the extracted parameters are equal to the ideal parameters, P should be zero. P will increase with the decrease of accuracy. The results of parameter extraction with SNR equal to 40 are presented in Figure 6.9(a)-(d). From these figures, the proposed approach of PSF parameter identification is accurate for those four common PSFs used. Note that all the identification results have a boundary effect. There are two possible reasons:

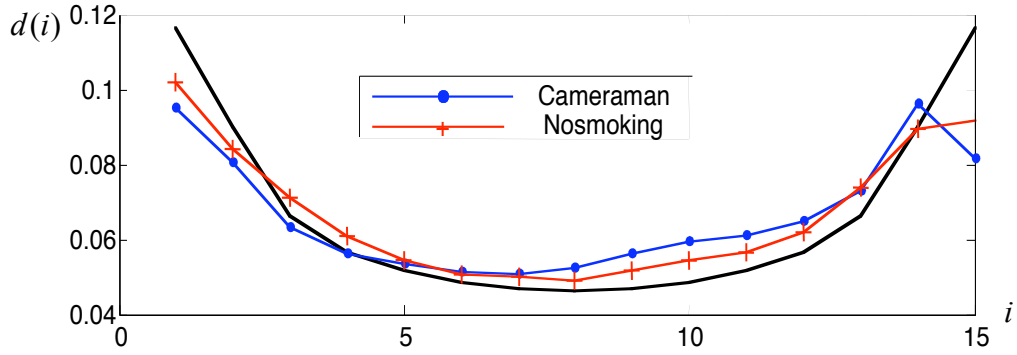
1. Information may not be sufficient to provide a good estimation of those points since they are located near the ends.

2. In Section 6.2.2, it was assumed that the supports of the edge-region and the smooth-region are much smaller than the image size. Furthermore, it is also assumed that $a>b$ in Eq. (6.8). If this assumption cannot be satisfied everywhere, it will also introduce some error at the boundary.

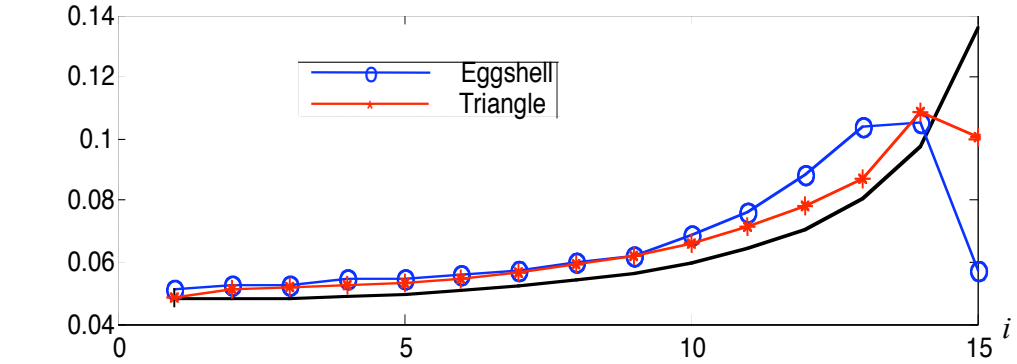
The proposed approach is verified at different values of SNR. The results are shown in Table 6-6 to 6-8. The similarity measurement of the true PSF and the estimated PSF is small when the SNR is high. Of course, when the SNR decreases, P will also increase, which means the difference between the true PSF and the estimated PSF becomes bigger. But note that the increase of P is much slower, for example, for Lena in Table 6-5, the SNR decreases from 60 *db* to 40 *db*, but P remains at the same value.



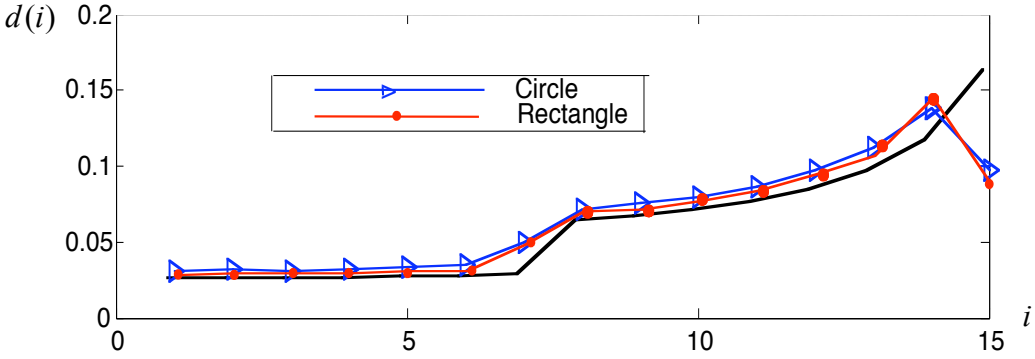
(a) Uniform Linear Motion PSF



(b) High Frequency Vibration PSF



(c) Continuous Low Frequency Vibration PSF



(d) Discontinuous Low Frequency Vibration PSF

Figure 6.10 Estimation of the PSF Coefficients in Spatial Domain

From these tables, when SNR = 40db, the results of extraction are accurate. When SNR < 30db, the blurred image would contain too much noise that the method is not able to extract useful information about the blurring process.

TABLE 6-5 PARAMETER EXTRACTION FROM BLURRED IMAGES IN SPATIAL DOMAIN (UNIFORM MOTION BLURRING - FIGURE 5.6(A))

SNR	Lena	Tools	Cameraman	Nonsmoking	Egg	Triangle	Circle	Rectangle
60 *db*	0.0094	0.0087	0.0091	0.0039	0.0087	0.0036	0.0075	0.0078
50 *db*	0.0094	0.0094	0.0080	0.0042	0.0087	0.0038	0.0074	0.0078
40 *db*	0.0094	0.0090	0.0075	0.0039	0.0075	0.0049	0.0068	0.0077
30 *db*	0.0124	-	-	0.0230	*	*	*	*
20 *db*	0.0099	-	-	-	-	-	-	-

TABLE 6-6 PARAMETER EXTRACTION FROM BLURRED IMAGES IN SPATIAL DOMAIN (HIGH FREQUENCY BLURRING - FIGURE 5.6(B))

SNR	Lena	Tools	Cameraman	Nonsmoking	Egg	Triangle	Circle	Rectangle
60 *db*	0.0150	0.0228	0.0181	0.0084	0.0165	0.0075	0.0136	0.0149
50 *db*	0.0257	0.0229	0.0193	0.0084	0.0166	0.0076	0.0138	0.0085
40 *db*	0.0256	0.0224	0.0110	0.0083	0.0160	0.0104	0.0118	0.0130
30 *db*	0.0272	0.0495	-	-	-	-	*	-
20 *db*	-	-	-	-	-	-	-	-

TABLE 6-7 PARAMETER EXTRACTION FROM BLURRED IMAGES IN SPATIAL DOMAIN (CONTINUOUS LOW FREQUENCY BLURRING - FIGURE 5.6(C))

SNR	Lena	Tools	Cameraman	Nonsmoking	Egg	Triangle	Circle	Rectangle
60 *db*	0.0235	0.0206	0.0159	0.0106	0.0233	0.0098	0.0160	0.0172
50 *db*	0.0232	0.0248	0.0185	0.0103	0.0229	0.0098	0.0162	0.0172
40 *db*	0.0223	0.0219	0.0139	0.0110	0.0208	0.0102	0.0145	0.0171
30 *db*	0.0299	-	*	0.0233	*	-	*	*
20 *db*	-	-	-	-	-	-	-	-

TABLE 6-8 PARAMETER EXTRACTION FROM BLURRED IMAGES IN SPATIAL DOMAIN (DISCONTINUOUS LOW FREQUENCY BLURRING - FIGURE 5.6(D))

SNR	Lena	Tools	Cameraman	Nonsmoking	Egg	Triangle	Circle	Rectangle
60 *db*	0.0297	-	0.0309	0.2448	0.0310	0.0130	0.0203	0.0213
50 *db*	0.0286	-	0.0324	0.1972	0.0318	0.0131	0.0203	0.0213
40 *db*	*	-	0.0286	0.1599	0.0305	0.0189	0.0218	0.0236
30 *db*	*	-	*	-	*	*	0.0935	0.1133
20 *db*	-	-	-	-	-	-	-	-

-: means can not find any exclusive peaks to identify the extent of the PSF

*: means can find the exclusive peaks to identify the extent of the PSF, but the result is not accurate.

6.4. Summary

In this Chapter, a spatial statistical blur identification approach is proposed to obtain the information on the PSF, such as the extent and the parameters of the PSF. This approach is also a statistical treatment for blur identification. Based on the edge region model of the image and the analysis in the previous sections, an algorithm to identify the PSF of a blurred image is developed. The results of the experiments with different images at different SNRs and under four common PSFs demonstrate that this proposed method is efficient and not sensitive to noise to a great extent. There are several issues that are worthwhile to discuss.

A major issue in the method proposed in this work is that the threshold of the shadow image may be of different values under different conditions (such as when the image is blurred by different PSFs). If the PSF is a linear motion blur, the values will not affect the identification of the extent of the blur. But if the shape of the PSF is similar to a parabola, the value of the threshold should be as low as possible. This is because too high a threshold value would truncate the extent. Paradoxically, when the threshold is very low, the shadow image would contain too much noise. The local exclusive peaks will be inundated by the noise. From the experiments, it is found that the optimized value of the threshold may be different for different images. The choice of the value of threshold is an important step in the method. In the experiments, the threshold value is determined dynamically. However, this may not be the best method to find the optimum threshold.

Secondly, the errors incurred in the extracted parameters and the boundary effects are due to two reasons.

- The assumption $a > b$ is not always true for some edge-regions in practice. This implies that the change of the gray level within the edge regions, other than the gray level changes of the

edges, will contaminate the extracted parameters. However, results of the experiments show that this assumption is acceptable.

- The effect of noise will influence the parameter extraction. The presented algorithm does not seem to be sensitive to noise, even when SNR is higher than 30 *db*, the method of parameter extraction is effective.

Chapter 7

Compact Discrete Polar Transform for Rotational Blurred Images

7.1. Introduction

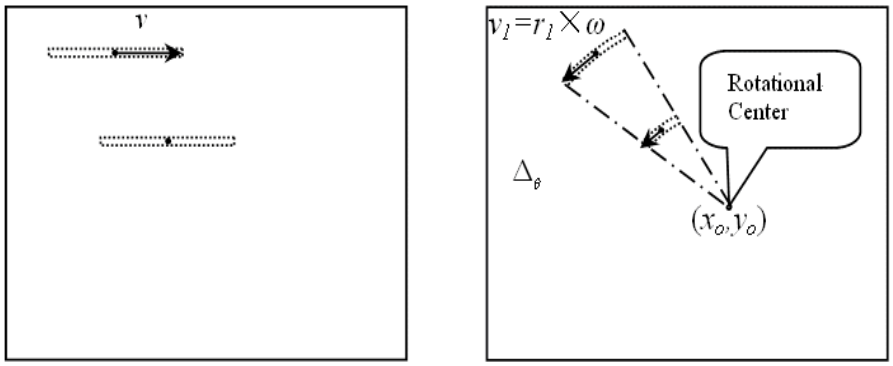
The restoration of an image which has been blurred due to the rotational motion of the camera is a much more complex task. The rotational camera motion blur, is complicated due to the fact that it is spatially variant in nature. Normal identification and restoration approach for linear camera motion blur might not be applicable since a digital image is usually sampled in Cartesian coordinates and hence is a spatially invariant problem. Image restoration techniques are different for spatially invariant and variant problem, with the latter being more difficult to handle.

There are a number of methods that describe how to deal with general spatially variant problems. However, the rotational blur problem is somewhat special, since it could be transformed to or approximated by a spatially invariant problem [113]~[114] with appropriate consideration to the transformation of coordinates systems. The Polar coordinates transform is one of the possible solutions [113]. In [113], two Polar coordinates systems are defined, and the discrete formulation of the transform is introduced. The under-sampling phenomenon is also noticed and discussed by the authors. Interpolation is used to solve the under-sampling problem. However, the discrete formulation presented in [113] is somewhat an analog and digital hybrid, since one pixel could be divided into several pieces, say, using internal interpolation. The optimal resolution of the interpolation is not provided.

Hong and Zhang [114] presented yet another coordinate transform restoration method to solve the problem of rotational blur. The concept of blurring path is presented, and three specific rules have

been devised to fetch the gray level of a given pixel into the new Polar coordinates. Since the calculation of the position of the next possible pixel in the new Polar coordinates is needed for every point and that the three rules must be respected, their method is not efficient. But, the most severe inconvenience is that the sampling interval is not uniform in the new coordinates.

7.2. Separation of the Spatially Variance and Invariance



(a). Linear Motion Blur (b). Rotational Motion Blur

Figure 7.1 Comparison of Linear Motion Blur and Rotational Motion Blur

One of the main objectives of the study is to simplify the restoration of the rotational blurred image from a spatially variant problem to a spatially invariant problem. In Cartesian Coordinates, the image $g(x, y)$ is sampled by uniform rectangular or square matrix. This mode of sampling is convenient for linear motion blur problems, but it is not suitable in the case of rotational blur problems, which are spatially variant in nature and are more complicated as shown in Figure 7.1, due to two reasons.

Firstly, the extent of the point spread function (PSF) for rotational motion blur is a function of the radius of rotation r, which is the distance between a given point (x, y) and the rotation centre (x_o, y_o). The radius r is given by:

$$r(x,y)=\sqrt{(x-x_o)^2+(y-y_o)^2} \tag{7.1}$$

For a given imaging system, the PSF will be determined by the motion function [114]. In the case of rotational motion of the camera, the velocity v of the point (x, y) at time t is given by:

$$v(x,y,t) = r(x,y) * \dot{\theta}(t) = r(x,y) * \omega(t) \tag{7.2}$$

$\dot{\theta}$ is the first time-derivative of θ, which is the angular velocity ω. From Figure 7.1 and Eq. (7.2), even when the angular velocity ω is a constant for every point in the image, *i.e.* spatially invariant, the velocity is still a function of x, and y, *i.e.* spatially variant.

Secondly, the intensity (or gray level value) of a point in a blurred image is not only decided by the prospective region where it is in, but is also influenced by the adjacent regions [1]. In linear motion blur, the adjacent regions, which are actually described by the extent of the PSF, are straight regions as shown in Figure 7.1(a). In rotational motion blur, the points with the same r will only have the same magnitude of velocity, while the directions are different. As shown in Figure 7.1(b), even if a constant ω is applied in the rotational motion such that $|v_1|=|v|$, the adjacent region, which will affect the proposed point, is an arc for the rotational motion blur shown in Figure 7.1(b). This arc was also called as the blurring path in [114].

From the above analysis, the PSF in Cartesian coordinates is spatially variant for the rotational motion blur, but the spatially variance and invariance are actually separable. As indicated in Eq. (7.2), only the first term r in the right hand side is a function of x and y. If in a new coordinates, r and ω are independent of each other, then the problem will become spatially invariant, *i.e.* separable. A method is proposed in this work to solve the spatially variant problem of the rotational motion blur, which includes three steps:

1. Separate the spatially variance and invariance by an appropriate coordinates transformation, represent the blurred image in the new coordinates and treat it as a spatially invariant problem approximately.

2. Identify the PSF and apply an image restoration algorithm in the new coordinates.

3. Using an appropriate inverse transform, obtain the deblurred image in Cartesian coordinates.

Polar coordinates system is the right candidate of coordinates transformation. With a carefully designed Cartesian to Polar coordinate transformation, the spatially variance and invariance can be separated. The Polar coordinates system defined in this chapter is shown in Figure 7.2. We must emphasize the motion of interest here is not the actual relative motion between the camera and objects, but rather the relative motion between the image plane and the camera lens. One of the significant differences from [113] is that only one Polar coordinates system is defined which is more convenient to implement. In Eq. (7.1) and Figure 7.2, it is shown that the radius *r* is the distance from the pixel in question to the centre of rotation where the rotational axis passes through. This axis could be located inside or outside of the image.

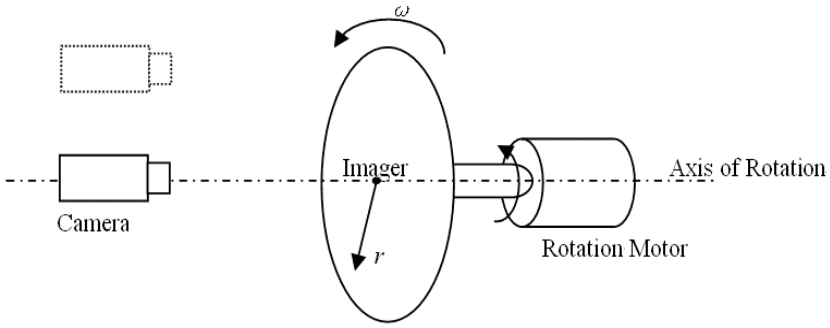


Figure 7.2 Polar Coordinates and the Axis of Rotation

7.2. Discrete Coordinates Transform between Cartesian and Polar

The coordinates transformation in continuous real domain $\mathbb{R}^2$ is denoted by $\mathbb{T}_{\mathbb{R}} : \mathcal{C} \to \mathcal{P}$, where $\mathcal{C}$ and $\mathcal{P}$ are the continuous image domain in Cartesian and Polar coordinates respectively and both $\mathcal{C}$ and $\mathcal{P} \subset \mathbb{R}^2$. $\mathbb{T}_{\mathbb{R}}$ is a vector function which maps $(x, y) \in \mathcal{C}$ into $(r, \theta) \in \mathcal{P}$. For the

2-D coordinates, generally the components of the vector function are defined as $\mathbb{T}_{\mathbb{R}} = < r = T_r(x,y)$, $\theta = T_\theta(x,y) >$, where $x, y, r, \theta \in \mathbb{R}$ and the inverse transform will be $\mathbb{T}_{\mathbb{R}}^{-1}: \mathcal{P} \rightarrow \mathcal{C} = < x = T_x(r,\theta)$, $\theta = T_y(r,\theta) >$. The subscript $\mathbb{R}$ stands for the real value domain.

For a given blurred image $g(x, y)$, suppose that the rotation centre is known and located at (x_o, y_o). Radius r is the measurement of the distance from the point in question at (x,y) to the centre of rotation. The distance between those two points in the image is important. the distance is defined by the Euclidean distance, which is:

$$r(x,y) = T_r(x,y) = \sqrt{(x-x_o)^2 + (y-y_o)^2} \qquad x, y \in \mathbb{R} \tag{7.3}$$

The definition of θ in this dissertation is given by:

$$\theta(x,y) = T_\theta(x,y) = \begin{cases} a\tan(y-y_o / x-x_o) & if\ x-x_o \neq 0 \text{ and } y-y_o > 0 \\ 2\pi + a\tan(y-y_o / x-x_o) & if\ x-x_o \neq 0 \text{ and } y-y_o < 0 \\ \pi/2 & if\ x-x_o = 0 \text{ and } y-y_o > 0 \\ 0 & if\ x-x_o = 0 \text{ and } y-y_o = 0 \\ 3\pi/2 & if\ x-x_o = 0 \text{ and } y-y_o < 0 \end{cases} \qquad x, y \in \mathbb{R} \tag{7.4}$$

In Eq. (7.4), the range of θ is from [0, 2π), instead of [-π, π). This definition is solely for the convenience of calculations. $\theta = 0$ corresponds to the positive x-axis in the Cartesian coordinates. Therefore, the inverse procedure is:

$$x = T_x(r,\theta) = x_o + r(x,y) \times \cos[\theta(x,y)] \quad r, \theta \in \mathbb{R} \tag{7.5}$$

$$y = T_y(r,\theta) = y_o + r(x,y) \times \sin[\theta(x,y)] \quad r, \theta \in \mathbb{R} \tag{7.6}$$

The coordinates transform is straightforward in $\mathbb{R}^2$ as shown above. The transformation function and its inverse in above equations are both bijective.

However the coordinates in digital images is the discrete variables, which means $\mathbb{T}_{\mathbb{Z}} =$

$< l = T_l(i,j)$, $k = T_k(i,j) >$, where $(i,j) \in \mathcal{C}_z$ and $(l,k) \in \mathcal{P}_z$. $\mathcal{C}_z$ and $\mathcal{P}_z$ are the discrete Cartesian and Polar domains. Both $\mathcal{C}_z$ and $\mathcal{P}_z \subset \mathbb{Z}^2$ and similar to the $\mathbb{T}_{\mathbb{Z}}^{-1}$. Notice that in $\mathcal{C}_z$ and $\mathcal{P}_z$, the above geometrical relationships in Eq. (7.3) ~ (7.6) are still valid. If directly replace the variables in Eq. (7.3), will obtain:

$$r(i,j) = \sqrt{(i-i_o)^2 + (j-j_o)^2} \qquad i,j \in \mathbb{Z} \tag{7.7}$$

$$\theta(i,j) = T_\theta(i,j) = \begin{cases} a\tan(j-j_o / i-i_o) & \text{if } i-i_o \neq 0 \text{ and } j-j_o > 0 \\ 2\pi + a\tan(j-j_o / i-i_o) & \text{if } i-i_o \neq 0 \text{ and } j-j_o < 0 \\ \pi/2 & \text{if } i-i_o = 0 \text{ and } j-j_o > 0 \\ 0 & \text{if } i-i_o = 0 \text{ and } j-j_o = 0 \\ 3\pi/2 & \text{if } i-i_o = 0 \text{ and } j-j_o < 0 \end{cases} \qquad i,j \in \mathbb{Z} \tag{7.8}$$

Therefore, in order to find the transform in discrete form, those variables both in Polar and Cartesian coordinates need to be digitized such that both $\mathbb{T}_{\mathbb{Z}} \subseteq \mathbb{Z}^2 \times \mathbb{Z}^2$ and $\mathbb{T}_{\mathbb{Z}}^{-1} \subseteq \mathbb{Z}^2 \times \mathbb{Z}^2$. The digitization of Cartesian coordinates is already done, since a digital image is generally a uniformly sampled $N \times M$ matrix in Cartesian coordinates. Eq. (7.7) and (7.8) should be digitized. It is actually a re-sampling procedure.

The discrete coordinates transform is to create a new matrix in Polar coordinates, say $L \times K$, which can express or approximate relationship described by Eq. (7.7) and (7.8) and its inverse. Generally, a pixel of a digital image is represented by a small square region, while sometimes a rectangle is used. Without the loss of generality, let us assume that the pixels are squares in this work. Since the variables need to be resampled in the new Polar coordinate domain, a new sampling interval must be determined. For r, its unit of measure is in terms of the number of pixels, and for θ is in degree. Digitization is necessary for both r and θ, while both variables are not linear; $r(i, j)$ is a quadratic function, while $\theta(i, j)$ is a discontinuous trigonometric function. Here let us define Polar Resampling Resolutions (*PRR*), Δ_r and Δ_θ , which are the resampling

interval of discrete Polar coordinates in r and θ, respectively. *PRR* is the smallest unit of the resampling in Polar coordinates, which must be positive. The r and θ will be digitized by using the sample intervals Δ_r and Δ_θ, respectively. In other words, r and θ will be approximated by the integral multiple of Δ_r and Δ_θ. Typically, there are three types of approximations for the digitization. They are nearest approximation, flooring approximation and ceiling approximation. In this dissertation, the nearest approximation will be applied since there is no significant difference between these three methods. The operators of the nearest approximation to describe the process of resampling are denoted by $S_n(r \mid \Delta_r)$ and $S_n(\theta \mid \Delta_\theta)$. In this work, Δ_r and Δ_θ are used to resample the given $r(i,j)$ and $\theta(i,j)$ represented by Eq. (7.7) and (7.8). If $(n\text{-}1)\ \Delta_r \leqslant r(i,j) < n\,\Delta_r$ and $(m\text{-}1)\ \Delta_\theta \leqslant\ \theta(i,j) < m\,\Delta_\theta$, where n and m are positive integral, then:

$$l_r = T_l(i,j \mid \Delta_r) = S(r(i,j) \mid \Delta_r) = \begin{cases} n-1 & if\ \mid r(i,j)-(n-1)\Delta_r \mid \leqslant \mid r(i,j)-n\Delta_r \mid \\ n & if\ \mid r(i,j)-n\Delta_r \mid \leqslant \mid r(i,j)-(n-1)\Delta_r \mid \end{cases} \quad (7.9)$$

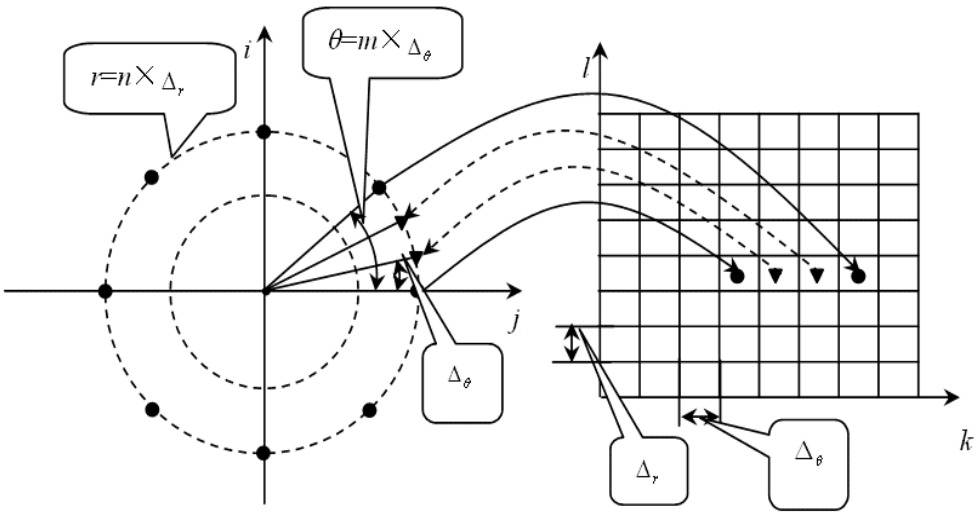
$$k_r = T_k(i,j \mid \Delta_\theta) = S(\theta(i,j) \mid \Delta_\theta) = \begin{cases} m-1 & if\ \mid \theta(i,j)-(m-1)\Delta_\theta \mid \leqslant \mid \theta(i,j)-m\Delta_\theta \mid \\ m & if\ \mid \theta(i,j)-m\Delta_\theta \mid \leqslant \mid \theta(i,j)-(m-1)\Delta_\theta \mid \end{cases} \quad (7.10)$$

where the $r(i,j)$ and $\theta(i,j)$ are determined by Eq. (7.7) and (7.8). $(l_r,\ k_\theta)$ is the pixel in discrete Polar Coordinates. The nearest approximation operator is also applied for the digitization of inverse transform, which is:

$$i = T_i(l_r, k_\theta \mid \Delta_r, \Delta_\theta) = i_o + S_n([l_r \times \Delta_r] \times \cos[k_\theta \times \Delta_\theta] \mid 1) \quad (7.11)$$

$$j = T_j(l_r, k_\theta \mid \Delta_r, \Delta_\theta) = j_o + S_n([l_r \times \Delta_\theta] \times \sin[k_\theta \times \Delta_\theta] \mid 1) \quad (7.12)$$

The smallest unit of approximation for the inverse transform is defined as 1. The transformation function in discrete form and its inverse are respectively defined by $\mathbb{T}_{\mathbb{Z}}$ in Eq. (7.9) and 7.10; $\mathbb{T}_{\mathbb{Z}}^{-1}$ is in Eq. (7.11) and (7.12).



(a). Cartesian Coordinates (b). Polar Coordinates

Figure 7.3 Support Points and Virtual Points

The transformation function and its inverse normally are both bijective in $\mathbb{R}^2$, however this constraint is too strict for discrete formulation in our applications and needs to be relaxed. For $\mathbb{T}_{\mathbb{Z}}$, an injective function would be sufficient, it is also known as One-to-One mapping, which could be expressed as:

- Uniqueness:

If a point in $\mathcal{P}_z$ satisfies $B = \mathbb{T}_{\mathbb{Z}}(A)$, there should be one and only one point A in $\mathcal{C}_z$ such that $B = \mathbb{T}_{\mathbb{Z}}(A)$ and $\mathbb{T}_{\mathbb{Z}}^{-1}(B) \to A$. (7.13)

The uniqueness means the transformation and its inverse will not introduce errors to the lattice. Similarly, $\mathbb{T}_{\mathbb{Z}}^{-1}$ should satisfy the condition of sufficiency:

- Sufficiency:

$\forall$ point $A \in \mathcal{C}_z$, there should exist one and only one corresponding point $B \in \mathcal{P}_z$ such that $B = \mathbb{T}_{\mathbb{Z}}(A)$.
$\forall$ point $B \in \mathcal{P}_z$, there exists one and only one corresponding point $A \in \mathcal{C}_z$ such that $A = \mathbb{T}_{\mathbb{Z}}^{-1}(B)$. (7.14)

In order to satisfy the uniqueness, Δ_r and Δ_θ must be carefully selected, since the resampling of the transform and its inverse could possibly introduce error. Fortunately, the error is bounded and controllable. For the transform from Cartesian to Polar, the errors will be bounded by Δ_r and Δ_θ for r and θ respectively. For the error of the inverse transform, let let's at the Eq. (7.11) and (7.12) again. If the *PRR* approach a positive but infinitesimally small value, both error for transform and its inverse could be eliminated, in other words, the Uniqueness can be satisfied by selecting a proper small Δ_r and Δ_θ .

The first condition in Eq. (7.14) is easy to satisfy, while the second one is too strict. It will be replaced by a weaker constraint in the next section, since the Sufficiency condition of $\mathbb{T}_{\mathbb{Z}}^{-1}$ cannot be fulfilled by selecting proper Δ_r and Δ_θ . There are two reasons why the second condition is too strict. Firstly, the under-sampling in Polar coordinates is inevitable. This phenomenon is more serious for θ, while it is not so obvious for r, although the problem still exists. Please note that $r(i, j)$ is a quadratic function, which changes slower when it is closer to the rotational centre. With the increase in r, the sampling space of θ between two consecutive points will become smaller; with the decrease of r, the sampling interval of θ is relatively sparse. This is the main reason for which the Sufficiency can not be assured during the coordinate transformation from Polar to Cartesian coordinates when r is small. Secondly, the digitization in Polar coordinates is another reason. The under-sampling is related to the choice of *PRR* and the process of the digitization. When a smaller *PRR* for re-sampling is selected, under-sampling problem becomes more severe. With the decrease in *PRR*, the approximation will become more accurate, while there are more "empty points" in Polar coordinates, as shown in Figure 7.3. The transformations presented in [113] and [114], also cannot guarantee the sufficiency of $\mathbb{T}_{\mathbb{Z}}^{-1}$. If Sufficiency is satisfied, the Uniqueness cannot be satisfied. Fortunately, the sufficiency of $\mathbb{T}_{\mathbb{Z}}^{-1}$ is somewhat redundant and it could be relaxed

7.3. Pixel Mapping between the Two Coordinates and the Interpolation of the Virtual Points

The pixels in $\mathcal{P}_z$, say (l, k), which are transformed from the points of Cartesian coordinates, such as (i, j), are called the Support Points (***SP***) in Polar coordinates. After the transformation and gray level mapping, the corresponding gray level of the *SP* in Polar coordinates can be obtained. Since the sufficiency condition is relaxed, some points in $\mathcal{P}_z$ are empty. Those "empty points" are called as virtual points (***VP***), which are between the support points. Suppose a digital blurred digital image $g(i, j)$ is represented by an $N \times M$ matrix. After the coordinates transformation, an $L \times K$ matrix in Polar coordinates for the given Δ_r and Δ_θ is created, as shown in Figure 7.3. The corresponding gray level of the *SP* in Polar coordinates is given by:

$$\Gamma\{S_n[r(i,j) \mid \Delta_r], S_n[\theta(i,j) \mid \Delta_\theta] \mid \Delta_\theta\} = g(i,j) \tag{7.15}$$

where Γ is the image in discrete Polar domain. The points in Eq. (7.15) are only the *SP*. Every pixel of $g(i,j)$ in Cartesian coordinates will be assigned to the corresponding *SP* in $\Gamma(l, k)$. Only the *SP* in $\Gamma(l, k)$ will be assigned gray values after the mapping. However, the *VP*s in the Polar coordinates are still empty, as mentioned previously. This blankness is not desired.

Figure 7.3 illustrates the pixel mapping from the Cartesian to the Polar coordinates. After the operation by using the nearest approximation mentioned earlier, a given $r = n \times \Delta_r$ is represented by the circles of dash lines in Figure 7.3(a). The round dots in Figure 7.3(b) are the pixels in $g(i,j)$. After the coordinates transformation, these points will be mapped into Polar coordinates, again represented by the round dots in Figure 7.3(b), which are defined as *SP*s. Virtual points are shown by the black triangles in Figure 7.3(b). If the inverse coordinates transform described by Eq. (7.11) and (7.12) is applied, these points will be located between the pixels in the Cartesian

coordinates. Obviously, the indices for the black triangular points in Cartesian coordinates will not be integers. Note these *VP*s do not really exist in the Cartesian coordinates. That is why they are called "Virtual Points" here. These *VP*s with the same *r* in Figure 7.3(b) between the two *SPs* will be uniformly spaced by the interval Δ_θ. The situation is similar to *r* axis. The uniform re-sampling strategy here will give us convenience in the further processing of the image. The algorithms for spatially invariant problem in Cartesian coordinates could be easily applied in the new Polar coordinates.

As analyzed above, $\Gamma(l, k)$ in $\mathcal{P}_z$ is not continuous after the gray value mapping since the *VP*s are not yet defined. The gray levels of the *VP*s are called the "missing data" in some literatures. Actually, it is a function regression problem, since the gray levels of some unseen data are needed. Continuity and smoothness are normal assumptions in the function regressions. For a given *l*, and with $r = l \times \Delta_r$, if there are *p VP*s between two *SPs*, it is assumed that the sampling rate of image in Cartesian coordinates is small enough such that linear interpolation based the two *SPs* can be applied to obtain the gray values of the *p VP*s. The situation is similar when dealing with θ. Theoretically the interpolation along either dimension can be done first. Since the under-sampling is more severe for θ, it is preferred to tackle θ first. The assumption is similar as in the case of *r*, the only difference is that some of the *VP*s already have gray values if the interpolation of θ has been done first. Therefore, for a given *k*, and $\theta = k \times \Delta_\theta$, if there are *q* undefined *VP*s between *SPs* and/or defined *VP*s, it is assumed that the sampling rate of image in Cartesian coordinates is small enough such that linear interpolation can be used to obtain two known points to obtain the gray value of the *q* unknown *VP*s. Admittedly, there are some advanced interpolation methods which could be better for *VP*s estimation, the linear interpolation using the above assumptions is solely for computational efficiency consideration. The boundary is an issue too, since the boundary points are empty and cannot be obtained by the internal interpolation. In

this chapter, the boundaries are determined by the mean value of Cartesian image.

After the gray level mapping, an image Γ in $\mathcal{P}_z$ can be obtained, in which the rotational blur could be treated as a spatially invariant problem approximately. In $\mathcal{P}_z$, the existed spatially invariant algorithm can be easily applied for the rotational blur identification and restoration. Of course, after the processing, the image will be transformed back to Cartesian coordinates for display and do further processing, such as edge detection, boundary tracing, *etc*.

7.4. Optimization of the PRR

The bijective transform functions in $\mathbb{R}^2$ are relaxed to be injective for $\mathbb{Z}^2$. For $\mathbb{T}_{\mathbb{Z}}^{-1}$, the Sufficiency condition cannot be assured. Notice that in pixel mapping between Cartesian and Polar coordinates, only the *SPs* are involved. $\mathbb{T}_{\mathbb{Z}}$ is a function, while strictly $\mathbb{T}_{\mathbb{Z}}^{-1}$ might not be one, since the *VP*s in $\mathcal{P}_z$ do not have corresponding point in $\mathcal{C}_z$. $\mathbb{T}_{\mathbb{Z}}^{-1}$ is only a mapping relation, but it is still called a function.

After the coordinates transformation and the mapping of gray levels, $N\times M$ *SPs* in $\mathcal{P}_z$ are obtained and there are about $(L\times K)-(N\times M)$ empty *VP* in $\Gamma(l, k)$ totally. Based on the assumption on continuity and smoothness, linear interpolation is applied to obtain the gray levels of those blank *VP*s in Polar coordinates. When the *PRR* approaches a positive but infinitesimally small value, the error of the transformation and its inverse will be minimized or eliminated. While if *PRR* is too small, the problem of under-sampling is more severe, which also means there are more "empty" *VP*s in $\mathcal{P}_z$. In other word, the Insufficiency is more serious when *PRR* is smaller.

The Sufficiency condition of the inverse transform function is relaxed in previous section, while the "Insufficiency" of the image in Polar coordinates should be minimized. The assumptions for

the linear interpolation will be doubtful if the distance between two *SPs* is too big. This optimization is actually a weaker constraint than the Sufficiency. The cost function for the optimization is the number of the "missing data", or the number of the *VP*s, which is:

$$P = (L \times K) - (N \times M) \tag{7.16}$$

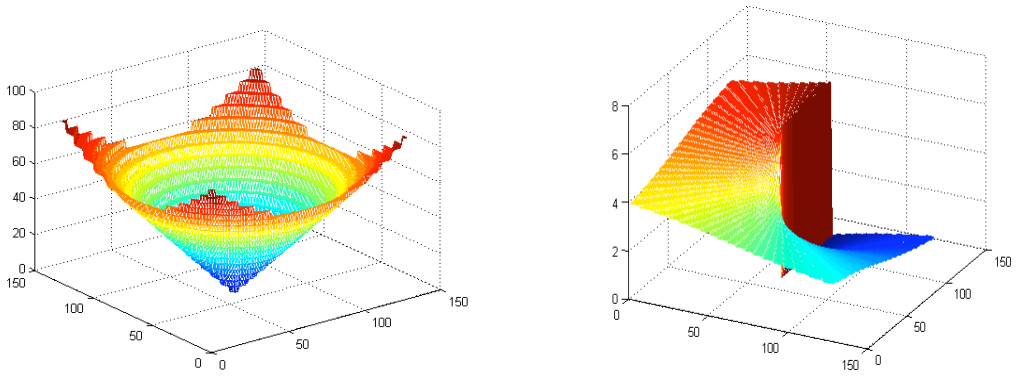
The expression of P is quit simple as shown in Eq. (7.16), but it is a function of N, M, i_o, j_o, Δ_r and Δ_θ. In Eq. (7.16), the contribution of the boundary points to P is ignored. Then the optimization problem is:

$$\underset{(\Delta_r, \Delta_\theta)}{Inf(P)} \quad \forall A \in \mathcal{C}_Z, exists \;\; only \;\; one \;\; unique \;\; B \in \mathcal{P}_Z \;\; S.T. \;\; B = \mathbb{T}_Z(A) \text{ and } A = \mathbb{T}_Z^{-1}(B) \tag{7.17}$$

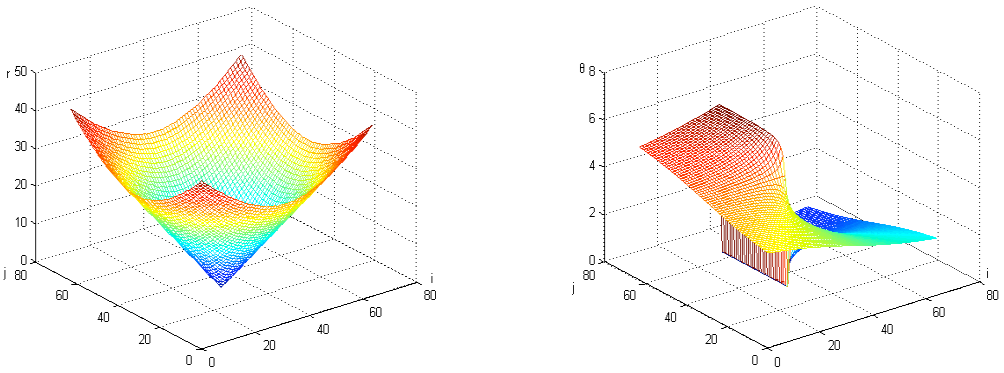
The meaning of Eq. (7.17) is actually a constrained optimization, *e.g.* to find the best *PRR* to minimize the number of *VP*s without violating the Uniqueness of $\mathbb{T}_{\mathbb{Z}}$. Theoretically, if P=0, the Sufficiency condition will be satisfied. Eq. (7. 17) is a weaker constraint than Eq. (7.14). After this optimization, the transformation is called as ***Compact Discrete Polar Transform***.

Numerical steepest gradient optimization methods could be applied to optimize Eq. (7.17). As stated above, the problem introduced by under-sampling is more serious in θ, optimization of the cost function is done in θ first, then based on the determined Δ_θ, optimized Δ_r is obtained.

The *PRR* are very important parameters in the compact discrete polar transform. Their influence in the discrete polar transformation is shown in Figure 7.4. In Figure 7.4, two different *PRRs* are applied for the discrete polar transform for a 128×128 lattice with the same rotation center at (64, 64).



(a) Digitized Radius and Angle in Discrete Polar Coordinates with ($\Delta_r = 5, \Delta_\theta = 5^\circ$)



(b) Digitized Radius and Angle in Discrete Polar Coordinates with ($\Delta_r = 0.9, \Delta_\theta = 0.3^\circ$)

Figure 7.4 $r(i, j)$ and $\theta(i, j)$ with Different *PRR* for a 128×128 Image Rotation Centre at (64, 64)

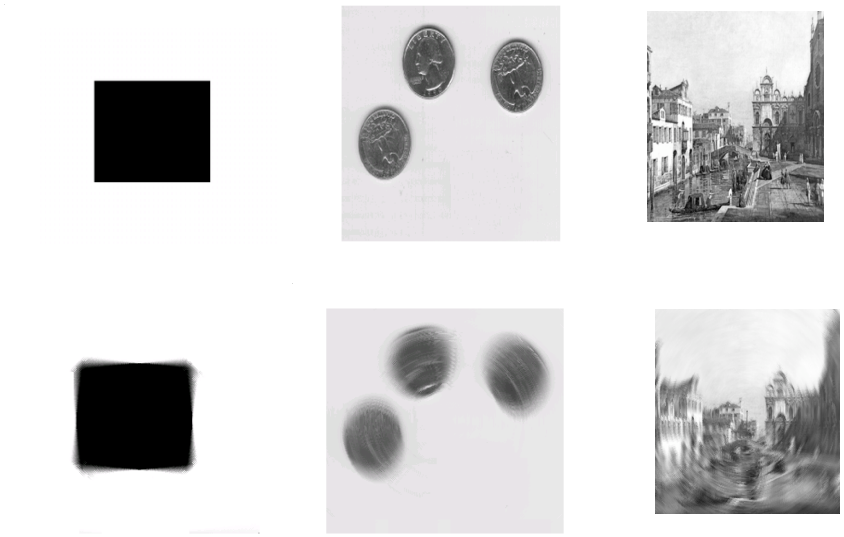
From Figure 7.4, it is obvious that the *PRR* in Figure 7.4(a) is too big since the r and θ, are discontinuous. There are obvious step artifacts in those two functions. Under this circumstance, the Uniqueness will be violated. Some different pixels in Cartesian coordinates will be mapped to a same point in the discrete polar coordinates, which is not acceptable. With the decrease in *PRR*, the digitized radius and angle in the Polar coordinates will become continuous, as shown in Figure 7.4(b).

7.5. Simulation and Experiments

As stated in the last section, the "insufficiency" in the transformation must be minimized. According to Eq. (7.17), the steepest gradient optimization algorithm is applied to determine the

optimized *PRR* for this given model, which is $\Delta_r = 0.9$ and $\Delta_\theta = 1/3°$. The results are shown in Figure 7.4(b). In this figure, both radius and angle are smooth. At the same time, the number of virtual points is minimized. In the discrete Polar coordinates, the gray levels of the virtual points will be determined by the linear interpolation.

As analyzed above, the rotational blurred image could be treated as a spatially invariant problem in $\mathcal{P}_z$ approximately. The formulation of blurring procedure and deconvolution are exactly the same in Cartesian coordinates after replacing the variables in the new coordinates. For the blur identification and restoration algorithm, please refer to [115][11] *etc*. for an overview. We also presented an algorithm on how to identify the extent of the PSF in [112].



(a)Square (Blurred degree=13.3°) (b) Coins (Blurred degree=16.7°) (c)Canal (Blurred degree=6.7°)

Figure 7.5 The Simulations of Rotational Motion Blur by the Compact Discrete Polar Coordinates Transform for 256×256 Lattices with Rotation Centre at (128,128)

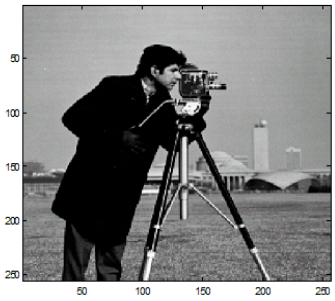
Since the convolution and the deconvolution are inverse operations, some images are simulated to obtain its rotational blurred version by using the coordinates transform described in this Chapter. Three images are blurred in $\mathcal{P}_z$ after the Compact Discrete Polar Transform is applied and their

Cartesian versions are displayed in Figure 7.5. The size of the images is 256×256 and the center of rotation is at the centre of the image, or at (128, 128). In this case, the optimized Δ_r and Δ_θ are 0.45 and 0.33° respectively, according to the steepest gradient optimization of the *PRR*. The non-blurred images in Cartesian coordinates are shown on the top and the simulated blurred images are at the bottom.

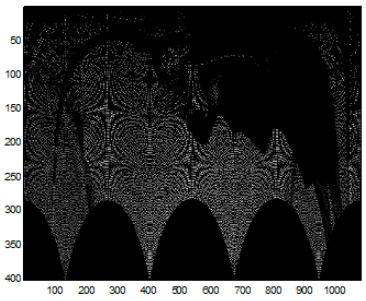
The PSF in $\mathcal{P}_z$ is assumed to be a constant angular velocity in the simulations. For more details about the more complicated PSF, which is not a constant angular velocity, please refer to [1]. All experimental results displayed in Figure 7.5 use the optimized *PRR* according to Eq. (7.17). When non-optimized *PRR* is applied for the transformation, there will be some artificial artifacts. The simulated blurred image will be discontinuous at some points in the Cartesian coordinates if the transformation is applied without optimizing the *PRR*. It is another interesting topic to discuss blurred image using the transformation with and without optimization of the *PRR*. However, this is beyond the scope of the current work.

In the next example, the noisy rotational blurred image of "Camera Man" is simulated and then restored in Polar coordinate domain. Finally, the restored image is shown in the Cartesian coordinates by using the inverse Polar transform. This procedure is also shown in Figure 7.6. The original image of the "Camera Man" is shown in Figure 7.6(a). The SPs of the image are shown in Figure 7.6(b). After the interpolation the image in Polar coordinate domain is shown in Figure 7.6(c). And its noisy rotational blurred version in the same domain is shown in Figure 7.6(d).

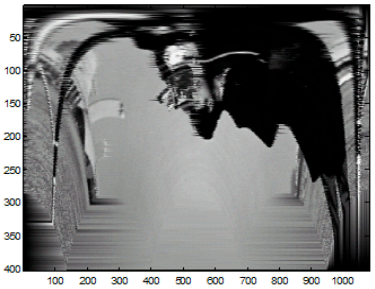
The noisy rotational blurred image in Cartesian coordinates is shown in Figure 7.7(a). The denoised image in Polar coordinate is shown in Figure 7.7(b). The deblurred image in Polar coordinate domain is shown in Figure 7.7(c); and finally, the restored image in Cartesian coordinate domain is shown in Figure 7.7(d).



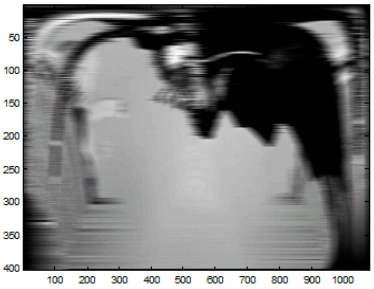
(a). Original Image of Cameraman



(b). SPs in Discrete Polar Domain

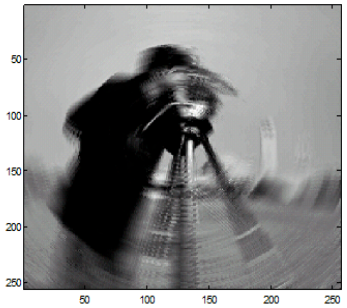


(c). Image in Discrete Polar Domain

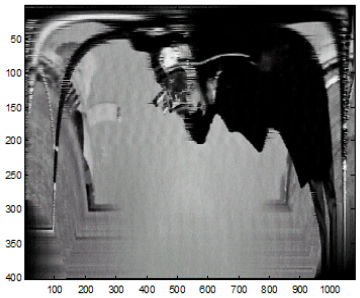


(d). Blurred Image in Polar Domain

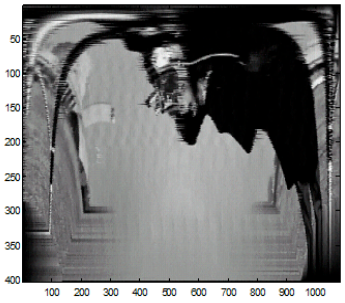
Figure 7.6. Original Image of Cameraman, the SPs in Polar Domain, the Image in Polar Domain and the Blurred image in Polar Domain (Blurred Degree=18°)



(a). Blurred Image in Cartesian Domain



(b) Denoised Image in Polar Domain



(c). Restored Image in Polar Domain

(d) Restored Image in Cartesian Domain

Figure 7.7 The Restoration of Rotational Blurred Image in Polar Domain and the Restored Image in Cartesian Domain

From this simulation, the Compact Discrete Polar Coordinates transform can successfully change a spatially variance problem of rotational blurred image to a more manageable spatially invariant problem in the new Polar domain. Both the blurring and deblurring procedures could be successfully simulated in Polar coordinates. The inverse transformation can also convert the image in Polar coordinate back to the Cartesian coordinates and achieve good visual results.

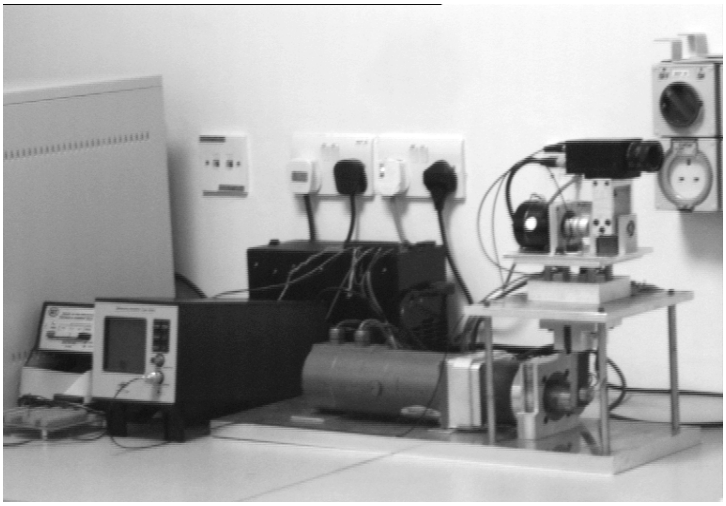


Figure 7.8 Setup of the Rotational Motion Platform and Imaging System

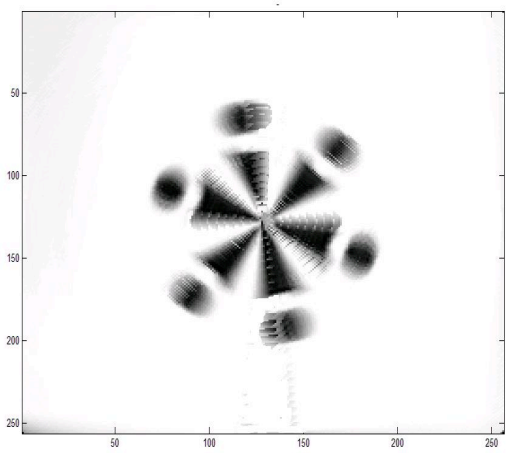
In order to test presented approach in real rotation blurred image restoration, an experimental platform was established by Mr. Lee Shay Liang[116], as shown in Figure 7.8. The calibration of the camera was done by Mr. Ng Wei Hiong [117].

(a) Image for the Rotation Centre Calibration (b) Blurred Image for Calibration

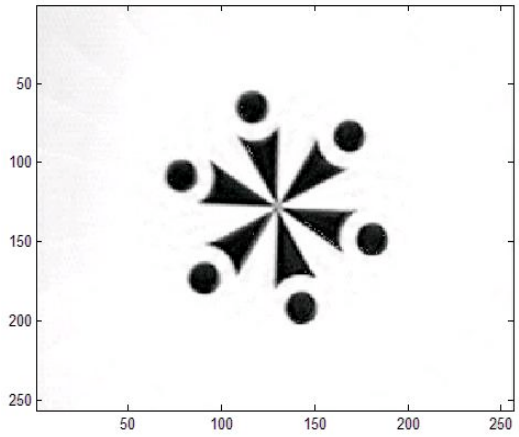
Figure 7.9 Calibration of Rotation Centre

Since the rotational centre should be known to apply the coordinates transformation, the rotational centre is obtained by the camera calibration instead of using other rotational centre identification methods to avoid the influence of the error of the location of the rotational centre. The centre of rotation is easily obtained by the artificial image. The image for calibration and the blurred image are shown in figure 7.9.

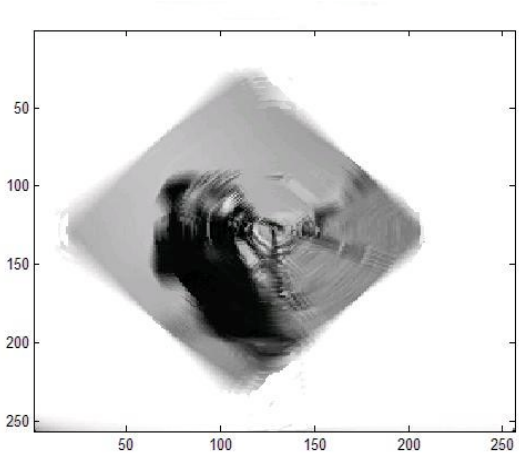
After the calibration, the rotation center is found to be at (126, 127) in the Cartesian coordinates. The image is mounted on the platform and then captured by a camera, which is in rotational motion. The captured images are shown in Figure 7.10(a) and 7.10(c).



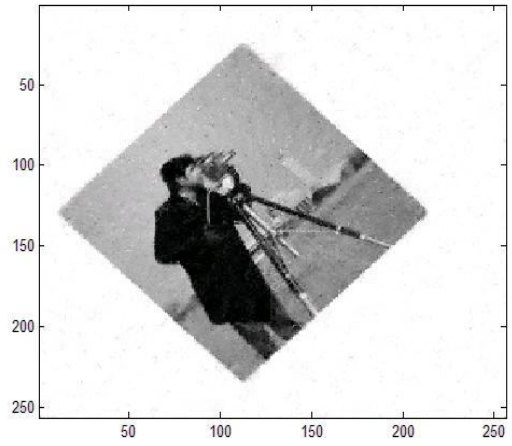
(a) Rotational Blur Image of "Star"



(b) Restored Image by Proposed Approach



(c) Rotational Blur Image of "Cameraman"



(b) Restored Image by Proposed Approach

Figure 7.10 Restoration of the Real Rotation Blurred Images

From the above simulations and experiments, the Compact Discrete Polar transformation is a good approach to simplify the spatially variant problem to a spatially invariant one. It can successfully restore the rotational blurred image. In the new Compact Polar coordinate domain, the spatially invariant formulation could be ready to solve the rotational blurred image, which is spatially variant in Cartesian coordinates.

7.6. Summary

In this chapter, a Discrete Compact Polar Coordinates Transformation is developed to handle the image restoration task of a rotational blurred image. By defining the coordinates transformation carefully, the initial spatially variant problem could be converted and simplified to a spatially invariant problem in the new coordinates. The transformation in continuous domain is straightforward; however, additional issues in the discrete domain are considerable. The linear interpolation is applied to deduce the values of the *VP*s for the consideration of computational efficiency based on the assumption made during sampling. The constraint of Sufficiency of the transformation is relaxed and replaced by a weaker constraint, *i.e.* the optimization of the *PRR* and minimization of the number of *VP*s. Since the number of the "empty points" is minimized, the transformation is so-called Compact Discrete Polar Transformation.

From the experimental results, images in discrete Cartesian lattices can be successfully rotationally blurred by simulation based the proposed transformation. The invariance and variance of the rotational blur problem are separable in the compact polar coordinates transform. Two real rotational blurred images captured by the established platform are used to validate the proposed approach. Both the simulation and the experiments achieved excellent results.

The proposed coordinates transformation has two advantages, uniform sampling and compact format. Uniform sampling allows the traditional spatially invariant image identification and

restoration algorithms in Cartesian coordinates to be readily applied to the spatially variant problem in the new coordinates by variable replacement according to the three steps stated in section 2 in this Chapter. The compactness will reduce the under-sampling in the Polar coordinates. The linear interpolation method is applied to deduce the gray level of the virtual points based on the assumptions made on the sampling interval in Cartesian.

Chapter 8

Conclusion

The objective of this dissertation is to develop approaches to handle image restoration problem for images blurred by motion of the camera or the objects in questions. Bayesian approaches have been chosen due to its robustness and accuracy. Its solid and established statistical foundation based on which efficient theoretical approaches could be built to solve our problems at hand.

Necessary mathematical representations of images have been established in Chapter 2. They are consistently used in the subsequent development of the proposed approaches and algorithms. Due to the importance of discontinuities in providing useful cue for the image restoration work, this work began with the identification of the edge points using the proposed approach on 1-D piecewise constant signals in order to gather better understanding of the potential difficulties at hand. Posterior evidence is proposed to select model orders since the prior distribution function is not proper. By assuming that the model is a Binomial distribution, the calculation of the evidence is simplified in Bayesian theory, and to avoid evaluating the posterior marginal likelihood function directly. Simulation and experiments proved that this approach is robust and accurate.

Subsequently, a novel Bayesian kernel approach on 1-D and 2-D discontinuous signal/image restoration is presented. Based on the assumption that the signals are Local Piecewise Linear, noise was removed and the discontinuities were identified and preserved. a novel convex cost function for image noise removal therefore is proposed. This doing helped to resolve the ill-conditioned problem of image restoration. The proposed cost function is proven to have only one global minimum. The cost function was minimized by tuning a proper hyper-parameter η through the steepest gradient optimization algorithm. The results are compared with those of LPA-ICI approach, as the latter is known to be recent and good. There are some other existing methods,

but their performance is not comparable to the proposed method nor that of LPA-ICI. The proposed approach achieved similar results: ISNR and PSNR is slightly low than ICI, the 1- and 2- Norm error are close, and the Infinite norm error is much smaller than that of LPA-ICI when noise level is high, *i.e.* at low SNR. However, it is slower than LPA-ICI, this is because of the necessity in evaluating the inverse of the Hessian matrix. LPA-ICI, on the other hand, requires convolution and de-convolution computation, which could be fast by using the well established FFT in the calculation.

Restoration of images blurred by motion of the camera and/or objects in the scene is also one of the aims of this dissertation. After the noise has been successfully removed with the edges preserved, the characteristics of the Point Spread Function (PSF) should be gathered with ease. The PSF will help to identify the motion blur, based on which the degraded image could be restored. A statistical approach for motion blur identification was presented in Chapter 6. Experimental results demonstrated its effectiveness even under low SNR conditions. It is also robust and accurate.

Finally, a more difficult problem is studied, the restoration of images under spatially variant condition. This would be the case when an image is blurred by rotation motion. In this situation, a normal approach would be to represent every pixel by a PSF before any restoration work could be performed. This is obviously not feasible as the computational work would be fastidious. With the innovative approach using a novel coordinate transformation, the spatially variant problem is reduced to a spatially invariant one. The formulation of the Discrete Polar Coordinates Transformation and its inverse are presented in Chapter 7, and it has two advantages, *viz*. Uniform Sampling Interval and Compactness (less missing data points after coordinate transformation). The Uniform Sampling allows the existing approaches of blur identification and image restoration methods to be applied to even rotational blur problem. Note that blurring due to rotational blur is originally a spatially variant problem. Experimental results convincingly show

the effectiveness of the proposed method. Furthermore, simulation shows that Compact Discrete Polar Transformation also can be applied for the problem of out-of-focus blurring problem, which is 1-D spatially invariant in the Polar domain. However, results are not presented in this dissertation due to the limitation of space.

For future works, some recommendations are as following:

1. The use of "traveling windows" in Chapter 5 for noise removal requires further refinement. Further works should be carried out to determine the size of the window in relation to the quality of the image in question. A correct size of the window will help to speed up the computation and hence the procedure as a whole. In addition, a global GML representation of image, without using windows, is also desired, but, an efficient algorithm with a manageable size of data must be developed.

2. A more accurate global noise estimation method and the associated computation algorithm are needed. With more accurate noise variance estimation, further performance improvement of the proposed restoration approach can be achieved. This will help in the optimization of the hyper-parameter η. Furthermore, new or improved optimization technique should be developed to reduce the residue of Cost function that theoretically should be zero.

3. In the work on the restoration of images blurred by rotational motion, the centre of rotation was based on calibration technique, which might not be accurate in some situations. A proper calibration technique should be established to estimate accurately the centre of rotation.

BIBLIOGRAPHY

[1] N.S. Kopeika, *A System Engineering Approach to Imaging*. SPIE Optical Engineering Press, 1998.

[2] R.C. Gonzalez and R E. Woods, *Digital Image Processing*, Addison-Wesley Publishing Company, 1993.

[3] I.M. Sezan and M.A. Tekalp, *Survey of Resent Developments in Digital Image Restoration*, Optical Engineering Vol. 29(5), pp. 393-404, May, 1990.

[4] G. Demoment, *Image Reconstruction and Restoration: Overview of Common Estimation Structure and Problems*, IEEE Transaction on Acoustics, Speech, and Signal Processing, Vol. ASSP-37(12), pp. 2024-2036, Dec. 1989.

[5] R.DL. Lagendijk and J. Biemond, *Iterative Identification and Restoration of Images*. Kluwer Academic Publishers, 1991.

[6] K. Miller, *Least-squares Method for Ill-posed Problem with a Prescribed Bound* SIAM Journal of Math. Anal., Vol. 1, pp. 52-74, 1970.

[7] T. F. Chan, S. Esedoglu, F. Park, M. H. Yip. *Recent Developments in Total Variation Image Restoration*. In "Handbook of Mathematical Models in Computer Vision". N. Paragios, Y. Chen, O. Faugeras, edts. Springer 2005.

[8] H.C. Andrews and B.R. Hunt, *Digital Image Restoration*, Prentice Hall Eaglewood Cliffs NJ, 1977.

[9] T.G. Stockham, T.M.Cannon and R.B. Ingebretsen, *Blind Deconvolution through Digital Signal Processing*, Proc. IEEE, Vol. 64, No. 4, pp. 678-692, 1975.

[10] M. Cannon, *Blind Deconvolution of Spatially Invariant Image Blurs with Phase*, IEEE Trans. Acoustics, Speech and Signal Processing, Vol. 24, pp. 58-63, 1976.

[11] G. Pavlovic, and A. M. Tekalp, *Maximum Likelihood Parametric Blur Identification Based on a Continuous Spatial Domain Model*, IEEE Transaction on Image Processing, Vol. 1, No. 4, October 1992.

[12] A.E. Savakis and H.J. Trussell, *Blur Identification by Residual Spectral Matching,* Image Processing, IEEE Transactions on, Vol. 2 , Issue: 2, pp. 141 - 151, April 1993

[13] G.B. Giannakis and R.W., Jr. Heath, *Blind Identification of Multichannel FIR Blurs and Perfect Image Restoration*, Image Processing, IEEE Transactions on, Vol. 9, Issue: 11, pp. 1877 - 1896, Nov. 2000.

[14] A.K. Katsaggelos and K. T. Lay, *Maximum Likelihood Blur Identification and Image Restoration Using the EM Algorithm.* IEEE transaction on Signal Processing, Vol. 39, No. 3, March 1991.

[15] A.M. Tekalp and H. Kaufman, *On Statistical Identification of a Class of linear Space-Invariant Image Blur Using Non-causal-Phase ARMA Models.* IEEE Transactions on Acoustics, Speech and Signal Processing, Vol. 36, pp. 1360-1363, August 1988

[16] R.L. Lagendijk, J. Biemond, and D.E. Boekee, *Regularized Iterative Image Restoration with Ringing Reduction.* IEEE Transactions on Acoustics, Speech and Signal Processing, Vol. 36, pp.1874 –1888, Dec. 1988.

[17] G. Pavlovic and A. M. Tekalp, *Maximum Likelihood Parametric Blur Identification Based on A Continuous Spatial Domain Model*, IEEE Transaction on Image Processing, Vol. 1, No. 4, Oct. 1992

[18] S.J. Reeves and R.M. Mersereau*, Blur Identification by the Method of Generalized Cross-Validation*, Image Processing, IEEE Transactions on, Vol. 1, Issue: 3, pp. 301 - 311, July 1992.

[19] K. Panchapakesan, D.G. Sheppard, M.W. Marcellin, and B.R. Hunt, *Blur Identification from Vector Guantizer Encoder Distortion*, Image Processing, IEEE Transactions on, Vol. 10 , Issue: 3,pp. 465 - 470,March 2001.

[20] R. Nakagaki, and A.K. Katsaggelos, *A VQ-based Blind Image Restoration Algorithm*, Vol. Issue: 9, pp. 1044 - 1053, Image Processing, IEEE Transactions on, Sept. 2003

[21] Y. Yu-Li and M. Kaveh, *A Regularization Approach to Joint Blur Identification and Image Restoration*, Image Processing, IEEE Transactions on, Vol.: 5 , Issue: 3, pp. 416 - 428,March 1996.

[22] L. Chen and Y. Kim-Hui, *A Soft Double Regularization Approach toParametric Blind Image Deconvolution*, Image Processing, IEEE Transactions on, Vol. 14, Issue: 5, pp. 624 - 633, May 2005.

[23] A.M. Tekalp, H. Kaufman, and J.W. Woods, *Identification of Image and Blur Parameters for the Restoration of Noncausal Blurs*. IEEE Transactions on Acoustics, Speech and Signal Processing, Vol. 34, pp. 963-972, August 1986

[24] J. Biemond, F.G. Putten, and J.M. Wood, *Identification and Restoration of Images with Symmetric Non-causal Blurs*, IEEE Transactions on Circuits and systems, Vol. 35, No. 4, pp. 385-393, Apirl 1988.

[25] S. Chitale, and W.T. Padgett, *Blur Identification and Correction for a Given Imaging System*. Southeastcon '99. Proceedings, IEEE , 1999.

[26] W. Norbert, *Extrapolation, Interpolation, and Smoothing of Stationary Time Series*. New York: Wiley. ISBN 0-262-73005-7, 1949.

[27] R. G. Brown and Y.C. Patrick, *Introduction to Random Signals and Applied Kalman Filtering,* 3rd Edition. New York: John Wiley & Sons. ISBN 0-471-12839-2, 1996

[28] J. Immerkaer: *Fast Noise Variance Estimation*, Computer Vision and Image Understanding, Vol. 64, No. 2, pp. 300-302, 1996

[29] J. Portilla, V. Strela, M.J. Wainwright, and E.P. Simoncelli. *Image Denoising Using Scale Mixtures of Gaussians in the Wavelet Domain*. IEEE Trans. on Image Processing, Vol. 12, Issue 11, pp. 1338--1351, Nov 2003.

[30] M.D. Grossberg and S.K. Nayar. *Modeling the Space of Camera Response Functions*. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 26, Issue 10, pp. 1272--1282, October 2004.

[31] C. Liu, W.T. Freeman, R. Szeliski and S.B. Kang. *Noise Estimation from a Single Image*. To appear in CVPR 2006.

[32] J. Besag, *Spatial Interaction and the Statistical Analysis of Lattice System.* Journal of the Royal Statistical Society, Series B (Methodological), Vol. 36, No.2. pp. 192-236 1974.

[33] S. Geman and D. Geman, *Stochastic Relaxation, Gibbs Distributions and the Bayesian Restoration of Images*, IEEE Transaction on Pattern Analysis and Machine Intelligence, Vol. PAM1.6, No.6 Nov. 1984.

[34] C. Bouman and K. Sauer, *A Generalized Gaussian Image Model for Edge-Preserving MAP Estimation*, IEEE Transaction on Image Processing, Vol. 2 No. 3. July 1993.

[35] R. Stevenson and E. Delp, *Fitting Curves with Discontinuities*, in Proc. of the 1st Int. Workshop. On Robust Computing. Vision, Seattle, WA. pp. 127-136. 1990.

[36] P.J. Green, *Bayesian Reconstructions from Emission Tomography Data Using a Modified EM Algorithm*, IEEE Trans. Medical Imaging, Vol. 9, pp. 84-93, Mar. 1990.

[37] K. Lange, *Convergence of EM Image Reconstruction Algorithm with Gibbs Prior*, IEEE Trans. Medical Imaging, Vol. 9, pp. 439-446, Dec. 1990.

[38] A. Blake and A. Zisserman A., *Visual Reconstruction*, Cambridge, MA: MIT Press, 1987.

[39] C. Bouman and B. Liu, *A Multiple Resolution Approach to Regularization*, in Proc. SPIE Conference on Visual Comm. and Image Processing., Cambridge, MA, pp. 512-520, 1988.

[40] D.A. Stephens, *Bayesian Retrospective Multi Changepoint Identification*, Applied Statistics, Vol. 43 No. 1, pp. 159-178, 1994.

[41] D.G.T. Denison, B.K. Mallick and Smith A. F. M., *Automatic Bayesian Curve Fitting*, J. R. Statistical Society B, Vol. 60, Part2, pp.333-350, 1998.

[42] J.G. Peter, *Reversible Jump Markov Chain Monte Carlo computation and Bayesian Model Determination*, Biometrika, Vol. 82-4, pp711-732, 1995.

[43] R. Molina, A.K. Katsaggelos, J. Mateos, A. Hermoso and C.A. Segall, *Restoration of Severely Blurred High Range Images Using Stochastic and Deterministic Relaxation Algorithms in Compound Gauss-Markov Random Fields*, Journal of Pattern Recognition Society, Vol. 33, pp 555-571, 2000

[44] T.S. Acton and C.B. Alan, *Piecewise and Local Image Model for Regularized Image Restoration Using Cross-Validation*, IEEE Transaction on Image Processing, Vol. 8, No. 5, 652-665, May 1999.

[45] A. Jalobeanu, L. Blanc-Feraud, J. Zerubia, *Hyper-parameter Estimation for Satellite Image Restoration Using a MCMC Maximum-likelihood Method*, Pattern Recognition, the Hournal of Pattern Recognition Society, Vol. 35, 341-352, 2002

[46] M. Sigelle, *A Cumulant Expansion Technique for Simultaneous Markov Random Field Image Restoration and Hyper-parameter Estimation*, International Journal of Computer Vision Vol. 37(3), pp. 275–293, 2000

[47] V. Katkovnik, K. Egiazarian and J. Astola, *A Spatial Adaptive Nonparametric Regression Image Deblurring*, IEEE Transaction on Image Processing, Vol. 14, No. 10, pp. 1469-1478, October 2005

[48] A. Goldenshluger and A. Nemirovski, *On Spatial Adaptive Estimation of Nonparametric Regression*, Math. Meth. Stat., Vol. 6, pp. 135-170. 1997.

[49] C.M.L. Thomas, *Automatic Smoothing for Discontinuous Regression Function, Statistica Sinica*, Vol. 12, pp. 823-842, 2002.

[50] K.P. Burnham and D.R. Anderson, *Model Selection and Inference: A Practical Information-Theoretic Approach*, Spring-Verlag, New York, 1998.

[51] G. Wahba, *Spine Models for Observational Data*, CBMS-NSF, Regional Conference Series in Applied Mathematics, SIAM. Philadelphia 1990.

[52] J. Rissanen, *Stochastic Complexity in Statistical Inquiry*. World Scientific, Singapore, 1989.

[53] E.A. Nadaraya, *On Estimating Regression, Theory of Probability and Its Applications*, Vol. 9, pp. 141-142, 1964

[54] G.S. Waston, *Smooth Regression Analysis*, Sankhya (Series A), Vol. 26, pp. 359-372, 1964.

[55] M.B. Priestley and M.T. Chao, *Non-parametric Function Fitting*, Journal of Royal Statistical Society (Series B), Vol. 34, pp. 385-392.1972

[56] T. Gasser and H.G. Mullier, *Kernel Estimation of Regression Function, in Smoothing Techniques for Curve Estimation*, Berlin, Springer-Verlag, pp. 23-68, 1979.

[57] K.F. Cheng, K.F. and P.E. Lin, *Nonparametric Estimation of a Regression Function*, Z. Wahrscheinlichkeistheorie Verw. Gebiete, Vol. 57, pp. 223-233, 1981.

[58] P. Qui, *Comparison of Several Local Smoothing Jump Detectors in One-dimensional Nonparametric Regression*, the 1999 ASA Proceeding of Statistical Computing Section, 1500155, 1999.

[59] I. Gilbels, P. Hall and A. Kneip, *On the Estimation of Jump Points in Smooth Curves*, Annals of the Institute of Statistical Mathematics, Vol. 51, pp. 231-251, 1999.

[60] D. Allen, *The Relationship Between Variable Selection and Data Augmentation and a Method of Prediction*, Technometrics, Vol. 16, pp. 125-127, 1974.

[61] C.L. Mallows, *Some Comments on Cp*, Technometrics, Vol. 15, pp. 661-675, 1973.

[62] T. Gausser, A. Kneip and W. Kohler, *A Flexible and Fast Method for Automatic Smoothing*, Journal of the American Statistical Association, Vol. 86, pp. 643-652, 1991.

[63] C. R Loader, *Bandwidth Selecion: Classical or Plug-in*, the Annals of Statistics, Vol. 27, pp. 415-438, 1999.

[64] L.I. Rudin, S. Osher and E. Fatemi, *Nonlinear Total Variation Based Noise Removal Algorithm*, Physica D, Vol. 60, pp. 259-268, 1992.

[65] Y.Y. Li and F. Santosa, *A computational Algorithm for Minimizing Total Variation in Image Restoration*, IEEE Transaction on Image Processing, Vol. 5, No. 6,pp. 987-995, June, 1996.

[66] Y.L. You and M. Kaveh, *Blind Image Restoration by Anisotropic Regularization*, IEEE Transaction on Image Processing, Vol. 8, No. 2, pp. 396-407, March 1999.

[67] P.L. Combettes and J.C. Pesquet, *Image Restoration Subject to a Total Constraint*, IEEE Transaction on Image Processing, Vol. 13, No. 9,pp. 1213-1222, September 2004

[68] M. David Strong and F. Tony Chan, *Spatially and Scale Adaptive Total Variation Based Regularization and Anisotropic Diffusion in Image Processing*, UCLA Image Processing Reports and Slides, November 1996.

URL: http://terminus.sdsu.edu/Research/CAMreports.html

[69] M. D. Strong, B. Peter and F. T. Chan, *Spatially Adaptive Local Feature-Driven Total Variation Minimizing Image Restoration*, UCLA Image Processing Reports and Slides, July 1997.

URL: http://terminus.sdsu.edu/Research/CAMreports.html

[70] J. Weickert, *Anisotropic Diffusion in Image Processing*. Stuttgart, Germany: Teubner-Verlag, 1998.

[71] D. Strong and T. Chan, *Relation of Regularization Parameter and Scale in Total Variation Based Image Denoising*, UCLA Math Department CAM Report, February 1996.

[72] D. Strong, T. Chan, *Exact Solutions to Total Variation Regularization Problems*, UCLA Math Department CAM Report, October 1996

[73] M. Cheriet, Shock *Filters for Character Image Enhancement and Peeling*, Proceedings of the Seventh International Conference on Document Analysis and Recognition (ICDAR 2003), 2003.

[74] T. Chan, A. Marquina and P. Mulet, *Second Order Differential Functions in Total Variation-Based Image Restoration*, August 1998.

[75] P. Blomgren T. F. Chen, *Total Variation Image Restoration: Numerical Methods and Extensions*, Proceedings of the 1997 IEEE International Conference on Image Processing, Vol. III, pp 384-387, 1997.

[76] A. Tonazzini and L. Bedini, *Monte Carlo Markov Chain techniques for Unsupervised MRF-based Image Denoise*, Pattern Recognition Letters 24, pp 55-64, 2003

[77] T. Rabie, *Robust Estimation Approach for Blind Denoise*, IEEE Transaction on Image Processing, Vol. 14, No. 11, pp 1755-1765, November, 2005

[78] R. Garnett, T. Huegerich, C. Chui and W.J. He, *A Universal Noise Removal Algorithm with Impulse Detector*, IEEE Transaction on Image Processing, Vol. 14 No. 11, November, 2005.

[79] S. H. Lee and J. K. Seo, *Noise Removal with Gauss Curvature-Driven Diffusion*, IEEE Transaction on Image Processing, Vol. 14, No. 7, pp 904-909, 2005.

[80] Z. L. Stan, *Roof-Edge Preserving Image Smoothing Based on MRFs*. IEEE Transactions on Image Processing. Vol. 9, No. 6, June, 2000

[81] Z. L. Stan, Y. H. Huang, and J. Fu. *Convex MRF potential functions*. In Proceedings of IEEE International Conference on Image Processing, Washington, D.C. Vol. 2, pp. 296--299, , pp. 23-26, October 1995.

[82] A. T. Figueiredo and M. N. Leitao, *Adaptive Discontinuity Location in Image Restoration*, ICIP-94. Proceedings, Vol. 2, pp 665-669, Nov 1994

[83] P. Charbonnier, L. Blance-Feraud, G. Aubert and M. Barlaud, *Deterministic Edge-Preserving Regularization in Computed Imaging*, IEEE Transaction on Image Processing, Vol. 6, No. 2, Feb, 1997.

[84] M. Ceccarelli, *Fast Edge Preserving Picture Recovery by Finite Markov Random Field*, Lecture Notes in Computer Science 3617, pp. 277-286.

[85] M. Cetin, W. C. Karl and A. S. Willsky, *Edge-Preserving Image Reconstruction for Coherent Imaging Application*, IEEE International Conference on Image Processing, Vol. 2, pp. 481-484, Rochester, New York, September 2002.

[86] J.A. Fessler, H. Erdogan and W.B Wu, *Exact Distribution of Edge-Preserving MPA Estimation for Linear Signal Models with Gaussian Measurement Noise*, IEEE Transaction on Image Processing, Vol. 9, No. 6, June 2000

[87] W. F. Chen, M. Chen and J. Zhou, *Adaptively Regularization Constrained Total Least-Squares Image Restoration*, IEEE Transaction on Image Processing, Vol. 9, No. 4. April, 2000

[88] F. Aghdasi and R. Ward, *Reduction of Boundary Artifacts in Image Restoration*, IEEE Transaction on Image Processing, Vol. 5, No.4, pp. 611-618, April, 1996

[89] S. J. Reeves, *Fast Restoration of PMMW Imagery without Boundary Artifacts, in Infrared and Passive Millimeter-Wave Imaging Systems: Design, Analysis Modeling and Testing*, R. Appleby, G.D. Holst, and D.A. Wikner, Eds. Orlando, FL: SPIE, 2002, Vol. 4719, SPIE, pp 289-295.

[90] W.M. Yu, K. B. Lim and S.L. Lee, *Ringing Reduction of Image Restoration*, 6th International Conference on Signal Processing 2002 Processing, BeiJing China, 2002

[91] S. J. Reeves, *Fast Image Restoration without Boundary Artifacts*, IEEE Transaction on Image Processing, Vol. 14, No. 10, pp. 1448-1453, October 2005.

[92] T.J. Hastie and R.T. Tibshirani, *Generalized Additive Model,* London: Chapman and Hall, 1990.

[93] T. Anderson, *An Introduction to Multivariate Analysis*. John Wiley, New York, 1958.

[94] J. O. Berger, *Statistical Decision Theory and Bayesian Analysis*. Springer-Verlag, New York,second edition, 1985.

[95] C.M. Bishop, *Neural Networks for Pattern Recognition*. Oxford University Press, 1995.

[96] C.J.C. Burges, *A Tutorial on Support Vector Machines for Pattern Recognition.* Data Mining and Knowledge Discovery, Vol. (2), pp. 121-167, 1998.

[97] W. Chu, S.S. Keerthi and C.J. Ong, *Bayesian Support Vector Regression Uusing a Unified Loss Function*, IEEE Transactions on Neural Networks, Vol. 15(1), pp. 29-44, 2004.

[98] Z. L. Stan, *Markov Random Field Modeling in Image Analysis*, ISBN 0-387-70145-1 Spinger-Verlag New York Berlin Heidelberg Tokyo, 1995.

[99] J. M. Hammersley and P. Clifford, *Markov Field on Finite Graphs and Lattice*, unpublished, 1971.

[100] C. Robert, *The Bayesian Choice: A Decision Theoretic Motivation*, Springer-Verlag, New York, 1994.

[101] J. Bernardo and A. Smith, *Bayesian Theory*, John Wiley & Sons, Chichester, United Kingdom, 1994.

[102] D. Mackay, *Bayesian methods for neural networks: Theory and applications*. Note for Neural networks summer school

URL: http://citeseer.nj.nec.com/mackay95 bayesian.html

[103] W. M. Thorburn, *Occam's Razor*, Mind, 24, pp. 287-288, 1915.

[104] W. M. Thorburn, *The Myth of Occam's Rrazor*, Mind, 27, pp. 345-353, 1918.

[105] A. Faul, and M.E. Tipping, *Sparse Bayesian Learning and Relevance Vector Machine*, Journal of Machine Learning Research Vol. 1, pp. 211–244, 2001.

[106] M.E. Tipping, *The Relevance Vector Machine*. Advances in Neural Information Processing Systems 12, pp. 652–658. Cambridge, Mass: MIT Press, 2000.

[107] A. Faul and M.E. Tipping, *Analysis of Sparse Bayesian Learning*, Proceedings of NIPS, 2001.

[108] M.E. Tipping and A. Faul, *Fast Marginal Likelihood Maximization for Sparse Bayesian Model*, In Proceedings of Artificial Intelligence and Statistics,2003.

[109] E. Previato, *Dictionary of Applied Math for Engineers and Scientists*, CRC Press, 2003.

[110] I.N. Bronshtein et al., *Handbook of Mathematics*, Berlin: Springer, 2004.

[111] J. Neter, W.Wasserman and G.A. Whitmore. *Applied Statistics*, Englewood Cliffs, N.J. Prentice Hall, 4th Edition, 1993.

[112] W. M. Yu, K. B. Lim and S. L. Lee, *Blur Identification In Spatial Domain*, the International Conference on Computer Vision, Pattern Recognition & Image Processing (CVPRIP'2002), USA, 2002.

[113] S. Ribaric, M. Milani, Z. Kalafatic: *Restoration of Images Blurred by Circular Motion*. First Int'l Workshop on Image and Signal Processing and Analysis (IWISPA 2000), Croatia, 2000.

[114] H. Hong and T. Zhang, *Fast Restoration Approach for Rotational Motion Blurred Image Based on Deconvolution along the Blurring Paths*. Opt. Eng. Vol. 42(12), pp. 3471-3486, Dec. 2003,

[115] O. Hadar, I. Dror and N. S. Kopeika, *Image Resolution Limits Resulting From Mechanical Vibration*. Part IV: Real-Time Numerical Calculation of the Optical Transfer Functions and Experimental Verification, SPIE, Optical Engineering 3Vol. 3(2), pp. 566-578, 1994.

[116] S. L. Lee, *Blur Identification and Image Restoration*, NUS Master of Engineering Thesis, 2003.

[117] W. H. Ng, *Restoration of Images Captured from the CCD Camera Mounted on a Vibrating Platform.* NUS Bachelor Final Year Project, Thesis, 2003.