LINE-FIELD BASED ADAPTIVE IMAGE MODEL FOR BLIND DEBLURRING

LE NGOC THUY
(Master of Engineering)

A THESIS SUBMITTED

FOR THE DEGREE OF DOCTOR OF PHILOSOPHY

DEPARTMENT OF MECHANICAL ENGINEERING

NATIONAL UNIVERSITY OF SINGAPORE

2010

Acknowledgement

I would like to express my deep gratitude to my supervisor, Professor Lim Kah Bin. His integral view on research and his untiring support have made a deep impression on me. It is a great pleasure for me to pursue my PhD degree under his supervision.

I am very grateful to the examiners of this thesis for their reviews and helpful feedbacks on this thesis.

I would like to thank Huynh Dinh Bao Phuong and Nguyen Minh Trung for many helpful discussions. I own my sincere thanks to my senior, Yu Weimiao, for his friendly help from the very first day I come to NUS. I also wish to warmly thank Mr. Yee Choon Seng, Ms. Ooi-Toh Chew Hoey, Ms. Tshin Oi Meng, and Ms. Hamidah Bte Jasman for their sympathetic help during my work in this Lab.

I would like to gratefully acknowledge the encouragement of my lab-mates and friends in Singapore - Zhao Meijun, Wang Qing, K. V. R. Subrahmanyam, Tran Thi Quynh Nhu, Dau Van Huan, Nguyen Tan Trong, and Do Tram Anh.

I owe the deepest gratitude to my mother and my husband for their love and supports. Furthermore, thanks my dear daughter, Chouchou, I am sorry for leaving her in the care of my mother during the last eight months. She gives me the motivation for going through difficult moments.

The financial support of the National University of Singapore is gratefully acknowledged.

Table of Contents

Summary

The results of analysing images reveal a lot of important information. In most cases, the information lies at the sharp transitions of intensity between pixels. When images are blurred, the information of images may be lost because the sharp transition of intensity between pixels becomes the smooth transitions of intensity in an area, thereby resulting in blurring. Deblurring has been an interesting problem during the last few decades in many areas such as: manufacturing industry, medical or satellite image analysis, and astronomy. However, deblurring is a challenging task because of its ill-posed inverse characteristics and lack of information about blurring phenomenon and its cause.

In this thesis, a new adaptive image model is introduced to deal with the deblurring problem. The proposed model which is constructed from a variant distributed line field is called LiFeAIM, which stands for Line Field based Adaptive Image Model. We use the model in a denoising algorithm to examine its goodness in image restoration. The experimental result is competent when comparing with the existing denoising algorithms. The convergent condition and convergent speed of the proposed denoising algorithm are also studied. We then use the model to construct blind deblurring algorithms by applying the Variational Bayesian approach developed in this thesis. In these blind deblurring algorithms, the covariance matrix of image is not assumed to be circulant and cannot be diagonalised by Fourier transform. Hence, the proposed deblurring algorithms must calculate the inversion of very huge matrices. To solve this numerical calculation problem, we propose and prove several

theorems to make the implementation of algorithms practical and to accelerate the computational speed. We also investigate the sensitivity of proposed algorithms to noise and initial parameters. Moreover, we apply the cross validation method to increase the accuracy of blurring estimation.

We make a comparison among the blind deblurring algorithms which use the Variational Bayesian approach and different image models such as Total Variation model, Simultaneous Auto-Regression model, and LiFeAIM. The experimental result show that the adaptive image models, Total Variation model and LiFeAIM, are more effective in deblurring.

Keywords: blind deblurring, ill-posed inverse problem, line field, LiFeAIM, Variational Bayesian approach, blurring estimation, original image estimation, circulant matrix, cross validation.

List of Figures

List of Tables

List of Symbols

The important symbols used in this thesis are listed here. The other terms are described later when they appear in the thesis.

C	the circulant matrix derived from the Laplacian operator
f	the original image
F	the left-wise circulant matrix whose first row is f'
g	the blurred observation
h	the blurring function, also called the Point Spread Function (PSF)
H	the circulant matrix whose first row is h'
ISNR	Improved Signal to Noise Ratio
$l(i,j)$	the line field which is imaginary random variables representing the bond between pixels i and j
M	the support size of blurring function which is lexicographically re-ordered into the vector form
n	the contaminated white Gaussian noise
N	the dimension of observation which is lexicographically re-ordered into the vector form
PSNR	Peak Signal to Noise Ratio
$T(k)$	the "temperature" parameter in the image model

β_{bl}	the parameter of blurring model
β_{im}	the parameter of image model
β_n	the inverse of noise variance
$\mu_\Delta(k)$	the mean of the line field distribution at step k
σ_i^2	the conditional variance of image model
σ_n^2	the variance of contaminated noise n
$\sigma_\Delta^2(k)$	the variance of the line field distribution at step k

Chapter 1
Introduction

1.1. Blurred image and point spread function (PSF)

The digital technology we have today allows us to capture a scene in a thousandth of a second. The graphic information we obtained is stored as a digital image. A digital image is a two-dimensional matrix of pixels which reflects a real scene at a specific view through an optical lens on the image plane of camera. However, sometimes, for various reasons (e.g. long shutter time of camera), each pixel of the captured image may end up as a combination of adjacent regions in the actual scene instead of a single region. When this happens, we get a blurred image of the captured scene and this combination is characterized by a kernel blurring function, called the Point Spread Function (PSF). On the blurred image, most details and patterns of the real scene are lost due to the reduction of intensity transition between pixels, which demarcates different individual regions in the scene. Consequently, we are unable to obtain the expected clear information from the blurred image.

This blurring phenomenon can happen due to different reasons. For example, we may get a blurred photographic image because the camera is not held steadily during the exposure. A blurred image may also be the result of the object movement or the out-of-focus phenomenon. Specifically, in astronomy, a blurred image can be caused by the movement of the air between the camera and the object. With various causes, the blurring problem is obviously an issue in many areas, such as in manufacturing, medical image registration, satellite domain, and astronomy.

To solve the blurring problem, the "original" image, reflecting the real scene without blurring phenomenon, must be estimated from captured image with some prior knowledge about the real scene and the PSF. This is known as the deblurring task which will be discussed in the next section.

1.2. Deblurring problem and noise effect

It is essential to model the blurring process first before dealing with the inverse problem, the deblurring process. The blurring process can be represented mathematically by the following equation:

$$g = h \otimes f \tag{1.1}$$

where g is the captured image; h is the PSF; and f is the original image.

From equation (1.1), we have only one equation with two unknown variables - the PSF and the original image - for solving the deblurring problem. Thus, to estimate the original image, we must know the PSF. Instead of finding the blurring kernel function, most previous studies assumed that the PSF was known. Then, the original image was estimated by solving the inverse problem in frequency domain [01-03], in time – frequency domain [04-08], or in spatial domain [09-17]. However, even if the PSF is known, deblurring is still not an easy task because it is an ill-posed inverse problem. For that reason, a small noise in the observed image is amplified and affects dramatically the deblurring result. When dealing with the deblurring problem, we should therefore consider the denoising problem at the same time. Unfortunately, these two tasks are conflicting with each other. While denoising tends to make the image less contrastive at some noisy pixels, deblurring increases the contrast of the image to make details clearer. This situation makes the deblurring problem more challenging for researchers during the last few decades.

However, the above mentioned studies [01-17] are incomplete because the PSF is unknown and needs to be estimated in all cases. Some researchers tried to solve the problem completely without making the assumption about PSF. Some studies tried to estimate the PSF in a separate algorithm for some specific cases, such as: camera moving uniformly in horizontal direction, and object being out-of-focus [18-27]. A few recent studies integrated the estimation of the PSF and the original image in a unique algorithm, called a blind deblurring algorithm [28-33]. These authors proposed an iterative algorithm in which the estimates are gradually improved.

Although estimating the PSF is a remarkable contribution of the above studies, none of these blind deblurring algorithms consider an adaptive image model which describes the high variation of intensity around the edges. It is well-known that the edges are the key elements of the image as the real scene can be sketched out by edges. However, the position of the edges is difficult to determine in a blurred image because the sharp transition at edges becomes smoother in an area, called the edge areas. Thus, it would be of interest to use an adaptive image model in the deblurring problem in order to carefully treat the edge areas in the deblurring problem. This thesis will propose a new adaptive image model based on the line field and use it to construct blind deblurring algorithms.

1.3. Objectives

The main objective of this thesis is to attempt to solve the deblurring problem using a new adaptive image model. We will estimate the clear image of the real scene from only one noisy blurred image of this scene. In our context, the blurring phenomenon is characterized by a spatially invariant PSF and the contaminated noise is an additive white Gaussian random process. The specific objectives of the thesis are:

- To construct an adaptive image model based on the line field model.
- To examine the proposed model's performance for image restoration by using it for the denoising problem.
- To solve the deblurring problem using the proposed model and the Variational Bayesian (VB) approach. The VB approach enables us to estimate both the original image and PSF. Thus, the deblurring problem can be solved as a whole.
- To demonstrate the efficiency of the adaptive image models in dealing with the deblurring problem by comparing the results of different deblurring algorithms which use the same approach but with different image models.

The proposed adaptive image model has two advantages in dealing with deblurring problem. Firstly, this model is implemented in the spatial domain that enables us to deal with denoising and deblurring at the same time. It is therefore well suited for this ill-posed inverse problem. Secondly, in our image model, the conditional variance, characterizing for the local variation of light intensity, is a varying parameter instead of a constant. This parameter is calculated from a random process - the line field of image. Therefore, it gives us a powerful tool to restore the edges, containing most of the lost information in the blurred image, by applying the stochastic theory in calculating the existence probability of edges. The stochastic theory is indispensable in this case because it is difficult to determine exactly the position of edges in a blurring problem.

To explore the efficiency of the proposed model in deblurring, our proposed blind deblurring algorithm will be compared with three other blind deblurring algorithms using the VB approach. Two among these algorithms are constructed from the Total Variation (TV) image model which is an adaptive image model. The other one, which uses Stimulate Autoregressive (SAR) model, is adopted from the work of Molina et al. [30]. These three algorithms use some approximation so that they can be implemented in the frequency domain. It is expected that the algorithms using adaptive image models, the TV model and the model proposed in this thesis, would yield better results.

1.4. Outline of the thesis

Chapter 2 reviews the state-of-art in deblurring. A lot of deblurring studies which have been done in the past few decades are classified following the domains that the deblurring process involved, such as: the spatial domain, the Fourier domain, and the wavelet domain. Chapter 3 introduces a new image model which is constructed from the line field. Since denoising is simpler and often incorporated into deblurring process, a denoising algorithm is constructed to examine the goodness of this model before it is used in Chapter 4 for deblurring. In Chapter 4, several theorems are also proposed and proven to help in accelerating the proposed deblurring algorithms. The experimental result of the proposed deblurring algorithms is presented in Chapter 5 with different types of blurring cause. The cross validation approach is also combined with the proposed algorithms to reduce the effect of noise during the estimation of blurring matrix. Chapter 6 compares the restoration results of four blind deblurring algorithms using the Variational Bayesian approach. Two among them are our proposed algorithms using the Total Variation model and the proposed image model in Chapter 3. The other two are the recent deblurring studies using the Simultaneous Auto-Regression model and the Total Variation model. The efficiency of these image models in deblurring is compared while they are used to construct the deblurring algorithms with the same approach and carry out experiments in the same condition. The work reported in this thesis is concluded in the last chapter, which also gives suggestions for future work.

Chapter 2
Literature Review

2.1. Introduction

There are three common kinds of blurring systems: single input- single output (SISO), single input – multi output (SIMO), and multi input – multi output (MIMO) system. In the SIMO system, one camera registers several images of the same scene under different environmental conditions. This case only occurs in some specific applications [34-37]. The most common case of MIMO blurring system is a blurred colour image [38-40]. The spectral channels of the colour image are, then, blurred by the same blurring function. However, the different channels may be contaminated by different noise signals. Depending on the correlated characteristics of the noise signals, these channel signals are processed dependently or independently. In the review of the state of the art below, we are only interested in the single input – single output (SISO) system because it is the blurring system of interest and the most common one in research, as well as in reality. In the SISO blurring system, the original image is restored from only one blurring grayscale image. It is also notable that the study of the SISO system is a basic step for solving the MIMO system when each channel of MIMO system is considered as a SISO system.

The blurring problem is a very common problem as blurring phenomenon occurs in many areas, such as: manufacturing industry, medical image registration, satellite domain, or astronomy. As a result, many researchers have studied the deblurring problem during the last few decades. The state-of-the-art of deblurring problem may be classified in many different ways. An image deblurring algorithm may be classified as a non-iterative or an iterative deblurring algorithm, a non-parametric or a parametric deblurring algorithm, and global or spatial deblurring algorithm [41]. Deblurring studies also can be classified following the methodology which is used, such as: *à priori* blur identification methods, ARMA parameter estimation methods, non-parametric methods based on high order statistics, methods using wavelet transform, methods using neural network [42-44].

In this chapter, the review of deblurring studies will be introduced following the domain in which the deblurring process is implemented. A deblurring algorithm is presented in section 2.3 where the deblurring process is implemented in the image domain, called the spatial domain. Meanwhile, a deblurring algorithm is presented in section 2.4 where the deblurring process is implemented in the frequency domain, also called the Fourier domain, or in the time – frequency domain, called the wavelet domain. However, all blind deblurring algorithms are described in a separate section, section 2.5, to show our interest in the blind deblurring problem. The general mathematical formulation of the blurring problem is briefly introduced in the next section.

2.2. Problem formulation of image deblurring

Denote g and f as the observed and original images, respectively, and h as a spatially invariant blurring function. Then the blurred image can be modeled by the following equation:

$$g(x,y) = \sum_{u,v} h(x-u, y-v) * f(u,v) = h \otimes f \qquad (2.1)$$

This inverse problem is an ill-posed inverse problem in which small errors (noise) in g will be dramatically amplified in the estimate of original image f. Hence, it is necessary that the blurring model should take noise into account, i.e.

$$g = h \otimes f + n \tag{2.2}$$

where $n \sim N(0, \sigma_n^2)$ is assumed to be a white Gaussian noise with zero mean and variance σ_n^2. A white Gaussian noise is an identical and independent distributed (*i.i.d*) Gaussian noise.

Beside a few studies dealing with spatially variant blurs [45-49], most deblurring studies are interested in the blurring problem caused by the spatially invariant blurring function because of its simplicity and wide application. In this case, the multiplying operator between h and f becomes a convolution. Since our work concerns the spatially invariant blurring function in this thesis, the “deconvolution stage” term is used, from now on, to indicate the inverse process in which a sharper image is estimated from the blurred observation g. This term is used to distinguish from the denoising stage in cases where the deblurring algorithm consists of two stages, the deconvolution and denoising stages. If the deblurring algorithm does not separate the deconvolution and denoising tasks, the “deconvolution” term is equivalent to deblurring.

To simplify the deblurring problem, many researchers have assumed that the blurring function was known. Hence the original image was estimated by constructing an inverse filter of h and using the observed image g as its input. As mentioned in the previous section, these deblurring studies can be classified into two main branches following different domains in which the deconvolution task is implemented. The first branch includes studies which implement the deconvolution task in the spatial domain, the original domain. The second branch includes studies which implement the deconvolution task in the frequency domain or in the time –frequency domain, the transformed domain. The studies of the first branch has an advantage in the possibility of combining the deconvolution task and the denoising task into a unique stage. The studies implementing the deconvolution task in the frequency domain take an advantage in the computational time with an assumption of circulant matrix. Meanwhile, the studies implementing the deconvolution task in the time – frequency domain have an advantage in suppressing the noise effectively while still preserve the detail of the image. Each of these branches will be introduced in the following sections with some examples of typical studies.

2.3. Deconvolution in the spatial domain

To implement the deconvolution and denoising tasks together, some authors have proposed deblurring algorithms in the spatial domain. As mentioned above, the Fourier domain is good for the deconvolution problem in terms of computation time while the wavelet domain is effective in the denoising problem. However, to restore a noisy blurred image, constructing a hybrid algorithm based on both transforms leads to the separate implementation of each task. Hence, the performance of the algorithm is limited. This limitation can be avoided by implementing deconvolution and denoising in the spatial domain at the same time. On the other hand, by adopting the implementation in the spatial domain, the important information of image, such as edges, can be carefully processed. This idea has been developed by many researchers and gives promising results. These studies can be classified in two main groups. One follows the regularised method, and the other employs the Bayesian framework.

2.3.1. Regularised methods

The regularised method is used in many ill-posed inverse applications. Each algorithm of this method is characterised by an energy function. The target of the regularised method is to find an estimate which minimises the energy function. In the image deblurring problem, the energy function is usually composed of two terms as follows:

$$J(f) = \|g - h \otimes f\|^2 + \lambda\phi(f) \qquad (2.3)$$

The first term of the right-hand side of the equation is the data fitting term which is related to noise affecting the data. The second term is the regularisation term which is the product of a regularisation coefficient λ and a non-negative potential function $\phi(f)$. The potential function $\phi(f)$ is used to guarantee the smoothness and sharpness of the restored image. It normally consists of a quadratic form of the differential between each pixel and its neighbouring pixels. This differential term helps to keep the smoothness at the smooth regions of the restored image in this ill-posed inverse problem. However, this term may also yield to over-smoothing the edges of the restored image. To achieve better deblurring result, regularised deblurring studies usually treat the edge regions of blurred images specifically or add some other terms into the potential function to sharpen the edges. These studies are called edge-preserving regularisation. Some examples of the added terms are the total variation of images [10], and the anisotropic diffusion equation [50].

In an edge – preserving algorithm, called ARTUR - [11], an auxiliary variable was added into the ordinary potential function $\phi(f)$ to make the optimum energy problem to be solved easily. The study provided the general form of the added term for $\phi(f)$, a strictly convex and decreasing function. The most important contribution of this study is the proving of convergence of the proposed algorithm under some assumptions. The study also described several deblurring experiments with three different edge-preserving potential functions and showed promising results.

While the ARTUR algorithm added the auxiliary variable to the potential function, the segmentation - based regularisation algorithm, proposed by Mignotte [13], used a segmentation technique to preserve the edges. In this algorithm, the potential function was constructed from the difference between a pixel and the average of partition regions instead of that between it and its neighbours. The partition regions were determined from an initial image which was estimated by the Wiener inverse filter.

The Total Variation model was assessed to be efficient in preserving the sharp contours and block features of images. By assuming that the total variation of images had an upper bound, the total variation of images was included in the potential function of a regularised deblurring algorithm [10]. The theory of sub-gradient projections was applied in this study to reduce the computational intensity of the optimisation problem.

It should be noted that the deblurring algorithm following this method must choose a suitable value for the regularisation coefficient λ. This is a challenge of the regularised method. Another challenge in using this method is to determine an appropriate potential function to preserve the edge of image as much as possible.

2.3.2. Bayesian methods

The main idea of Bayesian methods is to draw inferences which take into account of the prior distribution of parameters of interest. The Bayesian inferences are then used to make decision or to estimate the hidden data from a particular observed data set [51]. The most common methods using Bayesian inferences are Maximum Likelihood

(ML) and Maximum *à posteriori* (MAP). Some examples of deblurring studies using Bayesian methods are introduced in this section.

Note that *f* and *g* are the original and observed images, respectively, as stated above. The MAP approach is based on the basic Bayes' formula as given in the equation below:

$$p(f|g) = \frac{p(g|f)\,p(f)}{p(g)} \tag{2.4}$$

If there are unknown parameters in the above probability distributions, these parameters, denoted as Θ, are necessary to be estimated. The probability of unknown parameters Θ is added to the formula as in equation (2.5):

$$p(f,\Theta|g) = \frac{p(g|f,\Theta)\,p(f|\Theta)\,p(\Theta)}{p(g)} \tag{2.5}$$

In general, the probability of the observed image *g* given the original image *f* and the parameters Θ is the distribution of noise which is assumed to be white Gaussian. The probability of the original image *f* given Θ and the probability of Θ depend on the prior knowledge about the image and assumptions about the image model. As these probabilities are often in the exponential form, the criterion function of algorithms is constructed from their logarithm. The target of algorithms using MAP approach is to estimate *f* and Θ in order to optimise the likelihood probability or the posterior probability.

$$\left[f^{*},\Theta^{*}\right] = \arg\min_{f,\Theta} J_B(f,g,\Theta) \tag{2.6}$$

whereas,

$$J_B(f,g,\Theta) = -\log\left[p(g|f,\Theta)\,p(f|\Theta)\,p(\Theta)\right]$$

or:

$$J_B(f, g, \Theta) = -\log\left[p(g|f, \Theta)\right]$$

Depending on different assumptions about the image model, many studies have been developed in this framework.

Using the maximum *à posteriori* (MAP) approach, a deblurring algorithm was established with the modified Iterative Conditional Mode (ICM) and Simulated Annealing (SA) scheme [38]. The proposed deblurring algorithm was extended from the original ICM and SA algorithm which was investigated very widely in the denoising problem. The proposed algorithm used compound Gauss-Markov random fields, including the intensity field, the line field of the image, and the noise field. Although the global convergence of the original ICM-SA algorithm was proven, that of the modified ICM - SA algorithm was very complex to prove.

Another example of an algorithm using the line field in deblurring was the deblurring algorithm with a new adaptive image model [14]. The parameters of this image model were determined from four line processes which are oriented following the horizontal, vertical, diagonal, and sub-diagonal directions. The Gaussian distribution of these line fields was characterized by an inverse variance which was assumed to be a Gamma random variable and updated during the iterative steps of the algorithm. This assumption did not restrict the result of algorithm because the inverse variance parameter would be updated during the iterative steps of algorithm. This proposed algorithm had a challenge of determining the variation of parameters in the Gamma distribution during iterations to improve its convergence.

The MAP approach and Markov random field was also used in [09, 52] to construct a deblurring algorithm. This algorithm decomposed the blurred noisy observation into two sub-images and treated the edges and smooth regions of observed image separately. The shift-variant regularisation was applied at the edges while the shift-invariant regularisation was applied at the smooth regions. The Sherman-Morrison matrix inversion lemma was employed to reduce the computational complexity.

As mentioned in the previous section, the Total Variation model was known as an efficient model in preserving the sharpness of images. This model was also used to modeling the image in a deblurring algorithm following the Bayesian framework [16]. The unknown parameters in this study were assumed to be Gamma distributed random variables. Although the initial distributions of these parameters were given, they would not affect the final restored result as these distributions were updated during the iteration of algorithm.

As wavelet transformation is an efficient tool for denoising, combining the wavelet domain and the spatial domain in deblurring is an interesting idea. The study in [15] applied the MAP approach to deconvolve the blurred noisy image in the spatial domain and used wavelet shrinkage to remove the noise efficiently. The algorithm used Fourier transform as a tool for efficient numerical computation. The authors indicated that the algorithm performed well with various wavelet transforms such as orthogonal Discrete Wavelet Transforms (DWT) and undecimated DWT. The results of this algorithm relied on the initial image estimated by the standard Wiener inverse filter in the Fourier domain. In addition, the results were also affected by an adjustable parameter which was the ratio between noise suppressed in the deblurring step and in the denoising step.

Beside the regularized method and the approaches in the Bayesian framework, constructing the inverse filter is also an interesting direction for deblurring in the spatial domain [53]. It is also notable that the regularised method and the approaches in Bayesian framework sometimes yield the same algorithm. For instance, the logarithm form of the posteriori distribution in the MAP approach can be considered as the energy function of the regularised method. Examples of this analogue are studied in [54, 55] whose regularised functions can be interpreted by the MAP approach.

Deblurring in the spatial domain has an advantage in suppressing the noise and recovering the sharpness of the estimated image simultaneously. In the spatial domain, the detail of image can be recognized and treated with care. However, many researchers are still interested in seeking efficient deblurring algorithms in the other domains, such as the frequency domain and the time-frequency domain.

2.4. Deconvolution in the transformed domain

There are two transformed domains which are used for the deconvolution problem. One is the frequency domain, also called the Fourier domain, where the Fourier transformation is used to map data from the spatial domain to the frequency domain. The other is the time-frequency domain, called the wavelet domain, where the wavelet transformation is used to map data from the spatial domain to the time-frequency domain. Each domain has its own advantages in dealing with the deconvolution problem.

2.4.1. Deconvolution in the frequency domain

The Fourier transform is widely used in deblurring because the inverse of a blurring matrix can be found more easily in the frequency domain. With a spatially invariant PSF, the operator between the blurring function h and the original image f is a convolution which becomes an ordinary multiplication in the Fourier domain. Hence, the inversion problem can be implemented very rapidly by inverting scalar coefficients at each frequency. The deblurring studies using this approach often use the inverse or Wiener inverse filter in the Fourier domain (shown below) for the deconvolution process and another filter for the denoising process [01-03, 56].

The regularised inverse filter is given by:

$$F(\omega) = \frac{H(-\omega)}{|H(\omega)|^2 + \varepsilon} G(\omega) \tag{2.7}$$

and the regularised Wiener inverse filter by:

$$F(\omega) = \frac{H(-\omega)}{|H(\omega)|^2 + \varepsilon \frac{\sigma_n^2}{|\hat{F}(\omega)|^2}} G(\omega) \tag{2.8}$$

where, $F(\omega)$, $G(\omega)$, $\hat{F}(\omega)$, and $H(\omega)$ denote the Fourier transform of the original image *f*, observed image *g*, estimated image $\hat{f}$ and the blurring matrix *h*, respectively, and

σ_n^2 is the variance of the white Gaussian noise.

ε is the regularisation parameter.

As illustrated in the above equations, a regularization parameter is usually added to these inverse filters to avoid the division by zero error and to reduce the amplification of noise. However, the regularization parameter needs to be fine-tuned in order to achieve the compromise between suppressing noise and preserving image contents. As a consequence, these filters often are not able to effectively remove noise. It is crucial to perform piecewise-smoothing to the estimated image after deconvolution. For example, several algorithms use the Wiener filter with the Fourier transform for the deconvolution stage and the wavelet shrinkage for the denoising stage.

The wavelet transform is known as a powerful tool in denoising. Unfortunately, the wavelet transform is difficult to use directly for deconvolution because the problem becomes very complicated when the two-dimensional image is represented in four-dimensional space. Hence, the Fourier transform and wavelet transform have been combined into an algorithm to exploit their advantages in deconvolution and denoising. Some studies which have used this idea are introduced below.

An example of an algorithm which used the inverse filter in the frequency domain was ForWaRD algorithm, standing for Fourier –Wavelet Regularized Deconvolution algorithm [02]. This algorithm implemented the deconvolution process in the Fourier domain and the denoising process in the wavelet domain. It consisted of two shrinkage procedures. One was used for Fourier coefficientswhile the other was used for wavelet coefficients. It was a simple and effective algorithm in comparison with the existing studies. However, it was challenging to find the optimal value for the regularization parameter balancing between the Fourier and wavelet shrinkage. If the regularization parameter was high, the algorithm would suppress more noise but some image details would be lost and vice verse. Another example of combining the Fourier domain and the wavelet domain was the study in [03]. This study used the Wiener filter in the Fourier domain and applied a shrinkage process for Fourier coefficients. In the wavelet domain, a Bayesian approach applied to the hidden Markov model of wavelet coefficients.

Instead of denoising the image in wavelet domain, some studies implemented the denoising stage in the spatial domain while the deconvolution stage was implemented in the Fourier domain. For example, The LPA-ICI algorithm piecewise-smoothed the noisy blurred image by an adaptive Local Polynomial Approximation (LPA) method [01]. Firstly, the deconvolution process was solved in the frequency domain with a regularized inverse filter. An additional term of the filter was the Fourier transform of the approximation kernel. Secondly, the denoising process was implemented in the spatial domain based on the Intersecting Confidence Intervals (ICI) theory. In essence, a series of adaptive window sizes were chosen for each pixel from different noisy deconvolution estimates corresponding to different kernels. The final result was the weighted average of results in different directions, which might lead to a slight blurring in the obtained result. By using this result as an initial estimate, a similar algorithm in which the regularized inverse filter was replaced by the regularized Wiener inverse filter was suggested. The latter algorithm improved the preliminary result further. However, these results also depended on the regularization coefficients of inverse filters.

The studies introduced in this section have an advantage in computational time as the problem of inverting a big blurring matrix becomes the inverting of scalars. However, their performance is limited by the value of the regularization parameter of inverse filters which needs to be adjusted. The parameter must be fine-tuned in order to achieve the compromise between removing noise and preserving the image contents. Another disadvantage of these algorithms is that they often consist of two separate steps. The first step is deconvolution in the Fourier domain. The second is piece-wise smoothing the result of the first step in another domain, such as the wavelet domain or the spatial domain. Therefore, the effect of noise would be amplified through the first step. This will limit the performance of algorithms.

2.4.2. Deconvolution in the time - frequency domain

The wavelet transform is an effective and powerful tool for denoising. It is well-suited for denoising tasks because the noise is still white Gaussian, whereas the signal components are concentrated into a few coefficients in the wavelet domain, also called the time – frequency domain [57]. This important principle is capable of separating the signal from noise, thereby making the wavelet transform powerful for estimating data with sharp discontinuities such as edges. The efficiency of this denoising approach depends on choosing a proper shrinkage threshold. There were many techniques for estimating the shrinkage threshold such as RiskShrink [58] using a soft-threshold operator and minimizing the mean squared error; VisuShrink [58] as a global optimal threshold in the minimax sense of RiskShrink; SureShrink [59] minimizing Stein's unbiased risk estimate; or BayesShrink [60] performing a data-driven, subband-dependent threshold.

In the previous section, many deblurring algorithms use the wavelet transform for denoising after implementing the deconvolution stage in the spatial or the Fourier domain [02, 03, 15]. This section will introduce the deblurring algorithms which implement the deconvolution stage in the wavelet domain, the time – frequency domain [04-08].

Although the wavelet transform has an advantage in denoising in comparison with the Fourier transform, deblurring using the wavelet transform is more difficult because the convolution between the blurring function h and the original image f does not become a multiplication in the wavelet domain. Hence, the inversion problem is almost impractical in the wavelet domain. To deal with this computational problem, some studies simplify the problem by adding assumptions of wavelet coefficients and use iterative methods to solve the optimization problem.

Similar to deconvolution studies in the spatial domain, most deconvolution studies in the wavelet domain used the Bayesian framework or the regularised method. An example of studies using the Bayesian framework and the discrete wavelet transform was the generalised expectation maximisation deblurring algorithm [06]. This algorithm examined different types of Gaussian scale mixture densities to describe the prior distribution of wavelet coefficients, such as Laplace, Hardy, Jeffreys, generalized Gaussian, and garrote density. To solve the optimisation problem of MAP, this study used the expectation maximisation method and the second-order stationary iterative method.

Another example of studies applying the Bayesian framework for the wavelet coefficients of image is reported in [07]. This study used the MAP approach and the dual-tree complex wavelet transform. To simplify the problem, the prior distributions of wavelet coefficients of images are assumed to be independent. In addition, the variances of the real and imaginary parts of each wavelet coefficient are assumed to be equal. The conjugate gradient method is applied to solve the optimisation problem.

The regularised method in the time-frequency domain was used in an adaptive regularisation deblurring algorithm [05]. The weakness of the regularised method was how to choose the appropriate regularised coefficient. In this algorithm, the regularised coefficient was determined in the adaptively regularised constraint total least squares method. To reduce the computational effort, the study considered only one-level wavelet decomposition.

As described above, there are many deblurring studies which use different approaches and are implemented in different domains. Each algorithm has its own advantage in deblurring and gives promising restored results. However, the above mentioned studies are incomplete because they assume that the blurring function h was known. In fact, the blurring function is unknown and needs to be estimated in all cases. Some studies which try to solve the problem completely will be presented in the next section.

2.5. Blind deblurring - the dual problem

To estimate the original image from the observation, it is crucial to know the blurring function. In practice, the blurring function is unknown and it is very difficult to determine the blurring function from a degraded observation. The works which deal with this problem are called *blur identification*. However, blur identification and image restoration are two dual problems where one is estimated given the other and vice versa. Thus, we need a unified approach to solve the two problems jointly. The problem of restoring the original image without complete knowledge of blurring function is called blind deblurring.

There are two typical approaches for the blind deblurring problem. In the first approach, the blur identification procedure is realized in a separate step to estimate the blurring function. Then, any available deblurring method is used to estimate the original image. In the second approach, the blur identification and the image restoration procedure are incorporated in a unifying algorithm. They could be often estimated alternatively in an iterative algorithm. The precision of estimation will be improved through each step. These two approaches will be introduced below.

2.5.1. Blur identification

To deal with the blind deblurring problem, some studies estimated the blurring function, or the PSF, and used an available deblurring algorithm in the literature to examine the accuracy of PSF estimation through the restored image. In these studies, the PSF is often investigated as a specific case, such as the uniform horizontal moving blur, the out-of-focus blur, and the truncated Gaussian blur. These PSFs are assumed to have specified parametric forms and determined by one or several parameters. Their characteristic parameters may be the blur extent, the defocused radius, the blurring radius, or the variance of the coefficients. Some examples of specific blurring models are given below.

When there is the horizontally uniform relative movement between the camera and the captured object, the PSF has the following form:

$$h(x) = \begin{cases} \dfrac{1}{d} & 0 \le x \le d \\ 0 & otherwise \end{cases} \tag{2.9}$$

where d is the extent of the motion.

When there is the out-of-focus phenomenon in capturing the object, the PSF is characterized as:

$$h(x,y) = \begin{cases} \dfrac{1}{\pi R^2} & x^2 + y^2 \le R^2 \\ 0 & otherwise \end{cases} \tag{2.10}$$

where R is the radius of the out-of-focus function.

When the movement of the air between the camera and the object affects the process of image registration, called the air turbulence phenomenon, the PSF is modeled as follows:

$$h(x,y)=\begin{cases}\dfrac{1}{2\pi\sigma^2\left(1-\exp\left(-R^2/2\sigma^2\right)\right)}\exp\left(-\dfrac{x^2+y^2}{2\sigma^2}\right) & x^2+y^2\le R^2\\ 0 & otherwise\end{cases} \tag{2.11}$$

where R is the extent of the blur, and σ^2 is the variance of this distribution.

There were various approaches which were used in blur identification. Some approaches are listed here. For instance, the maximum likelihood method was applied in the three above described models to determine the PSF [2]. The autocorrelation of the shadowed image, which was constructed from the blurred observation, was used to estimate the blur extent of a horizontally uniform blur in [25]. The ADALINE neural network was used to determine the elements of PSF where the blur extent was roughly estimated [20]. Although the blurring model in this study was constructed in a general form theoretically, only the non-uniform straight motion blur was considered in their numerical experiments to limit the complexity of the network. The residual spectral matching approach was used to determined the blur extent of some one-parameter blurring models in [26, 27].

In all these studies, several specific mathematical types of PSF were considered. These studies were often limited and could hardly be generalized. On the other hand, the image restoration process was employed from an available work in the literature. Hence, they lack the interaction between the PSF estimation and the original image estimation, of which result would affect that of the other, and vice verse. A few studies filled this gap by integrating the estimation of the PSF and the original image in a unique algorithm, called a blind deblurring algorithm. This was often an iterative algorithm in which the estimates were gradually improved.

2.5.2. Blind deblurring- Unifying algorithms

The blind deblurring algorithms, in which both the blurring function and the original image were unknown and need to be estimated, were often derived in the spatial domain. There were also the blind deblurring algorithms derived in the Fourier domain [31], in which the blind deblurring problem in the study was equivalent to factorizing a two-dimensional polynomial. However, this algorithm was complex, unstable, and analysed only the noiseless observation. Some blind deblurring algorithms derived in the spatial domain would be introduced below.

Although the blind deblurring algorithms do not impose the assumption of a known PSF, they may require more prior knowledge about the original image. For example, they assume that the image is in the form of an object lying on a uniform contrast background and the object's support is known. Hence, the constraint is that the pixel outside the support would be replaced by a value corresponds to the grey level of background. The Iterative Blind Deconvolution (IBD) method [33] is one among the reported works using this assumption. The algorithm estimates the convolution matrix by a regularized Wiener inverse filter provided that the original image is approximately estimated; and vice versa. Each time the convolution matrix (image) is found in the Fourier domain, it is transformed to the spatial domain by the inverse Fourier transform to impose blur (image) constraints on it.

Another example of blind deblurring algorithms imposing special constraints on the original image is NAS-RIF algorithm [32], which stands for Non-negative And Support constraints Recursive Inverse Filter algorithm. This study assumed that the image showed an object on a uniform black, gray, or white background and that the object had a finite support. The cross validation method was employed in the case where the support size of the original object was unknown. Although the convergence of the algorithm was guaranteed, the restored result was not robust to noise.

A new approach used recently in blind deblurring studies is the Ensemble Learning approach. In this approach, not only the hidden data, the PSF and the original image, but also their model parameters are considered as random variables. All the prior distributions of the hidden data and the model parameters are given and approximated by simpler distributions. The approximated distributions are estimated by the Kullback-Leibler divergence [61]. Different blind deblurring algorithms will be derived when different prior distributions and approximated distributions are used. For instance, in [29, 30], the original image and PSF were modeled by simultaneous auto-regressive models and approximated by Gaussian distributions, while the model parameters were modeled and approximated by Gamma distributions. However, the covariance of the hidden data must be circulant to reduce the computational complexity. Slightly different to Ensemble Learning approach, the approach in this study, termed Variational Bayesian approach, updates the approximate distributions of model parameters through each iteration.

Similar to Ensemble Learning approach, a generalisation of Expectation-Maximisation is reported in [24] to construct a blind deblurring algorithm. This study uses the Kullback-Leibler divergence to bypass the main difficulty in applying the Expectation-Maximisation method. In this study, the model parameters are considered as the deterministic variables rather than the random variables. In fact, the result of this study is the same as that of the blind deblurring algorithm which uses Ensemble Learning approach and the uniform distributions of model parameters. Similar to the previous described algorithm, this algorithm also assumes that the covariance of hidden data and model parameters were circulant to reduce the computational complexity.

Although estimating the PSF is a remarkable contribution of the above studies, none of these blind deblurring algorithms consider an adaptive image model which describes the high variation of intensity at edge areas. It is well-known that edges are the key elements of the image as the real scene can be sketched out by edges. However, the position of the edges can hardly be determined in a blurred image because the sharp transition at edges becomes smoother in an area, called the edge areas. Thus, it would be of interest to use an adaptive image model in the deblurring problem in order to carefully treat the edge areas in the deblurring problem. That is our motivation to start the research which is reported in this thesis. In the course of our research, a blind deblurring algorithm using an adaptive image model is reported by Babacan *et al*. in 2009 [62]. The difference between this algorithm and the one reported in Chapter 6 is that this algorithm uses the conjugate gradient method to calculate the covariance matrices of hidden data. More details of this algorithm will be mentioned in Chapter 6.

There are some other techniques used in the blind deblurring problem, such as the neural network and the Vector Quantisation approach. However, these techniques are only applicable in some specific cases where the training database is available. For example, the training database is used to establish a codebook in the Vector Quantisation approach or to train the network in the neural network [63-65].

2.6. Summary

Numerous deconvolution studies with assumption of known PSF were introduced in the above sections with the advantages and disadvantages of their approaches stated.

In particular, the inverse filters in the frequency domain have the advantage in computational time but their result was limited due to the need in tuning the value of their regularization parameter. The regularized approach and Bayesian framework were used in the deconvolution studies in the spatial domain as well as the studies in the wavelet domain. The limitation of the regularized approach was due to the necessity to choose the appropriate value of regularization parameters while the Bayesian approach assumed the prior knowledge about the original image. Although many deblurring studies were reported in the literature, only few of them dealt with the blind deblurring problem.

In this thesis, the Bayesian inference is used to construct unifying algorithms for blind deblurring problem. The variational approach is combined with the Bayesian inference, named the Variational Bayesian approach, in order to minimize the effect of the prior information. Two adaptive image models are considered in this work as the prior information to construct different blind deblurring algorithms. The use of adaptive image models results in high computational load but it is useful in deblurring because these models treat the sharp intensity transition of image carefully.

Chapter 3
Denoising Using Line-Field Based Adaptive Image Model

3.1. Introduction

In Chapter 1, we have pointed out that denoising plays an important role in the deblurring problem. For example, the results in Figure 3-1 show the effect of noise on the deblurring process by using a standard Wiener filter to restore original images from two blurred images. Both observations are the blurred Lena image. One of them is noise-free while the other is contaminated by a white Gaussian noise. The restoration result of the latter is worse than that of the former as shown in Figure 3-1 (b) and (d). Hence, it is crucial to process noise pixels carefully when restoring a blurred image. In this chapter, we discuss the denoising of a purely noisy image while preserving its details before dealing with the deblurring problem in the next chapter. Specifically, we construct a new adaptive image model in this chapter and use it to deal with the denoising problem. Then, the proposed model will be employed in the next chapter for the deblurring task.

As mentioned in the last chapter, wavelet transform is a powerful tool for denoising [58-60, 66-70], thanks to the consistent characteristic of the white noise through the wavelet transform. However, image patterns are too complex to analyse in the transformed domain. Hence, denoising algorithms in wavelet domain may be difficult to integrate with the other image processing algorithms such as segmentation, deconvolution, and recognition. Besides, the implementation of wavelet and inverse wavelet transformation is also a time consuming computational process. Thus, the denoising approach in the spatial domain would be more competent in these applications because it is easier to be integrated with the other image processing tasks which are related to image pattern analysis. For these reasons, we choose to process the image in the spatial domain.

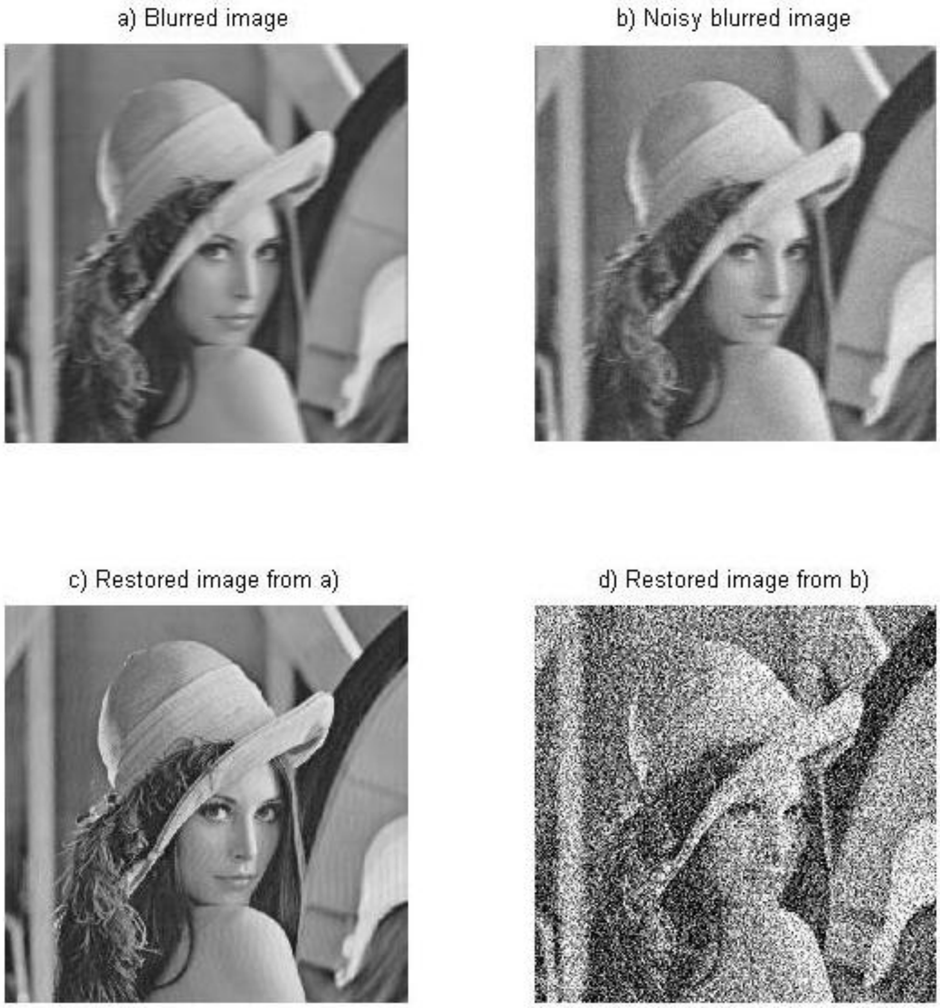


Figure 3-1. The effect of noise in deconvolution problem: the blurred image (a), the blurred noisy image (b) by the horizontally uniform blur with blurring extent *d=11* and noise variance $\sigma_n = 20$, and their deconvolution results (c), (d) by the standard inverse Wiener filter in Matlab.

The denoising algorithms in the spatial domain were based on the idea of locally smoothing the image with different smoothing coefficients. These algorithms could be regarded as spatially adaptive filters [71-74]. An example of smoothing filters in denoising was the median filter. The disadvantage of median filters was that some important details of image at transition areas may be lost. Considerable works have overcome this effect by switching among several median-filters based on some criteria [75-77]. Recently, Katkovnik [78] has proposed an efficient denoising method using the local polynomial approximation (LPA) with the adaptive window size estimated by the intersection of confidence intervals (ICI) rule. These algorithms might be applied to smooth out the image in various directions. The estimated image was finally determined by the average of the restored results following different directions to avoid the bias effect.

Getting inspiration from the similarity in the locally dependent characteristic of the image and the Markov chain, Besag [79] has proposed a probabilistic mathematical model for image processing tasks. By adding a virtual random process to this model, Geman and Geman [80] have made the model more powerful in removing the noise while preserving the details of the image since the added random process has driven the smoothing process appropriately. Applying different iterative schemes, such as Simulated Annealing (SA) scheme [80] or Iterated Conditional Modes (ICM) scheme [81], to the later model has resulted in efficient denoising algorithms that have had a better capability to preserve the details of the image [82, 83]. However, because of the convergence condition, these algorithms required hundreds of iterations with considerable computational time.

In this chapter, by using the Markov model with a variant distribution line field instead of the original line field, we propose an algorithm that may distinguish a pixel at the edge of image from noisy pixels. Hence, the noise is removed more effectively while details of the image are preserved. Specifically, from the modified line field, we reconstruct an adaptive image model based on Besag's model. In this model, its variance is not a constant but varies through the image. The fact that the variance is high at the "potential" edge pixel helps to preserve the high transition between edge pixels. This potential of each pixel is determined based on the line field. As a result, the denoising algorithm constructed from the proposed model has a good capability to detect the noise. Therefore, the convergence speed is accelerated and the computational time is reduced significantly.

3.2. Markov random field and image modeling

From the concept of the Markov chain, Besag [79] has developed a spatially interacting random process and proposed a valid probability structure for it. This spatial random process is called the Markov random field. For instance, we consider a 2D Markov chain $F = \{ f_i \}_{i=1..n}$, where *i* is a simple site index alternated for the two site indices. According to Besag's model, its conditional probability may have the form as given in eq. (3.1):

$$P\left(f_i \middle| f_j : j \neq i\right) = P(0)\exp\left\{ f_i G_i(f_i) + \sum_{j \neq i} f_i f_j G_{ij}(f_i, f_j) + \ldots + f_1 \cdots f_n G_{1..n}(f_1, \ldots, f_n) \right\} \tag{3.1}$$

where $P(0) = P(f_i = 0 | f_j : j \neq i)$

G-functions are chosen arbitrarily subject to the condition that they are only non-null at site *i* and its neighbours. The conception of neighbourhood will be described more clearly later when we apply this probability structure to modeling the image.

From this general form, we are able to construct more specific models. Considering a simple case in which the first order terms $G_i(f_i)$ and the second order terms $G_{ij}(f_i f_j)$ are linear, and the others are equal to zero, we obtain an auto-normal model:

$$P\left(f_i \middle| f_j : j \neq i\right) = \frac{1}{\sqrt{2\pi\sigma_i^2}} \exp\left\{-\frac{1}{2\sigma_i^2}\left(f_i - \sum_{j \neq i} \beta_{ij} f_j\right)^2\right\} \tag{3.2}$$

in which: β_{ij} is equal to zero unless *j* is a neighbour of *i*. β_{ij} implies the strength of an imaginary bond between *i* and its neighbour *j*.

σ_i^2 is the conditional variance of the image model which characterises the local difference around the site *i*.

Due to the analogy of locally dependent characteristics between the Markov random field and the image, this structure can be used to model the image. In image modeling, the intensity matrix of the image is considered as a Markov random field. Then, the conditional variance σ_i^2 of the above model is a parameter characterizing the local smoothness around the pixel *i*. Hence, it is obvious that σ_i^2 should be small at the smooth area and large at the edge of the image. The neighbours of a pixel are defined as the pixels around it. There are many neighbourhood models with different sizes of the neighbour set. For example, we have the first-, second-, or third-order neighbourhood model as shown in Figure 3-2. The first order model was preferred in many previous works since it required the least computational time. Unfortunately, using the first order model tends to cause the vertically and horizontally directional effect. Hence, in our studies, we consider the second order neighbourhood model to avoid this effect. Higher order neighbourhood models can be used but they will require more computational effort.

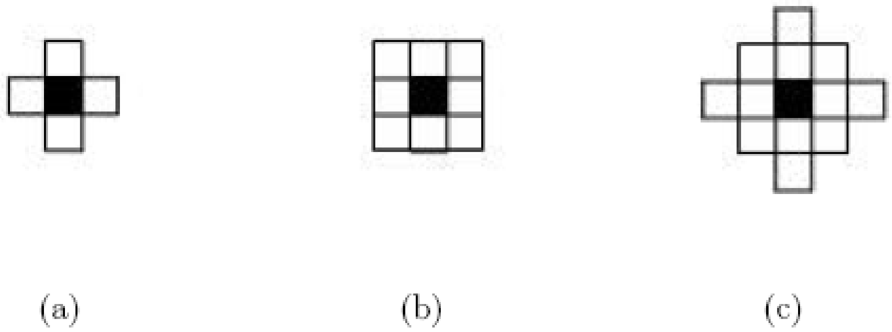


Figure 3-2. Different neighbourhood models: the first (a), second (b) and third (c) order neighbourhood model.

3.3. Line field with variant distribution

From Besag's model, Geman and Geman [80] have constructed a new model by combining the original model with another Markov random field. In this new model, the image is regarded as a pair of 2-D Markov random fields, the intensity field *F* and the line field *L*. While *F* is a real random process representing the intensity at each pixel, *L* is an imaginary random process representing the virtual bond between pixels. The line field of an image is constructed from the intensity field of that image. If there is "no difference" between the intensity of a pixel and that of its neighbour, it is said that there does not exist a bond between them $(l(i, j) = 0)$; otherwise $(l(i, j) = 1)$ (as shown in Figure 3-3). Then, the line field is a binary random field. By adding the line field to Besag’s image model, the pixel at the edge can be distinguished from pixels in the smooth areas and noisy pixels, thanks to the number of bonds between that pixel and its neighbours. The combination between the line field and Besag’s model is realized by connecting the original model with a Gibbs distribution which is represented in the following equation:

$$P(F = f, L = l | G = g) \propto \exp\left\{-\frac{U(f,l)}{T}\right\} \tag{3.3}$$

in which, G is the intensity field of the observed image;

F and L are the intensity field and the line field of the original image;

U is the energy function of the original image which is determined from the line field and neighbourhood system;

T is called the temperature parameter of the model.

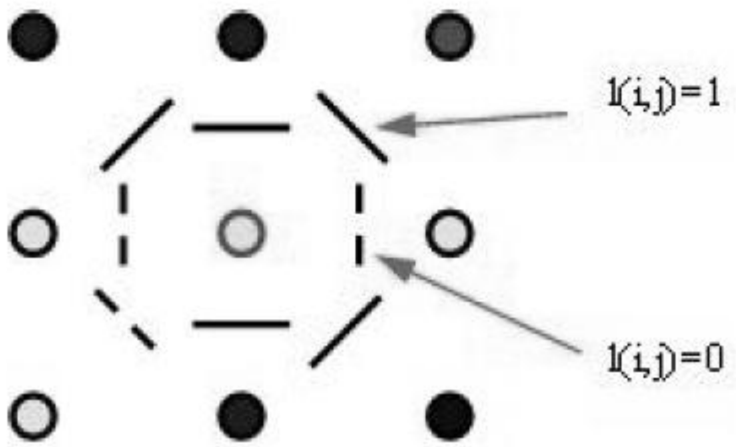


Figure 3-3. Line-field model: the neighbours of a pixel and the bonds between them *l(i,j)=1* if the bond exists between *i* and *j*; otherwise *l(i,j)=0.*

Connecting the posterior distribution in eq. (3.3) with the prior distribution in eq. (3.2) gives us an idea of calculating the parameters β_{ij} and σ_i^2 from the line field L. In this section, we first introduce our modified line field which will facilitate our effort in defining the image model in our own way. The next section will then clarify how to combine a Markov random field and a Gibbs distribution in image modeling.

The line field of an image is constructed from its intensity field. In the model suggested by Geman and Geman [80], the probability of the existence of a line between two pixels ($l(i, j) = 1$) is an invariant distribution which covers the whole variation interval of intensity difference. It may lead to the confusion between a noise and a pixel at the edge. For instance, the pixels of thin edges which lie within a smooth region will be highly and potentially considered as noisy pixels.

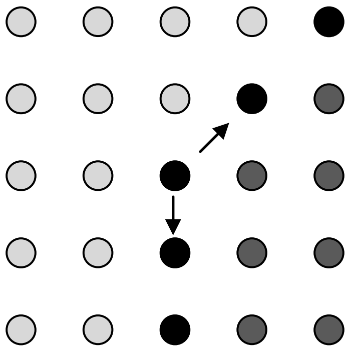


Figure 3-4. The smoothness of image at a pixel.

The inspiration of our line distribution comes from the fact that the intensity of each pixel in an image should be close to that of some pixels amongst its eight nearest neighbours. In other words, the image is smooth at each pixel in some direction. For example, in Figure 3-4 the image is smooth at the center pixel in two directions (North-East and South) which are shown by the arrows. Following this rule, pixels in an image can be classified into three categories:

- The pixel in the region where the pixels are of the same intensity is smooth in all eight directions;
- The pixel at the edge can be smooth in some directions, for instance, in the directions along the edge;
- A noisy pixel which is generally not smooth in all directions.

Therefore, a noise-free pixel could be distinguished from a noisy pixel based on the difference of intensity between a pixel and its neighbours; this idea is related to the line field concept. However, different from the line field suggested in [80], this thesis proposes a new virtual line field distribution. The 95% confidence interval of the existence of the proposed virtual line covers partially the variation interval of intensity difference. Since the 95% confidence interval covers only a limited interval of intensity difference we need to construct an iterative algorithm in this case. The probability distributions of the line cover different intervals of intensity difference at each iterative step *k* so that their combination must cover the whole variation interval of intensity difference (see Figure 3-5). The proposed distribution of the line has the following form:

$$P\big(l(i,j)\big| f, l(m,n): (m,n) \neq (i,j)\big) \propto \exp\left\{-\frac{1}{2\sigma_{\Delta}^{2}(k)}\big(\left|f_i - f_j\right| - \mu_{\Delta}(k)\big)^2\right\} \tag{3.4}$$

where, $\mu_{\Delta}(k)$ and $\sigma_{\Delta}^{2}(k)$ are the mean and the variance of the line field distribution at step *k*, respectively. As shown in Figure 3-5, $\sigma_{\Delta}^{2}(k)$ and $\mu_{\Delta}(k)$ decrease with respect to *k*. These parameters will be determined in our experimental work.

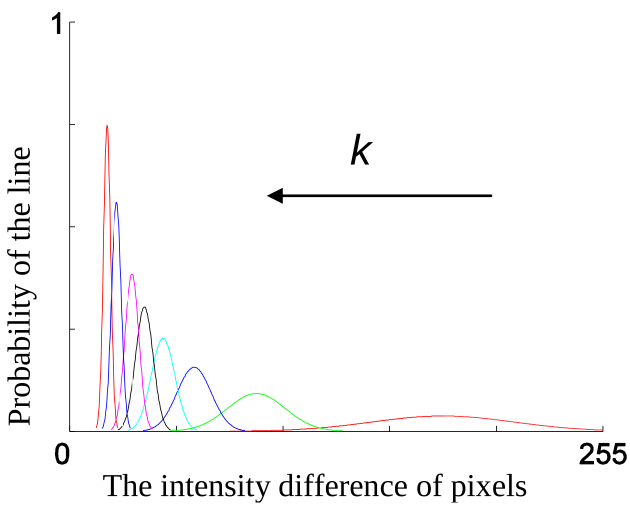


Figure 3-5. Probability distribution of the line at various iterative steps *k*.

3.4. Line-Field based Adaptive Image Model (LiFeAIM)

From the proposed line field above, we reconstruct the image model described in eq. (3.2) with parameters β_{ij} and σ_i^2 calculated from the proposed line field. As stated above, the fact that β_{ij} is non zero implies a bond between pixel i and pixel j $(l(i, j) = 1)$. Then, $l(i,j)$ could be used as a term included in β_{ij}. On the other hand, it is assumed that the observed value of intensity at a pixel is most reliable if there is no difference between its intensity and that of its neighbours. From eq. (3.2), we deduce the condition imposed on β_{ij} as follows:

$$\sum_{j \neq i} \beta_{ij} = 1 \tag{3.5}$$

Hence, the parameter β_{ij} could be written as:

$$\beta_{ij} = \frac{l(i, j)}{\sum_{j \neq i} l(i, j)} \tag{3.6}$$

We now determine the conditional variance σ_i^2, which characterizes the local smoothness around the pixel i. We should notice that σ_i^2 increases when scanning from a smooth area toward its boundaries, which are the edges within the image. Therefore, it is essential to identify whether or not a pixel lies on an edge. A new coefficient will be defined to distinguish between a point at the edge and a noisy pixel:

$$\alpha_i = \exp\left[\frac{\left(\sum_{j \neq i} l(i, j) - N\right)^2}{2\sigma_l^2}\right] \tag{3.7}$$

where N is the number of neighbours around the pixel i and σ_l^2 is the variance of the distribution of the number of lines around a pixel. α_i, called the noisy coefficient, varies in the interval [0, 1]. It is high if there is noise at the pixel i and low otherwise.

In contrast, σ_i^2 should be low at a noisy pixel. For instance, σ_i^2 may be chosen to be proportional to *(1-* α_i *)* as shown in the following equation:

$$\sigma_i^2 = \sigma_n^2(1-\alpha_i) \tag{3.8}$$

where σ_n is the standard deviation of the contaminated noise.

We now use the probability structure (3.2) for modeling the image with the parameters β_{ij} and σ_i^2 determined above. Similar to the model in eq. (3.3), the simulated annealing scheme is applied to this image model:

$$P(f_i | f_j : j \neq i) = \left(\frac{1}{\sqrt{2\pi\sigma_i^2}} \exp\left\{ -\frac{1}{2\sigma_i^2} \left[f_i - \sum_{j \neq i} \beta_{ij} f_j \right]^2 \right\} \right)^{\frac{1}{T(k)}} \tag{3.9}$$

The simulated annealing scheme is related to the temperature parameter *T(k)* which controls the convergence speed of the algorithm. When *T(k)* decreases slowly with respect to the iteration step *k*, the random process will be forced towards an minimal energy configuration. The meaning of *T(k)* comes from physical processes, such as the cooling down process of metal. The high temperature *T(k)* induces the chaotic phenomenon in which the neighbourhood elements has a loose bond. On the other hand, the low temperature *T(k)* induces tighter bonds between the neighbourhood elements which results in a more stable appearance. In the cooling down process, the temperature *T(k)* should decrease slow enough to let the elements (atoms) arrange into the right positions as in the metal crystallographic structure. Similarly, to guarantee the convergence of the iterative algorithm, it is shown that the temperature *T(k)* should satisfy the bound [80]:

$$T(k) \geq \frac{c}{\log(k+1)} \tag{3.10}$$

In this equation, the constant c is independent to the step k and is capable of controlling the speed of convergence. It is necessary to choose an appropriate value of c to achieve the desired precision while requiring as little effort as possible in computation. In this thesis, the parameter *T(k)* is assigned directly by the bound:

$$T(k) = \frac{c}{\log(k+1)} \tag{3.11}$$

3.5. Denoising algorithm using LiFeAIM

The denoising problem is modeled as follows:

$$g = f + n \tag{3.12}$$

in which, g is the observed image;

f is the noisy-free image;

n is the additive white Gaussian noise, $n \sim N(0, \sigma_n^2)$.

Hence, the conditional probability of the observed image given that the original image is a Gaussian distribution:

$$P(g_i | f_i) = \frac{1}{\sqrt{2\pi\sigma_n^2}} \exp\left\{ -\frac{1}{2\sigma_n^2} [g_i - f_i]^2 \right\} \tag{3.13}$$

The problem is solved by using the maximum *à posteriori* (MAP) approach. Following the MAP approach, we apply the Bayesian formula:

$$P(f_i | g_i, f_j : j \neq i) \propto P(g_i | f) P(f_i | f_j : j \neq i) \tag{3.14}$$

From equations (3.9), (3.13) and (3.14), we have the conditional probability of the intensity at a pixel given in eq. (3.15) below:

$$P(f_i|g_i, f_j : j \neq i) = const \times \exp\left\{ -\frac{(g_i - f_i)^2}{2\sigma_n^2} - \frac{\left(f_i - \sum_{j \neq i} \beta_{ij} f_j \right)^2}{2\sigma_i^2 T(k)} \right\} \qquad (3.15)$$

or:

$$-\log P(f_i|g_i, f_j : j \neq i) = const + \frac{(g_i - f_i)^2}{2\sigma_n^2} + \frac{1}{2\sigma_i^2 T(k)} \left(f_i - \sum_{j \neq i} \beta_{ij} f_j \right)^2 \qquad (3.16)$$

Since eq. (3.16) has a quadratic form, our optimisation problem is solved easily by taking the derivative of both sides of eq. (3.16) with respect to f_i and finding f_i. The optimal solution is obtained as follows:

$$f_i = \frac{T(k)(1-\alpha_i)g_i + \sum_{j \neq i} \beta_{ij} f_j}{T(k)(1-\alpha_i) + 1} \qquad (3.17)$$

In eq. (3.17) the noise variance term σ_n^2 has been omitted. In other words, to estimate the original image from the noisy one, it is not necessary to know its noise variance. Therefore, the algorithm which is established from this equation is a blind denoising algorithm. Moreover, it is known that $\sum_{j \neq i} \beta_{ij} = 1$. Eq. (3.17) may be rewritten as follows:

$$f_i = \frac{T(k)(1-\alpha_i)g_i + \sum_{j \neq i} \beta_{ij} f_j}{T(k)(1-\alpha_i) + \sum_{j \neq i} \beta_{ij}} \qquad (3.18)$$

Hence, we can see that the result is, in fact, a weighted-mean filter whose inputs are the observed intensity at the calculated pixel and its selective neighbours. Actually, the selected neighbours are its neighbouring pixels which have bonds with it

$(l(i,j)=1)$. Based on this filter, an iterative algorithm is proposed in the next section to denoise the 2D image.

3.6. Experimental results

To use eq. (3.17) in an iterative algorithm, we need to modify it to an appropriate form. The image estimate at step *k+1* will be calculated from the image estimate at step *k*. The modified formula is given below:

$$\hat{f}_i^{(k+1)} = \frac{T(k)(1-\alpha_i)\,\hat{f}_i^{(k)} + \sum_{j\neq i}\beta_{ij}\,\hat{f}_j^{(k)}}{T(k)(1-\alpha_i)+1} \tag{3.19}$$

Following the theoretical developments above, we now propose an iterative algorithm to solve the denoising problem:

Algorithm 1:

- Step 1: Set *k=1;*
- Step 2: Define the variance $\sigma_\Delta^2(k)$ and the mean $\mu_\Delta(k)$ of the modified line field distribution, and the temperature *T(k)*;
- Step 3: Calculate the binary line field *l(i,j)* following eq.(3.4);
- Step 4: Determine the parameters β_{ij} and σ_i^2 following eq. (3.6) and (3.8);
- Step 5: Estimate the intensity at each pixel from eq. (3.19);
- Step 6: Set *k:=k+1* and go to step 2 if the stop criterion is not satisfied. For simplicity, we define the maximum number of iteration steps as the termination criterion. In the other word, the algorithm is terminated after a specific loop number N_{loop}.

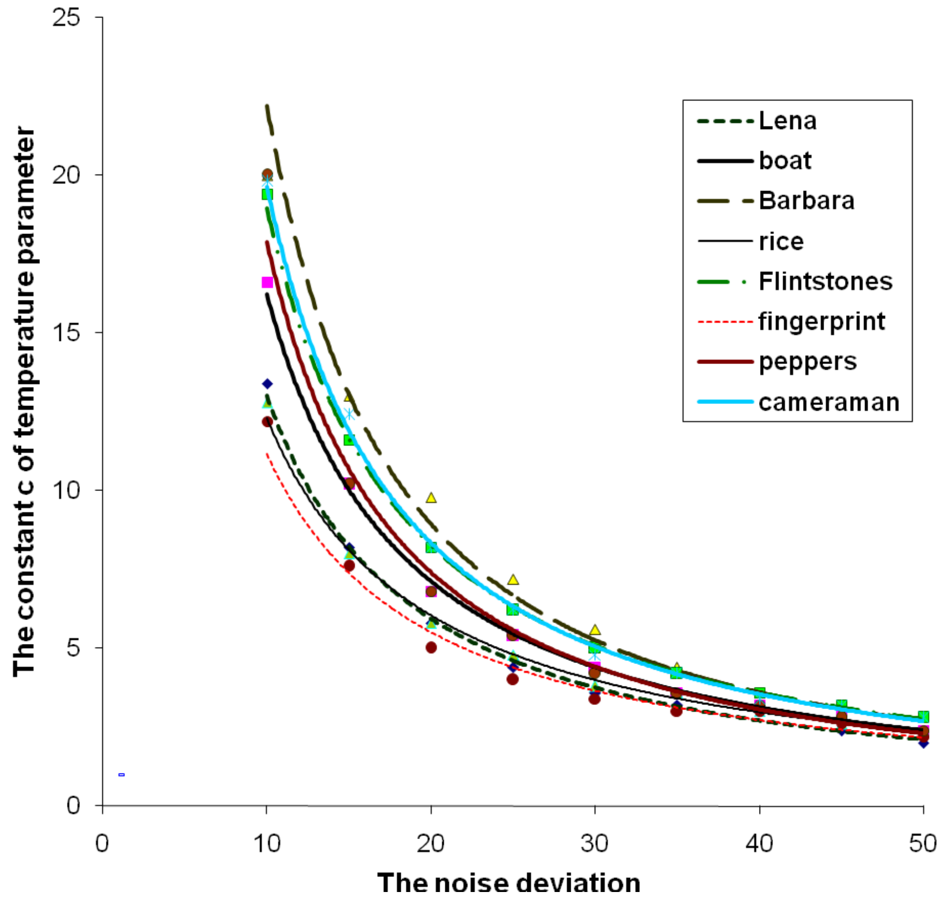


Figure 3-6. The relationship between the constant c of $T(k)$ and the noise deviation σ_n.

The accuracy and effectiveness of the algorithm are determined from choosing the appropriate parameters $T(k)$, $\sigma_{\Delta}^{2}(k)$ and $\mu_{\Delta}(k)$. In our experiments, the constant c of parameter $T(k)$ is changed according to the noise variance σ_n^2, while $\mu_{\Delta}(k)$ and $\sigma_{\Delta}^{2}(k)$ decrease with respect to k^{-2} as follows:

$$\mu_{\Delta}(k) = \frac{255}{k^2} \quad \text{and} \quad \sigma_{\Delta}^{2}(k) = \frac{\mu_{\Delta}(k)}{6} \tag{3.19}$$

Figure 3-7. The noise-free Lena image (top-left), the noisy image (top-right) σ_n=20 (PSNR=22.14dB), and the results of denoising processes using equation (30) with the original (bottom-right) (PSNR=29.70dB) and modified (bottom-left) (PSNR=30.77dB) line field.

Through experiments, we find the optimal relationship between the noise variance and the constant *c* of the "temperature" *T(k)* so that the algorithm gives the best denoising result. This relationship, which is shown in Figure 3-6, corresponds to eight experiments using the popular image such as: "Lena", "boat", "Barbara", "rice", "flinstones", "fingerprint", "peppers", and "cameraman". These images are available at [84]. The experiments were run with various noise variances to find the best value of temperature *T(k)* in each case. For all these cases, the relationship between the constant *c* and the noise variance is likely to be a hyperbolic curve. We generate a hyperbolic function which is approximated with the average value getting from these experiments. The generated function can be used to determine the parameters *T(k)* in the step 2 of our algorithm given the noise variance.

Table 3.1 PSNR[dB] results of VisuShrink [58], SureShrink [59], BayesShrink [60], equation (30) with Geman's line field and the proposed algorithm.

	$\sigma_n = 10$	$\sigma_n = 20$	$\sigma_n = 30$
Noisy Image	28.18	22.14	18.62
VisuShrink	28.76	26.46	25.14
SureShrink	33.28	30.22	28.83
BayesShrink	33.32	30.17	28.48
Equation (30)	31.78	29.70	28.12
Our algorithm	**34.18**	**30.77**	**28.95**

To compare effectively with the existing methods, the proposed algorithm is applied in denoising the Lena image corrupted by additive white noises with different variances (as shown in Figure 3-7). The image size is 512×512 pixels. The experimental results were compared with those of the other methods in term of Peak Signal Noise Ratio (PSNR):

$$PSNR = 10\log\frac{MAX^2}{\left\| f - \hat{f} \right\|^2}$$

where $\hat{f}$ is the estimates of the original image f, and *MAX* is the maximum possible value of the image intensity.

The quantitative performance comparison in Table 3.1 shows that our method is highly competent with denoising techniques appeared in the literature such as VisuShrink, SureShrink, BayesShrink. Moreover, our proposed algorithm, which is realized without involving the wavelet and wavelet inverse transformation, requires less effort on computation. In addition, the modified line field, whose distribution is changed at each iteration, helps to increase the convergence speed and to reduce the computational time significantly. Another advantage of the algorithm is that the terminating criterion is reached after about ten iterations. This fact makes our algorithm to have little effect on the edge area of images while removing effectively the noise.

To compare with denoising approaches in the spatial domain, the proposed algorithm is also implemented with various images such as: Lena, cameraman, bridge, boat, house, mountain, Zelda, rice, bird, goldhill, flinstones, library, frog, mandrill, washsat and compared with the LPA-ICI method in various criteria such as:

- Maximum absolute error (MAX): the infinity norm
- Mean Absolute Error (MAE): the 1-norm
- Root Mean Square Error (RMSE): the 2-norm (Euclidean norm)
- Mean Square Error (MSE): the square of Euclidean norm.
- Peak Signal to Noise Ratio (PSNR): the logarithm form of Euclidean norm.

Table 3.2. . Compare the denoising results of our proposed algorithm (printed in bold) and LPA-ICI algorithm [78].

Image	σ_n	PSNR		MSE		RMSE		MAE		MAX	
Lena	*10*	34.48	**34.30**	23.19	**24.18**	4.82	**4.92**	3.45	**3.52**	49.92	**49.59**
	15	32.41	**32.37**	37.32	**37.68**	6.11	**6.14**	4.31	**4.37**	71.49	**49.72**
	20	30.95	**30.96**	52.20	**52.15**	7.22	**7.22**	5.04	**5.07**	76.76	**77.23**
	25	29.78	**29.65**	68.43	**70.56**	8.27	**8.40**	5.71	**5.80**	138.63	**84.21**
	30	28.90	**28.45**	83.78	**92.95**	9.15	**9.64**	6.30	**6.55**	164.57	**122.60**
cameraman	*10*	33.07	**32.01**	32.06	**40.91**	5.66	**6.40**	3.75	**4.46**	45.98	**74.34**
	15	30.75	**29.87**	54.71	**66.98**	7.40	**8.18**	4.77	**5.54**	74.89	**66.68**
	20	29.33	**28.28**	75.93	**96.66**	8.71	**9.83**	5.55	**6.47**	74.70	**80.04**
	25	28.13	**26.83**	100.08	**134.99**	10.00	**11.62**	6.38	**7.53**	84.88	**104.57**
	30	27.00	**25.36**	129.79	**189.23**	11.39	**13.76**	7.21	**8.84**	123.55	**138.85**
bridge	*10*	29.13	**29.69**	79.39	**69.82**	8.91	**8.36**	6.93	**6.44**	51.94	**78.19**
	15	27.17	**27.51**	124.66	**115.33**	11.17	**10.74**	8.64	**8.31**	74.89	**62.72**
	20	25.85	**26.07**	169.25	**160.82**	13.01	**12.68**	10.02	**9.76**	72.24	**87.09**
	25	24.85	**24.88**	213.08	**211.43**	14.60	**14.54**	11.22	**11.18**	80.76	**80.69**
	30	24.06	**23.96**	255.06	**261.03**	15.97	**16.16**	12.26	**12.31**	85.33	**94.36**
house	*10*	34.55	**33.88**	22.80	**26.62**	4.77	**5.16**	3.40	**3.74**	34.73	**37.43**
	15	32.61	**32.08**	35.66	**40.32**	5.97	**6.35**	4.13	**4.44**	74.89	**47.23**
	20	31.29	**30.86**	48.33	**53.35**	6.95	**7.30**	4.74	**5.03**	68.27	**54.69**
	25	30.13	**29.45**	63.14	**73.87**	7.95	**8.60**	5.41	**5.77**	91.29	**81.80**
	30	29.06	**28.15**	80.68	**99.63**	8.98	**9.98**	6.04	**6.61**	123.55	**106.75**

mountain	*10*	25.69	**27.80**	175.44	**107.98**	13.25	**10.39**	9.68	**8.11**	85.77	**215.53**
	15	24.20	**25.88**	247.49	**167.75**	15.73	**12.95**	11.39	**9.82**	108.19	**216.33**
	20	22.99	**24.04**	326.44	**256.31**	18.07	**16.01**	12.93	**12.15**	132.23	**214.85**
	25	22.05	**22.61**	405.12	**356.19**	20.13	**18.87**	14.28	**14.24**	142.55	**214.97**
	30	21.35	**21.45**	476.22	**465.99**	21.82	**21.59**	15.39	**16.20**	184.74	**210.17**
zelda	*10*	35.07	**35.29**	20.23	**19.22**	4.50	**4.38**	3.43	**3.30**	49.92	**56.94**
	15	33.19	**33.68**	31.18	**27.89**	5.58	**5.28**	4.19	**3.92**	71.49	**63.58**
	20	31.85	**32.44**	42.44	**37.09**	6.51	**6.09**	4.84	**4.47**	76.76	**72.46**
	25	30.79	**31.27**	54.15	**48.48**	7.36	**6.96**	5.40	**4.98**	138.63	**83.95**
	30	30.01	**29.79**	64.82	**68.28**	8.05	**8.26**	5.89	**5.81**	164.57	**96.66**
boat	*10*	32.35	**32.21**	37.81	**39.10**	6.15	**6.25**	4.65	**4.76**	49.92	**56.15**
	15	30.38	**30.28**	59.56	**60.95**	7.72	**7.81**	5.69	**5.82**	71.49	**61.56**
	20	28.95	**28.83**	82.88	**85.12**	9.10	**9.23**	6.61	**6.75**	76.76	**86.97**
	25	27.87	**27.58**	106.14	**113.65**	10.30	**10.66**	7.38	**7.62**	138.63	**103.71**
	30	27.03	**26.54**	128.97	**144.22**	11.36	**12.01**	8.08	**8.52**	164.57	**114.45**
rice	*10*	32.19	**31.93**	39.26	**41.68**	6.27	**6.46**	4.78	**5.02**	43.51	**36.83**
	15	30.79	**30.48**	54.18	**58.16**	7.36	**7.63**	5.51	**5.83**	73.74	**48.96**
	20	29.55	**29.35**	72.14	**75.49**	8.49	**8.69**	6.26	**6.53**	80.69	**64.51**
	25	28.32	**28.02**	95.83	**102.68**	9.79	**10.13**	7.08	**7.40**	77.51	**85.38**
	30	27.25	**26.89**	122.58	**132.96**	11.07	**11.53**	7.98	**8.39**	123.55	**91.74**
bird	*10*	36.17	**35.88**	15.70	**16.77**	3.96	**4.10**	2.73	**2.86**	30.88	**39.19**
	15	33.99	**34.07**	25.93	**25.49**	5.09	**5.05**	3.39	**3.47**	74.89	**45.90**
	20	32.52	**32.45**	36.37	**36.98**	6.03	**6.08**	3.98	**4.08**	64.06	**65.49**
	25	31.36	**30.94**	47.58	**52.31**	6.90	**7.23**	4.56	**4.74**	67.87	**72.64**
	30	30.27	**29.48**	61.06	**73.28**	7.81	**8.56**	5.20	**5.60**	123.55	**91.44**
goldhill	*10*	30.78	**30.68**	54.33	**55.56**	7.37	**7.45**	5.62	**5.66**	52.48	**75.54**
	15	28.79	**28.69**	85.98	**87.91**	9.27	**9.38**	6.96	**7.09**	74.89	**79.62**
	20	27.38	**27.40**	118.88	**118.36**	10.90	**10.88**	8.14	**8.19**	72.47	**82.36**
	25	26.45	**26.27**	147.29	**153.33**	12.14	**12.38**	9.04	**9.22**	86.73	**91.74**
	30	25.69	**25.36**	175.59	**189.48**	13.25	**13.77**	9.84	**10.18**	116.22	**87.87**
library	*10*	28.98	**28.74**	82.31	**86.87**	9.07	**9.32**	6.26	**6.58**	63.62	**218.72**
	15	26.97	**26.74**	130.59	**137.74**	11.43	**11.74**	7.83	**8.63**	85.67	**223.86**
	20	25.50	**25.03**	183.45	**204.10**	13.54	**14.29**	9.25	**10.40**	102.94	**228.14**
	25	24.43	**23.56**	234.63	**286.21**	15.32	**16.92**	10.46	**12.23**	123.12	**212.23**
	30	23.44	**22.22**	294.62	**389.85**	17.16	**19.74**	11.67	**14.10**	177.30	**220.24**
frog	*10*	26.57	**29.23**	143.11	**77.62**	11.96	**8.81**	9.53	**6.93**	74.33	**75.56**
	15	25.97	**27.18**	164.34	**124.53**	12.82	**11.16**	10.14	**8.83**	76.51	**90.00**
	20	25.41	**25.92**	187.21	**166.51**	13.68	**12.90**	10.75	**10.19**	88.29	**77.53**
	25	24.92	**24.99**	209.60	**206.07**	14.48	**14.36**	11.29	**11.26**	114.50	**102.13**
	30	24.49	**24.29**	231.43	**242.06**	15.21	**15.56**	11.78	**12.09**	120.37	**109.82**
Flintstones	*10*	31.19	**30.87**	49.44	**53.25**	7.03	**7.30**	5.13	**5.47**	56.26	**93.12**
	15	29.08	**28.61**	80.38	**89.60**	8.97	**9.47**	6.36	**6.95**	80.89	**96.03**
	20	27.46	**27.00**	116.61	**129.74**	10.80	**11.39**	7.53	**8.27**	92.91	**89.42**
	25	26.19	**25.58**	156.49	**180.12**	12.51	**13.42**	8.66	**9.64**	138.63	**101.40**
	30	25.13	**24.35**	199.35	**238.82**	14.12	**15.45**	9.72	**11.05**	164.57	**125.25**
mandrill	*10*	28.57	**29.24**	90.41	**77.43**	9.51	**8.80**	7.40	**6.74**	55.81	**76.12**
	15	26.42	**26.94**	148.18	**131.50**	12.17	**11.47**	9.32	**8.81**	80.26	**96.26**
	20	24.87	**25.26**	211.86	**193.81**	14.56	**13.92**	11.01	**10.63**	90.14	**95.79**
	25	23.79	**23.99**	271.98	**259.29**	16.49	**16.10**	12.38	**12.19**	128.62	**120.01**
	30	23.03	**22.97**	323.30	**327.84**	17.98	**18.11**	13.45	**13.66**	164.57	**109.68**

washsat	*10*	33.70	**34.17**	27.75	**24.88**	5.27	**4.99**	3.96	**3.79**	45.81	**63.37**
	15	32.34	**32.70**	37.90	**34.93**	6.16	**5.91**	4.54	**4.45**	73.01	**96.75**
	20	31.51	**31.73**	45.92	**43.65**	6.78	**6.61**	4.97	**4.88**	88.47	**92.02**
	25	30.77	**30.87**	54.52	**53.24**	7.38	**7.30**	5.37	**5.26**	138.63	**87.04**
	30	30.26	**29.77**	61.26	**68.50**	7.83	**8.28**	5.68	**5.83**	164.57	**95.86**

From Table 3.2, we find that our results are as good as those of the LPA-ICI algorithm [78]. Some extracted results which are shown in Figure 3-8 demonstrate that the performances of the two algorithms are approximately the same. Visual observations show that the LPA-ICI result is often smoother than ours since it used the polynomial approximation approach, which tends to suppress minor variations in “flat regions”. However, if we use a standard Wiener filter to deconvolve the denoised image by our algorithm and LPA-ICI algorithm, our result is slightly better than that of LPA-ICI algorithm. This deconvolution experiment does not aime to hide the fact that LPA-ICI is better than our denoising algorithm in some cases as mentioned above.

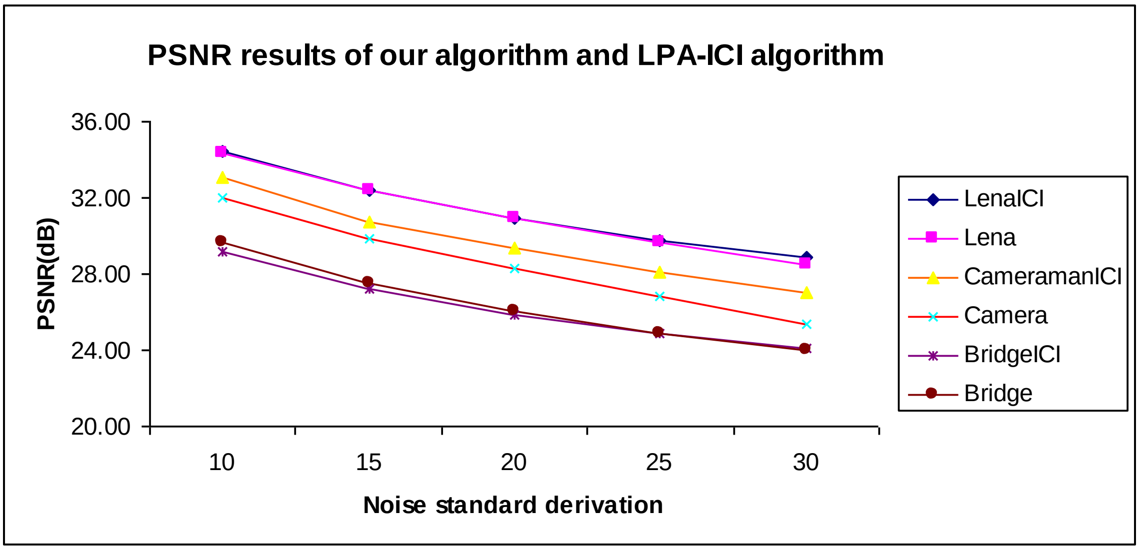


Figure 3-8. PSNR results of our proposed algorithm and LPA-ICI algorithm.

3.7. Concluding remarks

The Markov random field is an appropriate tool for modeling the image. Adding the line field to the model makes it more powerful in processing the image while preserving image details. Our suggested line field has a variant distribution whose sum covers the whole variation interval of intensity difference. It helps to distinguish between the noisy pixel and the edge pixels. Hence, reconstructing an adaptive image model based on the modified line field enables us to model the image more appropriately and effectively.

The experimental results show that our method is highly competent with the denoising techniques reported in the literature such as VisuShrink, SureShrink, BayesShrink [58-60]. Moreover, our proposed algorithm is fast because it is elaborated directly on the spatial domain and converges after about ten iterations. On the other hand, our experimental results are quite competent in comparison with those of the LPA-ICI algorithm, a very good denoising algorithm in spatial domain reported.

The adaptive image model constructed in this chapter, called LiFeAIM, will be used to deal with the deblurring problem in the next chapter.

Chapter 4
Deblurring Algorithms
Using the Proposed LiFeAIM
and Variational Bayesian Approach

4.1. Introduction

The blind deblurring problem can be considered as a kind of blind source separation problem. This problem is to separate the original signals from their observed mixtures while there is no information or little information about the original signals and the mixing process. It is often solved with the assumption that the original signals are independent.

A recent approach to solve the blind source separation is the Ensemble Learning approach [61], which is an approach in the Bayesian framework. This approach uses a set of hypotheses (or models) about the hidden data instead of only one hypothesis (or model) about it. When one hypothesis (or model) fails, the algorithm is still able to choose an appropriate hypothesis (or model) from the remaining ones. Hence, the Ensemble Learning approach is very effective for problems with little prior information such as the blind source separation problem. There are many successful applications in using the Ensemble Learning approach for blind source separation, such as: the cocktail party problem, music separation from a concert, and reflection removal.

There are two types of Ensemble Learning, the fixed-form Ensemble Learning and the free-form Ensemble Learning. The latter, also named the Variational Bayesian approach in this thesis, is more flexible than the former. The next section will introduce the development of the Variational Bayesian approach from the Bayesian formula and the application of this approach into the blind deblurring problem.

4.2. Variational Bayesian approach

4.2.1. Bayesian framework

In Bayesian framework, there are approaches finding the hidden data by maximizing the conditional probability of the hidden data given the observed data. This probability is calculated by the Bayesian formula of posterior probability. In our problem, it can be expressed as follows:

$$p(f,h|g)=\frac{p(g|f,h)p(f,h)}{p(g)} \tag{4.1}$$

where, the original image *f* and the kernel blurring function *h* are hidden data; and

the blurred image *g* is the observed data;

$p(f,h|g)$ is the posterior probability of the hidden data given the observed data;

$p(f,h)$ is the prior information about the hidden data;

$p(g|f,h)$ is called the likelihood which is the conditional probability of the observed data given the hidden data.

The classical approach of Bayesian framework which uses this formula is the Maximum à Posteriori (MAP). In this approach, the best hidden data is found to maximise the posterior probability $p(f,h|g)$. The problem of MAP approach is how to choose an appropriate prior distribution of the hidden data. If the prior information is not chosen properly, applying this approach may end up with a local optimum or divergence.

To overcome the above limitation of MAP approach when only a little prior information of hidden data is given, the prior distribution of the hidden data is not fix but varying in an iterative algorithm. Hence, the prior which approximates the distribution of the hidden data best would be chosen. Following this idea, the Ensemble Learning approach uses the prior distribution of the hidden data in parametric form. The prior distribution of the hidden data will be changed when the parameters of the prior of hidden data are re-estimated at each iterative step by maximising the posterior probability of the hidden data and the parameters given the observed data. Thus, both the prior information of the hidden data and the prior information of the parameters are required in the Ensemble Learning approach. The optimisation problem in Ensemble Learning approach is solved by using the Kullback-Leibler divergence, which will be described in the next section. The approach is named Ensemble Learning because both the hidden data and the parameters are re-estimated at each iterative step.

Once again, in the Ensemble Learning approach, the issue of choosing the best prior distribution of these parameters θ_i is raised. To approach this issue, the prior distribution of parameters θ_i is also written in the parametric form. The best prior distribution of these parameters will be selected from a class of distributions. By doing so, all the prior of hidden data and model parameters is variant. The theory of this free – form Ensemble Learning approach can be found in [85]. This approach, called Variational Bayesian approach in this thesis, is briefly described in the next section.

4.2.2. Variational Bayesian approach

In the Variational Bayesian approach, it is assumed that all hidden data and parameters are independent random variables. In this thesis, the hidden data are the original image *f* and the kernel blurring function *h*, while the parameters of the prior of hidden data are denoted θ_i. Hence, with the above-mentioned independent assumption, our prior information is represented by the following equation:

$$\begin{aligned} p(f,h,\Theta) &= p(f|h,\Theta)p(h|\Theta)p(\Theta) \\ &= p(f|\Theta)p(h|\Theta)\prod_i p(\theta_i) \end{aligned} \tag{4.2}$$

where Θ is the set of parameters θ_i.

The posterior probability, then, can be written as shown in eq. (4.3):

$$p(f,h,\Theta|g) = \frac{p(g|f,h,\Theta)p(f,h,\Theta)}{p(g)} = \frac{p(g|f,h,\Theta)p(f|\Theta)p(h|\Theta)\prod_i p(\theta_i)}{p(g)} \tag{4.3}$$

To solve our problem, we need to maximise the above equation. However, in general, there is no closed form solution for this optimisation problem. Hence, to solve the problem, the true posterior distribution $p(f,h,\Theta|g)$ is approximated by a simpler distribution $q(f,h,\Theta)$. The approximate distribution $q(f,h,\Theta)$ is also separable, i.e. $q(f,h,\Theta) = q(f)q(h)\prod_i q(\theta_i)$. The approximate solution of this optimisation problem can be found by using the Kullback-Leibler divergence [61].

The Kullback-Leibler divergence between the true posterior distribution $p(f,h,\Theta|g)$ and the approximate distribution $q(f,h,\Theta)$ is determined by the following equation:

$$
\begin{aligned}
D_{KL}(q\|p) &= \int_{f,H,\Theta} q(f,h,\Theta)\log\left[\frac{q(f,h,\Theta)}{p(f,h,\Theta|g)}\right]dfdhd\Theta \\
&= \int_{f,H,\Theta} q(f,h,\Theta)\log\left[\frac{q(f,h,\Theta)p(g)}{p(g|f,h,\Theta)p(f|\Theta)p(h|\Theta)\prod_i p(\theta_i)}\right]dfdhd\Theta
\end{aligned}
\tag{4.4}
$$

With the assumption about independence of the hidden data, the parameters, and the observation, the probability $p(g)$ of the observation can be taken outside the integral in the above equation. It is proven that the Kullback-Leibler divergence is non-negative and equal to zero when $q(f,h,\Theta) = p(f,h,\Theta|g)$. Hence, the approximate distribution $q(f,h,\Theta)$ is found by minimising the Kullback-Leibler divergence with respect to $q(f), q(h),$ and $q(\theta_i)$. The optimum solution of eq. (4.4) is given by [61]:

$$
\begin{aligned}
q(f) &= const \times \exp\left\{E\left[\log p(f|\Theta) + \log p(g|f,h,\Theta)\right]_{q(h)q(\Theta)}\right\} \\
q(h) &= const \times \exp\left\{E\left[\log p(h|\Theta) + \log p(g|f,h,\Theta)\right]_{q(f)q(\Theta)}\right\} \\
q(\theta_i) &= const \times p(\theta_i)\exp\left\{E\left[\log p(g|f,h,\Theta)p(f|\Theta)p(h|\Theta)\right]_{q(h)q(f)q(\Theta_i)}\right\}
\end{aligned}
\tag{4.5}
$$

where Θ_i is the subset of Θ after θ_i is removed.

It is noted that the optimum solution is not given directly in eq. (4.5) but obtained by iteratively estimating the hidden data *f*, *h* and the parameters Θ using eq. (4.5). Another notable remark of the above equation is that it does not give the value of the hidden data and the parameters directly but their approximate distribution. From these approximate distributions, we can estimate the hidden data and the parameters by their expectation and covariance. To calculate the approximate distributions in eq. (4.5), it is necessary to know the likelihood probability of the observed image g and the prior information of the original image f, the blurring function h, and the parameters Θ. All the prior information will be given in the following sections.

4.3. Prior information

4.3.1. Observation model

As we assume that the contaminated noise is a white Gaussian random signal, the likelihood of the observation can be represented as follows:

$$p(g|f,h,\beta_n) = const \times \beta_n^{N/2} \exp\left[-\frac{\beta_n}{2}\|g - h \otimes f\|^2\right] \tag{4.6}$$

in which, β_n^{-1} is the variance of the contaminated noise; g and f are the observed and original data written in the vector form $1\times N$ by lexicographically re-ordering the observed and original images; and h is the blurring function written in the vector form $1\times N$.

The observation g is assumed to be blurred by a spatially invariant blurring function h. Then, the convolution $h \otimes f$ may be rewritten in the matrix form as shown in the following equation:

$$h \otimes f = Hf \tag{4.7}$$

where H is a block-Toeplitz matrix [86] derived from the blurring function h.

Hence, the observation model is rewritten in eq. (4.8):

$$p\left(g \middle| f, h, \beta_n\right) = const \times \beta_n^{N/2} \exp\left[-\frac{\beta_n}{2}\|g - Hf\|^2\right] \tag{4.8}$$

As a block-Toeplitz matrix can be approximated and considered as a circulant matrix which is diagonalised by the Fourier transform. In this work, *H* is processed as a circulant matrix whose first row is h^T. The latter rows of *H* are created by shifting one element of its preceding row to the right.

The convolution $h \otimes f$ can also be rewritten in the matrix form as the product of the blurring function *h* and a matrix derived from *f*:

$$h \otimes f = Fh \tag{4.9}$$

where *F* is a matrix whose first row is f^T. The latter rows of *F* are created by shifting one element of their preceding row to the left. *F* is then called the left-wise circulant matrix. Although the left-wise circulant matrix is not diagonalised by the Fourier transform, we will prove later that it can be diagonalised by using the Fourier transformation matrix.

4.3.2. Image model

There are some popular image models in image restoration, such as the Auto-Regressive model, and the Total Variation model. Chapter 6 will introduce the deblurring algorithms using the Simultaneous Auto-Regressive model and the Total Variation model. In this chapter, we use our proposed adaptive image model to construct blind deblurring algorithms. This model, called the Line-Field based Adaptive Image Model (LiFeAIM), was proposed and its performance in denoising has been examined in Chapter 3. The model uses the line field, which is a virtual random process existing between each image pixel and its eight nearest neighbours, to calculate the variance of the image at that pixel. The model is represented by the following equation:

$$p\left(f|\beta_{im}\right) = const \times \left\{ \left\| \beta_{im} \right\|^{N/2} \exp\left[-\frac{\beta_{im}}{2} f^T B^T Bf \right] \right\}^{1/T_k} \qquad (4.10)$$

in which, β_{im} is the parameter of image model;

N is the size of image, the number of pixels;

$f^T B^T Bf$ is the matrix-form presentation of the following summation:

$$\sum_i \left[f_i - \sum_{j \in s_i} \beta_{ij} f_j \right]^2 ;$$

s_i is the set of eight nearest neighbours of pixel i;

$$\beta_{ij} = \frac{l_{ij}}{\sum_{j \in s_i} l_{ij}} ;$$

l_{ij} is the line random variable between the pixel i and its neighbour j.

T_k is the temperature parameter controlling the convergence.

Using such an adaptive image model help to handle the high transition of image intensities efficiently because the variance of the image model varies through pixels. The image model with a constant variance may flatten the intensity field of an image when the model is used in an iterative algorithm.

4.3.3. Blurring model

In this chapter, we will construct two deblurring algorithms with two different blurring models. One model is Gaussian distributed. The other is the Simultaneous

Auto-Regression model. If the former is used, the blurring function is modeled as follows:

$$p(h|\beta_{bl}) = const \times \beta_{bl}^{M/2} \exp\left[-\frac{\beta_{bl}}{2} h^T h\right] \qquad (4.11)$$

in which, β_{bl} is the parameter of blurring model;

M is the support size of blurring function.

If the later is used, the probability of blurring function is given in eq. (4.12):

$$p(h|\beta_{bl}) = const \times \beta_{bl}^{M/2} \exp\left[-\frac{\beta_{bl}}{2} h^T C^T C h\right] \qquad (4.12)$$

in which, β_{bl} is the parameter of blurring model;

C is the circulant matrix derived from the Laplacian operator;

M is the support size of blurring function.

From eq. (4.11) and eq. (4.12) it is found that the Gaussian model is only the special case of Simultaneous Auto-Regression model when the circulant matrix C in eq. (4.12) is an identity matrix. However, both models are used to construct the deblurring algorithms in this work to compare the performance of the proposed algorirhtms when more constraints are applied to the prior information.

4.3.4. Prior of parameters

The parameters β_n, β_{im}, and β_{bl} of observation, image, and blurring model are assumed to be Gamma distributed random variables. In our study, the Gamma distribution is chosen to model these parameters instead of the Gaussian distribution because the Gamma distribution is more appropriate to model the positive parameter than the Gaussian distribution. The prior distribution of parameters β_n, β_{im}, and β_{bl} is shown in the following equation:

$$p(\beta_x) = \frac{\left(b_x^o\right)^{a_x^o}}{\Gamma\left(a_x^o\right)} \beta_x^{\,a_x^o - 1} \exp\left(-b_x^o \beta_x\right), \; x \in \{im, bl, n\} \tag{4.13}$$

in which, $a_x^0 > 0$ is the shape parameter of Gamma distribution;

$b_x^0 > 0$ is the scale parameter of Gamma distribution;

$\Gamma\left(a_x^o\right)$ is the Gamma function.

4.4. Blind deblurring algorithms using LiFeAIM

In this section, we propose two blind deblurring algorithms using Variational Bayesian approach. Both algorithms use LiFeAIM as the image model and assume that the model parameters are Gamma distributed. The only difference between the two algorithms is the model of blurring function. One algorithm uses the Simultaneous Auto-Regression model as the prior information of blurring function. The other uses the Gaussian distribution for the blurring model. The former algorithm is called LF-SAR algorithm while the latter is called LF-G algorithm. Derivation of the estimates in LF-SAR algorithm is shown in detail below. For LF-G algorithm, only the final equations are shown since the derivation procedure is similar.

4.4.1. Estimation of image, blurring function and model parameters

By applying the Variational Bayesian approach introduced above, we obtain the optimum solution of the approximate distributions $q(f)$, $q(h)$, $q(\beta_n)$, $q(\beta_{im})$, and

$q(\beta_{bl})$. Then, the original image f, the blurring function *h*, and the parameters β_n, β_{im}, and β_{bl} are estimated by their expectation.

From the optimum solution in eq. (4.5), we have:

$$q(f) = const \times \exp\left\{E\left[\log p(f|\Theta) + \log p(g|f, h, \Theta)\right]_{q(h)q(\Theta)}\right\}$$

Hence,

$$\begin{aligned}\log q(f) &= const + E\left[-\frac{\beta_{im}}{2T_k} f^T B^T B f\right]_{q(h)q(\Theta)} + E\left[-\frac{\beta_n}{2}\|g - Hf\|^2\right]_{q(h)q(\Theta)} \\ &= const - \frac{E[\beta_{im}]}{2T_k} f^T B^T B f - \frac{E[\beta_n]}{2} E\left[\|g - Hf\|^2\right]_{q(h)}\end{aligned}$$

It is assumed that the approximate distributions $q(f)$ of the original image is Gaussian. Hence, the expectation and covariance of the original image f are given in the equations below:

$$E[f] = \left\{ f \left| \frac{\partial \log[q(f)]}{\partial f} = 0 \right.\right\}, \text{ and } \operatorname{cov}(f) = -\left(\frac{\partial^2 \log[q(f)]}{\partial f^2}\right)^{-1} \quad (4.14)$$

$$\frac{\partial \log[q(f)]}{\partial f} = -\frac{E[\beta_{im}]}{T_k} B^T B f - E[\beta_n] E\left[H^T H\right] f + E[\beta_n] E\left[H^T\right] g$$

$$E[f] = \left\{ f \left| \frac{\partial \log[q(f)]}{\partial f} = 0 \right.\right\} = \left[\frac{E[\beta_{im}]}{T_k} B^T B + E[\beta_n] E\left[H^T H\right]\right]^{-1} E[\beta_n] E\left[H^T\right] g$$

$$\begin{aligned}\operatorname{cov}(f) &= -\left(\frac{\partial^2 \log[q(f)]}{\partial f^2}\right)^{-1} \\ &= -\left(-\frac{E[\beta_{im}]}{T_k} B^T B - E[\beta_n] E\left[H^T H\right]\right)^{-1}\end{aligned}$$

Denote:

$$\operatorname{cov}(H) = E\left[(H - E[H])^T (H - E[H])\right]$$

Hence,

$$E\left[H^T H\right] = E\left[H^T\right]E[H] + \mathrm{cov}(H)$$

$$\boxed{\mathrm{cov}(f) = \left(E[\beta_n]E[H]^T E[H] + E[\beta_n]\mathrm{cov}(H) + \frac{E[\beta_{im}]}{T_k} B^T B \right)^{-1}} \tag{4.15}$$

$$\boxed{E[f] = E[\beta_n]\mathrm{cov}(f)E[H]^T g} \tag{4.16}$$

From eq. (4.5), the optimum approximate distribution of blurring function is as follows:

$$q(h) = const \times \exp\left\{ E\left[\log p(h|\Theta) + \log p(g|f, h, \Theta)\right]_{q(f)q(\Theta)} \right\}$$

As we now estimate the blurring function *h*, the convolution $h \otimes f$ in the observation model will be rewritten as *Fh*.

$$\begin{aligned} \log q(h) &= const + E\left[-\frac{\beta_{bl}}{2} h^T C^T C h \right]_{q(f)q(\Theta)} + E\left[-\frac{\beta_n}{2} \|g - Fh\|^2 \right]_{q(f)q(\Theta)} \\ &= const - \frac{E[\beta_{bl}]}{2} h^T C^T C h - \frac{E[\beta_n]}{2} E\left[\|g - Fh\|^2\right]_{q(f)} \end{aligned}$$

The approximate distribution $q(h)$ of the blurring function is also assumed to be the Gaussian distribution. Hence, the expectation and covariance of the blurring function *h* are given in the equation below:

$$E[h] = \left\{ h \middle| \frac{\partial \log[q(h)]}{\partial h} = 0 \right\} \text{, and } \mathrm{cov}(h) = -\left(\frac{\partial^2 \log[q(h)]}{\partial h^2} \right)^{-1} \tag{4.17}$$

Similarly, denote $\mathrm{cov}(F) = E\left[(F - E[F])^T (F - E[F])\right]$. The covariance matrix and the expectation of blurring function can be deduced as shown in eq. (4.18-4.19):

$$\boxed{\mathrm{cov}(h) = \left(E[\beta_n]E[F]^T E[F] + E[\beta_n]\mathrm{cov}(F) + E[\beta_{bl}]C^T C \right)^{-1}} \tag{4.18}$$

$$\boxed{E[h] = E[\beta_{bl}]\mathrm{cov}(h)E[F]^T g} \quad (4.19)$$

Now, we estimate the parameter of image, blurring, and observation prior. Estimation of the parameter β_{im} of image prior can be derived from the solution of approximate distribution $q(\beta_{im})$ given in eq. (4.5):

$$q(\beta_{im}) = const \times p(\beta_{im})\exp\left\{E\left[\log p\left(f|\beta_{im}\right)\right]_{q(f)}\right\} \quad (4.20)$$

The approximate distributions $q(\beta_x)$ of parameters are assumed to be Gamma distributions as presented in the following form:

$$q(\beta_x) = \frac{(b_x)^{a_x}}{\Gamma(a_x)}\beta_x^{\,a_x - 1}\exp(-b_x\beta_x),\ x \in \{im, bl, n\} \quad (4.21)$$

Substituting eq. (4.21) into eq. (4.20) and taking logarithm of eq. (4.20), we obtain the following equation:

$$\begin{aligned} &const + a_{im}\log b_{im} + (a_{im} - 1)\log\beta_{im} - b_{im}\beta_{im} \\ &\qquad = const + a_{im}^{o}\log b_{im}^{o} + (a_{im}^{o} - 1)\log\beta_{im} - b_{im}^{o}\beta_{im} + \frac{N}{2}\log\beta_{im} - \frac{\beta_{im}}{2T_k}E\left[f^T B^T Bf\right] \end{aligned} \quad (4.22)$$

Comparing the coefficients of both sides of eq. (4.22), we obtain the following results:

$$a_{im} = a_{im}^{0} + \frac{N}{2},\ b_{im} = b_{im}^{0} + \frac{1}{2T_k}E\left[f^T B^T Bf\right] \quad (4.23)$$

For the random variables with Gamma distributions, their expectation and variance are given as follows:

$$E[\beta_x] = \frac{a_x}{b_x}, \text{ and } Var[\beta_x] = \frac{a_x}{(b_x)^2} \quad (4.24)$$

where β_x is a Gamma random variable.

Hence,

$$\boxed{E[\beta_{im}] = \frac{a_{im}^0 + \frac{N}{2}}{b_{im}^0 + \frac{1}{2T_k} E\left[f^T B^T B f\right]}} \tag{4.25}$$

Similarly, we derive the expectation of the parameter β_{bl} and β_n as follows:

$$q(\beta_{bl}) = const \times p(\beta_{bl}) \exp\left\{E\left[\log p(h|\beta_{bl})\right]_{q(h)}\right\} \tag{4.26}$$

$$\begin{aligned} &const + a_{bl} \log b_{bl} + (a_{bl} - 1)\log \beta_{bl} - b_{bl}\beta_{bl} \\ &\qquad = const + a_{bl}^o \log b_{bl}^o + (a_{bl}^o - 1)\log \beta_{bl} - b_{bl}^o \beta_{bl} + \frac{M}{2}\log \beta_{bl} - \frac{\beta_{bl}}{2} E\left[h^T C^T C h\right] \end{aligned}$$

Hence,

$$a_{bl} = a_{bl}^0 + \frac{M}{2},\ b_{bl} = b_{bl}^0 + \frac{1}{2} E\left[h^T C^T C h\right] \tag{4.27}$$

$$\boxed{E[\beta_{bl}] = \frac{a_{bl}^0 + \frac{M}{2}}{b_{bl}^0 + \frac{1}{2} E\left[h^T C^T C h\right]}} \tag{4.28}$$

And,

$$q(\beta_n) = const \times p(\beta_n) \exp\left\{E\left[\log p(g| f, h, \beta_n)\right]_{q(f)q(h)}\right\} \tag{4.29}$$

$$\begin{aligned} &const + a_n \log b_n + (a_n - 1)\log \beta_n - b_n \beta_n \\ &\qquad = const + a_n^o \log b_n^o + (a_n^o - 1)\log \beta_n - b_n^o \beta_n + \frac{N}{2}\log \beta_n - \frac{\beta_n}{2} E\left[\|g - Hf\|^2\right]_{q(f)q(h)} \end{aligned}$$

Hence,

$$a_n = a_n^0 + \frac{N}{2},\ b_n = b_n^0 + \frac{1}{2} E\left[\|g - Hf\|^2\right]_{q(f)q(h)} \tag{4.30}$$

$$E[\beta_n] = \frac{a_n^0 + \frac{N}{2}}{b_n^0 + \frac{1}{2}E\left[\|g - Hf\|^2\right]_{q(f)q(h)}} \tag{4.31}$$

As we noted after eq. (4.5), the hidden data and model parameters are not given directly from the above estimation equations. These estimation equations are used to construct an iterative algorithm in which the hidden data and model parameters are re-estimated many times until the results converge. In these equations, the size of matrices is very large. We will prove several theorems in the next section, and apply some numerical techniques to make the computation easier and faster.

4.4.2. Numerical computation

We consider a blurred image which is lexicographically re-ordered into the vector form 1-by-*N*. To deblur this image, the proposed algorithm needs to manipulate *N*-by-*N* matrices in the above estimation equations. The algorithm must especially perform the inversion of the big matrices in equations (4.15-4.16) and (4.18-4.19) to estimate the covariance matrices of the original image *f* and the blurring function *h*. The calculation of these inverse matrices is computational intensive in the order $O(N^4)$. In the previous studies, some approximations or assumptions are often used to approximate these matrices by circulant matrices. Hence, this computation problem can be solved by using the Fourier transform to diagonalise the circulant matrices. The computational order then reduces to $O(N)$ in the Fourier domain. In this thesis, since the covariance matrix of image is not assumed to be circulant, the covariance matrix of image is not diagonalised in the Fourier domain. To deal with the inversion problem, this thesis applies the Matrix Inversion Lemma and proposes some new theorems (Theorem 2-4 in this section) to reduce the computational effort.

** <u>*Image estimation:*</u>

Following the previous section, the expectation and covariance matrix of the original image are given as:

$$\mathrm{cov}(f) = \left(E[\beta_n] E[H]^T E[H] + E[\beta_n] \mathrm{cov}(H) + \frac{E[\beta_{im}]}{T_k} B^T B \right)^{-1}$$

$$E[f] = E[\beta_n] \mathrm{cov}(f) E[H]^T g$$

The first term in the estimation equation of image covariance relates to the circulant matrix *H*. In the circulant matrix, each row is shifted one element to the right relative to its preceding row. We will prove later that the circulant matrix is diagonalised by the Fourier transformation.

Denote *C* as a circulant matrix. *C* can be represented as below in eq. (4.32) and can be written as a polynomial of matrix *R* as shown in eq. (4.33):

$$C = \begin{bmatrix} c_1 & c_2 & c_3 & \dots & c_N \\ c_N & c_1 & c_2 & \dots & c_{N-1} \\ c_{N-1} & c_N & c_1 & \dots & c_{N-2} \\ \vdots & \vdots & \vdots & & \vdots \\ c_2 & c_3 & c_4 & \dots & c_1 \end{bmatrix} \tag{4.32}$$

$$C = c_1 I + c_2 R + c_3 R^2 + \dots + c_N R^{N-1} \tag{4.33}$$

$$R = \begin{bmatrix} 0 & 0 & \dots & 0 & 1 \\ 1 & 0 & \dots & 0 & 0 \\ 0 & 1 & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & 1 & 0 \end{bmatrix} = \begin{bmatrix} e_2 & e_3 & \dots & e_N & e_1 \end{bmatrix}$$

where e_k is the k^{th} column of the identity matrix.

The Fourier transform matrix is represented in the following equation:

$$T = \frac{1}{\sqrt{n}}\begin{bmatrix} 1 & 1 & 1 & \dots & 1 \\ 1 & \omega & \omega^2 & \dots & \omega^{(N-1)} \\ 1 & \omega^2 & \omega^4 & \dots & \omega^{2(N-1)} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & \omega^{N-1} & \omega^{2(N-1)} & \dots & \omega^{(N-1)(N-1)} \end{bmatrix}$$

where $\omega = \exp(-2\pi i / N)$.

It is noted that $TT^T = T^T T = I$

Theorem 1: The circulant matrix C is diagonalised by the Fourier transformation as follows:

$$TCT^T = diag\left(\sqrt{N}Tc\right)$$

where c is the first column of matrix C: $c = \begin{bmatrix} c_1 \\ c_N \\ c_{N-1} \\ \vdots \\ c_2 \end{bmatrix}$

To prove this theorem we first prove that matrix R is diagonalised by the Fourier transformation:

Lemma 1: $TRT^T = D$

where $D = diag\left(1, \omega, \omega^2, \dots, \omega^{(N-1)}\right)$

Proof: (Lemma 1) The kl^{th} element of matrix TR is:

$$[TR]_{kl} = \frac{1}{\sqrt{N}}\begin{bmatrix} 1 & \omega^{(k-1)} & \omega^{2(k-1)} & \dots & \omega^{(N-1)(k-1)} \end{bmatrix} e_{l+1} = \frac{1}{\sqrt{N}}\omega^{l(k-1)}$$

The kl^{th} element of matrix DT is:

$$[DT]_{kl} = \omega^{k-1} e_k \frac{1}{\sqrt{N}} \begin{bmatrix} 1 \\ \omega^{(l-1)} \\ \omega^{2(l-1)} \\ \vdots \\ \omega^{(N-1)(l-1)} \end{bmatrix} = \frac{1}{\sqrt{N}} \omega^{l(k-1)}$$

Hence: $TR = DT$

Or: $TRT^T = D$

Proof: (Theorem 1) We have:

$$C = c_1 I + c_2 R + c_3 R^2 + \ldots + c_N R^{N-1}$$

So,

$$\begin{aligned} TCT^T &= c_1 TIT^T + c_2 TRT^T + c_3 TR^2 T^T + \ldots + c_N TR^{N-1} T^T \\ &= c_1 + c_2 TRT^T + c_3 \left(TRT^T\right)^2 + \ldots + c_N \left(TRT^T\right)^{N-1} \\ &= c_1 + c_2 D + c_3 D^2 + \ldots + c_N D^{N-1} \end{aligned}$$

Hence, TCT^T is diagonal. The kk^{th} element of TCT^T is:

$$\left[TCT^T\right]_{kk} = c_1 + c_2 \omega^{(k-1)} + c_3 \omega^{2(k-1)} + \ldots + c_N \omega^{(N-1)(k-1)}$$

Meanwhile, the k^{th} element of Tc is:

$$\begin{aligned} [Tc]_k &= \frac{1}{\sqrt{N}} \begin{bmatrix} 1 & \omega^{(k-1)} & \omega^{2(k-1)} & \ldots & \omega^{(N-1)(k-1)} \end{bmatrix} \begin{bmatrix} c_1 \\ c_N \\ c_{N-1} \\ \vdots \\ c_2 \end{bmatrix} \\ &= \frac{1}{\sqrt{N}} \left(c_1 + c_2 \omega^{(k-1)} + c_3 \omega^{2(k-1)} + \ldots + c_N \omega^{(N-1)(k-1)}\right) \end{aligned}$$

Thus, $TCT^T = diag\left(\sqrt{N} Tc\right)$

The second term in the estimation equation of image covariance matrix can be represented as a linear function of the blurring covariance matrix. More precisely:

Theorem 2: H is a circulant matrix whose first row is h^T.

$$H = \begin{bmatrix} h_1 & h_2 & h_3 & \dots & h_N \\ h_N & h_1 & h_2 & \dots & h_{N-1} \\ h_{N-1} & h_N & h_1 & \dots & h_{N-2} \\ \vdots & \vdots & \vdots & & \vdots \\ h_2 & h_3 & h_4 & \dots & h_1 \end{bmatrix} \text{ and } h = \begin{bmatrix} h_1 \\ h_2 \\ h_3 \\ \vdots \\ h_N \end{bmatrix}$$

The covariance matrix of h is assumed to be circulant. Then:

$$\operatorname{cov}(H) = N \operatorname{cov}(h)$$

where N is the size of the column vector h

Proof: (Theorem2)

As H is the circulant matrix whose first row equals the vector h^T, each row of $H^T H$ is likely a convolution of h and itself. Therefore, each element (i,j) of $H^T H$ can be represented as following:

$$\left[H^T H\right]_{(i,j)} = \sum_{\substack{k,l \\ \operatorname{mod}(k-l,N)=i-j}} \left(hh^T\right)$$

Hence, we have the expectation:

$$E\left[H^T H\right]_{(i,j)} = \sum_{\substack{k,l \\ \operatorname{mod}(k-l,N)=i-j}} E\left[hh^T\right]$$

$$\left[E\left[H^T H\right]\right]_{(i,j)} = \sum_{\substack{k,l \\ \operatorname{mod}(k-l,N)=i-j}} \left(E[h]E\left[h^T\right] + \operatorname{cov}(h)\right)$$

$$= \sum_{\substack{k,l \\ \operatorname{mod}(k-l,N)=i-j}} \left(E[h]E\left[h^T\right]\right) + \sum_{\substack{k,l \\ \operatorname{mod}(k-l,N)=i-j}} \operatorname{cov}(h)$$

On one hand, $E[H]$ is a circulant matrix whose first row equals to $E[h^T]$, we also have:

$$\sum_{\substack{k,l \\ \mathrm{mod}(k-l,N)=i-j}} \left(E[h]E[h^T]\right) = \left[E[H^T]E[H]\right]_{(i,j)}$$

On the other hand, we assume that $\mathrm{cov}(h)$ is a circulant covariance matrix. It means that:

$$[\mathrm{cov}(h)]_{(k,l)} = [\mathrm{cov}(h)]_{(i,j)} \quad \forall k,l : \mathrm{mod}(k-l,N) = i-j$$

So, we can deduce:

$$\sum_{\substack{k,l \\ \mathrm{mod}(k-l,N)=i-j}} \mathrm{cov}(h) = N[\mathrm{cov}(h)]_{(i,j)}$$

Then,

$$\left[E[H^T H]\right]_{(i,j)} = \left[E[H^T]E[H]\right]_{(i,j)} + N[\mathrm{cov}(h)]_{(i,j)}$$

Or:

$$\begin{aligned} \mathrm{cov}(H) &= E[H^T H] - E[H^T]E[H] \\ &= N\,\mathrm{cov}(h) \end{aligned}$$

The last term in the estimation equation of image covariance matrix is related to a big matrix *B* of size *N*-by-*N* with *N* having large value. As this term is not diagonalised by the Fourier transformation in general, its inversion calculation is highly computational intensive. Fortunately, this matrix can be decomposed into simpler and smaller matrices which can be inversed much faster. So, the computational efficiency of the algorithm will be improved. *B* can be decomposed in the following way:

$$B = I - B^* \tag{4.34}$$

where $B^* = \overline{Q}^T \overline{B^*}\,\overline{Q}$, $\overline{Q}\overline{Q}^T = I$

Thus, the covariance of the original image can be rewritten as:

$$\mathrm{cov}(f) = \left[L + \frac{E[\beta_{im}]}{T_k} \overline{Q}^T \left(\overline{B^*}^T \overline{B^*} - \overline{B^*}^T - \overline{B^*} \right) \overline{Q} \right]^{-1} \tag{4.35}$$

where: $L = E[\beta_n] E[H]^T E[H] + E[\beta_n] N \,\mathrm{cov}(h) + \dfrac{E[\beta_{im}]}{T_k}$

The covariance in eq. (4.35) can be calculated rapidly by applying the Matrix Inversion Lemma.

<u>Lemma 2:</u> Matrix Inversion lemma:

$$(S + UTV)^{-1} = S^{-1} - S^{-1}U\left(T^{-1} + VS^{-1}U\right)^{-1} VS^{-1}$$

<u>Proof:</u> (Lemma 2)

On one hand, we have:

$$\begin{aligned}
&(S + UTV)\left[S^{-1} - S^{-1}U\left(T^{-1} + VS^{-1}U\right)^{-1} VS^{-1}\right] \\
&= I + UTVS^{-1} - \left[U + UTVS^{-1}U\right]\left(T^{-1} + VS^{-1}U\right)^{-1} VS^{-1} \\
&= I + UTVS^{-1} - UT\left[T^{-1} + VS^{-1}U\right]\left(T^{-1} + VS^{-1}U\right)^{-1} VS^{-1} \\
&= I + UTVS^{-1} - UTVS^{-1} \\
&= I
\end{aligned}$$

On the other hand,

$$\begin{aligned}
&\left[S^{-1} - S^{-1}U\left(T^{-1} + VS^{-1}U\right)^{-1} VS^{-1}\right](S + UTV) \\
&= I + S^{-1}UTV - S^{-1}U\left(T^{-1} + VS^{-1}U\right)^{-1}\left[V + VS^{-1}UTV\right] \\
&= I + S^{-1}UTV - S^{-1}U\left(T^{-1} + VS^{-1}U\right)^{-1}\left[T^{-1} + VS^{-1}U\right]TV \\
&= I + S^{-1}UTV - S^{-1}UTV \\
&= I
\end{aligned}$$

Hence,

$$\left(S+UTV\right)^{-1}=S^{-1}-S^{-1}U\left(T^{-1}+VS^{-1}U\right)^{-1}VS^{-1}$$

Applying the Matrix Inversion Lemma into the eq. (4.35), we have:

$$\operatorname{cov}(f)=L^{-1}-L^{-1}\overline{Q}^{T}\left(\frac{T_k}{E[\beta_{im}]}\left(\overline{B^{*}}^{T}\,\overline{B^{*}}-\overline{B^{*}}^{T}-\overline{B^{*}}\right)^{-1}+\overline{Q}L^{-1}\overline{Q}^{T}\right)^{-1}\overline{Q}L^{-1} \qquad (4.36)$$

** <u>*Blurring estimation:*</u>

The covariance matrix and expectation of blurring function are written as the following equations:

$$\operatorname{cov}(h)=\left(E[\beta_n]E[F]^T E[F]+E[\beta_n]\operatorname{cov}(F)+E[\beta_{bl}]C^T C\right)^{-1}$$

$$E[h]=E[\beta_{bl}]\operatorname{cov}(h)E[F]^T g$$

The first term in blurring estimation equation relates to the special matrix F whose row is shifted one element to the left relative to its preceding row.

$$F=\begin{bmatrix} f_1 & f_2 & f_3 & \cdots & f_N \\ f_2 & f_3 & f_4 & \cdots & f_1 \\ f_3 & f_4 & f_5 & \cdots & f_2 \\ \vdots & \vdots & \vdots & \iddots & \vdots \\ f_N & f_1 & f_2 & \cdots & f_{N-1} \end{bmatrix} \qquad (4.37)$$

Although F is not a circulant matrix and can not be diagonalised by the Fourier transformation, we use another way to diagonalise this matrix by using the Fourier transformation matrix. We called F a left-wise circulant matrix. It is worth noting that a real left-wise circulant matrix is a symmetric matrix: $F=F^T$

F can be written as a polynomial of matrices R and $\tilde{I}$ as follows:

$$F=f_N\tilde{I}+f_{N-1}\tilde{I}R+f_{N-2}\tilde{I}R^2+\ldots+f_1\tilde{I}R^{N-1} \qquad (4.38)$$

$R=[e_2 \quad e_3 \quad \ldots \quad e_N \quad e_1]$, and $\tilde{I}=[e_N \quad e_{N-1} \quad \ldots \quad e_2 \quad e_1]$

where e_k is the k^{th} column of the identity matrix.

Theorem 3: The left-wise circulant matrix F is diagonalised by the following transformation:

$$TFT = diag\left(\sqrt{N}Tf\right)$$

where f is the first column of matrix F: $f = \begin{bmatrix} f_1 \\ f_2 \\ f_3 \\ \vdots \\ f_N \end{bmatrix}$

To prove this theorem, we need to prove that the matrices R and $\tilde{I}$ are diagonalised by some transformations in Lemma 2 and Lemma 3.

Lemma 3: $T^T RT = D^T$

where $D = diag\left(1, \omega^1, \omega^2, \ldots, \omega^{(N-1)}\right)$

Proof: (Lemma 3)

The kl^{th} element of matrix RT is:

$$[RT]_{kl} = e_{k-1}^T \frac{1}{\sqrt{N}} \begin{bmatrix} 1 \\ \omega^{(l-1)} \\ \omega^{2(l-1)} \\ \vdots \\ \omega^{(N-1)(l-1)} \end{bmatrix} = \frac{1}{\sqrt{N}} \omega^{(k-2)(l-1)}$$

The kl^{th} element of matrix TD^T is:

$$\left[TD^T\right]_{kl} = \frac{1}{\sqrt{N}}\left[1 \quad \omega^{(k-1)} \quad \omega^{2(k-1)} \quad \ldots \quad \omega^{(N-1)(k-1)}\right]\omega^{-(l-1)} e_l = \frac{1}{\sqrt{N}} \omega^{(k-2)(l-1)}$$

Hence,

$$RT = TD^T$$

Or:

$$T^T RT = D^T$$

Lemma 4: $T\tilde{I}T = \tilde{D}$

where $\tilde{D} = diag\left(1, \omega^{N-1}, \omega^{2(N-1)}, \ldots, \omega^{(N-1)(N-1)}\right)$

Proof: (Lemma 4)

The kl^{th} element of matrix $T\tilde{I}$ is:

$$\left[T\tilde{I}\right]_{kl} = \frac{1}{\sqrt{N}}\left[1 \quad \omega^{(k-1)} \quad \omega^{2(k-1)} \quad \ldots \quad \omega^{(N-1)(k-1)}\right]e_{N-l+1} = \frac{1}{\sqrt{N}}\,\omega^{(N-l)(k-1)}$$

The kl^{th} element of matrix TR is:

$$\left[\tilde{D}T^T\right]_{kl} = \omega^{(k-1)(N-1)}e_k\,\frac{1}{\sqrt{N}}\begin{bmatrix} 1 \\ \omega^{-(l-1)} \\ \omega^{-2(l-1)} \\ \vdots \\ \omega^{-(N-1)(l-1)} \end{bmatrix} = \frac{1}{\sqrt{N}}\,\omega^{(N-l)(k-1)}$$

Hence,

$$T\tilde{I} = \tilde{D}T^T$$

Or:

$$T\tilde{I}T = \tilde{D}$$

Proof: (Theorem 3)

We have:

$$F = f_N\tilde{I} + f_{N-1}\tilde{I}R + f_{N-2}\tilde{I}R^2 + \ldots + f_1\tilde{I}R^{N-1}$$

So,

$$
\begin{aligned}
TFT &= f_N T\tilde{I}T + f_{N-1}\left(T\tilde{I}T\right)\left(T^T RT\right) + f_{N-2}\left(T\tilde{I}T\right)\left(T^T R^2 T\right) + \ldots + f_1\left(T\tilde{I}T\right)\left(T^T R^{N-1}T\right) \\
&= T\tilde{I}T\left[f_N + f_{N-1}\left(T^T RT\right) + f_{N-2}\left(T^T RT\right)^2 + \ldots + f_1\left(T^T RT\right)^{N-1}\right] \\
&= \tilde{D}\left[f_N + f_{N-1}D^T + f_{N-2}\left(D^T\right)^2 + \ldots + f_1\left(D^T\right)^{N-1}\right]
\end{aligned}
$$

Since the matrices $\tilde{D}$ and D are diagonal, TFT is diagonal. The kk^{th} element of TFT is:

$$
\begin{aligned}
\left[TFT\right]_{kk} &= \omega^{(k-1)(N-1)}\left[f_N + f_{N-1}\omega^{-(k-1)} + f_{N-2}\omega^{-2(k-1)} + \ldots + f_1\omega^{-(N-1)(k-1)}\right] \\
&= f_N\omega^{(k-1)(N-1)} + f_{N-1}\omega^{(N-2)(k-1)} + f_{N-2}\omega^{(N-3)(k-1)} + \ldots + f_1
\end{aligned}
$$

Meanwhile, the k^{th} element of Tf is:

$$
\begin{aligned}
\left[Tf\right]_k &= \frac{1}{\sqrt{N}}\begin{bmatrix} 1 & \omega^{(k-1)} & \omega^{2(k-1)} & \ldots & \omega^{(N-1)(k-1)} \end{bmatrix}\begin{bmatrix} f_1 \\ f_2 \\ f_3 \\ \vdots \\ f_N \end{bmatrix} \\
&= \frac{1}{\sqrt{N}}\left(f_1 + f_2\omega^{(k-1)} + f_3\omega^{-2(k-1)} \ldots + f_N\omega^{(k-1)(N-1)}\right)
\end{aligned}
$$

Thus,

$$
TFT = diag\left(\sqrt{N}Tf\right)
$$

<u>Theorem 4:</u> F is a left-wise circulant matrix whose first column is f.

$$
F = \begin{bmatrix} f_1 & f_2 & f_3 & \cdots & f_N \\ f_2 & f_3 & f_4 & \cdots & f_1 \\ f_3 & f_4 & f_5 & \cdots & f_2 \\ \vdots & \vdots & \vdots & \iddots & \vdots \\ f_N & f_1 & f_2 & \cdots & f_{N-1} \end{bmatrix} \text{ and } f = \begin{bmatrix} f_1 \\ f_2 \\ f_3 \\ \vdots \\ f_N \end{bmatrix}
$$

The covariance matrix of f is assumed to be circulant. Then:

$$
\operatorname{cov}(F) = N\operatorname{cov}(f)
$$

where N is the size of the column vector f

Proof: (Theorem 4)

As F is the left-wise circulant matrix whose first column is the vector f, each row of $F^T F$ is a convolution of f and itself. Therefore, each element (i,j) of $F^T F$ can be represented as following:

$$\left[F^T F\right]_{(i,j)} = \sum_{\substack{k,l \\ \mathrm{mod}(k-l,n)=i-j}} \left(ff^T\right)$$

Hence, we have the expectation:

$$\begin{aligned}\left[E\left[F^T F\right]\right]_{(i,j)} &= \sum_{\substack{k,l \\ \mathrm{mod}(k-l,N)=i-j}} E\left[ff^T\right] \\ &= \sum_{\substack{k,l \\ \mathrm{mod}(k-l,N)=i-j}} \left(E[f]E\left[f^T\right]\right) + \sum_{\substack{k,l \\ \mathrm{mod}(k-l,N)=i-j}} \mathrm{cov}(f)\end{aligned}$$

$E[F]$ is a left-wise circulant matrix whose first column equals to $E[f]$. It means that:

$$\sum_{\substack{k,l \\ \mathrm{mod}(k-l,N)=i-j}} \left(E[f]E\left[f^T\right]\right) = \left[E\left[F^T\right]E[F]\right]_{(i,j)}$$

As $\mathrm{cov}(f)$ is assumed to be a circulant covariance matrix, it can be deduced:

$$\sum_{\substack{k,l \\ \mathrm{mod}(k-l,N)=i-j}} \mathrm{cov}(f) = N\left[\mathrm{cov}(f)\right]_{(i,j)}$$

Then,

$$\left[E\left[F^T F\right]\right]_{(i,j)} = \left[E\left[F^T\right]E[F]\right]_{(i,j)} + N\left[\mathrm{cov}(f)\right]_{(i,j)}$$

Or:

$$\begin{aligned}\mathrm{cov}(F) &= E\left[F^T F\right] - E\left[F^T\right]E[F] \\ &= N\,\mathrm{cov}(f)\end{aligned}$$

Appling Theorem 4 into the estimation equation of blurring covariance, we have:

$$\operatorname{cov}(h) = \left(E[\beta_n]E[F]^T E[F] + E[\beta_n]N\operatorname{cov}(f) + E[\beta_{bl}]C^T C\right)^{-1} \tag{4.39}$$

To reduce the computational requirement, we also approximate the estimation of $\operatorname{cov}(h)$ by replacing $\operatorname{cov}(f)$ with L in eq. (4.39):

$$\operatorname{cov}(h) = \left(E[\beta_n]E[F]^T E[F] + E[\beta_n]NL + E[\beta_{bl}]C^T C\right)^{-1} \tag{4.40}$$

** <u>*Parameter estimations:*</u>

The estimate of the image parameter is:

$$E[\beta_{im}] = \frac{a_{im}^0 + \frac{N}{2}}{b_{im}^0 + \frac{1}{2T_k}E\left[f^T B^T Bf\right]}$$

in which, $E\left[f^T B^T Bf\right] = E[f]^T B^T BE[f] + trace\left(B^T B\operatorname{cov}(f)\right)$

Note that matrix $B^T B$ can be analysed as:

$$B^T B = \left(I - \overline{Q}^T \overline{B^*}\,\overline{Q}\right)^T \left(I - \overline{Q}^T \overline{B^*}\,\overline{Q}\right) \tag{4.41}$$

where $\overline{Q}\,\overline{Q}^T = I$

$$B^T B = I + \overline{Q}^T \left(\overline{B^*}^T \overline{B^*} - \overline{B^*}^T - \overline{B^*}\right)\overline{Q} \tag{4.42}$$

Besides, the covariance matrix of image $\operatorname{cov}(f)$ can be approximated by L. Hence, the image parameter is approximated by the following equation:

$$E[\beta_{im}] = \frac{a_{im}^0 + \frac{N}{2}}{b_{im}^0 + \frac{1}{2T_k}\left[f^T f + (\overline{Q}f)^T \left(\overline{B^*}^T \overline{B^*} - \overline{B^*}^T - \overline{B^*} \right)(\overline{Q}f) + trace\left(L + \overline{Q}^T \left(\overline{B^*}^T \overline{B^*} - \overline{B^*}^T - \overline{B^*} \right)\overline{Q}L \right) \right]} \tag{4.43}$$

The estimate of the blurring parameter is:

$$E[\beta_{bl}] = \frac{a_{bl}^0 + \frac{M}{2}}{b_{bl}^0 + \frac{1}{2}E\left[h^T C^T Ch\right]}$$

or,

$$E[\beta_{bl}] = \frac{a_{bl}^0 + \frac{M}{2}}{b_{bl}^0 + \frac{1}{2}\left\{E[h]^T C^T CE[h] + trace\left(C^T C \operatorname{cov}(h)\right)\right\}} \tag{4.44}$$

Hence, the expectation of blurring parameter can be calculated easily in the Fourier domain as all component matrices in eq. (4.44) are circulant.

The estimate of the noise parameter is:

$$E[\beta_n] = \frac{a_n^0 + \frac{N}{2}}{b_n^0 + \frac{1}{2}E\left[\|g - Hf\|^2\right]_{q(f)q(h)}}$$

in which,

$$\begin{aligned}
E\left[\|g - Hf\|^2\right]_{q(f)q(h)} &= g^T g - g^T E[H]E[f] - E[f]^T E[H]^T g + E\left[f^T H^T Hf\right]_{q(f)q(h)} \\
&= \|g - E[H]E[f]\|^2 - E[f]^T E[H]^T E[H]E[f] + E\left[f^T H^T Hf\right]_{q(f)q(h)} \\
&= \|g - E[H]E[f]\|^2 + E[f]^T \operatorname{cov}(H)E[f] - E[f]^T E\left[H^T H\right]E[f] \\
&\qquad + E[f]^T E\left[H^T H\right]E[f] + trace\left(E\left[H^T H\right]\operatorname{cov}(f)\right) \\
&= \|g - E[H]E[f]\|^2 + E[f]^T \operatorname{cov}(H)E[f] \\
&\qquad + trace(\operatorname{cov}(H)\operatorname{cov}(f)) + trace\left(E[H]^T E[H]\operatorname{cov}(f)\right)
\end{aligned} \tag{4.45}$$

Approximating $\mathrm{cov}(f)$ by L and calculating $\mathrm{cov}(H)$ following $\mathrm{cov}(h)$, we obtain the following equation which can be implemented in the Fourier domain:

$$E[\beta_n] = \frac{a_n^0 + \dfrac{N}{2}}{b_n^0 + \dfrac{1}{2}\left[\|g - E[H]E[f]\|^2 + NE[f]^T \mathrm{cov}(h)E[f] + Ntrace(\mathrm{cov}(h)L) + trace\left(E[H]^T E[H]L\right)\right]} \tag{4.46}$$

4.4.3. Proposed algorithms

4.4.3.1. LF-SAR algorithm

Finally, we propose a new blind deblurring algorithm by using the optimum solution derived in section 4.4.1 and applying the theorems stated in section 4.4.2. In particular, the computation of circulant and left-wise circulant matrices in the optimum solution is feasible by applying Theorem 1 and Theorem 3 while the covariance of circulant and left-wise circulant matrices is replaced by simple expressions following Theorem 2 and Theorem 4. The proposed theorems are really helpful for the implementation of the proposed algorithm because calculating the big *N*-by-*N* matrix is replaced by calculating and storing the 1-by-*N* vector following these theorems. The proposed deblurring algorithm, called LF-SAR, is represented below with three iterative steps.

LF-SAR algorithm:

- Step 1: estimate the covariance and the expectation of f

$$\mathrm{cov}(f) = \left(E[\beta_n]E[H]^T E[H] + E[\beta_n]N\,\mathrm{cov}(h) + \frac{E[\beta_{im}]}{T_k}B^T B\right)^{-1}$$

$$E[f] = E[\beta_n]\mathrm{cov}(f)E[H]^T g \tag{4.47}$$

- Step 2: estimate the covariance and the expectation of *h*

$$\mathrm{cov}(h) = \left(E[\beta_n]E[F]^T E[F] + E[\beta_n]N\,\mathrm{cov}(f) + E[\beta_{bl}]C^T C\right)^{-1}$$

$$E[h] = E[\beta_{bl}]\mathrm{cov}(h)E[F]^T g \tag{4.48}$$

- Step 3: estimate the expectation of the parameters β_n, β_{im}, and β_{bl}

$$E[\beta_{im}] = \frac{a_{im}^0 + \frac{N}{2}}{b_{im}^0 + \frac{1}{2T_k}\left[\begin{array}{c} f^T f + \left(\overline{Q}f\right)^T\left(\overline{B^*}^T\,\overline{B^*} - \overline{B^*}^T - \overline{B^*}\right)\left(\overline{Q}f\right) \\ + trace\left(\mathrm{cov}(f) + \overline{Q}^T\left(\overline{B^*}^T\,\overline{B^*} - \overline{B^*}^T - \overline{B^*}\right)\overline{Q}\,\mathrm{cov}(f)\right) \end{array}\right]}$$

$$E[\beta_{bl}] = \frac{a_{bl}^0 + \frac{M}{2}}{b_{bl}^0 + \frac{1}{2}\left\{E[h]^T C^T C E[h] + trace\left(C^T C\,\mathrm{cov}(h)\right)\right\}}$$

$$E[\beta_n] = \frac{a_n^0 + \frac{N}{2}}{b_n^0 + \frac{1}{2}\left[\left\|g - E[H]E[f]\right\|^2 + NE[f]^T\mathrm{cov}(h)E[f] + trace\left(N\,\mathrm{cov}(h)\mathrm{cov}(f) + E[H]^T E[H]\mathrm{cov}(f)\right)\right]} \tag{4.49}$$

It is noted that to reduce the computational complexity, the term $\mathrm{cov}(f)$ in step 2 and step 3 is approximated by a circulant matrix $\overline{\mathrm{cov}(f)}$ as follows:

$$\overline{\mathrm{cov}(f)} = \left(E[\beta_n]E[H]^T E[H] + E[\beta_n]N\,\mathrm{cov}(h) + \frac{E[\beta_{im}]}{T_k}\right)^{-1} \tag{4.50}$$

4.4.3.2. LF-G algorithm

Similarly, we construct another deblurring algorithm, called LF-G algorithm, which uses the Gaussian distribution as the prior information of blurring function instead of the SAR model. Performing the derivation procedure in section 4.3.1 again and applying the theorems stated in section 4.3.2, we have the LF-G algorithm which also consists of three iterative steps as shown below:

LF-G algorithm:

- Step 1: estimate the covariance and the expectation of *f*

$$\mathrm{cov}(f) = \left(E[\beta_n] E[H]^T E[H] + E[\beta_n] N\,\mathrm{cov}(h) + \frac{E[\beta_{im}]}{T_k} B^T B \right)^{-1}$$

$$E[f] = E[\beta_n] \mathrm{cov}(f) E[H]^T g \tag{4.51}$$

- Step 2: estimate the covariance and the expectation of *h*

$$\mathrm{cov}(h) = \left(E[\beta_n] E[F]^T E[F] + E[\beta_n] N\,\mathrm{cov}(f) + E[\beta_{bl}] \right)^{-1}$$

$$E[h] = E[\beta_{bl}] \mathrm{cov}(h) E[F]^T g \tag{4.52}$$

- Step 3: estimate the expectation of the parameters β_n, β_{im}, and β_{bl}

$$E[\beta_{im}] = \frac{a_{im}^0 + \frac{N}{2}}{b_{im}^0 + \frac{1}{2T_k} \left[\begin{array}{l} f^T f + (\overline{Q} f)^T \left(\overline{B^*}^T \overline{B^*} - \overline{B^*}^T - \overline{B^*} \right) (\overline{Q} f) \\ \qquad + trace\left(\mathrm{cov}(f) + \overline{Q}^T \left(\overline{B^*}^T \overline{B^*} - \overline{B^*}^T - \overline{B^*} \right) \overline{Q}\,\mathrm{cov}(f) \right) \end{array} \right]}$$

$$E[\beta_{bl}] = \frac{a_{bl}^0 + \frac{M}{2}}{b_{bl}^0 + \frac{1}{2} E[h]^T E[h]}$$

$$E[\beta_n] = \frac{a_n^0 + \frac{N}{2}}{b_n^0 + \frac{1}{2}\left[\| g - E[H]E[f] \|^2 + N E[f]^T \mathrm{cov}(h) E[f] + trace\left(N\,\mathrm{cov}(h)\mathrm{cov}(f) + E[H]^T E[H] \mathrm{cov}(f) \right) \right]} \tag{4.53}$$

The term $\mathrm{cov}(f)$ in step 2 and 3 of LF-G algorithm is also approximated by the circulant matrix $\overline{\mathrm{cov}(f)}$ as shown in eq. (4.50) to reduce the computational complexity of the algorithm. It is noted that $\mathrm{cov}(f)$ is calculated directly in step 1 of the algorithm to guarantee the precise estimation of the image. The approximation of this term in step 2 and 3 is a tradeoff between the computational complexity of proposed algorithm and the performance of algorithm. To estimate the blurring function model parameters more precisely, the direct calculation of $\mathrm{cov}(f)$ in step 2 and 3 by using the conjugate gradient method is considered as the future task to develop this work.

4.4.3.3. Initial value dependence

It is noted that the parameters $a_n^0, a_{im}^0, a_{bl}^0, b_n^0, b_{im}^0,$ and b_{bl}^0 of the prior distribution of $\beta_n, \beta_{im},$ and β_{bl} appear in the parameter estimation equations (4.49) and (4.53). Hence, we now analyse the initial value dependence of the two proposed algorithms. Eq. (4.49) and eq. (4.53) can be rewritten in term of the initial value of parameters $\beta_n, \beta_{im},$ and β_{bl}. Since these parameters are Gamma distributed, their initial values can be determined by the following equation:

$$\beta_x^0 = \frac{a_x^0}{b_x^0} \ , \ x \in \{im, bl, n\}$$

Hence, the parameter estimation equations above are rewritten in term of $\beta_n^0, \beta_{im}^0,$ and β_{bl}^0 and confidence coefficients $\gamma_{\beta_n}, \gamma_{\beta_{im}},$ and $\gamma_{\beta_{bl}}$ instead of $a_n^0, a_{im}^0, a_{bl}^0, b_n^0, b_{im}^0,$ and b_{bl}^0:

$$E[\beta_{im}] = \left[\frac{\gamma_{im}}{\beta_{im}^0} + \frac{(1-\gamma_{im})}{N} \frac{1}{T_k} \left\{ f^T f + \left(\overline{Q} f\right)^T \left(\overline{B^*}^T \overline{B^*} - \overline{B^*}^T - \overline{B^*} \right) \left(\overline{Q} f\right) + trace\left(\mathrm{cov}(f) + \overline{Q}^T \left(\overline{B^*}^T \overline{B^*} - \overline{B^*}^T - \overline{B^*} \right) \overline{Q}\, \mathrm{cov}(f) \right) \right\} \right]^{-1}$$

$$E[\beta_{bl}] = \left[\frac{\gamma_{bl}}{\beta_{bl}^0} + \frac{(1-\gamma_{bl})}{M} \left\{ E[h]^T C^T C E[h] + trace\left(C^T C\, \mathrm{cov}(h) \right) \right\} \right]^{-1}$$

$$E[\beta_n] = \left[\frac{\gamma_n}{\beta_n^0} + \frac{(1-\gamma_n)}{N} \left\{ \| g - E[H] E[f] \|^2 + N E[f]^T \mathrm{cov}(h) E[f] + N trace(\mathrm{cov}(h) \mathrm{cov}(f)) + trace\left(E[H]^T E[H] \mathrm{cov}(f) \right) \right\} \right]^{-1} \qquad (4.54)$$

The confidence coefficients $\gamma_{\beta_n}, \gamma_{\beta_{im}}$, and $\gamma_{\beta_{bl}}$ represent how much the initial values β_n^0, β_{im}^0, and β_{bl}^0 should be trusted during the estimation of parameters β_n, β_{im}, and β_{bl}. The confidence coefficients are calculated by the following equations:

$$\gamma_x = \frac{a_x^0}{a_x^0 + \frac{N}{2}}, \; x \in \{\beta_{im}, \beta_{bl}, \beta_n\}$$

It is interesting that the value of these confidence coefficients depends only on the initial parameters a_n^0, a_{im}^0, and a_{bl}^0. Hence, if a_n^0, a_{im}^0, and a_{bl}^0 are chosen so that the confidence coefficients are approximated to zero, there is almost no confidence in the initial values β_n^0, β_{im}^0, and β_{bl}^0. It means that the finally estimated value of model parameters β_n, β_{im}, and β_{bl} depends entirely on the estimation process.

In the next chapter, two proposed algorithms, LF-SAR and LF-G, are used to carry out deblurring experiments with different types of blurring function and various levels of noise. The sensitivity of the algorithms to the initial values is also investigated by carrying out many experiments with several sets of initial values β_n^0, β_{im}^0, and β_{bl}^0 and confidence coefficients $\gamma_{\beta_n}, \gamma_{\beta_{im}}$, and $\gamma_{\beta_{bl}}$.

Chapter 5

Experimental Studies for Deblurring

5.1. Introduction

The LF-SAR and LF-G algorithms, developed in this thesis, which were introduced in Chapter 4, will be used in this chapter to restore blurred images caused by different types of PSF, such as: the Gaussian-shape PSF, the out-of-focus PSF, and the horizontally uniform PSF. All experiments carried out in this chapter use “Lena” image of size 512×512 pixels. Besides, some other images, such as: “cameraman”, “boat”, “Barbara”, “montage”, and “Flintstones” images (see Appendix A) , are also used to show the performance of proposed algorithms in wide range of image patterns. The contaminated noise used in most of these experiments is the identically independently Gaussian random process with variance 10^{-6}. Meanwhile, the experiments in section 5.5 are carried out with different levels of contaminated noise to study the effect of noise on the deblurring result. In all these experiments, the initial covariance matrices of image and blurring function are zero.

These algorithms are iterative and stop if one among the two following criteria is satisfied:

- the difference in magnitude between the image estimates of two subsequent steps is less than 10^{-6};
- the number of bond (line) between pixels in the whole image is larger than a pre-defined threshold, 1000 in our algorithms. Since the number of edge pixels

in an image is limited, there is a possibility that the algorithm will end up with a conventional solution if the number of bond between pixels becomes so high during the implementation of algorithm. Thus, the number of bond (line) between pixels is chosen as a termination criterion so that the computational effort will not be wasted in this case.

The deblurring results of these two algorithms are compared in term of ISNR (Improved Signal Noise Ratio) and ISNR_h indices:

$$ISNR = 10\log\frac{\|f-g\|^2}{\|f-\hat{f}\|^2}$$

and,

$$ISNR\_h = 10\log\frac{\|h-h_{init}\|^2}{\|h-\hat{h}\|^2}$$

where $\hat{f}$, $\hat{h}$ are estimates of the original image f and the blurring function h, respectively;

g is the noisy blurred observation;

h_{init} is the initial value of blurring function h.

5.2. Image deblurring with the Gaussian-shape PSF

The Gaussian-shape PSF has the general form represented in the following equation:

$$h(x,y) = \begin{cases} \dfrac{1}{2\pi\sigma^2\left(1-\exp\left(-R^2/2\sigma^2\right)\right)}\exp\left(-\dfrac{x^2+y^2}{2\sigma^2}\right) & x^2+y^2 \le R^2 \\ 0 & otherwise \end{cases} \tag{5.1}$$

where R is the radius of PSF;

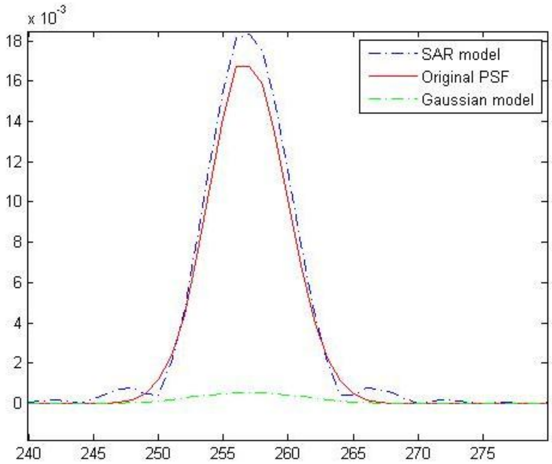
σ^2 is the variance of Gaussian-shape PSF;

x, y are the indices of blurring matrix h. These indices are determined in the local coordinate system of each pixel.

a)

b)

c)



d)

Figure 5-1. Deblurring results using LF-SAR algorithm and LF-G algorithm. a) the noisy blurred observation (Gaussian-shape PSF with variance 9, noise variance 10^{-6}); b) deblurring result by LF-SAR; c) deblurring result by LF-G; d) a slice cut of PSF estimates and the real PSF.

The experiments in this section are carried out with six images in the same blurring and noisy condition. The real PSF is the Gaussian-shape PSF with variance 9 while the initial PSF is the Gaussian-shape PSF with variance 4. The initial parameters of the image model, the blurring model, and the noise variance are assigned the same values as those used in experiment 2 of paper [30], *i.e* $\beta_n^0 = 0.22, \beta_{bl}^0 = 10^{-6}, \beta_{im}^0 = 93.6$. The confidence coefficients of these parameters are $\gamma_n = 0.001, \gamma_{bl} = 0.001$, and $\gamma_{im} = 0.5$, respectively.

Table 5.1. The ISNR and ISNR_h [dB] of the image and PSF estimated by LF-SAR algorithm and LF-G algorithm with the observation blurred by a Gaussian-shape PSF.

Image	LF-SAR		LF-G	
	ISNR	ISNR_h	ISNR	ISNR_h
Lena	3.18	8.33	1.53	-3.18
Cameraman	2.35	7.68	0.93	2.00
Boat	2.82	8.08	0.61	0.35
Barbara	1.18	8.29	0.34	0.97
Montage	2.27	7.78	0.83	2.04
Flintstones	3.36	6.66	0.93	3.41

Table 5.1 indicates that the restored image and PSF estimate of LF-SAR algorithm are more accurate than those of LF-G algorithm. The image shown in Figure 5-1 (b) estimated by LF-SAR is clearer than the image Figure 5-1 (c) estimated by LF-G under visual inspection. A slice cut of PSF estimate shown in Figure 5-1 (d) also indicates that LF-SAR algorithm estimates the PSF much more accurately than LF-G algorithm does.

These interesting findings can be explained by the difference between the models of blurring function which are used in these two algorithms. In the LF-SAR algorithm, the Simultaneous Auto-Regression model is used to model the distribution of PSF. As a result, the elements of the kernel blurring matrix vary in different ways which depend on each element’s neighbours. In the LF-G algorithm, the independent Gaussian distributions are used to model the PSF. Since the covariance matrix of PSF is assumed to be circulant, the variation at all elements of the kernel blurring matrix are the same. Thus, the LF-G algorithm is less flexible than the LF-SAR algorithm in approximating the real PSF. As a result, a better restored image is produced by the LF-SAR algorithm, which is the algorithm using the more appropriate PSF model.

5.3. Image deblurring with the horizontally uniform PSF

The horizontally uniform PSF is simply described in eq. (5.2):

$$h(x) = \begin{cases} \dfrac{1}{d} & 0 \le x \le d \\ 0 & otherwise \end{cases} \tag{5.2}$$

where d is the extent of the motion.

Table 5.2. ISNR and ISNR_h [dB] of the image and PSF estimated by LF-SAR algorithm and LF-G algorithm with the observation blurred by a horizontally uniform PSF.

Image	LF-SAR		LF-G	
	ISNR	ISNR_h	ISNR	ISNR_h
Lena	4.58	13.86	4.07	24.20
Cameraman	2.39	23.58	3.95	26.39
Boat	2.55	24.12	4.44	24.67
Barbara	1.40	24.23	2.70	24.68
Montage	1.67	22.72	4.60	27.54
Flintstones	1.67	20.81	3.33	21.66

The horizontally uniform PSF in these experiments has a square support 9×9. The initial parameters are $\beta_n^0 = 10^6, \beta_{bl}^0 = 10^{-6}$, and $\beta_{im}^0 = 93.6$, while their confidence coefficients are $\gamma_n = 0.001, \gamma_{bl} = 0.001$ and $\gamma_{im} = 0.001$, respectively. The initial PSF is also the Gaussian-shape PSF with variance 4. In this case, because the support size of PSF is limited, the PSF estimate will be shrunk by a square window of size 9×9 and normalised after each iterative step.

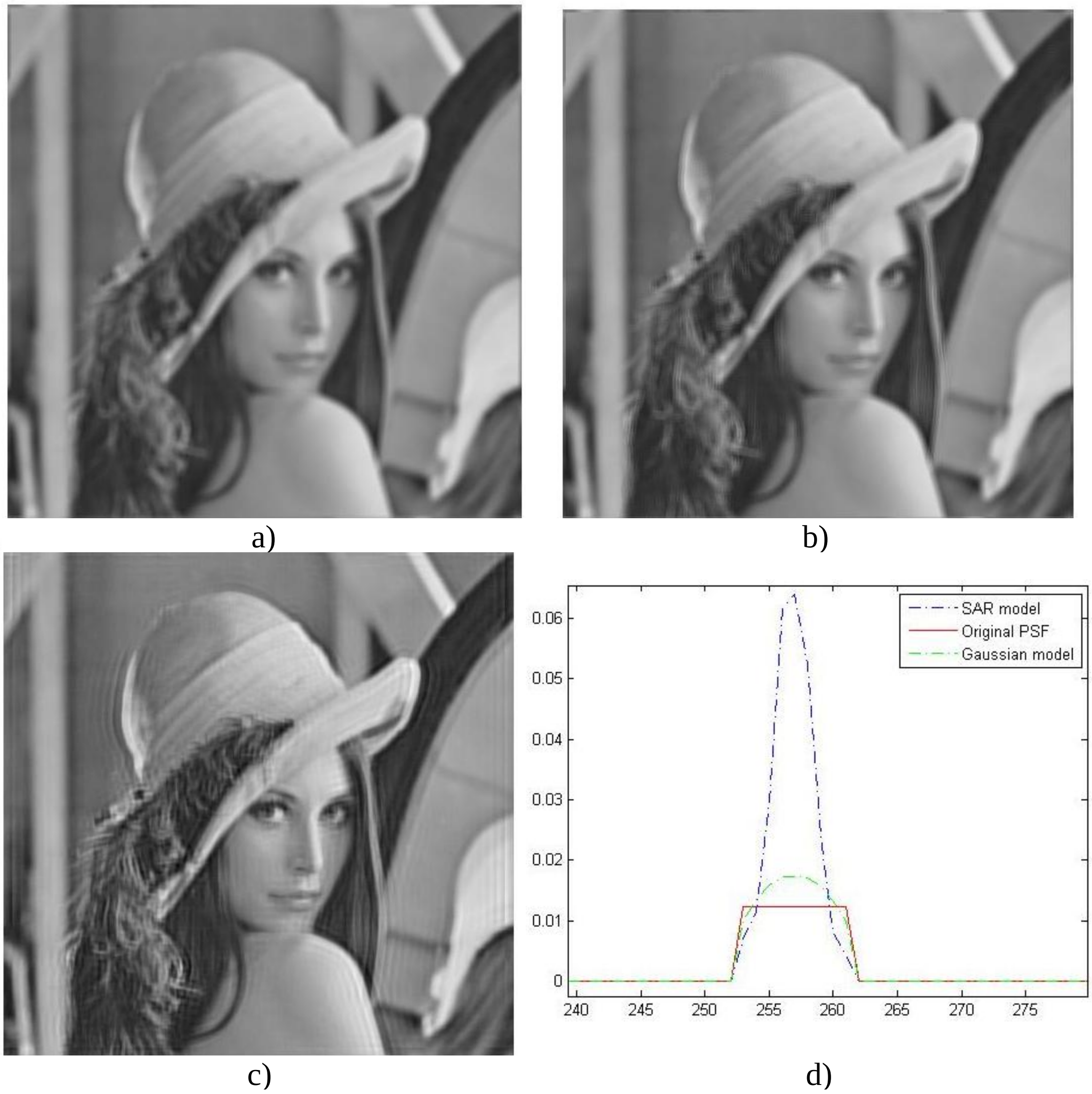


Figure 5-2. Deblurring results using LF-SAR algorithm and LF-G algorithm. a) the noisy blurred observation (horizontally uniform PSF with the support size 9×9, noise variance 10^{-6}); b) deblurring result by LF-SAR; c) deblurring result by LF-G; d) a slice cut of PSF estimates and the real PSF.

In contrast with the result shown in Table 5.1 in the previous section, Table 5.2 shows that the LF-G algorithm produces better results than the LF-SAR algorithm in this case where the blurred observation is caused by a horizontally uniform blurring function. This finding can have the similar explanation as presented in the previous section. It is due to the difference between the models of blurring function which are used in the two algorithms. As the covariance matrix of PSF is assumed to be circulant, the independently Gaussian distributed elements of PSF always have the same variance. Hence, the LF-G algorithm tends to have an advantage in dealing with the horizontally uniform blurring function.

5.4. Image deblurring with the out-of-focus PSF

The kernel blurring function of out-of-focus phenomenon is modeled by the following equation:

$$h(x,y) = \begin{cases} \dfrac{1}{\pi R^2} & x^2 + y^2 \le R^2 \\ 0 & otherwise \end{cases} \tag{5.3}$$

where R is the radius of the out-of-focus PSF.

Table 5.3. ISNR and ISNR_h [dB] of the image and PSF estimated by LF-SAR algorithm and LF-G algorithm with the observation blurred by an out-of-focus PSF.

Image	LF-SAR		LF-G	
	ISNR	ISNR_h	ISNR	ISNR_h
Lena	2.95	9.60	1.75	5.82
Cameraman	2.22	7.90	2.14	6.32
Boat	2.35	7.75	1.68	6.14
Barbara	2.03	9.49	1.53	5.81
Montage	2.17	7.93	2.10	6.46
Flintstones	2.85	9.46	2.11	7.06

The real PSF in this series of experiments is an out-of-focus PSF with the support size 7×7. The initial PSF is assigned by an out-of-focus PSF with the support size 3×3. The initial parameters and their confidence coefficients are

$\beta_n^0 = 10^6$, $\beta_{bl}^0 = 10^{-6}$, $\beta_{im}^0 = 93.6$, $\gamma_n = 0.001$, $\gamma_{bl} = 0.001$, and $\gamma_{im} = 0.5$, respectively. In these experiments, the PSF estimate is also shrunk by a square window 9×9 and normalised after each iterative step. The result in Table 5.3 and the visual appearance in Figure 5-3 show that the LF-SAR algorithm is marginally better than the LF-G algorithm in this case.

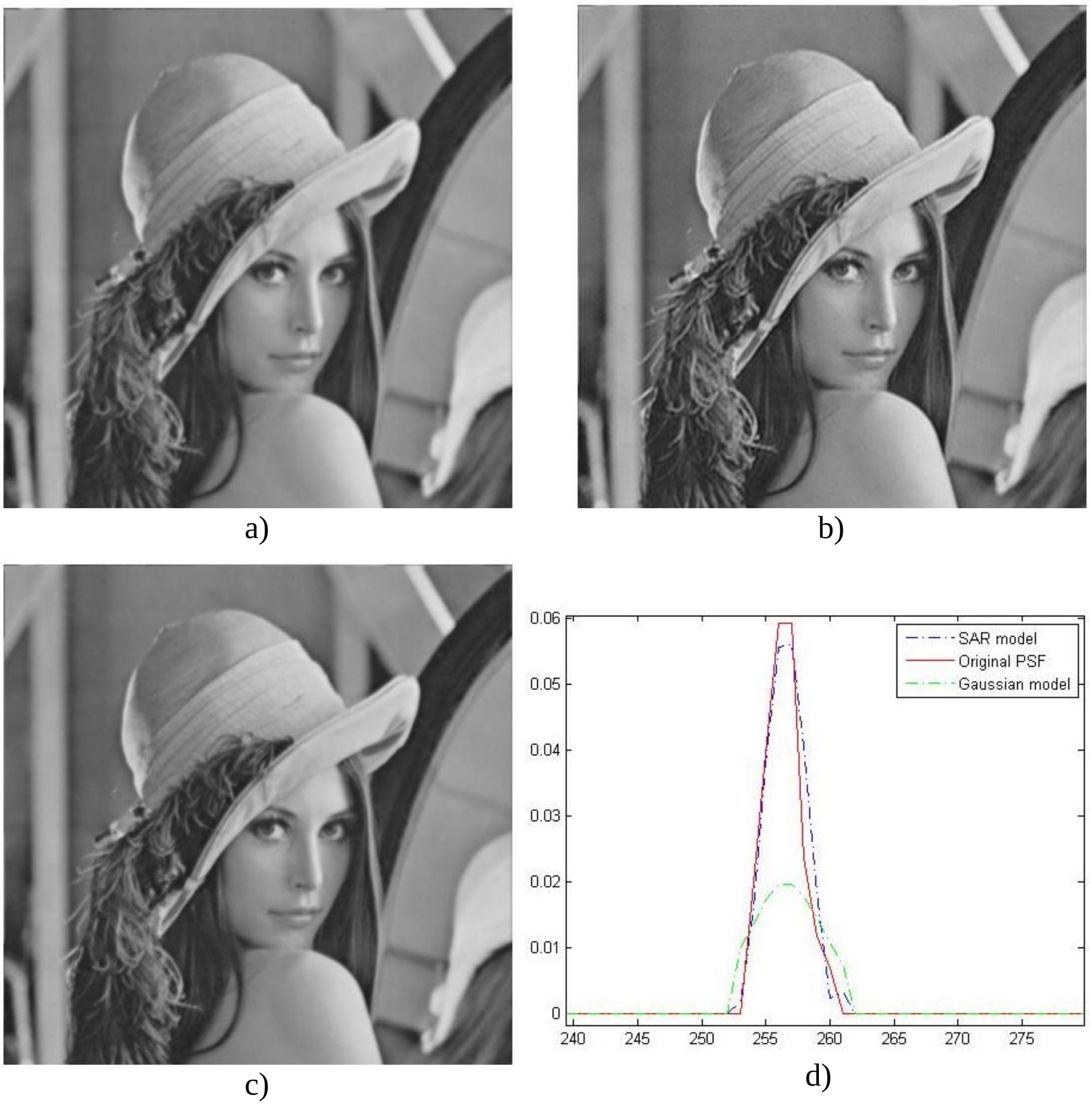


Figure 5-3. Deblurring results using LF-SAR algorithm and LF-G algorithm. a) the noisy blurred observation (out-of-focus PSF with the size support 7×7, noise variance 10^{-6}); b) deblurring result by LF-SAR; c) deblurring result by LF-G; d) a slice cut of PSF estimates and the real PSF.

Comparing the deblurring results of three experiments above, the PSF estimation of these two latter experiments seems better than that of the first one. It is mainly due to the shrinkage and normalisation of PSF after each iterative step. To do this adjustment, it must be supposed that the support size of PSF was known or roughly estimated in these two latter cases.

Besides, in all these three experiments, the least improved image is always the "Barbara" image. This fact may be explained by the image pattern. Since "Barbara" image is full of small strips and checked patterns, the alternative strips of different intensities are mixed up during the blurring process. Hence, it is difficult to relocate the sharp transitions between them.

5.5. The robustness of algorithm with the initial parameters

Table 5.4. Experiments with different initial parameters and confidence coefficients.

Group	Experiment	β_n^0	β_{bl}^0	β_{im}^0	γ_{β_n}	$\gamma_{\beta_{bl}}$	$\gamma_{\beta_{im}}$
1	2	0.22	10^{-6}	93.6	0.001	0.001	0.001
	3	15.7	$10^{-7}/2.15$	206	0.001	0.001	0.001
	4	1	1	1	0.001	0.001	0.001
2	5	0.22	10^{-6}	93.6	0.5	0.5	0.5
	6	15.7	$10^{-7}/2.15$	206	0.5	0.5	0.5
	7	1	1	1	0.5	0.5	0.5
3	8	0.22	10^{-6}	1	0.5	0.5	0.001
	9	1	10^{-6}	93.6	0.001	0.5	0.5
	10	0.22	1	93.6	0.5	0.001	0.5
4	11	0.22	1	1	0.5	0.001	0.001
	12	1	10^{-6}	1	0.001	0.5	0.001
	13	1	1	93.6	0.001	0.001	0.5
	14	0.22	10^{-6}	93.6	0.001	0.001	0.5
	15	0.22	10^{-6}	93.6	0.001	0.5	0.001
	16	0.22	10^{-6}	93.6	0.5	0.001	0.001

To examine the effect of the initial parameters on the restored results, a series of experiments were carried out. Table 5.4 shows the initial parameters and their confidence coefficients used in these experiments, which are divided into four groups following the confidence coefficients $\gamma_{\beta_n}, \gamma_{\beta_{im}}$, and $\gamma_{\beta_{bl}}$:

- In the first group, the experiments have very small confidence coefficient (0.001).
- In the second group, these coefficients have higher values.
- In the third group, one among these three confidence coefficients is close to zero while the other two have higher values.
- In the last group, two among these three coefficients are close to zero.

In each group of experiments, the initial parameters β_n^0, β_{im}^0, and β_{bl}^0 are assigned by three basic parameter sets and their permutations. Two among these basic parameter sets are initial parameter sets used in paper [30]. The other is a conventional parameter set where all the initial parameters are equal to one.

From the previous sections, we choose the best cases to carry out the experiments in this part of the experiments; the LF-SAR algorithm is used in these experiments to restore an observation blurred by a Gaussian-shape PSF.

Table 5.5 shows that the restoration results are sensitive to the initial parameters. For instance, the conventional parameter set $\beta_n^0 = \beta_{im}^0 = \beta_{bl}^0 = 1$ yields the worse results than the other initial parameter sets no matter what value of the confidence coefficients is used. In addition, the restoration result also depends on the confidence coefficients in some of the experiments. With the same initial parameter set, the experiments 2, 14, 15, and 16 produce different results because they are carried out with different levels of confidence on the initial parameters.

Table 5.5. ISNR and ISNR_h [dB] of the image and PSF estimated from a blurred noisy observation (Gaussian-shape PSF with variance 9, $\beta_n=10^6$) by LF-SAR algorithm with different initial parameters and confidence coefficients shown in Table 5.4.

Group	Experiment	LF-SAR& Gaussian-shape PSF	
		ISNR	ISNR_h
1	2	2.72	8.69
	3	2.45	8.53
	4	1.53	3.32
2	5	1.70	4.04
	6	2.03	7.30
	7	1.53	2.86
3	8	1.70	4.04
	9	1.80	6.35
	10	1.53	2.72
4	11	1.53	2.72
	12	1.80	6.79
	13	1.53	3.32
	14	3.18	8.33
	15	1.80	6.79
	16	2.66	7.79

5.6. The noise effect

It is known that the deblurring problem is an ill-posed inverse problem which is very sensitive to noise. In the previous sections, we only study the blurred observation contaminated by a small level of noise. Hence, in this section, experiments are carried out with higher levels of contaminated noise to investigate how the noise affects the deblurring results.

Table 5.6 shows the deblurring results of blurred noisy images of Lena and Cameraman images with the Gaussian distributed noise with variances 10^{-4}, 10^{-2}, 1, and 10^2. There are two types of blurring functions which are involved in these experiments, the Gaussian-shape PSF and the horizontally uniform PSF. The LF-SAR algorithm is used when the observation is blurred by the Gaussian-shape PSF. The LF-G algorithm is used when the observation is blurred by a horizontally uniform blurring function. In most of these experiments, the better estimations of image and PSF are produced when the contaminated noise is lower. This result is predictable and can be easily understood. In the case of uniform PSF, the estimate of image is still consistent with this rule "lower noise, better estimation", but the rule seems inapplicable in estimating the blurring function. This exception is resulted from shrinking the kernel blurring function and normalising it at each iterative step in the LF-G algorithm.

Table 5.6. ISNR and ISNR_h [dB] of the image and PSF estimated by LF-SAR algorithm (Gaussian-shape PSF with variance 9) and LF-G algorithm (horizontally uniform PSF with size support 9×9) at different levels of noise.

Image	Variance	LF-SAR& Gaussian-shape PSF		LF-G & Uniform PSF	
		ISNR	ISNR_h	ISNR	ISNR_h
Lena	10^{-4}	3.16	8.33	3.41	10.60
	10^{-2}	2.81	8.31	3.41	10.60
	1	2.13	8.21	2.98	11.26
	10^{2}	2.09	4.68	1.92	13.63
cameraman	10^{-4}	2.35	7.69	3.35	11.96
	10^{-2}	2.22	7.88	3.35	11.97
	1	1.68	8.85	3.19	11.38
	10^{2}	1.28	6.08	1.18	14.31

5.7. PSF estimation using cross validation method

As stated in the first chapter, the thesis only considers the spatially invariant blurring function. In this case, the kernel blurring matrices are the same at every pixel. Hence, if the image is divided into smaller sub-images, the kernel blurring matrix can be estimated from the sub-image by using the proposed algorithms. The kernel blurring matrix estimated from the sub-image may be slightly different from the one estimated from the full image because of the bound effect. When the observed image is noisy, the estimation of blurring matrix will have an error even if it is estimated from the full image or sub-images. By dividing the image into sub-images, we can have some blurring estimates which are affected by noise in different ways. Taking the weighted average of these blurring estimates gives us an opportunity to get a better blurring estimate.

In addition, dividing the image into sub-images also helps to reduce the computational time of the proposed algorithms. For instance, an observation g which is lexicographically re-ordered in the vector form $1 \times N$ will require the inverse calculation of covariance matrix $N \times N$ in our deblurring algorithm. The order of this calculation is $O(N^4)$. If the observed image is divided into m sub-images, the similar process requires $O(N^4/m^4)$ computational time for each sub-image. Hence, for m sub-images, the order of inverse calculation is $O(N^4/m^3)$. It means that the computational time should decrease when the number of division increases.

Table 5.7 shows the error of blurring estimation in percentage when each dimension of 2D image being divided into one to eight equal intervals. The experiments in this section use the LF-SAR algorithm and are carried out with the Gaussian-shape PSF. In general, the error of PSF estimation decreases when the number of sub-images increases. It is noted that the PSF estimate shown in Table 5.7 is the average of PSFs estimated from sub-images. When the number of division increases, although the PSF estimate shown in Table 5.7 is smaller, the error of each PSF estimate from sub-images does not decrease. The experimental result also confirms that the computational time of algorithm reduces when the number of division increases from one to eight.

Table 5.7. Errors of PSF estimation when the image is divided into sub-images.

No. sub-images	Error of PSF estimation (%)
1	6.56
4	7.94
9	5.43
16	5.52
25	3.87
36	3.56
49	3.54
64	3.68

Unfortunately, by applying the cross validation method, even though the PSF estimation is better by taking averages, the images estimated from sub-images are totally different and cannot be improved by taking averages. In addition, the error of image estimation from sub-images is even worse than that of restored result using whole image. This may be due to the bound effect which is an obvious consequence of image division. Hence, the PSF estimate using the cross validation method could only be used as a reference for the other deblurring algorithms which require knowledge about the blurring function.

5.8. Concluding remarks

From the experimental results presented above, we can derive the following notable remarks.

Firstly, the performance of the proposed algorithm was sensitive to the initial parameters.

Secondly, the result of blurring estimation might be improved by applying the cross validation in spatially invariant blurring problem. The division of observed image into sub-images also helps to accelerate the computational speed of the proposed algorithms. However, the deblurring result of the original image which was estimated from sub-images was not good due to the bound effect.

Last but not least, the Gaussian distribution and SAR model were used in our algorithms, as well as in most of the existing blind deblurring studies in stochastic approach, to model the PSF. As a result, the algorithms performed well only when the blurring phenomena were caused by a smooth-shape PSF because these models are not capable of dealing with the PSF consisting of sharp transition. When the PSF had sharp transitions, the experimental result of proposed algorithms were improved by shrinking and normalising the PSF estimate.

Chapter 6

Blind Deblurring Algorithms

Using Variational Bayesian Approach

6.1. Introduction

In this section, we compare four different algorithms, all of them using the Variational Bayesian approach:

- One uses the Simulated Auto-Regression (SAR) model as the prior information of the original image. This algorithm, called SAR algorithm, is introduced in [30] by Molina *et al*.
- The other one is the TV algorithm which uses the Total Variation model as the prior information of the original image. This model is used in [16] to estimate the original image from its blurred observation with a known blurring function. We extend the work to blind deconvolution.
- The next one is LF-SAR algorithm which uses our proposed image model in [87], called LiFeAIM.
- The last one, which is proposed in [62], is similar to our TV algorithm. The only difference between these two algorithms is that the algorithm in [62] calculates the covariance matrix of image by the conjugate gradient method while the TV algorithm approximates it by a simpler matrix. The deblurring result of the algorithm in [62], denoted as TV_CG in this thesis, will be quoted to compare with the results of the three above algorithms which are coded by

ourselves. The algorithm proposed in [30] is implemented by ourselves instead of quoting the results shown in [30] because the authors admited that there were some mistyping errors in [30]. Moreover, while comparing a new algorithm with the algorithm in [30], the authors of [61] also showed the results of algorithm in [30] by implementing their own code which were very different from the results shown in [30] and similar to those implemented by us.

It is important to note that the only difference in ideas amongst the four compared algorithms is the image model. In all these algorithms, the Gamma distribution and the SAR model are used as the prior information of the model parameters and the blurring function, respectively. To guarantee the same experimental condition for all algorithms, the LF-G algorithm proposed in Chapter 4 is not used in the experiments in this chapter because it uses Gaussian distribution for the blurring model. In this chapter, two algorithms using TV model are represented in section 6.3. The SAR model of SAR algorithm is briefly described in section 6.2. More details of SAR algorithm can be found in [30].

6.2. Modeling image by Simulated Auto-Regression (SAR) model

The Auto-Regression models are used by many researchers in image restoration [20, 2, 23]. The Simultaneous Auto-Regression (SAR) model is one among them. This model can be represented by the following equation:

$$p\left(f \middle| \beta_{im}\right) = const \times \beta_{im}{}^{N/2} \exp\left[-\frac{\beta_{im}}{2} f^T C^T C f\right] \qquad (6.1)$$

in which, N is the size of lexicographically re-ordered image f;

β_{im} is the parameter of image model;

C is the Laplacian operator.

The SAR model is used in the blind deblurring algorithm of Molina *et al.* [30]. In this algorithm, the SAR model is also used as the prior information of blurring function, while the model parameters are assumed to be Gamma distributed. The deblurring result of this algorithm, called SAR algorithm, is used to compare with our restoration result in the thesis. However, the results of SAR algorithm, shown later in section 6.4 of this thesis, differ from those shown in [30] because they are coded by us. The difference may be due to the different orders of Fourier transforms which are used in their programme and ours. In the derived equations of SAR algorithm, the two-dimension image and blurring function are lexicographically re-ordered into the vector form. However, in their programme, the image and blurring function are kept in the original form, the matrix form. Thus, the 2D Fourier transform is used. In our programme, the image and blurring function are lexicographically re-ordered into the vector form. Then, we use the 1D Fourier transform whose order is the product of two dimension of the original image.

6.3. Modeling image by Total Variation model

6.3.1. Total Variation model

In the Total Variation (TV) model [88], the variance of image at each pixel is calculated by the horizontal and vertical first order difference. The TV model can be approximated by the following equation:

$$p(f|\beta_{im}) = const \times \beta_{im}{}^{N/2} \exp[-\beta_{im} TV(f)] \tag{6.2}$$

where, $TV(f) = \sum_i \sqrt{\left(f_i - f_i^h\right)^2 + \left(f_i - f_i^v\right)^2}$ (6.3)

f_i^h and f_i^v are the intensities of the nearest left and above neighbours of pixel i, respectively.

N is the size of the lexicographically re-ordered image f;

β_{im} is the parameter of image model.

6.3.2. Blind deblurring algorithms using TV model

We now introduce blind deblurring algorithms using TV model by applying the Variational Bayesian (VB) approach which is described in section 4.2. The prior information of blurring function and that of model parameters are the same as those described in section 4.3.3 and 4.3.4. However, when we apply exactly the same process represented in Chapter 4, it leads to an optimisation problem of the posterior probability $p(f, h, \Theta|g)$:

$$p(f, h, \Theta|g) = \frac{p(g|f, h, \beta_n)p(f|\beta_{im})p(h|\beta_{bl})p(\beta_n)p(\beta_{im})p(\beta_{bl})}{p(g)} \tag{6.4}$$

where $\Theta = \{\beta_{im}, \beta_{bl}, \beta_n\}$;

$p(g|f, h, \beta_n)$ is the Gaussian distribution of the observation model;

$p(h|\beta_{bl})$ is the SAR model of blurring function;

$p(\beta_n)$, $p(\beta_{im})$, and $p(\beta_{bl})$ are Gamma distributions of model parameters.

An obstacle would occur in solving this optimisation problem because $TV(f)$ term in the TV model $p(f|\beta_{im})$ is not a quadratic form of f. To make the optimisation problem easier to solve, $TV(f)$ term in this model can be bounded as proposed in [88] by a quadratic form of f:

$$TV(f) \le \sum_i \frac{\left(f_i - f_i^h\right)^2 + \left(f_i - f_i^v\right)^2 + w_i^2}{2w_i} \tag{6.5}$$

where w_i is an arbitrary positive scalar. The equality of this inequality happens when we have:

$$w_i = \sqrt{\left(f_i - f_i^h\right)^2 + \left(f_i - f_i^v\right)^2} \tag{6.6}$$

Following this idea, the prior information of image $p\left(f|\beta_{im}\right)$ has a lower bound $\phi\left(f|\beta_{im}\right)$ as shown in eq. (6.7):

$$\begin{aligned} p\left(f|\beta_{im}\right) &\ge const \times \beta_{im}{}^{N/2} \exp\left[-\beta_{im}\sum_i \frac{\left(f_i - f_i^h\right)^2 + \left(f_i - f_i^v\right)^2 + w_i^2}{2w_i}\right] \\ &= \phi\left(f|\beta_{im}\right) \end{aligned} \tag{6.7}$$

Hence, we define a distribution function $\Phi\left(f, h, \Theta|g\right)$ which is the lower bound of $p\left(f, h, \Theta|g\right)$ as follows:

$$\begin{aligned} \Phi\left(f, h, \Theta|g\right) &= \frac{p\left(g|f, h, \beta_n\right)\phi\left(f|\beta_{im}\right)p\left(h|\beta_{bl}\right)p\left(\beta_n\right)p\left(\beta_{im}\right)p\left(\beta_{bl}\right)}{p(g)} \\ &\le p\left(f, h, \Theta|g\right) \end{aligned} \tag{6.8}$$

The distribution function $\Phi\left(f, h, \Theta|g\right)$ is approximated by $q\left(f, h, \beta_{im}, \beta_{bl}, \beta_n\right)$ using the Kullback-Leibler divergence:

$$\begin{aligned} D_{KL}\left(q\|\Phi\right) &= \int_{f,h,\Theta} q\left(f, h, \Theta\right)\log\left[\frac{q\left(f, h, \Theta\right)}{\Phi\left(f, h, \Theta|g\right)}\right]dfdhd\Theta \\ &= \int_{f,h,\Theta} q\left(f, h, \Theta\right)\log\left[\frac{q\left(f, h, \Theta\right)p(g)}{p\left(g|f, h, \beta_n\right)\phi\left(f|\beta_{im}\right)p\left(h|\beta_{bl}\right)p\left(\beta_n\right)p\left(\beta_{im}\right)p\left(\beta_{bl}\right)}\right]dfdhd\Theta \end{aligned} \tag{6.9}$$

where $q\left(f, h, \Theta\right) = q(f)q(h)q\left(\beta_{im}\right)q\left(\beta_{bl}\right)q\left(\beta_n\right)$;

$q(f)$ and $q(h)$ are assumed to be Gaussian distributions;

$q\left(\beta_{im}\right)$, $q\left(\beta_{bl}\right)$ and $q\left(\beta_n\right)$ are assumed to be Gamma distributions.

The optimum solution of the approximate distributions $q(f), q(h), q(\beta_{im})$, $q(\beta_{bl})$ and $q(\beta_n)$ are calculated following eq. (4.5). When the lower bound $\Phi(f, h, \Theta|g)$ is approximated by $q(f, h, \Theta)$, the distributions of blurring function and parameters, $p(h|\beta_{bl})$, $p(\beta_n)$, $p(\beta_{im})$, and $p(\beta_{bl})$, are approximated by $q(h)$, $q(\beta_{im})$, $q(\beta_{bl})$, and $q(\beta_n)$, respectively. Meanwhile, $\phi(f|\beta_{im})$ is approximated by $q(f)$. Since $\phi(f|\beta_{im})$ is the lower bound of the image distribution $p(f|\beta_{im})$, the image distribution can be approximated by $q(f)$ if the equality of eq. (6.8) occurs. The condition under which this equality occurs is represented in eq. (6.6), $w_i = \sqrt{(f_i - f_i^h)^2 + (f_i - f_i^v)^2}$.

Now, we estimate the original image, the blurring function and the model parameters following eq. (6.6) and the optimum solution shown in eq. (4.5).

** *Image estimation:*

Following eq. (4.5), the optimum approximate distribution of the original image is given below:

$$q(f) = const \times \exp\left\{E\left[\log\phi(f|\beta_{im}) + \log p(g|f, h, \beta_{im}, \beta_{bl}, \beta_n)\right]_{q(h)q(\beta_{im})q(\beta_{bl})q(\beta_n)}\right\}$$

$$\log q(f) = const + E\left[\log\phi(f|\beta_{im}) + \log p(g|f, h, \beta_n)\right]_{q(h)q(\beta_{im})q(\beta_n)}$$

$$= const - E[\beta_{im}]\sum_i \frac{(f_i - f_i^h)^2 + (f_i - f_i^v)^2 + w_i^2}{2w_i} - \frac{E[\beta_n]}{2} E\left[\|g - Hf\|^2\right]_{q(h)} \tag{6.10}$$

The summation $\sum$ in the above equation can be rewritten in the matrix form:

$$\sum_i \frac{(f_i - f_i^h)^2 + (f_i - f_i^v)^2 + w_i^2}{2w_i} = W\Delta^{v^T}\Delta^v + W\Delta^{h^T}\Delta^h$$

where $W = diag(w_i^{-1})_{i=1..N}$;

Δ^v and Δ^h denote matrices such that $\Delta^v f = \left[f_i - f_i^v\right]_{i=1..N}$ and $\Delta^h f = \left[f_i - f_i^h\right]_{i=1..N}$.

Hence,

$$\log q(f) = const - E[\beta_{im}] W \Delta^{v^T} \Delta^v - E[\beta_{im}] W \Delta^{h^T} \Delta^h - \frac{E[\beta_n]}{2} E\left[\|g - Hf\|^2\right]_{q(h)}$$

Since $q(f)$ is a Gaussian distribution, the expectation and covariance matrix of the image are determined by eq. (6.11):

$$E[f] = \left\{ f \left| \frac{\partial \log[q(f)]}{\partial f} = 0 \right. \right\} \text{, and } \operatorname{cov}(f) = -\left(\frac{\partial^2 \log[q(f)]}{\partial f^2} \right)^{-1} \tag{6.11}$$

$$\frac{\partial \log[q(f)]}{\partial f} = -E[\beta_{im}] W \Delta^{v^T} \Delta^v f - E[\beta_{im}] W \Delta^{h^T} \Delta^h f - E[\beta_n] E\left[H^T H\right] f + E[\beta_n] E\left[H^T\right] g$$

Hence,

$$\begin{aligned} \operatorname{cov}(f) &= -\left(\frac{\partial^2 \log[q(f)]}{\partial f^2} \right)^{-1} \\ &= \left(E[\beta_{im}] W \Delta^{v^T} \Delta^v f + E[\beta_{im}] W \Delta^{h^T} \Delta^h f + E[\beta_n] E\left[H^T H\right] \right)^{-1} \end{aligned} \tag{6.12}$$

And,

$$E[f] = \left(E[\beta_{im}] W \Delta^{v^T} \Delta^v f + E[\beta_{im}] W \Delta^{h^T} \Delta^h f + E[\beta_n] E\left[H^T H\right] \right)^{-1} E[\beta_n] E\left[H^T\right] g$$

$$E[f] = \operatorname{cov}(f) E[\beta_n] E\left[H^T\right] g \tag{6.13}$$

** <u>*Blurring estimation:*</u>

Similarly, the approximate distribution of blurring function has the expectation and covariance matrix shown in the following equations:

$$\operatorname{cov}(h) = \left(E[\beta_n] E[F]^T E[F] + E[\beta_n] \operatorname{cov}(F) + E[\beta_{bl}] C^T C \right)^{-1} \tag{6.14}$$

$$E[h] = E[\beta_{bl}] \operatorname{cov}(h) E[F]^T g \tag{6.15}$$

in which, F is the left-wise circulant matrix whose first row is f^T.

** <u>Parameter estimation:</u>

The optimum approximate distribution of parameters has the following form:

$$q(\theta_i) = const \times p(\theta_i) \exp\left\{ E\left[\log p(g|f,h,\Theta) p(f|\Theta) p(h|\Theta)\right]_{q(h)q(f)q(\Theta_i)} \right\}$$

where $\theta_i \in \{\beta_{im}, \beta_{bl}, \beta_n\}$

Comparing the coefficients of θ_i in both sides of the above equation, we obtain the following results:

$$a_{im} = a_{im}^0 + \frac{N}{2},\ b_{im} = b_{im}^0 + \frac{1}{2T_k} E\left[f^T B^T Bf\right] \tag{6.16}$$

$$a_{bl} = a_{bl}^0 + \frac{M}{2},\ b_{bl} = b_{bl}^0 + \frac{1}{2} E\left[h^T C^T Ch\right] \tag{6.17}$$

$$a_n = a_n^0 + \frac{N}{2},\ b_n = b_n^0 + \frac{1}{2} E\left[\|g - Hf\|^2\right]_{q(f)q(h)} \tag{6.18}$$

The expectation of model parameters is given by the following equation:

$$E[\beta_x] = \frac{a_x}{b_x},\ x \in \{im, bl, n\}$$

By applying the theorems in Chapter 4 into the above results, the iterative deblurring algorithm, denoted TV algorithm, is introduced with four following steps:

** <u>TV algorithm:</u>

- Step 1: estimate the covariance and the expectation of f

$$\begin{aligned} \mathrm{cov}(f) &= \left(\beta_n E[H]^T E[H] + \beta_n N \,\mathrm{cov}(h) + \beta_{im} W \Delta^{v^T} \Delta^v + \beta_{im} W \Delta^{h^T} \Delta^h\right)^{-1} \\ E[f] &= \beta_n \,\mathrm{cov}(f) E[H]^T g \end{aligned} \tag{6.19}$$

- Step 2: determine the condition under which the posterior probability reaches its lower bound in eq. (6.8)

$$w_i = \sqrt{\left(f_i - f_i^{h}\right)^2 + \left(f_i - f_i^{v}\right)}$$

$$W = diag(w_i^{-1})_{i=1..N} \tag{6.20}$$

- Step 3: estimate the covariance and the expectation of h

$$\mathrm{cov}(h) = \left(\beta_n E[F]^T E[F] + \beta_n N\,\mathrm{cov}(f) + \beta_{bl} C^T C\right)^{-1}$$

$$E[h] = \beta_{bl}\,\mathrm{cov}(h) E[F]^T g \tag{6.21}$$

- Step 4: estimate the expectation of the parameters β_n, β_{im}, and β_{bl}

$$a_{im} = a_{im}^0 + \frac{N}{2},\; b_{im} = b_{im}^0 + \sum_i \sqrt{w_i}$$

$$a_{bl} = a_{bl}^0 + \frac{M}{2},\; b_{bl} = b_{bl}^0 + \frac{1}{2} E\left[h^T C^T C h\right]$$

$$a_n = a_n^0 + \frac{N}{2},\; b_n = b_n^0 + \frac{1}{2} E\left[\|g - Hf\|^2\right]$$

$$E[\beta_{im}] = \frac{a_{im}}{b_{im}}, E[\beta_{bl}] = \frac{a_{bl}}{b_{bl}}, E[\beta_n] = \frac{a_n}{b_n} \tag{6.22}$$

To reduce the computational effort of this algorithm, the covariance matrices $\mathrm{cov}(f)$ and $\mathrm{cov}(h)$ are assumed to be circulant. Hence, to ensure this circulant assumption, the matrix $W(w)$ in our TV algorithm is approximated by:

$$\overline{W(w)} = \frac{1}{N} trace(W(w)) I \tag{20}$$

Meanwhile, the TV_CG algorithm in [62] calculates the matrix $W(w)$ in eq. (6.20) by the conjugate gradient method. The terminate criterion of TV_CG algorithm is that the gradient descent is less than 10^{-5}.

6.4. Comparison among blind deblurring algorithms using Variational Bayesian approach

This section compares four blind deblurring algorithms using Variational Bayesian approach and different image models. They are:

(i) SAR algorithm which uses SAR model as the prior information of the image;

(ii) TV algorithm which uses Total Variation model as the prior information of the image;

(iii) LF-SAR algorithm which uses our proposed image model in Chapter 3 as the prior information of the image;

(iv) TV-CG algorithm [62] which also uses Total Variation model as the prior information of the image.

The restored images of these algorithms are compared in term of ISNR (Improved Signal Noise Ratio) index:

$$ISNR = 10\log\frac{\|f-g\|^2}{\|f-\hat{f}\|^2}$$

where $\hat{f}$ is an estimation of the original image f

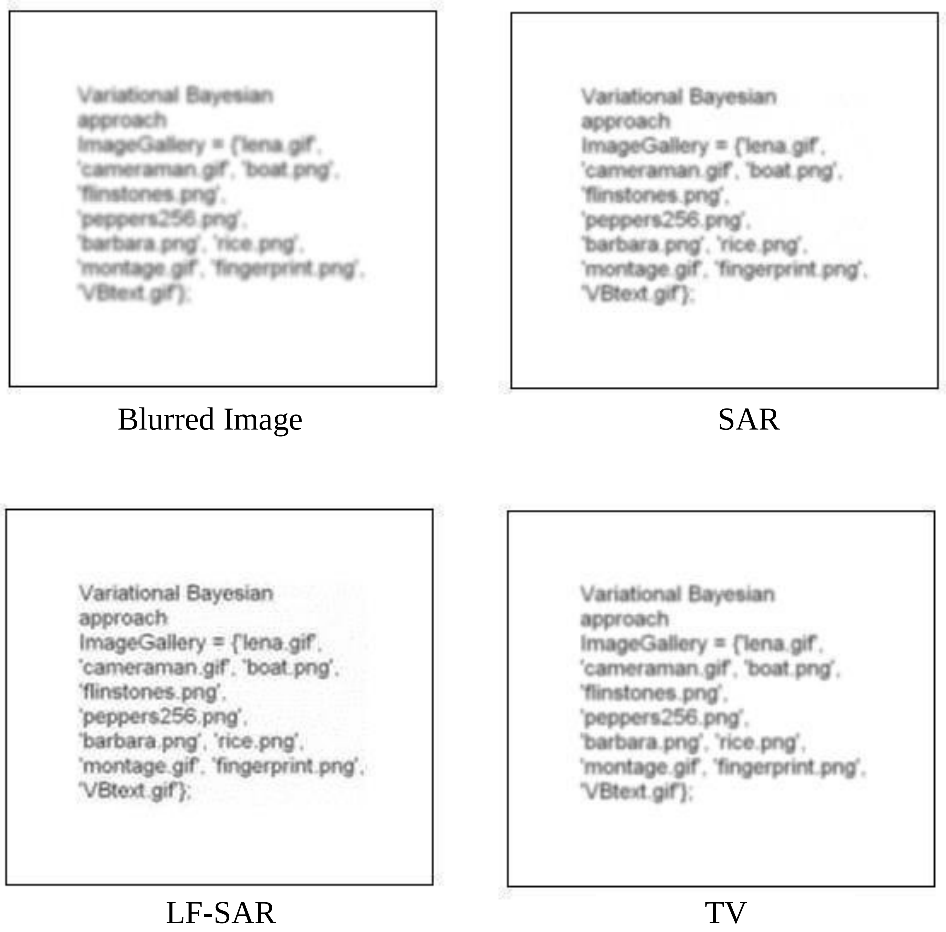


Figure 6-1. The blurred noisy Text image and its restored results by SAR algorithm (ISNR=0.48dB), TV (ISNR=0.78dB), and LF-SAR (ISNR=1.37dB).

Our experiments use three images:

- The Text image, created by ourselves, contains some words in order to compare the visual appearance among the restored images easily.
- The second one is the image of Lena which is used in a series of experiments to investigate the effect of the initial parameters on the restoration result.
- The last one is a synthesized image, Shepp-Logan phantom image, which is generated by Matlab.

In all these experiments, the blurring function is a Gaussian function with variance 9. The support size of blurring function is assumed to be equal to the image size. In most of the experiments, the variance of contaminated noise is $\beta_n^{-1} = 0.23$, or otherwise stated.

Four sets of experiments are carried out in this chapter. The first set of experiments uses an image containing text (Figure 6-1) to compare the visual appearance of restored image besides comparing the ISNR index. The second set of experiments is to investigate the effect of initial parameters and confidence coefficients on the restoration result. We also examine how the noise affects the deblurring result by carrying out experiments with different levels of noise contamination in our third set of experiments. In the first three sets of experiments, only three among the four algorithms mentioned above are compared. This is because in the first three sets of experiments there is no similar experiment of TV-CG algorithm reported in [62] for comparison with those of the other three algorithms. All these four algorithms are compared in our last set of experiments in which the confidence coefficients are zero.

For our first experiment, Figure 6-1 shows the restored results of the Text image. The ISNR index shows that our deblurring result (ISNR=1.37dB) is better than that of the TV algorithm (ISNR=0.78dB) and SAR algorithm (ISNR=0.48dB). Ours also has the best visual appearance among the three restored images.

The second set of experiments is to examine the effect of the initial model parameters on the restoration result. A series of experiments are carried out with initial parameters and their confidence coefficients are given in Table 5.4 in Chapter 5. These initial parameter sets consist of three specific parameter sets and their permutations. The confidence coefficients of these parameters are ranged from low to high (0.001 to 0.5).

Table 6.1. The ISNR[dB] of the restored result of SAR, TV, LF-SAR, and LF-SAR2 with different initial parameters shown in Table 5.4.

Group	Experiment	SAR	TV	LF-SAR	LF-SAR2
1	2	1.36	2.19	2.32	2.31
	3	1.36	2.18	2.30	2.30
	4	0.61	1.24	1.30	1.54
2	5	1.15	0.97	1.61	2.36
	6	0.90	2.03	2.03	2.42
	7	0.55	1.21	1.33	1.32
3	8	1.01	0.58	1.60	2.33
	9	1.04	1.99	1.80	2.42
	10	1.02	0.82	1.33	1.52
4	11	0.83	0.40	1.30	1.54
	12	0.87	1.99	1.73	2.32
	13	0.86	1.25	1.30	1.51
	14	1.38	2.19	2.41	2.34
	15	0.87	1.72	1.98	2.32
	16	1.43	1.02	2.23	2.32

The corresponding deblurring results of these experiments are shown in Table 6.1 (Please refer to Table 5.4 for the parameters of the experiments). From Table 6.1, by comparing between group 1 and 2, we find that with the same initial parameters, the better restored images are produced if the confidence coefficients are close to zero. Meanwhile, groups 3 and 4 show how effective each algorithm are in estimating each parameter. The best result in these experiments is that of experiment 14 where the initial parameters are $\beta_n^0 = 0.22, \beta_{im}^0 = 93.6,\ \beta_{bl}^0 = 10^{-6}$ and the confidence coefficients $\gamma_{\beta_n} = 0.001, \gamma_{\beta_{im}} = 0.5, \gamma_{\beta_{bl}} = 0.001$. It is found that in experiment 14 of LF-SAR algorithm, the initial values are close to the final estimated parameters. Hence, we introduce another algorithm, called LF-SAR2, which is the same as our

LF-SAR but with the initial parameters resulting from LF-SAR. The confidence coefficients in LF-SAR2 algorithm are also fixed to those of experiment 14, $\gamma_{\beta_n} = 0.001, \gamma_{\beta_{im}} = 0.5,$ and $\gamma_{\beta_{bl}} = 0.001$. The restoration result of LF-SAR2 is found to be very good and it is the best among most of the experiments. Another interesting finding is that the results of our adaptive image models, TV algorithm and LF-SAR algorithm, are better than those of SAR algorithm in most experiments.

TV LF-SAR Blurred image SAR

Figure 6-2. The blurred noisy Lena images and their restored results by SAR, TV, and LF-SAR with low level (first row) and high level (second row) noise.

In the third set of experiments, we investigate the robustness of four deblurring algorithms above with different levels of noise. The images in Figure 6-2 show the restored images of two blurred and noisy images. In the first row of Figure 6-2, the contaminated noise is low, β_n^{-1}= 0.23. In the second row, the contaminated noise is high, β_n^{-1}= 16. These experiments use the initial value set of experiment 16 in Table 5.4. As stated above, the results of TV and LF-SAR algorithms are better than those of SAR algorithm at the low level noise. However, the TV algorithm is not as good at the moderate and high level of noise. LF-SAR is still better than SAR algorithm at the moderate level of noise (see Table 6.2). The experimental results show that our deblurring algorithm outperforms the other two algorithms using the Variational Bayesian approach, TV and SAR, at the low and moderate levels of noise (BSNR>20dB). However, if the level of contaminated noise is high, further investigation of choosing the initial parameters is needed to improve the performance of our proposed algorithm.

Table 6.2. The ISNR[dB] of the restored result of SAR, TV, LF-SAR, and LF-SAR2 with different levels of noise.

β_n^{-1}	SAR	TV	LF-SAR
0.23	1.39	2.19	2.43
1.00	1.12	-0.82	2.13
4.00	1.09	-13.58	1.55
16.00	1.37	-23.08	1.37

Although the proposed LF-SAR algorithm is better than the TV and SAR algorithms in term of ISNR, it requires longer computational time. Since LF-SAR does not assume that the covariance matrix of the image model was circulant, it must be implemented in the spatial domain instead of the Fourier domain as most of the studies did. Hence, it must deal with the inverse problem of a huge matrix. The calculation of the inverse matrix takes several seconds each time although we did some improvements to reduce the size of the matrix by a few thousand times.

Table 6.3. The ISNR[dB] of the restored result of SAR, TV, TV_CG, and LF-SAR without confidence in the initial parameters.

Image	β_n^{-1}	SAR	TV	TV_CG	LF-SAR
Lena	0.16(40dB)	1.56	1.78	2.53	2.23
	16(20dB)	1.26	-16.72	2.62	1.30
Shepp-Logan	0.18(40dB)	1.62	2.11	3.07	2.10
	18(20dB)	1.57	-19.66	2.47	1.83

The last set of experiments in this thesis is an ideal case where all confidence coefficients γ_x are set to zero, corresponding to $a_x^0 = 0, b_x^0 = 0$ with $x \in \{\beta_{im}, \beta_{bl}, \beta_n\}$. In this case, it is unnecessary to choose the initial parameters $\beta_n^0, \beta_{im}^0,$ and β_{bl}^0 since they will not affect the estimation results. The experiments are carried out at two levels of noise (BSNR = 40dB and BSNR = 20dB) with both synthesized and real images, the Shepp-Logan phantom and Lena images. Table 6.3 shows that the results of LF-SAR are still better than those of SAR and TV algorithms. Besides, it also shows that the results of TV_CG are the best among those of four compared algorithms. Although TV_CG is better than LF-SAR in term of ISNR, TV_CG algorithm requires intensive computation as it uses the conjugate gradient method. LF-SAR algorithm is about ten times faster than TV_CG algorithm.

Chapter 7
Conclusions and Future Works

7.1. Conclusions

In this thesis, the blind deblurring algorithms using a new adaptive image model has been developed. The proposed image model was called LiFeAIM, which stands for Line Field based Adaptive Image Model. The model was represented by a probability distribution whose standard deviations were different at each pixel. The standard deviations were calculated from the new line field whose distribution was varying through iterations. The new line field gave the proposed image model the ability to distinguish edge pixels from noisy pixels. Since the deblurring problem is very sensitive to noise, using this image model to construct deblurring algorithms led to interesting restored results.

As the deblurring problem is a complex problem that embraced the denoising problem, LiFeAIM was also used to construct a denoising algorithm to examine its performance in denoising. Our proposed denoising algorithm used LiFeAIM and the maximum *à posteriori* approach. By comparing its result with that of the denoising algorithm using the original line field, it was demonstrated that our new line field helped to construct an efficient denoising algorithm. Moreover, we showed that the proposed algorithm was competent in comparison with the existent denoising algorithm using the wavelet transform or the hidden Markov model, such as BayesShrink, VisuShrink, and HMTs. This good performance resulted from the high capacity of our model for distinguishing between the noisy pixels and the edge pixels of the images. Many experiments were carried out to investigate the speed of convergence of the proposed denoising algorithm. It was proven that one of the convergent conditions of our denoising algorithm was:

$$T(k) \geq \frac{c}{\log(k+1)}$$

Hence, the constant c in the lower bound term of the temperature parameter *T(k)* could be used to control the convergence speed of our denoising algorithm. We have determined the relationship between the "best" value of constant c and the standard deviation of the contaminated noise. This relationship was used to generate an approximate function which can be used later to select the appropriate parameter *T(k)* corresponding to any level of the noise. Another interesting finding was that the speed of convergence of the algorithm also depended on the smoothness of images. We also discussed how to choose an appropriate value of the temperature parameter *T(k)* to control the convergence of algorithm faster.

After examining the quality of LiFeAIM in denoising, we dealt with the blind deblurring problem by using LiFeAIM and the Variational Bayesian method. To show the performance of the proposed deblurring algorithms in a wide variety of blurring types and image patterns, many experiments using these algorithms were carried out with three different types of blurring functions and several images. Some experiments were also carried out with various levels of contaminated noise to examine the sensitivity of proposed algorithms to noise. One of our significant contributions was that the covariance matrix of our image model was not assumed to be circulant. This assumption is unrealistic even though it was often used in other deblurring algorithms. Nevertheless, because it helped to solve the problem faster in the Fourier domain in which the problem of matrix inversion became the simple problem of scalar inversion. Our other contribution was that we have developed and proven theorems to accelerate the computational speed of our deblurring algorithms.

The proposed deblurring algorithms using LiFeAIM was compared with those using TV and SAR models. At the low level of noise, the results of these algorithms showed that the adaptive image models, including the TV and proposed models, were more effective than the SAR model in the deblurring problem. Moreover, our algorithm produced the best result among compared algorithms when the level of contaminated noise was low or moderate. We also found that the performance of these algorithms highly dependent on the initial parameters.

Although the algorithms using adaptive image models outperformed the algorithm using SAR model in deblurring, the algorithms using adaptive image models developed in this thesis required longer computational time than the algorithm using SAR model does. Since the algorithms using adaptive image models did not assume that the covariance matrix of the image model to be circulant, the inversion of the covariance matrix must be implemented in the spatial domain instead of in the Fourier domain. Hence, these algorithms needed time to deal with the inverse problem of a huge matrix. In the proposed algorithms, the calculation of the inverse matrix took several seconds each time although we did some improvements to significantly reduce the computational complexity.

To reduce the computational time, we also tried to divide the image into several sub-images. By dividing the image into smaller ones, it not only helped to reduce the computational time of our deblurring algorithm but also increased the accuracy of blurring estimation. Since the kernel blurring function was invariant in our studies, the kernel blurring functions for sub-images were the same. Hence, the noise effect on the kernel blurring function could be removed effectively by estimating the blurring function based on sub-images and applying the cross validation method. Our experiments with cross validation method showed that even though the estimation of blurring function was more accurately, the estimate of the original image unfortunately became worse because of the bound effect.

7.2. Future works

Although the thesis has filled the gap by using adaptive image model, developed in this thesis, in the blind deblurring problem, some challenges require further investigations in the future.

Firstly, in the implementation of the proposed blind deblurring algorithms, some approximations which were applied to reduce the computational complexity can be replaced by direct calculations. In particular, the covariance matrix of the original image which was approximated by a circulant matrix in step 2 and step 3 of the proposed algorithms (see section 4.4.3) would be numerically calculated by the conjugate gradient method. Direct calculation of the covariance matrix should lead to more precise image estimation. However, it makes the algorithms much more computational intensive because the estimation of covariance matrix of the blurring function is not a circulant matrix anymore.

Secondly, the speed of convergence of the proposed algorithms was controlled by the temperature parameter *T(k)* which was proportional to the inverse of logarithm function. The parameter should be studied further to investigate how its variation affects the speed of convergence of the deblurring algorithms.

Lastly, a common limitation of the existing blind deblurring algorithms and our proposed algorithms was that they did not perform well when the blurring function had sharp transitions. To deal with this problem, more complicated model should be used to model the blurring function. For instance, in our proposed algorithms, the SAR model or the Gaussian model may be replaced by the TV model in modeling the blurring function. It is notable that the more complicated the model is, the higher would be the computational effort.

Bibliography

[01] V. Katkovnik, K. Egiazarian, and J. Astola, **A Spatially Adaptive Nonparametric Regression Image Deblurring**, *IEEE Trans. Image Process.*, vol. 14 (10), pp: 1469- 1478, 2005.

[02] R. Neelamani, H. Choi, and R. Baraniuk, **ForWaRD: Fourier-Wavelet Regularized Deconvolution for Ill-Conditioned Systems**, *IEEE Trans. Signal Processing*, vol. 52 (2), pp: 418- 433, 2004.

[03] R. Li, **Image Restoration using A Hybrid Fourier-Wavelet domain Hidden Markov Model**, *Southeastern Symposium on System Theory*, pp: 279-282, Mercer University, USA, 2007.

[04] X. Cao, M. Yi, X. Wang, and S. Peng, **Image restoration with edge-preserving regularization in wavelet domain**, *Networking, Sensing and Control, Proceedings IEEE*, pp: 543- 548, Arizona, USA, March 2005.

[05] X. Zhang and W. P Zhu, **Wavelet Domain Image Restoration using Adaptively Regularized Constrained Total Least Squares,** *Proc. Intelligent Multimedia, Video and Speech Processing*, pp: 567- 570, Hongkong, Oct 2004.

[06] J. M. B. Dias, **Bayesian wavelet-based image deconvolution: a GEM algorithm exploiting a class of heavy-tailed priors,** *IEEE Trans. Signal Processing,* Vol. 15(4), pp: 937-951, 2006.

[07] P. D. Rivaz and N. Kingsbury, **Bayesian Image Deconvolution and Denoising Using Complex Wavelets**, *CIP*, vol. 2, pp: 273-276, Greece, 2001.

[08] M. A. T. Figueiredo and R. D. Nowak, **A bound optimization approach to wavelet-based image deconvolution**, *ICIP*, vol. 2, pp: 782-785, Genova, 2005.

[09] R. Pan and S. J. Reeves, **Fast Restoration with Edge-preserving Regularization**, *Proceedings of 37th Southeastern Symposium on System Theory*, Tuskegee AL, March 2005.

[10] P. L. Combettes and J. C. Pesquet, **Image restoration subject to a total variation constraint**, *IEEE. Trans. Image Processing*, vol. 13 (9), 2004.

[11] P. Charbonnier, L. Blance-Feraud, G. Aubert, et al., **Deterministic edge-preserving regularization in computed image**, *IEEE Trans. Image Process.*, vol. 6(2), pp: 298-311, 1997.

[12] R. Molina, *et al.*, **Restoration of Severely Blurred High Range Images Using Stochastic and Deterministic Relaxation Algorithms in Compound Gauss-Markov Random Field**, *Pattern Recognition*, vol. 33, pp: 555-571.

[13] M. Mignotte, **A Segmentation-Based Regularization Term for Image Deconvolution**, *IEEE. Trans. Image Processing*, vol. 15 (7), pp: 1973-1984, 2006.

[14] G. K. Chantas, N. P. Galatsanos, and A. C. Likas, **Bayesian Restoration Using a New Nonstationary Edge-Preserving Image Prior**, *IEEE. Trans. Image Processing*, vol. 15 (10), pp: 2987-2997, 2006.

[15] M. A. T. Figueiredo and R. D. Nowak, **An EM algorithm for wavelet-based image restoration**, *IEEE Trans. Image Process.*, Vol. 12, pp. 906–916., 2003.

[16] S. D. Babacan, R. Molina, and A. K. Katsaggelos, **Total Variation Image Restoration and Parameter Estimation using Variational Posterior Distribution Approximation**, *ICIP*, vol. 1, pp: I – 97-I -100, 2007.

[17] M. R. Bhatt and U. B. Desai, **Robust Image Restoration Algorithm using Markov Random Field Model**, *CVGIP- Graphical Models and Image Processing*, pp. 61-74, 1994.

[18] L. D. Cai, **Objective Assessment to Restoration of Global Motion-blurred Images Using Traveling Wave Equations**, *Third International Conference on Image and Graphics Proceedings*, pp: 6- 9, Hong Kong, 2004.

[19] R. L. Lagendijk, A. M. Tekalp, and J. Biemond, **Maximum likelihood image and blur identification: a unifying approach**, *Opt. Eng.*, vol.29, pp. 422-435, 1990.

[20] W. G. He, S. F. Li, and G. W. Hu, **Blur Identification Using an Adaptive ADALINE Network**, *Machine Learning and Cybernetics*, vol. 9, pp: 5314- 5317, 2005.

[21] G. Pavlovic and A. M. Tekalp, **Maximum Likelihood Parametric Blur Identification Based on a Continuous Spatial Domain Model**, *ICASSP'91*, vol. 1, pp: 496- 504, Canada, 1991.

[22] L. Chen and K. H. Yap, **Identification of Blur Support Size in Blind Image Deconvolution**, *Proc. Int. Conf. Information, Communications, Signal Processing and Pacific-Rim Conf. Multimedia*, vol. 1, Singapore, pp: 503- 507, 2003.

[23] L. Chen and K. H. Yap, **Efficient Discrete Spatial Techniques for Blur Support Identification in Blind Image Deconvolution**, *IEEE Trans. Signal Processing*, vol. 54, pp: 1557- 1562, 2006.

[24] S. J. Reeves and R. M. Mersereau, **Blur Identification by the Method of Generalized Cross Validation**, *IEEE Trans. Image Processing*, vol. 1, pp: 301-310, 1992.

[25] Y. Yitzhaky and N. S. Kopeika, **Identification of Motion Blur for Blind Restoration**, *Convention of Electrical and Electronics Engineers*, pp: 3.4-3.5, 1995.

[26] A. E. Savakis and H. J. Trussell, **Blur Identification by Residual Spectral Matching**, *IEEE Trans. Image Processing*, vol. 2, pp: 141-151, 1993.

[27] A. E. Savakis and H. J. Trussell, **Blur Identification by Statistical Analysis**, *ICASSP'91*, vol. 4, pp: 2493-2496, 1991.

[28] A. C. Likas, N. P. Galatsano, **A Variational Approach for Bayesian Blind Image Deconvolution**, *IEEE Trans. Signal Processing*, vol. 52, pp: 2222-2233, 2004.

[29] J. Mateos, R. Molina, and A. K. Katsaggelos, **Approximation of Posterior Distribution in Blind Deconvolution using Variational Methods**, ICIP (2) 2005: pp: 770-773

[30] R. Molina, J. Mateos, and A. K. Katsaggelos, **Blind Deconvolution Using Variational Approach to Parameter, Image, and Blur Estimation**, *IEEE. Trans. Image Processing*, vol. 15 (12), pp: 3715-3727, 2006.

[31] D. C. Ghiglia, L. A. Romero, and G. A. Mastin, **Systematic approach to two-dimensional blind deconvolution by zero-sheet separation**, *J. Opt. Soc. Amer. A*, vol. 10, no. 5, pp. 1024–1036, 1993.

[32] D. Kundur and D. Hatzinakos, **A Novel Blind Deconvolution Scheme for Image Restoration Using Recursive Filtering**, *IEEE Trans. Signal Processing*, vol. 46, no. 2, pp: 375, 1998.

[33] E. Y. Lam and J. W. Goodman., **An iterative statistical approach to blind image deconvolution**, *Journal of the Optical Society of America A*, vol. 17(7), pp: 1177–1184, 2000.

[34] F. Sroubek and J. Flusser, **Multichannel blind deconvolution of spatially misaligned images,** *IEEE Trans. Image Processing*, vol. 14(7), pp: 874-883, 2005.

[35] S. Yanga, G. Erryb, S. Nemethb, S. Mitraa, and P. Soliz, **Image restoration with high resolution adaptive optical imaging system**, *CBMS*, pp: 479- 484, USA, 2004.

[36] J. Zhou and M. N. Do, **Multi-channel FIR Exact Deconvolution in Multiple Variables**, *ICASSP*, vol. 4, pp: 1001-1004, 2005.

[37] N. K. Bose, M. K. Ng, and A. C. Yau, **Super – Resolution Image Restoration from Blurred Observations**, *IEEE Trans. Image Processing*, pp: 6296-6299, 2005.

[38] R. Molina, J. Mateos, A. K. Katsaggelos, and M. Vega, **Bayesian Multichannel Image Restoration Using Compound Gauss-Markov Random Fields**, *IEEE Trans. Image Process.*, vol. 12, pp: 1642- 1654, 2003.

[39] Y. P. Guo, H. P. Lee, and C. L. Teo, **Multichannel Image Restorations using An Iterative Algorithm in Space Domain**, *Image and Vision Computing*, vol. 14, pp: 389-400, 1996.

[40] P. Bar, A. Brook, N. Sochen, and N. Kiryati, **Deblurring of Color Images Corrupted by Impulsive Noise**, *IEEE Trans. Image Processing*, vol. 16, pp: 1101-1110, 2007.

[41] R. C. Puetter, T. R. Gosnell, and A. Yahil, **Digital Image Reconstruction: Deblurring and Denoising**, *Annual Review of Astronomy and Astrophysics*, vol. 43, pp: 139-194, 2005.

[42] M .R. Banham and A. K. Katsaggelos, **Digital image restoration,** *IEEE Signal Processing Magazine***,** vol. 14 (2), pp: 24 –41, 1997.

[43] D. Kundur and D. Hatzinakos, **Blind image deconvolution**, *IEEE Signal Process. Mag.*, vol. 13, pp: 43–64, 1996.

[44] D. Kundur and D. Hatzinakos, **Blind image deconvolution revisited**, *IEEE Signal Processing Magazine*, vol. 13(6), pp: 61–63, 1996.

[45] Y. P. Guo, H. P. Lee, and C. L. Teo, **Blind Restoration of Images Degraded by Space-Variant Blurs using Iterative Algorithms for Both Blur Identification and Image Restoration**, *Image and Vision Computing*, vol. 15, pp: 399-410, 1997.

[46] D. Calvetti, B. Lewis, and L. Reichel, **Restoration of images with spatially variant blur by the GMRES method**, *Advanced Signal Processing Algorithms, Architectures, and Implementations X, Proc. of the Society of Photo-Optical Instrumentation Engineers, SPIE*, vol. 4116, The International Society for Optical Engineering, Bellingham, WA, 2000, pp. 364-374.

[47] J. G. Nagy and D. P. O'Leary, **Restoring images degraded by spatially-variant blur**, *SIAM Journal on Scientific Computing*, vol. 19, pp: 1063-1082, 1998.

[48] A. Marquina, and S. Osher, **A new time dependent model based on level set motion for nonlinear deblurring and noise removal**, *Lecture Notes in Computer Science*, vol. 1682, pp. 429-434, 1999.

[49] M. Welk, D. Theis, and J. Weickert, **Variational deblurring of images with uncertain and spatially variant blurs**, *Pattern Recognition, Lecture Notes in Computer Science*, vol. 3663, pp.485–492, 2005.

[50] E. B. Braiek, H. Amiri, and F. Benzarti, **Motion Blurred Image Deconvolution with Anisotropic Regularization**, *International Symposium on Control, Communications and Signal Processing*, pp: 443- 446, Tunisia, March 2004.

[51] P. H. Garthwaite, I. T. Jolliffe, and B. Jones, **Statistical Inference,** *Prentice Hall,* 1995.

[52] R. Pan and S. J. Reeves, **Efficient Huber-Markov Edge – Preserving Image Restoration**, *IEEE Trans. Image Processing*, vol. 15, pp: 3728-3735, 2006.

[53] A. Asif, **Fast Rauch – Tung – Striebel Smoother-based Restoration for Non-causal Images**, *IEEE Signal Processing Letters*, vol. 11, pp: 371-374, 2004.

[54] N. B. Karayiannis and A. N. Venetsanopoulos., **Regularization theory in image restoration-the stabilizing functional approach**, *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 38(7), pp: 1155, 1990.

[55] G. Chinn and S. C Huang, **Image Reconstruction and Restoration: Overview of Common Estimation Structures and Problems**, *IEEE Trans. Acoustics Speech and Signal Processing,* vol. 37 (12), 1989.

[56] A. M. Tekalp, H. Kaufman, and J. W. Woods, **Edge-Adaptive Kalman Filtering for Image Restoration with Ringing Suppression**, *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 37(6), pp: 892-899, 1989.

[57] C. Taswell, **The what, how, and why of wavelet shrinkage denoising**, *Technical Report CT-1998-09*, Computational Toolsmiths, Stanford, www.toolsmiths.com, 1999.

[58] D. L. Donoho and I. M. Johnstone, **Ideal spatial adaptation via wavelet shrinkage**, *Biometrika*, vol. 81, pp: 425-455, 1994.

[59] D. L. Donoho and I. M. Johnstone, **Adapting to unknown smoothness via wavelet shrinkage,** *Journal of American Stat. Assoc.*, vol. 90, pp: 1200-1244, 1995.

[60] G. Chang, B. Yu, and M. Vetterli, **Adaptive wavelet thresholding for image denoising and compression**, *IEEE Trans. Image Processing*, pp: 1532-1546, 2000.

[61] J. Miskin, **Ensemble Learning for Independent Component Analysis**, 2000.

[62] S. D. Babacan, R. Molina, and A. K. Katsaggelos, **Variational Bayesian Blind Deconvolution using A Total Variation Prior**, *IEEE Trans. Image Processing*, vol. 18, pp: 12-26, 2009.

[63] R. Nakagaki and A. K.,Katsaggelos, **A VQ-based Blind Image Restoration Algorithm,** *IEEE Trans. Image Processing*, vol. 12, pp: 1044–1053, 2003.

[64] Asmatullah, A. M. Mirza, and A. Khan, **Blind Image Restoration Using Multilayer Back-propagator**, *INMIC 2003*, pp: 55-58, Islamabad, 2003.

[65] E. Y. M. Lam, **A Nonlinear Image Restoration Framework Using Vector Quantization**, *Third International Conference on Image and Graphics*, pp: 2-5, Hong Kong, 2004.

[66] A. Pižurica and W. Philips, **Estimating the probability of the presence of a signal of interest in multiresolution single- and multiband image denoising**, *IEEE Transactions on Image Processing*, vol. 15, no. 3, pp. 654–665, 2006.

[67] M. A. T. Figueiredo and R. D. Nowak, **Wavelet-based image estimation: an empirical Bayes approach using Jeffrey's noninformative prior,** *IEEE Trans. Image Processing*, pp: 1322-1331, 2001.

[68] G. Y. Chen, T. D. Bui, and A. Krzyzak, **Image Denoising Using Neighbouring Wavelet Coefficients**, *ICASSP*, vol. 2, pp. 917-920, 2004.

[69] L. Sendur and I. W. Selesnick, **Bivariate Shrinkage Functions for Wavelet-Based Denoising Exploiting Interscale Dependency**, *ICASSP*, vol. 50, pp: 2744-2756, 2002.

[70] N. T. Le and K. B. Lim, **Image Denoising Using a New Line-Field,** *Image and Vision Computing New Zealand 2006*, pp: 305-309, New Zealand, Nov 27-29, 2006.

[71] W. Li, D. Zhang, Z. Liu, and X. Qiao, **Fast block-based image restoration employing the improved best neighborhood matching approach**, *IEEE Transactions on Systems, Man, and Cybernetics*, Part A , vol. 35(4), pp: 546-555, 2005.

[72] P. L. Combettes and J. C. Pesquet, **Estimating first-order finite-difference information in image restoration problems**, *ICIP*, vol. 1, pp. 321-324, Singapore, 2004.

[73] X. Limin, **Image Restoration Based on Particle Filters**, *7th International Conference On Signal Processing Proceedings*, vol. 1 (3), Honolulu, Hawaii, USA 2004.

[74] S. T. Acton and A. C. Bovik, **Nonlinear image estimation using piecewise and local image models**, *IEEE Trans. Image Processing*, vol. 7, pp. 979-991, 1998.

[75] A. Kondu and J. Zhou, **Combination median filter**, *IEEE Trans. Image Processing*, vol. 1, pp: 422-429, 1992.

[76] A. B. Hamza and H. Krim, **Image Denoising: A Nonlinear Robust Statistical Approach**, *IEEE Trans. Image Processing*, vol. 49, pp: 3045-3054, 2001.

[77] P. E Ng and K. K. Ma, **A switching median filter with boundary discriminative noise detection for extremely corrupted images**, *IEEE Trans. Image Processing*, vol. 15, pp: 1506-1516, 2006.

[78] V. Katkovnik, K. Egiazarian, and J. Astola, **Adaptive Window Size Image Denoising Based on Intersection of Confidence Intervals (ICI) Rule**, *J. Math. Imag. Vis.*, vol. 16, no. 3, pp: 223–235, 2002.

[79] J. Besag, **Spatial interaction and the statistical analysis of lattice systems**, *J. Royal Statist. Soc., Ser. B*, vol. 36, pp: 192-236, 1974.

[80] S. Geman and D. Geman., **Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images**, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 6, pp: 721-741, 1984.

[81] J. Besag, **On the statistical analysis of dirty pictures**, *J. Royal Statistical Soc., Series B*, vol. 48, no. 3, pp: 259-302, 1986.

[82] F. J. Jeng and J. W. Woods, **Simulated Annealing in Compound Gaussian Random Fields**, *IEEE Trans. Info. Theory*, vol. 36, no. 1, pp. 94-107, 1990.

[83] J. Park and L. Kurz, **Image enhancement using the modified ICM method**. *IEEE Trans. Image Processing,* vol. 5, pp: 765-771, 1996.

[84] J. Portilla, **Javier Portilla's Home Page**, *http://decsai.ugr.es/javier/index.html*, visited on 29/09/2006.

[85] D. J. C. MacKay, **Information theory, inference, and learning algorithms**, *Cambridge Univ. Press*, 2003.

[86] R. M. Gray, **Toeplitz and Circulant Matrices: A review**, *Now Publishers Inc*, 2006.

[87] N. T. Le and K. B. Lim, **Image Denoising by Line-Field Based Filter and Wavelet Shrinkage**, *Wavelet and Applications Conference 2006,* pp: A-43, Switzerland, Jul 10-14, 2006.

[88] J. M. B-Dias, M. A. T. Figueiredo, and J. P. O, **Total Variation based Image Deconvolution: A Majorization – Minimization Approach**, *IEEE Int. Conf. on Acoustics, Speech, and Signal Processing* 2004.

Appendix A – Images Used for Experiments

Figure A- 1. “Lena” image 512×512 pixels

Figure A- 2. “Cameraman” image 256×256 pixels.

Figure A- 3. “Montage” image 256×256 pixels.

Figure A- 4. “Bridge” image 256×256 pixels.

Figure A- 5. “House” image 256×256 pixels.

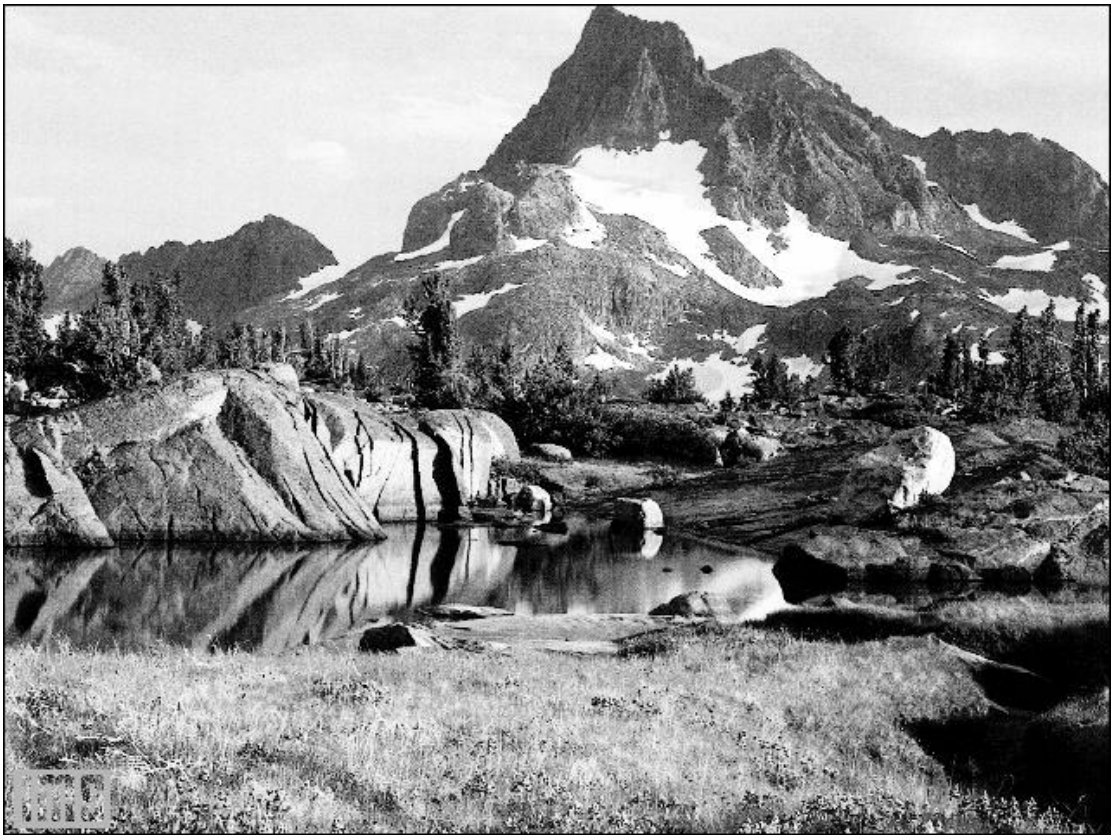


Figure A- 6. “Mountain” image 640×480 pixels.

Figure A- 7. “Zelda” image 512×512 pixels.

Figure A- 8. “Boat” image 512×512 pixels.

Figure A- 9. “Bird” image 256×256 pixels.

Figure A- 10. “Goldhill” image 256×256 pixels.

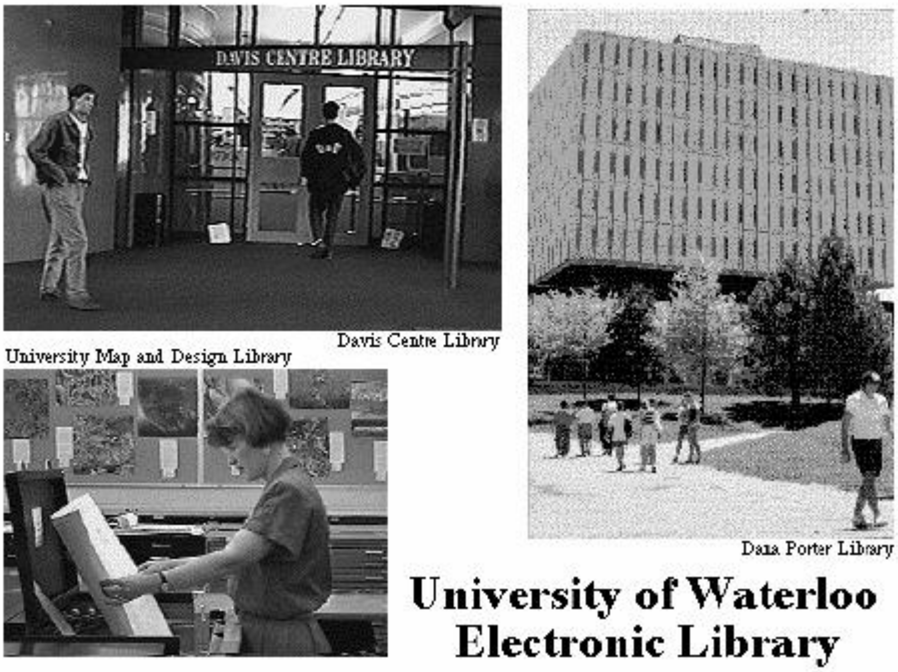


Figure A- 11. “Library” image 464×352 pixels.


Figure A- 12. “Frog” image 621×498 pixels.

Figure A- 13. “Flinstones” image 512×512 pixels.

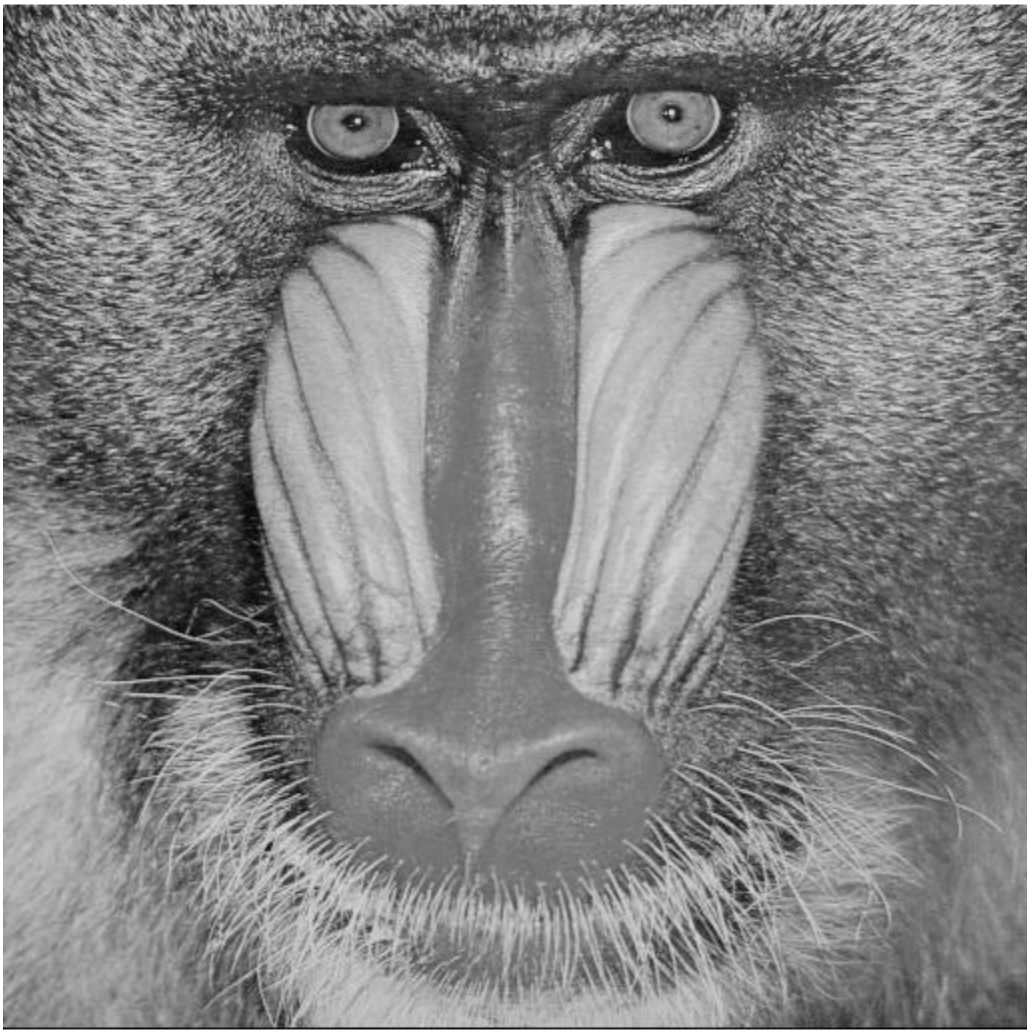


Figure A- 14. “Mandrill” image 512×512 pixels.

Figure A- 15. “Washsat” image 512×512 pixels.

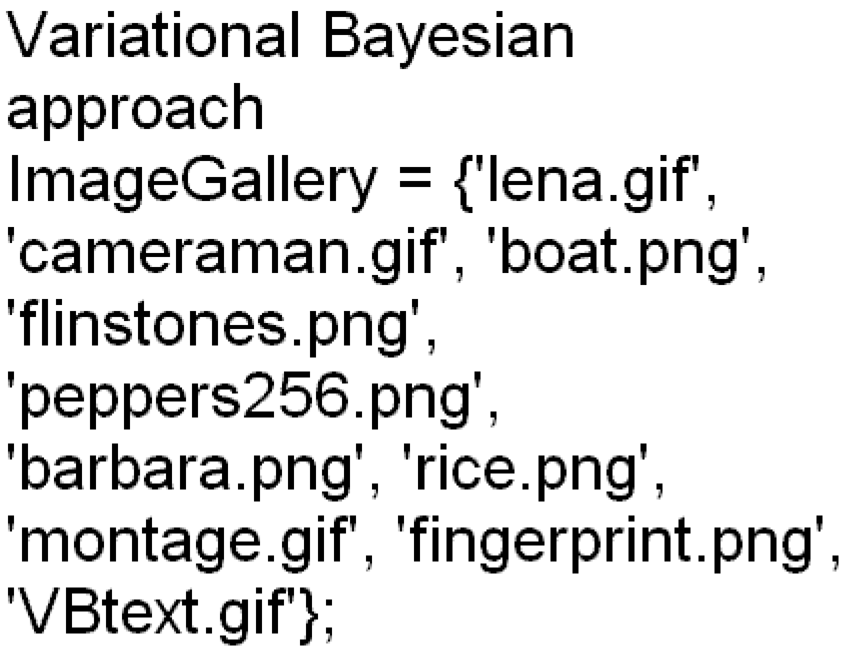


Figure A- 16. “Text” image 512×512 pixels.

Figure A- 17. “Barbara” image 512×512 pixels.

Appendix B – Deblurred Images

I. Experimental results with Gaussian - shape PSF

The images in this section are the noisy blurred images and the deblurred images of experiments in section 5.2 using LF-SAR algorithm.

Figure B - 1. The noisy blurred image of Lena image and its restored image.

Figure B - 2. The noisy blurred image of “Cameraman” image and its restored image.

Figure B - 3. The noisy blurred image of “Boat” image and its restored image.

Figure B - 4. The noisy blurred image of Barbara image and its restored image.

Figure B - 5. The noisy blurred image of “Montage” image and its restored image.

Figure B - 6. The noisy blurred image of “Flintstones” image and its restored image.

II. Experimental results with horizontally uniform PSF

The images in this section are the noisy blurred images and the deblurred images of experiments in section 5.3 using LF-G algorithm.

Figure B - 7. The noisy blurred image of Lena image and its restored image.

Figure B - 8. The noisy blurred image of “Cameraman” image and its restored image.

Figure B - 9. The noisy blurred image of “Boat” image and its restored image.

Figure B - 10. The noisy blurred image of Barbara image and its restored image.

Figure B - 11. The noisy blurred image of “Montage” image and its restored image.

Figure B - 12. The noisy blurred image of “Flintstones” image and its restored image.

III. Experimental results with out-of-focus PSF

The images in this section are the noisy blurred images and the deblurred images of experiments in section 5.4 using LF-SAR algorithm.

Figure B - 13. The noisy blurred image of Lena image and its restored image.

Figure B - 14. The noisy blurred image of “Cameraman” image and its restored image.

Figure B - 15. The noisy blurred image of “Boat” image and its restored image.

Figure B - 16. The noisy blurred image of Barbara image and its restored image.

Figure B - 17. The noisy blurred image of “Montage” image and its restored image.

Figure B - 18. The noisy blurred image of “Flintstones” image and its restored image.