MOTION DEBLURRING FOR OPTICAL CHARACTER RECOGNITION

QI XING YU

NATIONAL UNIVERSITY OF SINGAPORE

2004

MOTION DEBLURRING
FOR
OPTICAL CHARACTER RECOGNITION

QI XING YU

A THESIS SUBMITTED

FOR THE DEGREE OF MASTER OF SCIENCE

DEPARTMENT OF COMPUTER SCIENCE

SCHOOL OF COMPUTING

NATIONAL UNIVERSITY OF SINGAPORE

2004

Acknowledgement

First and foremost, I would like to express my sincere gratitude to my supervisor, Associated Professor Tan Chew Lim. Without his valuable advice, guidance and encouragements, this thesis would not have been finished.

I also thank my senior Zhang Jie. His constant assistance and support on my research is always of great help to me. I really appreciate that he has spent considerable time for sharing his experience and clarifying my thoughts.

Last but not least, I am grateful to my family who support me any time when I am in need and give me confidence for finishing my work.

Contents

List of Figures

List of Tables

Summary

Motion blur is the one dimensional distortion when the relative velocity between different objects in the scene and camera is relative large compared with the camera's exposure time in the resulting image. Optical Character Recognition (OCR) performance of document images, e.g. Name Cards is severely downgraded if blurring exists in the images. To improve OCR results, we need to precisely estimate the two motion blur parameters – Orientation and Extent from the information in a single blurred image and apply image restoration algorithms, e.g. Wiener Filter to deblur.

I.Rekleitis has proposed an algorithm to estimate the optical flow fields of an image based on the motion blur interpretation. This algorithm can estimate blur parameters but the processing time and considerable errors in the estimation make it not suitable for deblurring name card images. In this thesis, an algorithm based on I.Rekleitis' method has been proposed. It works for both synthetic and real world motion blurred images. The algorithm first assumes the blur in the image is uniform linear motion blur. Two blur parameters are successfully extracted from the blurred image. OCR results can be improved to certain extent based on the estimation of blur. Then more severe motion – uniform acceleration motion blur is analyzed and method to estimate such blur has been proposed from the expansion of the previous case. OCR performance is enhanced for those blurred images, which do not have results in the first attempt. Finally, analysis on the Wiener Filter has shown the correct procedure to deblur name card images with blur estimation.

1 Introduction

In this chapter, we briefly introduce the motion deblurring in the application of OCR, followed by the motivation of current research and the outline of this thesis.

1.1 Introduction

The process to take images from real life scene is not perfect. Three major types of degradation will occur – blurring, point wise nonlinearities and noise. The definition of blur is a cause of imperfect vision or more formally blur is a form of bandwidth reduction of the image in the image formation process. Some common types of blurring are motion blur, out-of-focus blur and simple harmonic vibration blur. Motion blur is caused by relative motion between the camera and the object. The motion can be modeled by either constant velocity or velocity with uniform acceleration. Out-of-focus blur is caused by a defocused lens system with a circular aperture and the vibration blur by its name, is caused by harmonic vibrations of the camera. In this thesis, only motion blur is studied and analyzed in blurred images.

The problem of estimating the motion blur has received much attention because of its many different applications. Application varies from aerial photographs that are produced for remote sensing where the blur are introduced by atmospheric turbulence, aberrations in the optical system and relative motion between the camera and the ground to electro micrographs that are corrupted by the aberrations of the electro lenses or even in the criminology where photos of evidence are blurred by accidents. One particular application which we are interested in is document image. Motion blur will severely affect the performance of optical character recognition (OCR) results on blurred document images. Here OCR is the recognition of printed or written text characters by a computer. This involves photo scanning of the text character-by-character, analysis of the scanned-in image, and then translation of the character image into character codes, such as ASCII, commonly used in data processing. Nowadays sophisticated OCR software have been developed and put into markets. Common ones are ScanSoft OmniPage series.

In this thesis, we focus on name card images obtained from handheld cameras. These images are provided by courtesy of HotCard Company. HotCard is a pioneering local company in the development of compact sized, multilingual and high accuracy OCR for name card recognition. Their cutting-edge products, Scan Pen and Name Card Scanner, are widely used by business people who collect a large amount of name cards every day. They scan name cards, perform OCR operation, and transfer images into editable text. A typical name card is of rectangular shape and has characters or textures in any region as shown in Figure 1.1. Normally, the image captured has some disturbing backgrounds and distortions due to the relative position of the camera and the name card. However, they show negligible impact on the final OCR results.

Figure 1.1 Typical Name Card Image

A typical blurred name card is shown in Figure 1.2. We can hardly recognize anything by eyes except those large and bold fonts. Thus an effort has been put to “decode” the motion blur information in the blurred name cards and attempted to deblur it for better OCR performance.

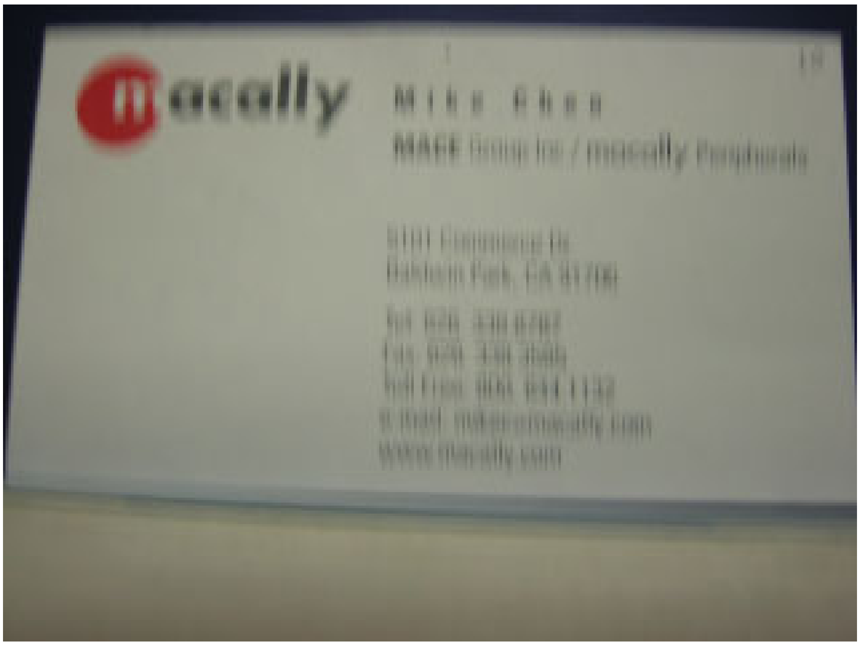


Figure 1.2 Typical Motion Blurred Name Card Image

1.2 Motivation

Scanning a name card may take seconds or even minutes. The long processing time has made scanning a time-wasting task. Instead, digital camera can be used to capture photos of name cards and transfer them to OCR software for recognition. Unfortunately, possible degradations especially motion blur have been encountered in this alternative approach. A practicable solution to deblur these blurred name cards need to be raised.

I.Rekleitis' method [Rekleitis, 1995] (will be analyzed in detail in chapter 2) has been found to provide a practicable way to estimate the motion blur orientation and extent to improve the OCR results to some extent. However, his method is designed mainly for estimating the optical flow field in a single image. The long processing time and considerable errors have made it difficult to use for blurred name cards.

Previous work by J.Zhang [Zhang, 2004] has been done based on I.Rekleitis' algorithm to resort to a process of re-estimation to recover the actual blur distance. The process of re-estimation itself is time consuming. Besides, the earlier work assumes only uniform linear motion blur occurred in the blurred image. As such, for more severe or irregular motion, the OCR results are not too satisfactory. Thus another issue to examine is to find whether there is a way to estimate acceleration. Literature will be further surveyed to see if there is any related attempt, while at the same time, with a deeper understanding of I.Rekleitis' algorithm, we might be able to find some solution ourselves.

With the aim to reduce computational cost and achieve precise motion blur estimation thus improve OCR results, we have our own algorithm based on I.Rekleitis' presented in chapter 3.

1.3 Thesis Structure

This thesis is divided into five chapters.

Chapter 1, *Introduction* – Introduces the motion deblurring in OCR applications and the motivation of this research and finally the thesis organization.

Chapter 2, *Background and Literature* – Explains the motion blur problem. The mathematics behind is presented and conventional approach to solve this problem is studied. Recent work that has been addressed to estimate the motion blur parameters and deblur the blurred images is surveyed. Finally the chapter defines our algorithm requirements on motion deblurring for OCR.

Chapter 3, *Algorithm* – Describes our modified methods based on I.Rekleitis' algorithm in the first part and how we expand this algorithm to more complex uniform acceleration motion blur in the second part. Besides, theory of uniform acceleration motion blur and Wiener filter is explained.

Chapter 4, *Experiments* – Examines the results of the proposed algorithm on both synthetic and real world motion blurred images. Sufficient experiments have been conducted to prove that this algorithm has achieved the precise motion blur parameter estimation. Final OCR results are presented to measure the overall performance.

Chapter 5, *Conclusions* – Summarizes the contributions of this research work and the difficulties encountered together with the limitations. Finally future research directions are proposed.

2 Background and Literature

In this chapter, we will discuss the motion blur problem. The mathematics behind is presented and conventional approach to solve this problem is studied. Recent work that has been addressed to estimate the motion blur parameters and deblur the blurred images is surveyed. In the last section, we define our algorithm requirements on motion deblurring for OCR.

2.1 Motion Blur Definition

When a moving object is observed by a camera, the image captured will suffer degradation of blurring if the exposure time is large enough. A number of scene points will contribute to the final intensity of a single image pixel during the capture process. The resulting intensity value for pixel $P_{i,j}$ can be illustrated in equation 2.1,

$$P_{i,j} = \frac{1}{N}\sum_{n=1}^{N} P_n \ , \tag{2.1}$$

where we assume the N scene points has intensity $P_1 \dots P_n$. The blurring of the image exists only in the direction of the relative motion between camera and object, so the one dimensional blur is called Motion Blur. A random dot image blurred in the horizontal direction is shown in Figure 2.1.

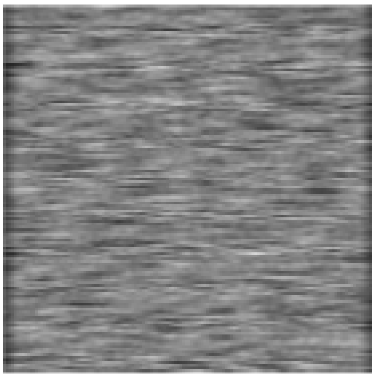


Figure 2.1 Motion Blurred Random Dot Image

We can model the blurring as a spatially linear invariant system. If we assume the object translates at a constant velocity *V* during the exposure time *T* at α angle from the horizon, the distortion is one-dimensional. We use *d* = *VT* and define the point spread function (PSF) as

$$h(x, y) = \begin{cases} 1/d, & 0 \leq |x| \leq d * \cos(\alpha), y = \sin(\alpha) * d \\ 0, & otherwise \end{cases} \tag{2.2}$$

The PSF of motion blur gives the number of original scene points that affect a specific pixel in the blurred image. Now we can define uniform linear motion blur mathematically as the result of a linear filter

$$g(x, y) = h(x, y) * f(x, y) + n(x, y), \tag{2.3}$$

where *g(x, y)* denotes the blurred image, *f(x, y)* denotes the original image and *n(x, y)* denotes additive noise as shown in Figure 2.2. Note (*) here is used to denote 2-D convolution. If the object does not translate at a constant velocity, then the PSF is more complex. We will study this problem called uniform acceleration motion blur in chapter 3. In motion deblurring problem, we need to estimate *h(x, y)* component in equation (2.3) then use image restoration algorithms to recover the original image. Our research thesis focuses on the estimation of α and *d* from the blurred image (name cards) to interpret uniform linear motion blur and expands to uniform acceleration case.

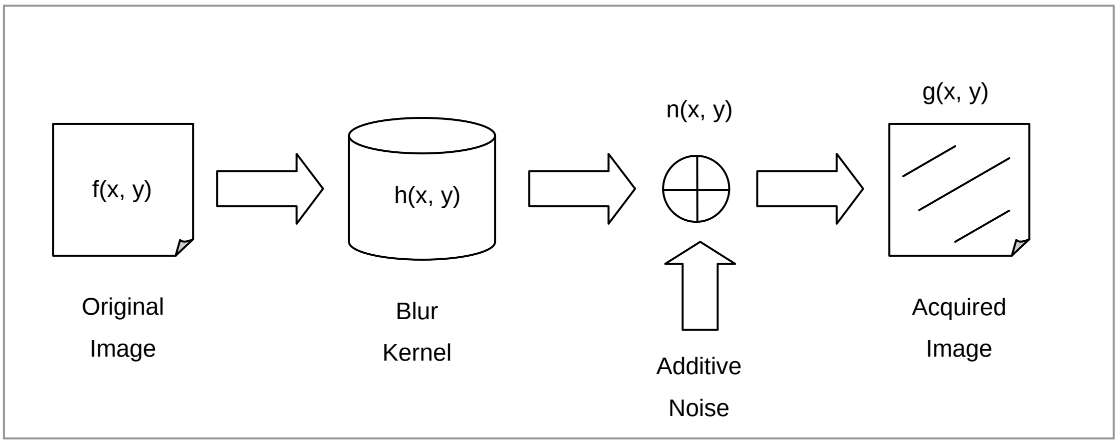


Figure 2.2 Image Acquisition

2.2 Conventional Approach for Motion Deblurring

In practice, to deblur an image, the degradation is rarely known, so the blur must be identified from the blurred image itself. For uniform linear motion blur, it is only necessary to estimate the two parameters of the PSF, i.e. the orientation of the blur and the blurring extent. Classic approach to this problem involves frequency and cepstral domain analysis.

2.2.1 Frequency Domain Method

Equation 2.3 is transformed to

$$G(u,v) = H(u,v)F(u,v) + N(u,v) \tag{2.4}$$

by using Fourier Transform, where $H(u,v)$ is the 2-D frequency response of the PSF and G, F, N are the transforms of the blurred image, original image and noise respectively. One approach to determine $h(x, y)$ is to identify $H(u,v)$ component in equation 2.5.

$$H(u,v) = \frac{\sin(\pi du)}{\pi du} = \sin c(\pi du)\,. \tag{2.5}$$

H(u,v) is the well known *sinc* function as shown in Figure 2.3.

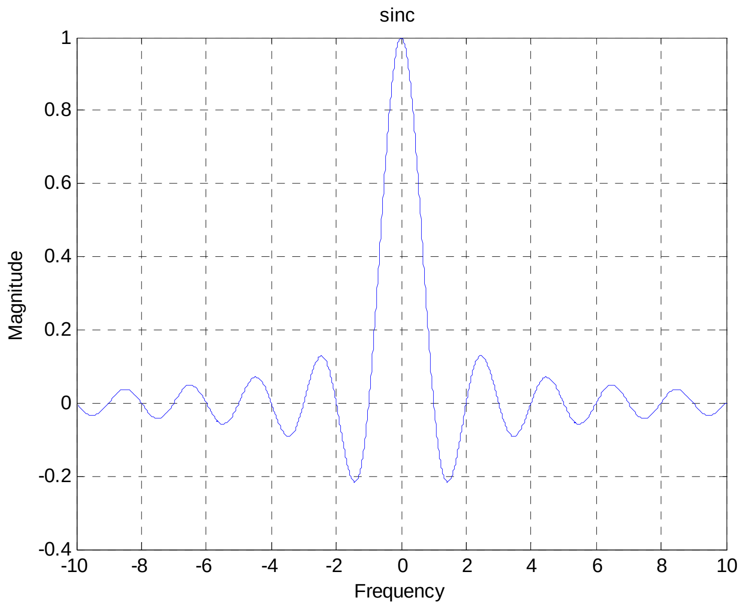


Figure 2.3 *sinc* Function

Estimating the blur extent can be done by searching the zero crossings of the frequency response of PSF, i.e. the magnitude of *sinc* function. If the noise is negligible, the zero crossings of *H(u,v)* are the same as *G(u,v)*. In the case of uniform linear motion blur, these zeros occur along the lines perpendicular to the orientation of the blur and spaced at intervals of *1/d* [Gennery, 1973].

2.2.2 Cepstral domain method

The cepstrum of the blurred image is defined as

$$C_g(p,q) = F\{\log|G(u,v)|\}, \tag{2.6}$$

where *F* denotes Fourier Transform. In Fourier analysis the independent variables in the transform domain (u,v) are called frequencies and have the physical dimension of $1/x$ (x being the dimension of the spatial independent variable). The independent variables in the cepstral domain (p,q) are called quefrencies and have the physical dimension of $1/u = x$. By the property of cepstrum, the convolutional effects of $h(x,y)$ are additive in the cepstral domain [Rom, 1975]. Again if the noise is negligible, we derive

$$C_g(p,q) = C_h(p,q) + C_f(p,q)\,. \tag{2.7}$$

Periodic zeros in $H(u,v)$ lead to large negative spikes in $C_h(p,q)$. For example, the zero crossings in linear motion blur are spaced $1/d$ apart. This periodic pattern results in a negative spike in $C_h(p,q)$, a distance d from the origin as shown in Figure 2.4. The negative spikes are always accompanied by spikes with less magnitude at each period. The amount this spike has rotated around the origin is the orientation of the blur. This approach to locate the spikes in $C_h(p,q)$ from $C_g(p,q)$ is not successful because of the overlay structure $C_f(p,q)$.

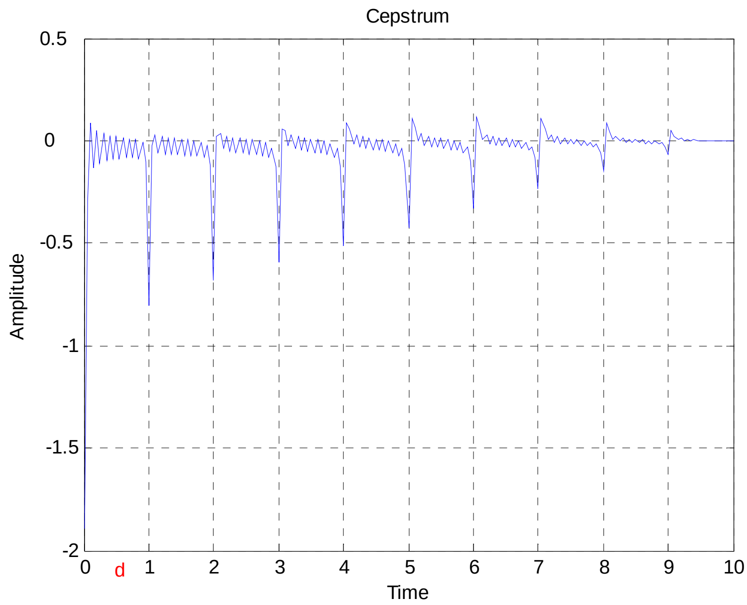


Figure 2.4 Cepstrum of *sinc* Function

Though these two approaches for motion blur are well defined theoretically, we find they are usually not practicable because of the extreme randomness of random noise and the blurred image. However, they form the theoretical basis of those proposed methods in recent work. Our algorithm presented in chapter 3 makes use of both frequency and cepstral domain methods as well.

2.3 Related Works

Motion deblurring algorithms usually can be divided into two categories,

I. Identify the blur parameters then apply well known restoration algorithm, e.g. Wiener Algorithm to deblur the image.

II. Incorporate the identification procedure into the restoration algorithm.

2.3.1 Blur Identification Methods

M.Cannon [Cannon, 1976] proposed the following technique to identify the blur parameters. He broke the blurred image into many sub images. Each sub section is multiplied by 2D Hamming window to reduce the edge effects and the average of power spectra is used to estimate the power spectrum of the PSF [Welch, 1967]. An alternative way is to compute the power cepstrum of each sub section. R.Fabian *et al.* [Fabian and Malah, 1991] proposed another method based on M.Cannon's approach. The algorithm first employs a form of spectral subtraction method to reduce high level noise. The resulting enhanced spectral magnitude function is transformed to the cepstral domain and identification procedure is completed using an adaptive, quefrency-varing, "comb-like" window. This algorithm works for both uniform linear motion blur and out of focus blur. Y.Yitzhaky *et al.* [Yitzhaky, Mor, Lantzman and Kopeika, 1998] proposed a method by making the observation that image characteristics along the direction of motion are different from the characteristic in other directions. The main idea is that the smearing effect in the motion direction acts as a low-pass filter in the spatial frequency domain. Therefore implementation of a high-pass filter, e.g. simple image derivative filter should suppress more image intensities than other directions. Motion direction can be identified by measuring the direction where the power spectrum of the image derivative is lowest. Then autocorrelation function (ACF) of each image derivative line in the motion direction is performed, and the average of the ACF of these lines is calculated. The blur extent is the distance between the location of the minimum and the center of the average based on the assumption that the average ACF of the image derivative lines resembles the ACF of the PSF derivative. This algorithm has proved to be practicable in the uniform acceleration blur. It distinguishes with others by estimating a complete blur function but not parameters. M.Chang *et al.* [Chang, Tekalp and Erdem, 1991] proposed a method, which is an extension of the classical methods for identification using power spectrum or power cepstrum of the blurred image to the bispectrum domain. It is assumed the observation noise is Gaussian and independent from the original image, so the zeros of the PSF can be obtained in the "central slice" of the bispectrum of the blurred image. K.C.Tan *et al.* [K.C.Tan, Lim and B.T.G.Tan, 1991] developed a procedure for estimating asymmetrical PSFs of real world motion blurred images. The procedure starts with a preliminary restoration using a ramp PSF. The result will indicate the true PSF based on the restoration errors and the blur extent is estimated. Note that all the algorithms mentioned above do not provide an effective way to determine the blur orientation.

I.Rekleitis proposed a new method to estimate the optical flow map of a blurred image using only information from the motion blur. His algorithm consists of two parts. The first part estimates the orientation of the blur by the following steps:

1. Apply Gaussian Masking and Zero Padding on the blurred image. (Optional)
2. Use steerable filter (2nd derivative of 2D Gaussian function) to identity the orientation of the motion blur from the logarithm of the Fourier Spectrum of the blurred image.

The second part uses the estimated orientation as the input and estimate the extent of the blur using the following steps:

1. Collapse the 2D logarithm of the power spectrum of the blurred image into 1D signal along the line indicating the orientation of the blur.
2. Calculate cepstrum of the 1D signal obtained in step 1. The negative peak in the real part of the cepstrum is used to estimate the blur extent.

This approach has been proved to work for both synthetic and real world motion blurred images. Experiments data provided in his thesis (Table 2.1) show average estimation error of blur orientation is 2.2 degrees. However, the average estimation error of blur extent is as large as 5.7 pixels for synthetic motion blurred images. The 64p and 128p stands for window size. Note that even provided with the correct angle, the blur extent error is very large. Besides there is no performance measure for real world blurred images. In J.Zhang's master thesis, he points out that I.Rekleitis' method has bad performance on blurred name cards. There is certain error characteristic between the detected angle and the actual angle.

Error Estimation	Gaussian Padded		Zero Padding		Gaussian Masking		No Preprocessing		Known Angle	
	64p	128p	64p	128p	64p	128p	64p	128p	64p	128p
Mean angle	2.4^o	2.1^o	1.8^o	2.0^o	2.0^o	1.6^o	1.6^o	1.9^o	-	-
Mean |angle|	**3.0^o**	**2.2^o**	2.8^o	2.5^o	3.0^o	2.0^o	2.5^o	2.0^o	-	-
S. Dev. Angle	0.5^o	0.4^o	0.2^o	0.2^o	0.6^o	0.5^o	0.3^o	0.2^o	-	-
Max angle	10^o	5^o	14^o	10^o	11^o	5^o	12^o	7^o	-	-
Min angle	-8^o	-2^o	-9^o	-5^o	-10^o	-3^o	-7^o	-2^o	-	-
Mean length	-2.7	-1.5	-1.3	-0.2	-2.2	-3.1	-1.5	0.5	-3.2	-5.2
Mean |length|	**4.1**	**5.7**	4.7	6.6	5.0	6.0	5.6	6.3	**4.6**	**6.2**
S. Dev. length	0.8	1.1	1.0	1.7	1.2	1.4	1.0	1.4	1.1	2.1
Max length	16	14	17	17	17	12	13	17	15	15
Min length	-8	-8	-8	-8	-8	-8	-8	-8	-8	-8

Table 2.1 Experiment Data on a Single Synthesized Motion Blurred Image in I.Rekleitis' Thesis

All the blur identification methods described above can successfully extract the motion blur parameters from the blurred image only. I.Rekleitis' method is the first algorithm that has been proven to work for a large variety of blur orientations and extents by our experimental results. After the blur estimation, image restoration algorithms will be employed to deblur images. In recent years, iterative image restoration has been well established. Popular methods include inverse filtering, minimum mean square error filtering and constrained least squares filtering [Biemond, Lagendijk and Mersereau, 1990]. Wiener filter has computational edges over others when the blur function is known. We will discuss it in detail in chapter 3.

2.3.2 Blur Identification with Restoration Methods

Recent developments in blur identification relate the identification process with the restoration process. ARMA parameter estimation methods involves modeling the true image as a two-dimensional autoregressive (AR) process and the PSF as a two-dimensional moving average (MA) process. Based on these models, the blurred image is represented as an autoregressive moving average (ARMA) process. The true image and PSF are estimated by identification of the ARMA parameters [Kundur and Hatzinakos, 1996]. Unfortunately, these methods have assumed some *a prior* knowledge of the original image and normally suffer high computational cost.

The existing methods of this class differ in how the ARMA parameters are estimated. Maximum-likelihood (ML) estimation and generalized cross-validation (GCV) methods are the most successful for image processing applications so far. The ML method is applicable to general symmetric PSFs. The restriction to symmetric PSFs is due to the fact that the phase of the PSFs can not be determined by ML. One new approach by A.Savakis [Savakis and Trussell, 1993] is presented, which does not directly utilize the ARMA modeling approach, but it can incorporate such models if desired. The PSF estimate is selected to provide the best match between the restoration residual power spectrum and its expected value from a collection of candidate PSFs constructed from experimental results.

2.4 OCR for Motion Blurred Images

Motion blur has severe degradation on the performance of OCR. Characters in the name cards are not recognized by OCR software even in the case of minor blurring. There are ways to enhance the performance such as background removal and character sharpening. In this thesis, we focus on the motion deblurring only.

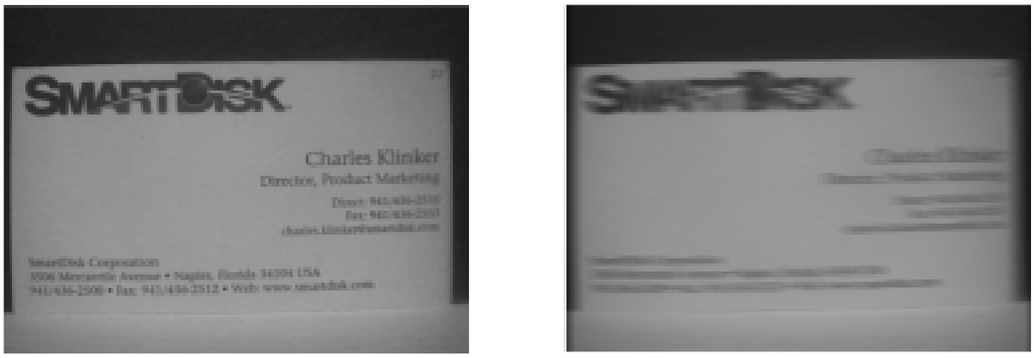
2.4.1 Precise Motion Blur Parameter Estimation

First, we define two performance measures of OCR.

$$\text{Precision} = \frac{\text{No. of characters correctly recognized}}{\text{Total No. of characters recognized}}, \tag{2.8}$$

$$\text{Recall} = \frac{\text{No. of characters correctly recognized}}{\text{Total No. of characters in a name card}}. \tag{2.9}$$

The precision and recall on real world motion blurred images is nearly 0 by our testing results. Thus we need to estimate the precise motion blur parameters from the blurred image and apply well tuned restoration filter to deblur it. Figure 2.5 shows two synthetic motion blurred name cards with blur extent 3 pixel and 25 pixel respectively. We observe that the blurring effect in Figure 2.5(a) is almost undetectable. It is confirmed by the OCR results. Thus we conclude motion blur with extent less than 4 pixels will not affect OCR performance. Since name cards are mainly captured by handheld cameras, we assume the blur extent will not exceed 25 pixel as in Figure 2.5(b).

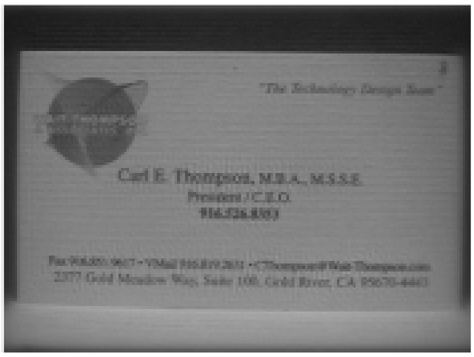


(a) (b)

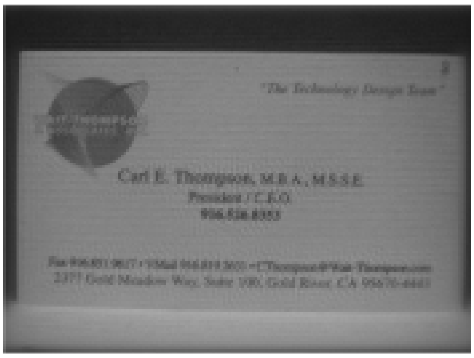
Figure 2.5 (a) Synthesized Motion Blurred Name Card with Blur Extent = 3 pixel (b) Synthesized Motion Blurred Name Card with Blur Extent = 26 pixel

Next, we observe the influence of the restoration error to OCR performance. Figure 2.6 shows blurred name card restored with four combinations of blur parameters. Figure 2.6(a) is restored with the correct orientation $= 45°$ and extent = 10 pixel. Though small ringing artifacts has occurred around the character, the restored image looks as clear as the original image. Figure 2.6(b) is restored with the orientation error equal to 5 degree. Obviously more ringing artifacts occur. OCR results show that most characters can still be recognized. Figure 2.6(c) is restored with the extent error equal to 1 pixel. OCR can tolerate such error. The precision and recall is above 80%. Finally, Figure 2.6(d) is restored with hybrid errors in both orientation and extent. Serious ringing artifacts and some “ghosting” effects have degraded the OCR performance. Recall is less than 20% in most cases.

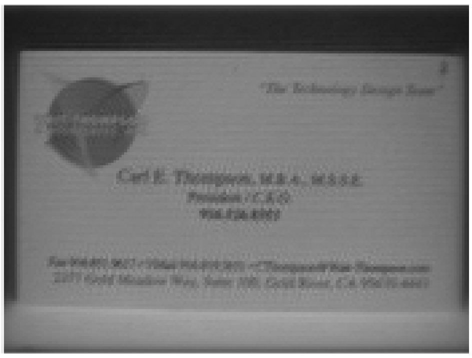
(a)



(b)



(c)



(d)

Figure 2.6(a) Synthesized Blurred Name Card Restored with Correct Blur Orientation and Extent (b) Restored with Orientation Error = 5 Degree (c) Restored with Blur Extent = 1 Pixel (d) Restored with Hybrid Error

We conclude that OCR is more sensitive to errors in blur extent. Now we define our precise motion blur parameter as the average error in blur orientation is less than 5 degree and the average error in blur extent is less than 1 pixel. However, blur extent usually can only be estimated when the blur orientation is known. This has increased the necessity of precise blur orientation estimation.

2.4.2 Algorithm Requirement

From the results in the section 2.4.1, a successful motion deblurring algorithm for OCR should report blur orientation in the range of $0°$ to $179°$ and blur extent in the range of 4 pixel to 25 pixel with average error less than 5 degree and 1 pixel respectively. Obviously, none of the algorithm in section 2.3.1 has fulfilled this requirement. We present our own algorithm in chapter 3. Experiments data in chapter 4 has proved that this modified algorithm based on I.Rekleitis' method is practicable for motion blurred name cards.

To sum up, motion blur is the distortion when the relative velocity between the objects in the scene and the camera is larger than the exposure time in the resulting image. Various methods have been proposed since decades ago. Most of them are able to make estimations of the blur parameters and deblur the images to certain extent. I.Rekleitis' algorithm meets the requirements of OCR most closely. Unfortunately the existence of the larger errors in blur extent has failed the purpose. This leads to our research in proposing new methods based on his idea for application in the name card image recognition, which will be discussed in detail in the next chapter.

3 Algorithm

In this chapter, a new approach of motion deblurring based on I.Rekleitis' algorithm is described in the first part and expanded to work on more complex motion – uniform acceleration motion blur in the second part. Finally, theory of Wiener filter is explained and an optimal form of the filter on document images is derived.

3.1 New Approach of Motion Deblurring on Uniform Linear Motion Blur

I.Rekleitis' method computes the optical flow map for a single blurred image. To obtain a precise blur function, we have made modifications on both blur orientation and extent estimations. To estimate the blur orientation of uniform linear motion blur, we apply three steps on the blurred image. In the first step, Gaussian masking is used to obtain better results with the initial Fourier transform. This step is the optional step of I.Rekleitis' algorithm. As opposed to his method, we apply Gaussian mask on the whole blurred image and eliminate the zero padding. In the new second step, thresholding techniques are used to make the blur smear lines (parallel ripples) clearer in the spectrum. The 2^{nd} derivative of a Gaussian and its Hilbert transform is applied as bandpass filter to the modified spectrum as opposed to the steerable filter used in I.Rekleitis' algorithm in the last step. The orientation that maximizes the frequency response of this filter is returned as the blur orientation. To estimate the blur magnitude of uniform linear motion blur, we again apply three new steps on the blurred image. The blur angle is required in the magnitude estimation. In the first step, differential filter is used in the direction perpendicular to the blur orientation to decorrelate the blurred image. In the second step, Radon transform is applied to the Fourier spectrum of the blurred image in the blur orientation to get a collapsed 1D signal. Cepstral domain analysis is performed in the last step. The position corresponding to the local peak of the cepstrum is the blur magnitude. The algorithm is required to report the correct blur orientation with tolerable errors in the range from $0°$ to $179°$ under variety of blur magnitudes and the correct blur magnitude with tolerable errors in the range from 5 to 25 pixels under all blur orientations.

The outline of the new approach is shown in Figure 3.1. Each step will be explained in depth in the following subsections. Note the “M” beside the text box indicates it is a modified step, otherwise a new step.

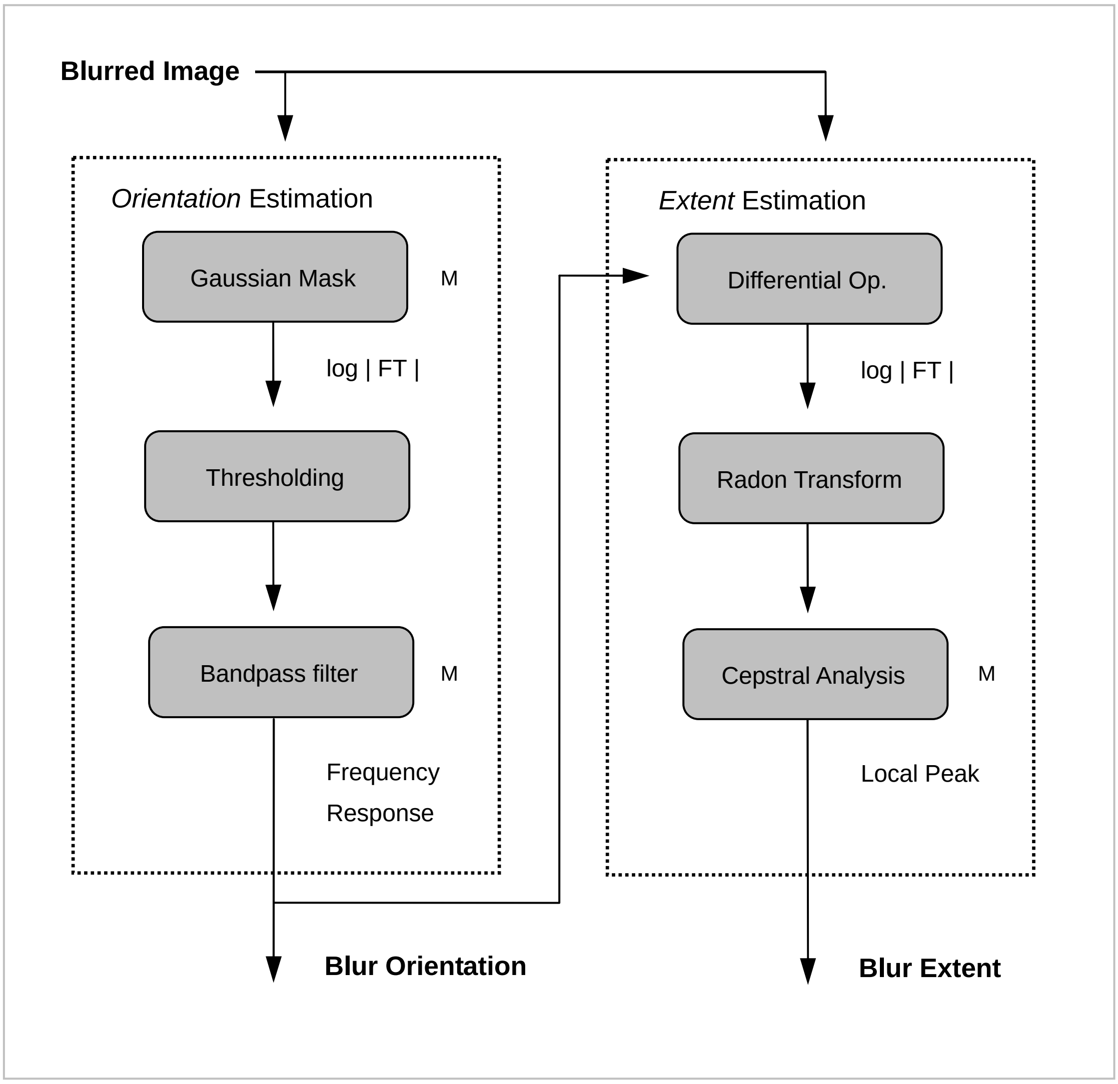


Figure 3.1 Outline of the Motion Deblurring Algorithm

3.1.1 Gaussian Mask

To analyze the spectrum of the blurred image, we always encounter the problem called boundary effect. It is usually caused by the sudden change of pixel intensities at image boundaries, which creates false edge signals in the image spectrum after applying Fourier transform. One approach to this problem is to consider only taking a patch of the image, i.e. mask the image with a window function that has value at the area of interest and zero everywhere else.

The simplest type of window function is called Rectangular window as shown by equation 3.1,

$$f(n) = \begin{cases} 1 & 0 \le n \le M - 1 \\ 0 & otherwise \end{cases}. \tag{3.1}$$

Window function also creates disturbing artifacts in the spectrum. The more abrupt the change into zeros of the functions, the more severe ripples will appear in the frequency domain. The ripples are caused when we convolve the two frequency domain representations together. By using a mask, we also require the original signal to be kept maximally unchanged. There is lots of research on the choice of best mask functions. The most common types are listed in Table 3.1. The functions have zero outside the range [0, M – 1].

Hamming Function	$0.54 - 0.46\cos(2\pi n / M - 1),\ \ 0 \le \mathrm{n} \le \mathrm{M}-1$
Hanning Function	$1/2(1 - \cos(2\pi n / M - 1)),\ \ 0 \le \mathrm{n} \le \mathrm{M}-1$
Blackman Function	$0.42 - 0.5\cos(2\pi n / M - 1) + 0.08\cos(4\pi n / M - 1),$ $0 \le n \le M - 1$
Bartlett Function	$2n/(M-1),\ \ \ 0 \le \mathrm{n} \le (\mathrm{M}-1)/2$ $2 - 2\mathrm{n}/(\mathrm{M}-1),\ \ (\mathrm{M}-1)/2 \le \mathrm{n} \le 1$
Gaussian Function	$\exp^{-(\frac{n-(M-1)/2}{(M-1)/4})^2},\ 0 \le \mathrm{n} \le \mathrm{M}-1$

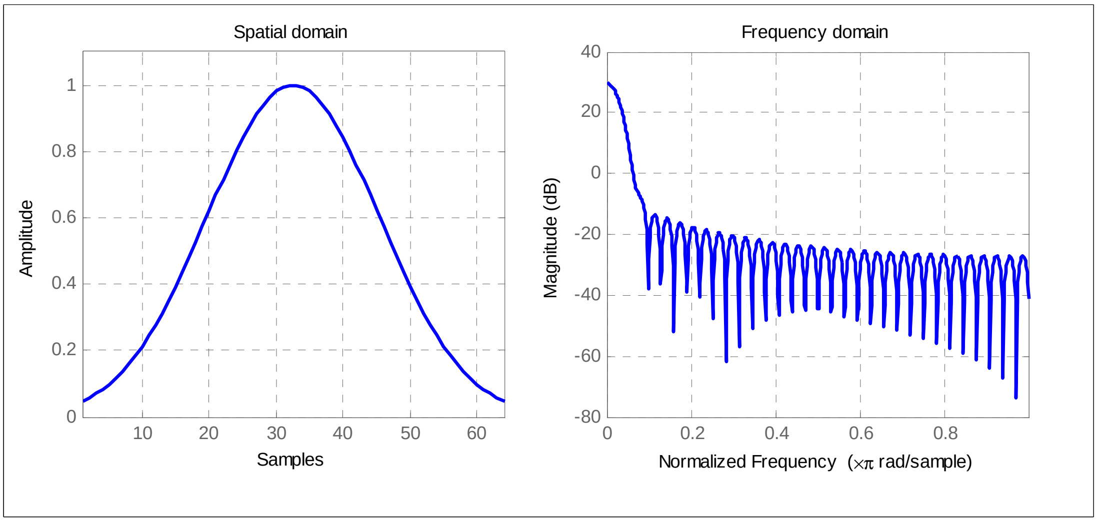
Table 3.1 Windowing Functions

All the functions can be easily transferred to two dimensions and applied to images. For name card images, the boundary effect is not so obvious since they usually have uniform color backgrounds. We can use any of the windowing functions except the Rectangular window. The Rectangular window can create strong artifacts in the frequency domain, while others have approximate same minimal ringing effect.

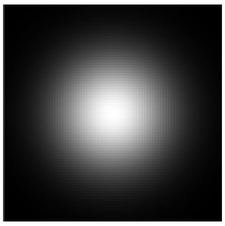
In this algorithm, we use Gaussian mask. The Gaussian mask is a circularly symmetrical or isotropic mask, such that any cross-section through its centre yields a weight profile that has the form of a Gaussian or normal curve. The spatial domain and frequency domain representation is shown in Figure 3.2(a). We can get a 2-D Gaussian mask by multiplying elements of the 1-D mask together,

$$f(n) = \begin{cases} \exp^{(-x^2 - y^2)}, & -(M-1)/2 \le x, y \le (M-1)/2 \\ 0, & otherwise \end{cases}. \tag{3.2}$$

The Gaussian window is shown in Figure 3.2(b). A typical blurred name card image is masked with this Gaussian window and shown in Figure 3.3. The Fourier transform of the blurred image without windowing and with Gaussian window are shown for comparison in Figure 3.4(a), (b). It is clear that the masked spectrum has minimal boundary effects, while the unmasked spectrum has false edge signals at the border of the spectrum.



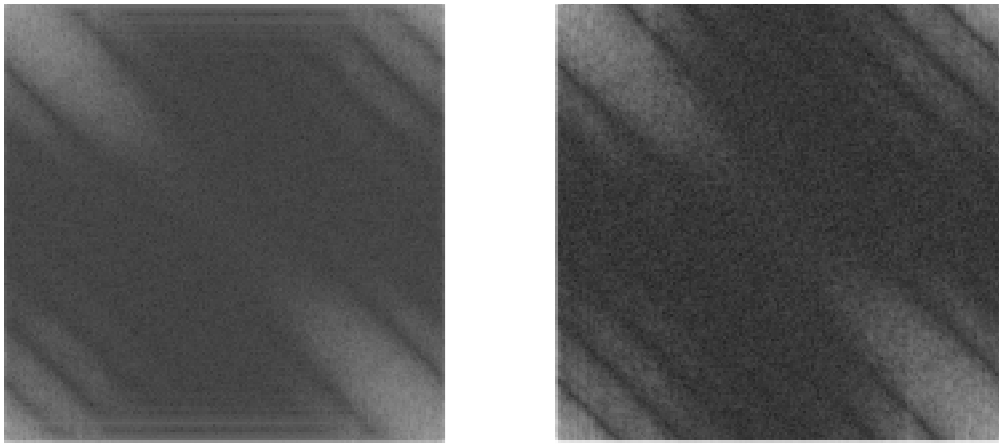
(a)



(b)

Figure 3.2(a) Gaussian Function (b) Gaussian Mask

Figure 3.3 Gaussian Masked Name Card



(a) (b)

Figure 3.4(a) Unmasked Fourier Spectrum (b) Masked Fourier Spectrum

3.1.2 Thresholding

The Fourier spectrum of a blurred image can be viewed as an image itself. In this step, we apply thresholding techniques to the spectrum in order to make the motion blur information clearer to extract, or more specifically, the techniques reduce unnecessary signals for blur orientation estimation.

In Figure 3.4(b), we have seen the spectrum of a blurred image. We first use the MATLAB command *fftshift* to shift zero-frequency component to the center of spectrum in Figure 3.5. It is useful to visualize the smear effect of motion blurs. From the spectrum, we state that the orientation of blur is the direction perpendicular to the smear lines in the spectrum. It is because that the motion blur effectively performs a lowpass on the image in the direction of the blur, even in case of minor image distortion, thus high frequency components diminish significantly in this direction. We also observe that when the blur magnitude is larger, the smear effect is more severe as the ripples occurring in the spectrum are narrower.

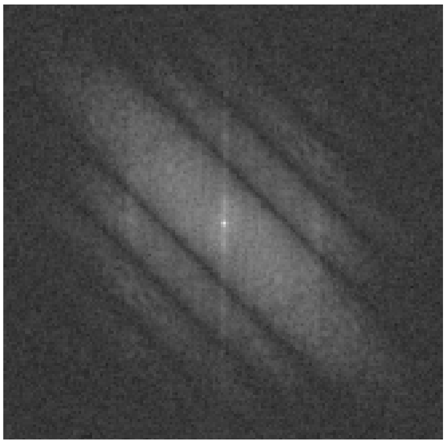
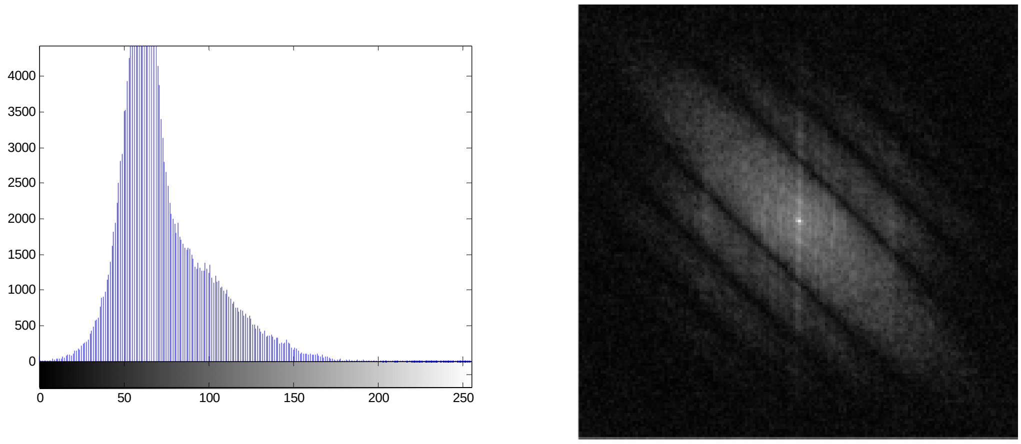


Figure 3.5 Fourier Spectrum with Zero Components Shifted to the Center

An intuitive way to distinguish the smear lines from the dark background is to use contrast stretching. This simple technique attempts to improve the contrast of the image by stretching the range of pixel intensities it contains. The modified intensity of each pixel in the spectrum is defined by

$$p_{new} = scale \times \frac{p_{old} - MIN}{MAX - MIN}, \tag{3.3}$$

where MAX and MIN represent the maximum and minimum pixel values in the spectrum. The problem with this formula is that a single outlying pixel with either a very high or very low value can severely affect the value of MAX or MIN and this could lead to very unrepresentative scaling. We use a more robust approach here by taking a look of the spectrum histogram in Figure 3.6(a). MIN is chosen as the 5th percentile (i.e. 5 percent of the pixels will have a value lower than the MIN) and MAX the 95th in the histogram. Now the scaling is more successful and the smear lines are brighter compared to the dark area as in Figure 3.6(b).



(a) (b)

Figure 3.6(a) Histogram of Fourier Spectrum (b) Fourier Spectrum with Contrast Stretching

After stretching the contrast of the image, our next task is to extract the smear lines from the spectrum. Normally, there will be numbers of parallel lines perpendicular to the blur orientation depends on the blur magnitude. Only the main smear line in the center is used for estimation, others are simply ignored since they do not represent much information. The intuitive idea is to start from the center pixel and gradually expands its surrounding pixels until any pixel falls below the threshold defined by the average pixel values of the spectrum. Thresholding itself is a research field with many efforts [Sezgin and Sankur, 2004]. In this application, we use the simple mean value of the spectrum since our main objective is to remove unwanted information and it does yield satisfactory results.

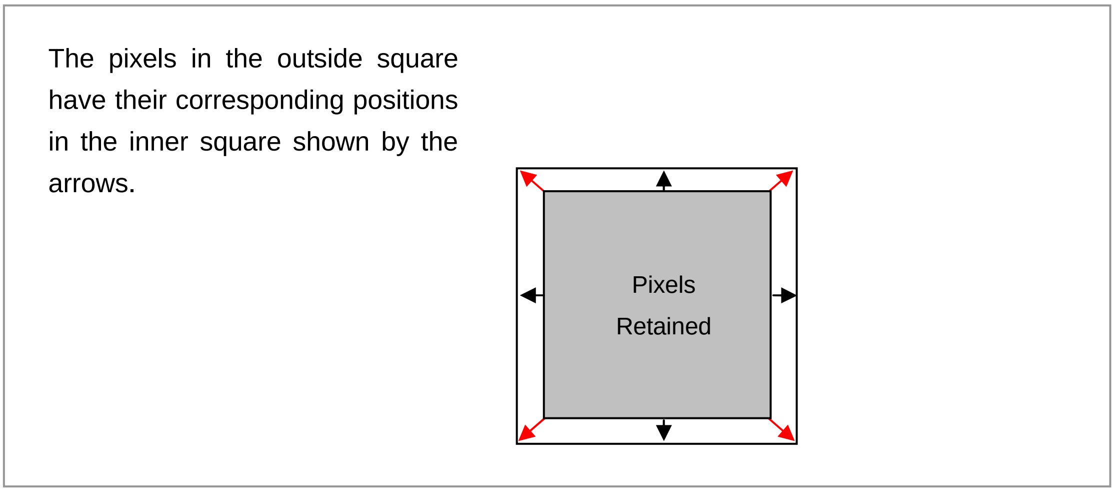


Figure 3.7 Expanding Process

When the average neighbor pixel values of a specific pixel *(i, j)* defined by,

$$\begin{aligned}\hat{p} = 1/8\sum[&p(i-1, j-1) + p(i-1, j) + p(i+1, j) + p(i, j-1) \\ &+ p(i, j+1) + p(i+1, j-1) + p(i+1, j) + p(i+1, j+1)],\end{aligned} \tag{3.4}$$

is below the average of the spectrum, we set $p(i, j) = 0$. In the expanding process, the pixel for calculation first checks its corresponding neighboring pixel in the inner square as in Figure 3.7. If the inside pixel has already been set to 0, then the current pixel is considered to be outside the range of smear lines. Otherwise, we use equation 3.4 to decide whether to retain this pixel for estimation. After the expansion reaches the border, only the spectrum pixel values inside the main smear line are kept while other pixels are set to black. The result of this process can be found in Figure 3.8. In Figure 3.8(a) and 3.8(c), the image is blurred with 4 pixels and 10 pixels magnitude respectively. The success of this method depends on how accurately we extract the smear line from the spectrum. The zigzag structures at the border of extraction may pose problems in the estimation. It is especially severe when the magnitude is small.

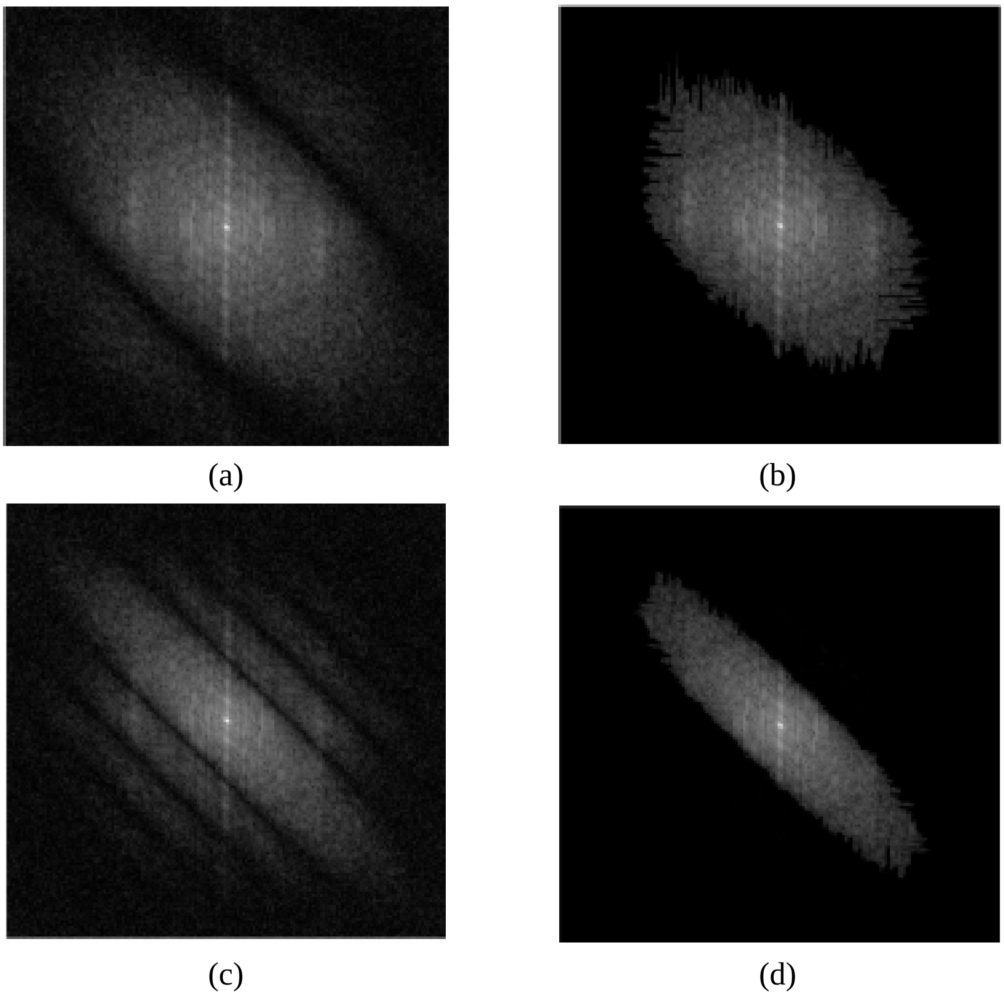


Figure 3.8(a) Fourier Spectrum with Blur Magnitude = 4 pixels (b) Fourier Spectrum after Smear Line Extraction (c) Fourier Spectrum with Blur Magnitude = 10 pixels (d) Fourier Spectrum after Smear Line Extraction

3.1.3 Steerable Filter

Orientated filters are used in many vision and image processing tasks, such as edge detection, segmentation, texture analysis and motion analysis. To find the response of a filter at many orientations, we have to apply many versions of the same filter, each different from the others by small differences in angle. The more efficient approach is to apply some filters corresponding to some angle and interpolate between the responses. “Steerable Filter” is the name to describe such class of filters in which a filter of arbitrary orientation is synthesized as a linear combination of a set of “basis filters" [Freeman and Adelson, 1991].

Gaussian derivatives are useful functions for image analysis and a steerable quadrature pair of them is used here to find the blur orientation in the spectrum. By taking the 2nd derivative of a Gaussian and its Hilbert transform as our bandpass filters, we have

$$E_2(\theta) = [G_2^\theta]^2 + [H_2^\theta]^2 . \tag{3.5}$$

The basis filters and interpolation functions of 2nd derivative of Gaussian and its Hilbert transform are listed in Table 3.2. The 2D view of these functions is shown in Figure 3.9.

$G_{2a} = 0.9213(2x^2 - 1)e^{-(x^2+y^2)}$	$k_a(\theta) = \cos^2(\theta)$
$G_{2b} = 1.843xye^{-(x^2+y^2)}$	$k_b(\theta) = -2\cos(\theta)\sin(\theta)$
$G_{2c} = 0.9213(2y^2 - 1)e^{-(x^2+y^2)}$	$k_c(\theta) = \sin^2(\theta)$
$H_{2a} = 0.9780(-2.254x + x^3)e^{-(x^2+y^2)}$	$k_a(\theta) = \cos^3(\theta)$
$H_{2b} = 0.9780(-0.7515 + x^2)(y)e^{-(x^2+y^2)}$	$k_b(\theta) = -3\cos^2(\theta)\sin(\theta)$
$H_{2c} = 0.9780(-0.7515 + y^2)(x)e^{-(x^2+y^2)}$	$k_c(\theta) = 3\cos(\theta)\sin^2(\theta)$
$H_{2d} = 0.9780(-2.254y + y^3)e^{-(x^2+y^2)}$	$k_d(\theta) = -\sin^3(\theta)$

Table 3.2 Basis Filters and Interpolation Functions

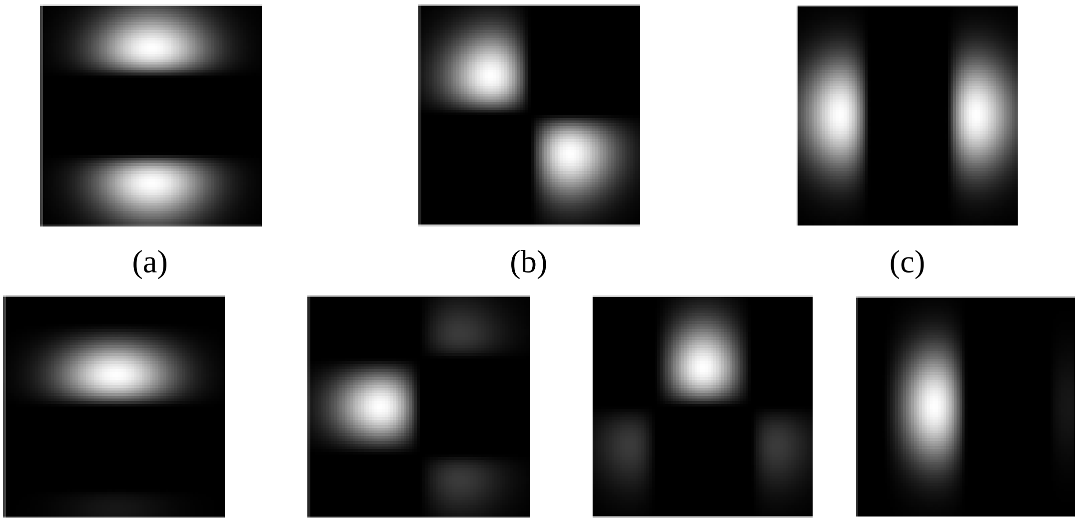


Figure 3.9(a) G_{2a} (b) G_{2b} (c) G_{2c} (d) H_{2a} (e) H_{2b} (f) H_{2c} (g) H_{2d}

To calculate the response of our bandpass filter, we first take the logarithm of the Fourier spectrum *FI*. Then we calculate the response of each of the basis filters for *FI*. We use these responses to create the responses for all the orientations in the range $0° \text{ to } 179°$ in equation 3.6 and 3.7,

$$G_2^\theta = k_a(\theta)\sum\sum FIG_{2a} + k_b(\theta)\sum\sum FIG_{2b} + k_c(\theta)\sum\sum FIG_{2c} \tag{3.6}$$

$$\begin{aligned} H_2^\theta &= k_a(\theta)\sum\sum FIH_{2a} + k_b(\theta)\sum\sum FIH_{2b} + k_c(\theta)\sum\sum FIH_{2c} \\ &+ k_d(\theta)\sum\sum FIH_{2d} \end{aligned} \tag{3.7}$$

The orientation that maximizes equation 3.5 is the local dominant orientation in the image. The measurement of orientation angle was made directly from the basis filter outputs without performing the steering operation. The computation cost for an $n \times n$ image is O(N) if the image size is not large. Now the output angle is used as the input for the magnitude estimation in the next section.

3.1.4 Differential Filter

Real world image is characterized by high spatial correlation. Pixels that are next to each other in an image are highly positively correlated, i.e. bright pixels tend to be next to other bright pixels and dark pixels tend to be next to other dark pixels. Pixels that are close to each other are still correlated, but not as much as next to each other. Name card images exhibit strong correlation properties in the background and the text area. Such property leads to difficulties in the cepstral domain analysis of the blurred image since the overlying structure of the original image in the cepstrum significantly frustrates the identification of the PSF.

To suppress the correlation, we normally use differential operation as the decorrelating filter. The differential operation is simply to replace an original pixel with the difference between the pixel and its adjacent pixel. In the uniform linear blur, the pixels close to each other in the direction of the blur are also correlated because they are mostly imaging the same set of pixels in the original image. Such property shows the characteristics of the blur PSF. To maximally not affect this information, we use the differential filter in the direction perpendicular to the blur orientation. Since we deal with digitized images, the neighboring pixel to $p(i, j)$ in the direction α where $0 \le \alpha \le 45°$ is approximated by intensities from both $p(i, j + 1)$ and $p(i + 1, j + 1)$ as the black frame in Figure 3.10. Other directions are done in the similar way.

The position of the black dashed-line frame between two neighboring pixels depends on the direction α . E.g. When $\alpha = 0°$, p(i’, j’) = p(i, j + 1)

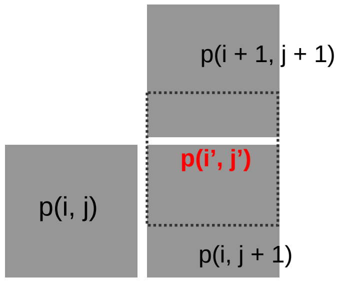


Figure 3.10 Approximation used in Differential Operation

The detailed formula for the differential operation is given in equation (3.8),

$$p(i,j) = \begin{cases} \tan\alpha \times \mathrm{p(i+1, j+1)} \\ + (1 - \tan\alpha) \times \mathrm{p(i, j+1) - p(i, j)} & 0 \le \alpha \le 45° \\ \tan(90° - \alpha) \times \mathrm{p(i+1, j+1)} \\ + (1 - \tan(90° - \alpha)) \times \mathrm{p(i+1, j) - p(i, j)} & 45° < \alpha \le 90° \\ \tan(\alpha - 90°) \times \mathrm{p(i+1, j-1)} \\ + (1 - \tan(\alpha - 90°)) \times \mathrm{p(i+1, j) - p(i, j)} & 90° < \alpha \le 135° \\ \tan(180° - \alpha) \times \mathrm{p(i+1, j-1)} \\ + (1 - \tan(180° - \alpha)) \times \mathrm{p(i, j-1) - p(i, j)} & 135° < \alpha < 180° \end{cases}, \tag{3.8}$$

where *p (i, j)* is the pixel of the blurred image and α is the blur orientation.

As we see in Figure 3.11, after applying the differential operation, the highly correlated background of the original image is almost removed, while the blur characteristics remain intact. This operation is similar to those we apply in the edge detection. It is used mainly to suppress the correlation features from the blurred image here.

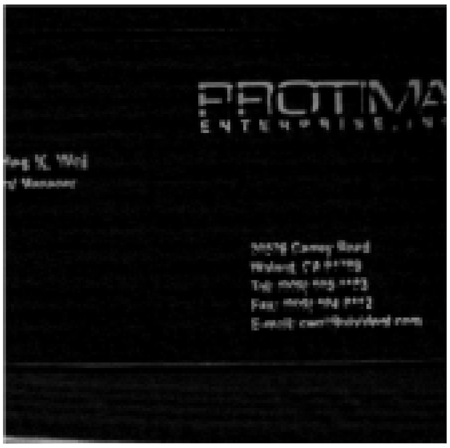
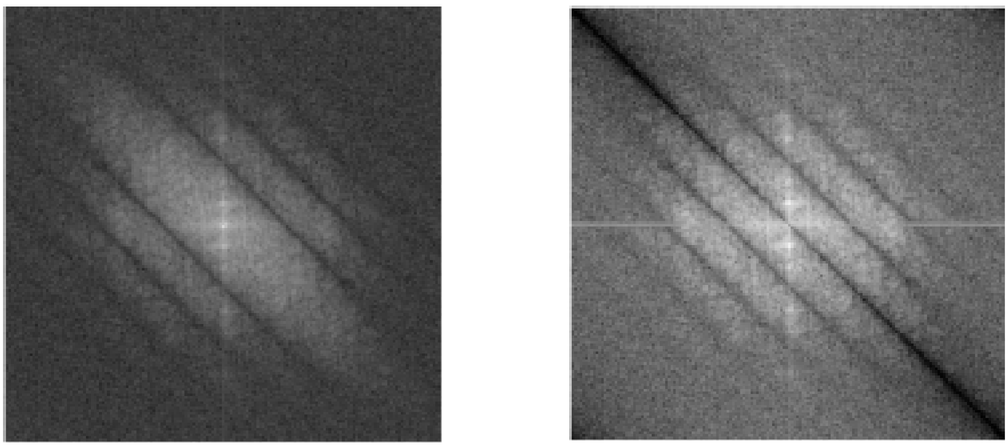


Figure 3.11 Synthesized Blurred Name Card after Decorrelation

The effect in the frequency domain is shown in Figure 3.12. We take the Logarithm of the spectrum of the blurred image without decorrelation and with decorrelation. The differentiation operation tends to whiten the spectrum at a glance. This reduces the dynamic range of the data plotted. The ripples are clearly separated by zeros and show strong periodicity. Notice the central ripple is halved by a ‘black strip’ indicating the differential operation performed.



(a) (b)

Figure 3.12(a) Fourier Spectrum without Decorrelation (b) Fourier Spectrum with Decorrelation

3.1.5 Radon Transform

In the uniform linear motion blur, the PSF is in rectangular shape. Its representation in the frequency domain is the well known *sinc* function as in Figure 2.3. The space between the zero crossings of the *sinc* function is the blur magnitude. The larger the magnitude, the narrower the *sinc* function and more zeros appears. The Fourier transform of the blurred image is the multiplication of the Fourier transform of the original image and the *sinc* function. The result exhibits clear ripples along the blur direction as in Figure 3.12. An intuitive idea is to extract the ripples and collapse them into 1D signal to get the blur magnitude from the underlying periodicity.

Radon transformation can project 2D objects into one line. Let us look at a coordinate system shown in Fig 3.13. The function $g(s,\theta)$ is a projection of $f(x, y)$ on the axis s of θ direction. The function $g(s,\theta)$ is obtained by the integration along the line whose normal vector is in θ direction. The value $g(s,\theta)$ is defined that it is obtained by the integration along the line passing the origin of (x, y)-coordinate. Since the points on the line whose normal vector is in θ direction and passing the origin of (x, y)-coordinate satisfy

$$\frac{y}{x} = \tan(\theta + 90°) = \frac{-\cos\theta}{\sin\theta}, \tag{3.9}$$

we get

$$x\cos\theta + y\sin\theta = 0. \tag{3.10}$$

Similarly, it follows from equation (3.10) that the line whose normal vector is in θ direction and whose distance from the origin is s satisfying the following equation,

$$(x - s\cos\theta)\cos\theta + (y - s\sin\theta)\sin\theta = 0,$$
$$x\cos\theta + y\sin\theta = s. \tag{3.11}$$

Now we can define the Radon Transform with the help of σ function as

$$g(s,\theta) = \int\int_{-\infty}^{\infty} f(x,y)\sigma(x\cos\theta + y\sin\theta - s)dxdy\,, \tag{3.12}$$

where σ is the unit impulse function located at the origin. The Radon Transform has been widely used in line detection algorithm within image processing, computer vision. Another best known application is computerized tomography (CT), a medical diagnostic procedure which yields high contrast images of thin slices of the human body. Radon transformation is used to reconstruct a 2D object from its integral values along the lines of all directions passing the point.

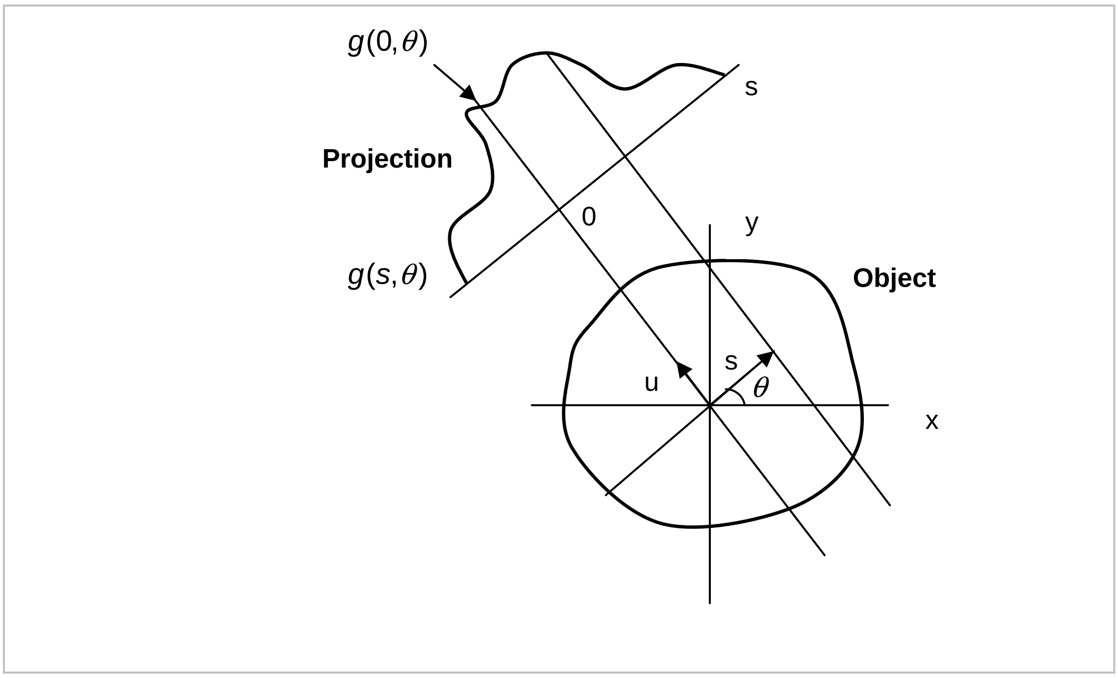
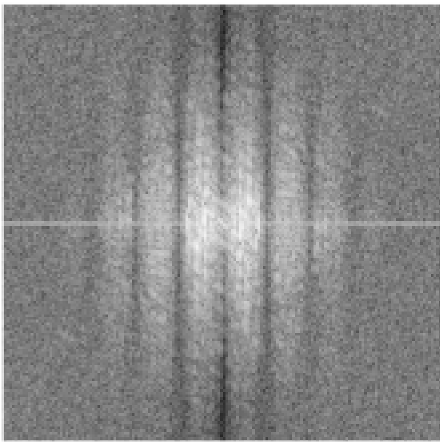
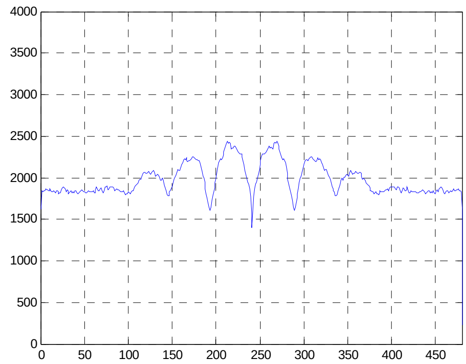


Figure 3.13 Radon Transform

To use Radon transformation in our algorithm, we just need to apply this formula in the direction of the blur. The resulting 1D signal represents the projection of the 2D

Fourier spectrum. The central slice theorem states that the Fourier transform of the projection is identical to the spectrum of the original object on a plane normal to the direction of the projection plane. So the 1D signal can be used as an approximation to the 1D PSF in frequency domain. The correctness of the approximation is decided by two factors, i.e. the windowing effect even after the use of Gaussian mask window and the overlying structure of the Fourier transform of the original image even after decorrelation. The 1D signal is shown in Figure 3.14. The boundary values are removed.





(a) (b)

Figure 3.14(a) Fourier Spectrum (b) Projected 1D Spectrum in Blur Orientation

3.1.6 Cepstral Domain Analysis

As we have transformed the logarithm of the Fourier spectrum of the blurred image into a 1D signal in step 2, we can apply classic cepstral domain analysis to estimate the length of the ripple, i.e. the blur magnitude.

The 1D signal has different lengths under different projection angles. We truncate 1D signal to uniform length since the values at two sides are less important. The uniform length is the width of the image here. Then we take the real part of the Fourier transform of the 1D signal as the definition of cepstrum. We need to identify the correct peak to estimate the blur magnitude. The approximated 1D cepstrum of a blurred image with 21 pixel blur magnitude is shown in Figure 3.15.

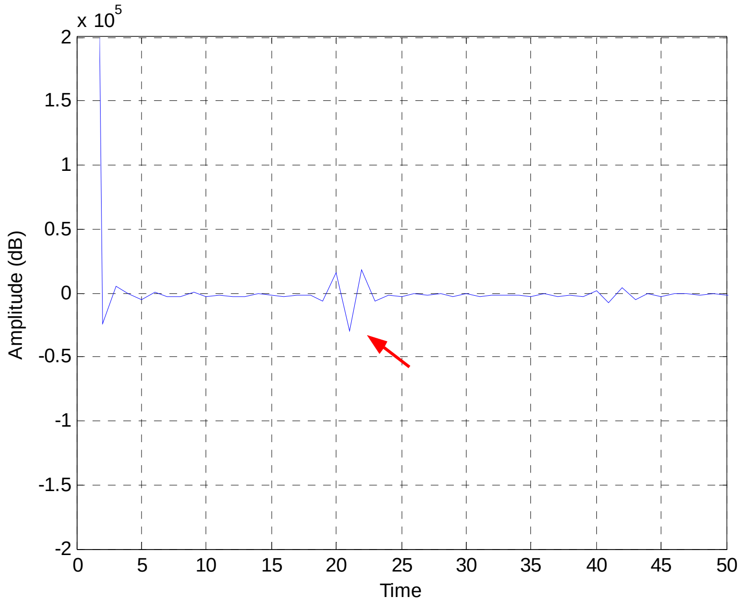


Figure 3.15 Cepstrum with a Local Negative Peak

The desired peak in the cepstral sequence is always accompanied by its replicas (another peak occurs in position 42 with smaller amplitude) and false pulses from the original image. As we analyze Figure 3.15, the desired peak is labeled with the red arrow. The spikes in the first few positions of the cepstrum are caused by the DC component. The cepstral sequence tends to oscillate around zero amplitude until suddenly a local peak appears. Our peak searching process is to identify this local peak within the possible magnitude range. This is simply done by recording the even and odd positions of the cepstral sequence. If we find the pulse in any odd position is smaller than the previous odd position and next odd position, then it is identified as a negative peak. Similarly, if the pulse in any even position is larger than the previous even position and next even position, then it is identified as a positive peak. Notice sometimes this peak appears as the global peak sometimes not, so we can not simply take the largest absolute value within the range. Finally, this identified position (positive peak or negative peak) is estimated as the blur magnitude.

3.1.7 Complexity Analysis

The complexity of this algorithm mainly depends on the size of the blurred image. We define N = Width $\times$ Height in the following analysis. For angle estimation, Gaussian masking in the 1st step takes the multiplication of the blurred image and the mask. This is O(N) time. In the 2nd step, we apply thresholding to the Fourier spectrum. The spectrum is of the same size as the blurred image, so it is O(N). In the 3rd step, we calculate the maximum response for the band filter. This is O(N) for 7 basis filters and O(1) for each angle. The most computational expensive part is the Fourier transform. By using FFT, we have O(N log N) time complexity. For distance estimation, decorrelation filtering in the 1st step replaces each pixel with the difference between adjacent pixels. This is O(N) time. In the 2nd step, Radon transform performs on the known projection angle, so it is O(N). In the 3rd step, we calculate the cepstrum of the projected signal. This is O(N log N). Note that we cancel the Zero padding step in I.Rekleitis algorithm. Zero padding increases N by 4-fold and dramatically increases the computational time.

The total deblurring process including blur parameter estimation and inverse filtering on a 640 $\times$ 480 image is less than **5** seconds. The experiment is done on a laptop with CPU Pentium IV 2.66 GHz, memory 512 Mb RAM. The software used is MATLAB 6.50, R13.

3.2 Blur Estimation Procedure for Uniform Acceleration Motion Blur

The previous section has described a motion deblurring method assuming uniform linear motion blur. Such an assumption is usually not valid in real world motion blurred images. In this section, we look into the more complex motion – uniform acceleration motion blur. The estimation procedure has used the result from the algorithm in section 3.1.

3.2.1 Mathematics Background

The relative motion between image and camera during exposure can be expressed in terms of the rectangular spread function and the corresponding *sinc* transfer function. It is valid only in the uniform linear motion case. For the general case, i.e. motion with non-uniform velocity, S.C.Som [Som, 1971] has derived an expression for the optical transfer function (OTF). The expression is given by

$$L(u) = \int_{-\infty}^{\infty} l(s)\exp(-i2\pi us)ds\,, \tag{3.13}$$

where

$$l(s) = \begin{cases} 1/[Tg(s)] & 0 \le s \le x \\ 0 & otherwise \end{cases}. \qquad (3.14)$$

The function *l(s)* represents the effective spread function, in which *g(s)* represents the velocity of smear at *s*, *s* is the extent of smear at time *t* from the start and *x* is the maximum displacement.

In the uniform linear motion case, *g(s)* is the constant velocity. Equation (3.14) becomes

$$l(s) = \begin{cases} 1/x & 0 \le s \le x \\ 0 & otherwise \end{cases}. \qquad (3.15)$$

Equation (3.15) agrees with the known result. It only represents motion on one side of the image, so the rectangular pulse is asymmetric. From equation (3.13), we get the transfer function as

$$L(u) = \frac{1}{x}\int_0^x \exp(-i2\pi us)ds = \sin c(\pi ux)\exp(-i\pi ux)\ . \qquad (3.16)$$

Again equation (3.16) is consistent to the known as we discuss in chapter 2, except the additional phase factor. In most cases, the shift is of no importance.

In the case of smear due to uniform acceleration, the displacement is given by

$$s = V_0 t + \frac{1}{2}at^2\ , \qquad (3.17)$$

together with

$$V = V_0 + at\ , \qquad (3.18)$$

we get

$$g(s) = (V_0^2 + 2as)^{1/2}\ . \qquad (3.19)$$

Consequently from equation (3.13), the spread function is

$$l(s)=\begin{cases}\dfrac{1}{T(V_0^2+2as)^{1/2}} & 0\le s\le x \\ 0 & otherwise\end{cases}, \quad (3.20)$$

where a is the uniform acceleration and V_0 is the initial velocity. We make the assumption that a and V_0 are in the same direction here, and a is always positive.

The transfer function is complex as we do not show here. The phase factor can not be neglected since the spread function is not uniform over the extent of smear. For each frequency, the net phase of the transfer function is a *nonlinear* function of the frequency. This indicates that phase distortion is an additional image degradation we should take into consideration in the uniform acceleration motion blur.

As orientation and magnitude in the uniform linear case, the ratio

$$R=V_0^2/a\ . \quad (3.21)$$

is an important parameter in the analysis of uniform acceleration case. When R is very large, i.e. $V_0 >> a^{1/2}$, the spread function will be close to uniform linear. On the other hand, when R is small, i.e. $V_0 << a^{1/2}$, the difference is obvious. We can normalize equation (8) by using R and $l(0)$,

$$l(s)=[1+(2s/R)]^{-1/2} \quad 0\le s\le x\ . \quad (3.22)$$

3.2.2 Creation of Uniform Acceleration Motion Blur

MATLAB does not have built-in functions for uniform acceleration motion blur, we need to create our own blur kernels. The spread function is first generated using equation (3.22) then rotated to the blur orientation using bilinear interpolation. Since the smallest unit of digitized image is pixel, the displacement s is in pixels. We plot the PSF of uniform acceleration motion blur in Figure 3.16.

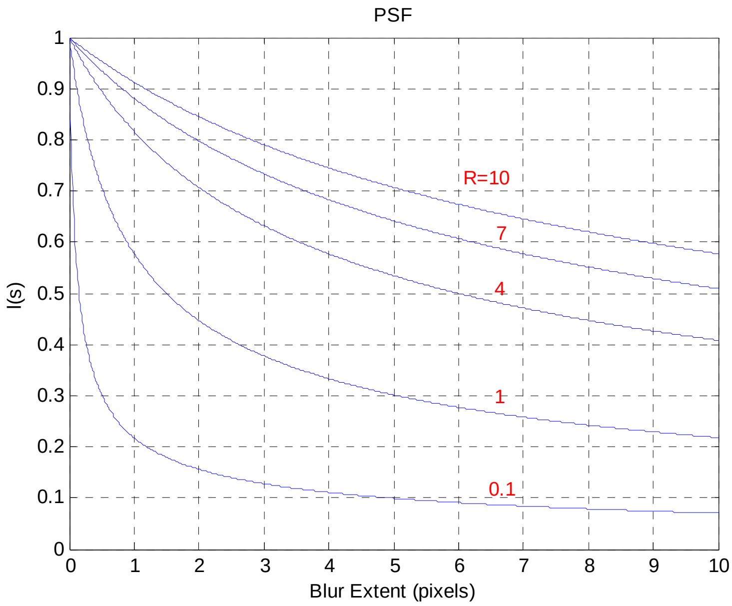


Figure 3.16 PSFs of Uniform Acceleration Motion Blur

From Figure 3.16, we can see that when R is larger, the wave gradually becomes square shape. The uniform acceleration motion blur is closer to linear. A normalized blur kernel created with blur extent = 10 pixels, blur orientation = $45°$ and R = 0.10 is shown in Table 3.3. This kernel is used to convolve with the original image to obtain the blurred image.

0	0	0	0	0	0	0	0.0102	0
0	0	0	0	0	0	0.0195	0.0643	0.0102
0	0	0	0	0	0.0213	0.0696	0.0195	0
0	0	0	0	0.0236	0.0763	0.0213	0	0
0	0	0	0.0268	0.0856	0.0236	0	0	0
0	0	0.0320	0.0990	0.0268	0	0	0	0
0	0.0417	0.1235	0.0320	0	0	0	0	0
0.0316	0.1819	0.0417	0	0	0	0	0	0
0	0.0316	0	0	0	0	0	0	0

Table 3.3 Blur Kernel of Uniform Acceleration Motion Blur

The modulus of the normalized transfer function is defined as modulation transfer function (MTF). MTF is the spatial frequency response of an imaging system or a component; it is the contrast at a given spatial frequency relative to low frequencies. Spatial frequency is typically measured in cycles or line pairs per millimeter, which is analogous to cycles per second (Hertz) in audio systems. High spatial frequencies correspond to fine image detail. The more extended the response, the finer the detail and the sharper the image.

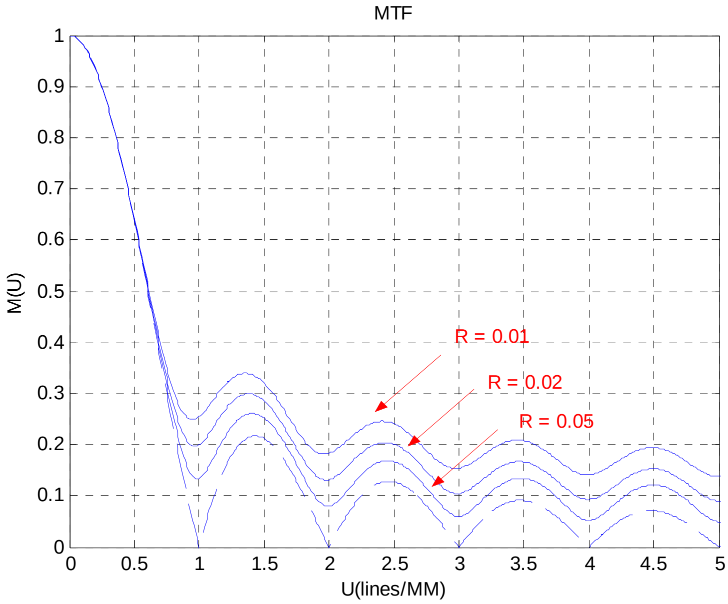


Figure 3.17 MTF

As is Figure 3.17, the modulus of the transfer function increases with decreasing *R*. It implies for the same amount of smear, the image resolution should be better when the acceleration is greater than the initial velocity. Another apparent result in Figure 3.17 shows that the MTF due to uniform acceleration is equal or greater than that due to uniform velocity (dash line) in all *R*. It implies that for the same amount of smear, blur due to uniform acceleration causes less distortion to images compared to that due to uniform velocity. But in the former case, phase of the transfer function (PTF) leads to additional degrading effects as shown in Figure 3.18. Finally, we can see the zero crossings no longer exist in the MTF of uniform acceleration, so the usually method of blur estimation discussed in chapter 2 on uniform linear blur by measuring the difference between adjacent zero crossings will not work.

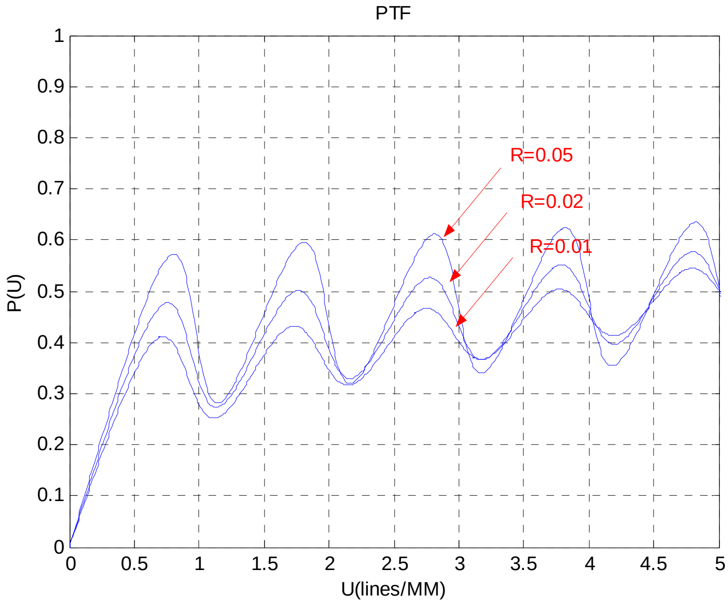


Figure 3.18 PTF

A comparison of uniform acceleration blurred image and uniform linear blurred image is shown in Figure 4. Figure 3.19(a)(b)(c) is blurred with R = 0.1, 10, infinity respectively. (d) is blurred with the same orientation and extent but uniform linear motion blur. As we observe, when R tends to infinity, acceleration blurred image is close to linear one. When R is small, blurred image shows much higher resolution then others.

(a)

(b)

(c)

(d)

Figure 3.19(a) Synthesized Uniform Acceleration Motion Blurred Image with R = 0.1 (b) R = 10 (c) R = infinity (d) Synthesized Uniform Linear Motion Blurred Image with Same Blur Orientation and Extent

3.2.3 Estimation Procedure

In Figure 3.17, we observe the MTF of uniform acceleration blur also exhibits strong periodicity of the blur extent. Periodicity diminishes when R approaches 0, i.e. the acceleration is at infinity. However, in this case the image is virtually not blurred. In real world, acceleration is always bounded in a range, thus we define the possible values of R within [0.1, 50] here. Such assumption is reasonable as the PSF shape usually varies in a small margin when R approaches two ends of this range. We can consider the motion blur is lowly accelerated when R is at the higher end and highly accelerated when R is at the lower end. The final conclusion here is that the methods for uniform linear motion blur works on acceleration case as well, except that we need to estimate the possible range of *R*.

PSF of acceleration blur is usually asymmetric. For the same blur extent, we define two different acceleration PSFs as in Figure 3.20. *R* is 0.1 is (a)(c) and 50 in (b)(d). Since the spread function is not uniform over the extent of the smear, we have both ‘forward’ and ‘backward’ acceleration depends on how the image is blurred. Note that the PSFs in Figure 3.20 have yet to be normalized. To determine the PSF curve, we make use of autocorrelation functions (ACF). The autocorrelation function *K(n)* of an *M* pixel image line *l* is defined as,

$$K(n) = \sum_{i=-M}^{M} l(i+n)l(i) \quad n \in [-M, M], \tag{3.23}$$

where *l(i)* = 0 outside the image line range. Equation 3.23 describes how pairs of pixels at particular displacements from each other are correlated. It is high where they are well correlated and low where poorly correlated. For a normal image, the ACF will be some function of distance from the origin plus random noises. But for a motion blurred image, the ACF will decline much more slowly in the direction of the blur than in other directions. This is because pixels those are close together in the blurred direction are mostly imaging the same real world points. Obviously, when acceleration is higher namely *R* is larger, ACF will decline faster in the blur direction since the difference between neighboring pixels increases.

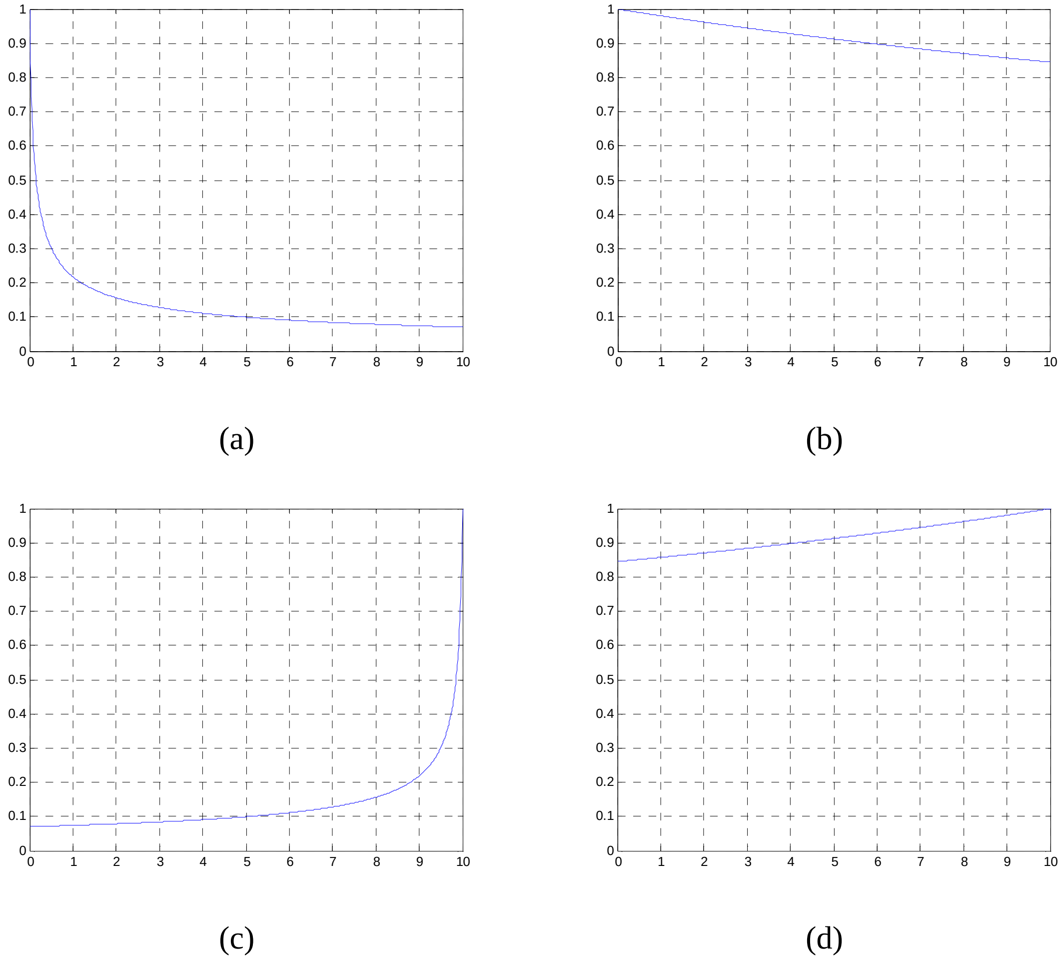


Figure 3.20(a) Forward Highly Accelerated PSF (b) Forward Lowly Accelerated PSF (c) Backward Highly Accelerated PSF (d) Backward Lowly Accelerated PSF

To sum up, our estimation procedure on real world blurred images is as follows:

1. Apply algorithm for uniform linear motion blur to estimate the blur orientation and extent.
2. Use the average autocorrelation function of the image lines in the blur orientation to adjust *R*.
3. Create both forward acceleration and backward acceleration PSF based on the estimated parameters in 1, 2.
4. Use the created PSF in 3 to restore the blurred image. If the restoration from

both forward acceleration and backward acceleration shows clear inverted scene imposed on the true scene, the image is blurred by a symmetric PSF, i.e. uniform linear motion blur; otherwise use the PSF that yields better OCR results.

5. (Optional) Adjust the parameter in Wiener filter for better OCR results.

3.3 Wiener Filter

As the blur estimation is done, we use image restoration filter to deblur. The simplest restoration filter is inverse filter. This linear filter is given by the inverse of the blurring function. Unfortunately, several problems exist with this approach. Firstly, as in uniform linear motion blur, the frequency response of the PSF is 0 at some frequencies. The inverse does not exist. Even the frequency response does not go to 0 in uniform acceleration motion blur, there are problems caused by excessive noise amplification at high frequencies. The reason is that the power spectrum of the blurred image is higher at low frequencies and lower at higher frequencies by the lowpass nature of the PSF. The spectrum of the noise, on the contrary, typically contains more high frequency components. Thus at high frequencies, e.g. the characters in the name cards, the restored image is dominated by the inverse of noises.

A number of restoration filters have been developed to overcome the noise sensitivity in inverse filtering. They are collectively called least square filters. The most common one is Wiener filter, that incorporates both the point spread function and statistical characteristics of noise into restoration process. The objective of the filter is to find an estimate *F'* of the original image *F* such that the mean square error given by equation (3.23) is minimized.

$$Er = E(|F(u,v) - F'(u,v)|^2) . \tag{3.24}$$

The solution to this minimization problem is given by

$$F'(u,v) = \frac{H^*(u,v)}{|H(u,v)|^2 + \frac{P_n(u,v)}{P_f(u,v)}} G(u,v) , \tag{3.25}$$

where P_n and P_f denote the power spectra of the noise and the original image respectively and $H^*(u,v)$ is the complex conjugate of the PSF. Since P_f is rarely known, the expression is approximated by

$$F'(u,v) = \frac{H^*(u,v)}{|H(u,v)|^2 + C} G(u,v) , \tag{3.26}$$

where *C* is a specified constant. When *C* is 0, Wiener filter approximates the inverse filter.

Restoration errors of Wiener filter have 2 major causes. One large error component, called Edge error [K.C.Tan, H.Lim and B.T.G.Tan, 1991] that arises due to the fact that real images seldom have the periodicity assumed by DFT. The error occurs at the boundary of the restored image. We blur the edges of motion blurred image using Gaussian PSF. The output image is the weighted sum of the original blurred image and its newly-blurred version. This operation tapers the discontinuity along the edge of the images and helps to reduce the Edge error.

The deviation of the Wiener filter from the exact inverse at frequencies where the exact inverse tends to large values is the second cause. We can describe the error in the following equations.

$$F'(u,v) = L(u,v)[H(u,v)F(u,v) + N(u,v)]\,, \tag{3.27}$$

where *L(u,v)* is the frequency response of Wiener filter and *N(u,v)* is the noise term. By mathematics manipulation, we have

$$F'(u,v) - F(u,v) = [L(u,v)H(u,v) - 1]F(u,v) + L(u,v)N(u,v)\,, \tag{3.28}$$

The first term on the right side of equation (3.27) is image dependent and the second term is noise dependent. If we assume noise is negligible in the restoration, ringing artifacts present in the restored image mainly come from the first term. In the case where *H(u,v)* has periodic zeros, the first term evaluates to *–F(u,v)*, which in turn leads to negative echo of the intensity transition in the restored image, that is, ringing artifacts.

Two possible solutions of reducing ringing effects are proposed. The first method incorporates deterministic *a prior* knowledge of the original image. The second method adaptively regulates the noise magnification and regulation errors on the local edge content of the image. Tekalp *et al.* [Tekalp, Kaufman and Woods, 1989] describe a multiple image model Kalman restoration filter in which a number of image models are used to filter an image in agreement with the local edge orientations. Lagendijk *et al.* [Lagendijk, Biemond and Boekee, 1988] propose a regularized iterative image restoration algorithm by making use of the theory of the projections onto convex sets and the concept of norms in a weighted Hilbert space.

Since our ultimate goal is to improve the OCR results for name card images but not perfect restoration, we only need to minimize restoration errors to tolerable extent. Noise is assumed to be negligible, so our focus is on the regulation errors occurring around high frequencies, especially characters in the blurred name card. By regularizing these regions less strongly, i.e. constant C in equation (3.25) is smaller, the local regularization error and hence the severity of the ringing is reduced. The resolution is enhanced at the same time. Experiments will be performed in Chapter 4. It shows that C = [0.005, 0.01] yields the best results.

To sum up, in this chapter we have presented a practicable motion deblurring procedure for document images. The performance of the algorithm will be evaluated in the next chapter.

4 Experiments

In this chapter, the proposed algorithm is evaluated on both synthetic and real world motion blurred name cards. OCR results are presented to measure the overall performance for real world blur when the actual PSF function is unknown.

I.Rekleitis' algorithm can obtain an approximation of the optical flow map for a single blurred image in most cases but without accuracy. This makes the algorithm unsuitable for applications like deblurring name cards, which needs an accurate estimation of the motion blur PSF. We have defined the algorithm requirements in section 2.4.2, so the experiments in this chapter will demonstrate how our algorithm fulfills the requirements.

4.1 Synthetic Motion Blurred Images

We use name cards provided by Hot Card Company as our experiment source. There are in total over 100 name cards, some of them are acquired with possible blur. In this section, we use name cards that do not have degradations initially and add on intended blur ourselves. Algorithm in section 3.1 is used to estimate the two blur parameters, i.e. the blur orientation in degrees and the extent in pixels, and compare the findings with the actual value.

First, we perform the blur angle estimation. The error analysis that is presented in the next few pages is created using the following methods. We first blur the name cards in the orientation from 0° to 89° at a given blur magnitude. We only test half range of the blur orientation because the symmetric properties of PSF. The blurred images are inputted to our algorithm and we record down the result for each orientation. We run the algorithm in two variations. The first variation does not use thresholding techniques on Fourier spectrum. The second uses the complete algorithm. For each experiment, there exist five measurements of the error. The first is the mean value of the total number of errors in the angle estimation within given range. This gives us the idea whether this algorithm works well in certain ranges. Since there are positive and negative errors, the second measurement is the mean value of the absolute error. This is the most important measurement as we know how the angle estimation errors are going to affect the accuracy of OCR software The third measures the standard deviation and the fourth and fifth gives the max and min error respectively. Max and min error is worth noting because it shows the largest error that could occur. All the errors are measured in degrees. We have more than 100 name card images in hand. To evaluate the performance on each of them requires time. Without the loss of generality, we pick 10 images randomly each time to average the errors in all the experiments.

In the first experiment, we blur the group of name cards using two methods, i.e. a simple antialiasing convolution matrix shown in Table 4.1 and MATLAB function *fspecial*. In the real world, the blur is created before digitization. To reproduce the result in the discrete space, we need an accurate blur kernel. Since we want to know whether the implementation of the MATLAB functions resembles the true blur function, we use a common blur creation method to compare the results. The name card is blurred with 10 pixels magnitude.

0.5	1	0.5	0	0	0	0	0	0	0
0	0.5	1	0.5	0	0	0	0	0	0
0	0.5	1	0.5	0	0	0	0	0	0
0	0	0.5	1	0.5	0	0	0	0	0
0	0	0	0.5	1	0.5	0	0	0	0
0	0	0	0	0.5	1	0.5	0	0	0
0	0	0	0	0.5	1	0.5	0	0	0
0	0	0	0	0	0.5	1	0.5	0	0
0	0	0	0	0	0	0.5	1	0.5	0
0	0	0	0	0	0	0.5	1	0.5	0
0	0	0	0	0	0	0	0.5	1	0.5

Table 4.1 Simple Antialiasing Convolution Matrix

ERROR ESTIMATIONS	SIMPLE ANTIALIASING MATRIX					MATLAB FUNCTION FSPECIAL				
	0-17	18-35	36-53	54-71	72-89	0-17	18-35	36-53	54-71	72-89
Mean	2.17	-2.20	-1.09	-2.79	-3.39	2.44	2.72	0.28	-3.01	-2.11
|Mean|	**3.23**	**3.44**	**2.29**	**3.65**	**3.28**	**2.44**	**2.94**	**0.95**	**3.11**	**2.11**
Max	7	6	4	5	5	5	6	3	1	0
Min	-4	-7	-5	-7	-9	0	-2	-2	-5	-4
S.Dev	3.34	3.14	1.77	3.25	4.19	1.60	1.61	0.76	2.03	1.17

Table 4.2 Blur Orientation Error Estimation for Name Card Images with Two Different Artificial Blur Creation Methods

By observing the results, we notice the MATLAB blur function returns much better estimation in almost every range. It proves that this function is superior to common blur creation method. So we will use the 2nd approach as our synthesizing method in the later experiments.

In the second experiment, we blur the group of name cards with 4, 10, 16 pixels magnitude respectively. The objective of this experiment is to compare the algorithm results under different blur magnitudes. As we state in section 2.4, we choose the smallest magnitude as 4 pixels here because any magnitude less than 4 will minimally affect the OCR results. The magnitudes here can be considered as light, medium and severe uniform linear motion blur.

ERROR ESTIMATIONS	GAUSSIAN MASK					GAUSSIAN MASK WITH THRESHOLDING				
	0-17	18-35	36-53	54-71	72-89	0-17	18-35	36-53	54-71	72-89
Mean	5.57	-1.44	-1.06	-2.89	4.39	4.97	-2.87	-0.83	-1.27	-3.28
|Mean|	**5.57**	**3.64**	**3.06**	**3.67**	**4.39**	**4.97**	**3.37**	**1.17**	**3.49**	**3.59**
Max	9	6	5	4	-3	6	6	2	4	1
Min	1	-3	-4	-5	-7	2	-2	-2	-5	-4
S.Dev	4.26	3.14	1.78	2.28	1.76	1.37	2.34	1.03	2.36	2.25
Blur Magnitude = 4 pixels										

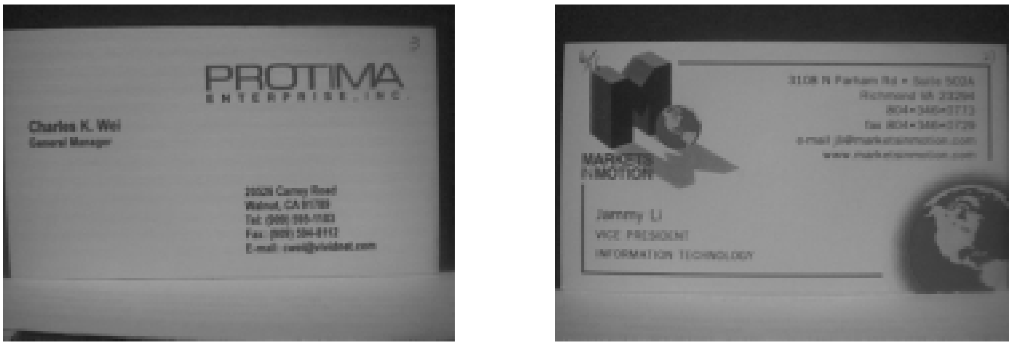
ERROR ESTIMATIONS	GAUSSIAN MASK					GAUSSIAN MASK WITH THRESHOLDING				
	0-17	18-35	36-53	54-71	72-89	0-17	18-35	36-53	54-71	72-89
Mean	3.72	2.75	0.27	-1.05	-3.57	2.44	2.72	0.28	-3.01	-2.11
|Mean|	**3.72**	**3.23**	**2.34**	**3.25**	**3.57**	**2.44**	**2.94**	**0.95**	**3.11**	**2.11**
Max	8	6	4	2	-7	5	6	3	1	0
Min	1	-1	-4	-4	-3	0	-2	-2	-5	-4
S.Dev	3.21	2.52	1.23	2.01	2.16	1.60	1.61	0.76	2.03	1.17
Blur Magnitude = 10 pixels										

ERROR ESTIMATIONS	GAUSSIAN MASK					GAUSSIAN MASK WITH THRESHOLDING				
	0-17	18-35	36-53	54-71	72-89	0-17	18-35	36-53	54-71	72-89
Mean	3.61	3.06	0.39	-1.78	3.39	1.83	1.89	0.61	-2.22	-2.06
|Mean|	**3.61**	**3.17**	**2.28**	**2.32**	**3.39**	**1.83**	**1.89**	**0.92**	**2.33**	**2.06**
Max	6	5	6	3	-1	3	3	4	1	0
Min	1	-1	-4	-4	-6	0	0	-1	-3	-5
S.Dev	1.74	1.42	1.32	1.36	1.56	0.60	0.96	0.93	1.07	1.76
Blur Magnitude = 16 pixels										

Table 4.3 Blur Orientation Error Estimation for Name Card Images Blurred with Magnitude 4, 10 and 16 pixels

By observing the results, we notice that when blur magnitude is larger, the algorithm tends to work better. This is easily verified in the image spectrum where larger magnitudes lead to more severe lowpass effect. The smear lines are narrower with clearer orientation in larger blur magnitudes. We read unusually big standard deviations when the blur magnitude is 4. This is due to single large error occurred in the error estimation. When the magnitude is larger, standard deviations are smaller. This means the algorithm is more reliable with larger blur magnitudes. The thresholding techniques on the spectrum prove to be useful in all the blur orientations. The average error improves by considerable margin in almost all magnitudes compared to use Gaussian mask only. We find that the algorithm works extremely well in the range $35° \text{ to } 53°$. We explain this interesting observation by arguing that our extraction of smear lines in this range is very successful. The maximum error of using Gaussian mask only is $10°$, while the maximum error with additional thresholding is $6°$. The average error is all below $5°$ under all the magnitudes. This is acceptable for OCR software as we show in section 2.4. Note that previous research work states that I.Rekleitis' algorithm on angle estimation returns unacceptable large errors in certain range of orientations. This phenomenon disappears in our algorithm instead the error is randomly distributed in all orientations.

In the third experiment, we blur two groups of name card images with 12 pixels magnitude. The first group of images is taken with fewer textures while the second group with complicated backgrounds. Typical name cards in these two groups are shown in figure 4.1.



(a) (b)

Figure 4.1 (a) Name Card with Plain Background (b) Name Card with Complex Background

The objective of this experiment is to compare the algorithm results under images with different extent of textures.

ERROR ESTIMATIONS	GAUSSIAN MASK					GAUSSIAN MASK WITH THRESHOLDING				
	0-17	18-35	36-53	54-71	72-89	0-17	18-35	36-53	54-71	72-89
Mean	3.22	-1.07	-1.72	-2.97	-3.72	2.67	-2.65	-0.83	2.09	-2.28
|Mean|	**3.22**	**3.27**	**2.73**	**3.06**	**3.72**	**2.67**	**2.77**	**0.96**	**2.49**	**2.39**
Max	6	4	3	1	-3	6	3	1	4	1
Min	0	-4	-5	-6	-6	0	-5	-3	-2	-4
S.Dev	2.34	3.14	1.58	2.17	1.39	1.91	1.54	0.91	1.66	1.21

Table 4.4 Blur Orientation Error Estimation on Name Card with Plain Background Blurred with Magnitude 12 pixels

ERROR ESTIMATIONS	GAUSSIAN MASK					GAUSSIAN MASK WITH THRESHOLDING				
	0-17	18-35	36-53	54-71	72-89	0-17	18-35	36-53	54-71	72-89
Mean	3.77	-3.72	-3.06	-2.89	-4.39	2.76	-1.78	-2.27	-2.61	-2.27
|Mean|	**3.77**	**4.05**	**3.06**	**3.67**	**4.39**	**2.93**	**2.78**	**2.27**	**2.61**	**2.50**
Max	7	2	0	2	-1	5	4	0	0	1
Min	0	-7	-4	-6	-7	-2	-3	-4	-5	-5
S.Dev	3.34	3.54	0.78	2.28	1.56	1.89	1.23	0.87	1.74	1.25

Table 4.5 Blur Orientation Error Estimation on Name Card with Complex Background Blurred with Magnitude 12 pixels

By observing the results, we notice when the name card has more textures, the result is less accurate. Though more textures create more information in the frequency domain, it is not favorable to the orientation analysis of the spectrum. The method using Gaussian mask only is more vulnerable to the complicatedness of textures in the name cards. Their average error degrades by a larger margin compared to the method with thresholding techniques. This is due to the nature of our bandpass filter. In the process of extracting smear lines, we treat the spectrum as in spatial domain. Information with no orientation energies has been removed. However, we also remove useful frequency domain components. The error estimate depends primarily on the successfulness of extraction. This can be seen in the range $35° \text{ to } 53°$. When the smear lines are difficult to extract, our algorithm suffers. However, even in the case of complicated textures, our algorithm returns satisfying result. We still have improvements in almost all the orientations over the method without extraction. The average mean error is less than $3°$ with a 12 pixels blur magnitudes.

One concern is that certain extreme error in angle estimation does occur. Since magnitude estimation relies on the accuracy of angle estimation, the correctness of the estimated angle is of uttermost importance to us. As we have kept the average error below 5 degrees, the magnitude estimation shown in the later experiment already achieves satisfactory results. Unfortunately certain intolerable error can totally fail the estimation. Since the angle estimation algorithm shows random distributed errors in different blurred images, we can not predict where such errors could occur.

One way to handle usually large error in the angle estimation is to use averaging. We break the blurred images into several smaller sections and apply full algorithm on each section. The averaged estimated angle from all the sections is taken as the final result. The reason we do not include this process into the algorithm is that it slow down the speed of the estimation. We have another concern on the nature of name cards. Since name card images usually have uniform backgrounds, blur information is hardly to detect if the chosen section does not contain any textures. A compromised solution is to choose the central 4 subsections of the blurred image as in Figure 4.2.

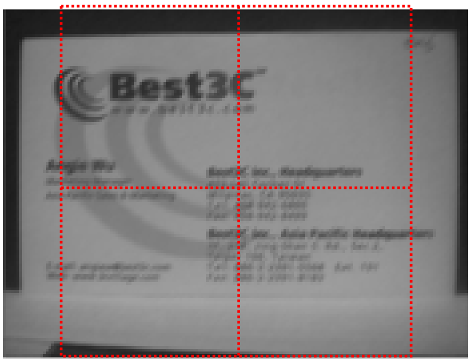


Figure 4.2 Synthesized Blurred Name Card Divided in 4 Sections

In Figure 4.3(a), the result is derived by applying the algorithm on the full image. In Figure 4.3(b), the result is the average estimation from 4 subsections.

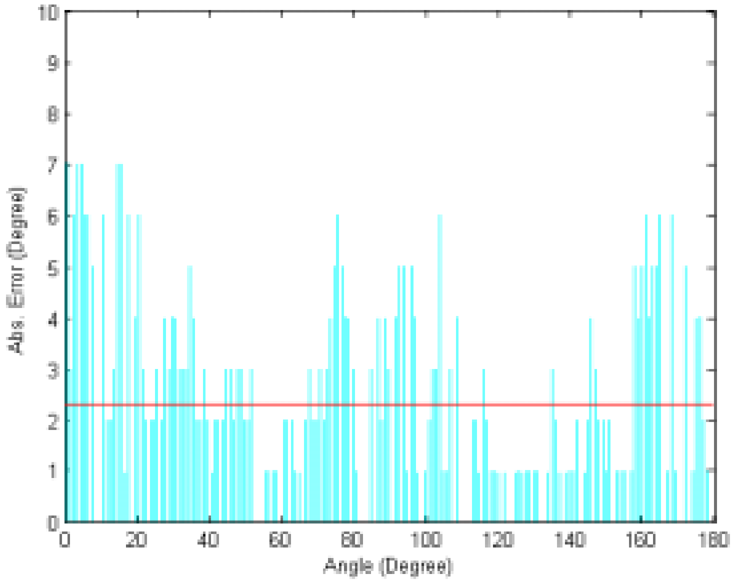


Figure 4.3(a) Angle Estimation Errors

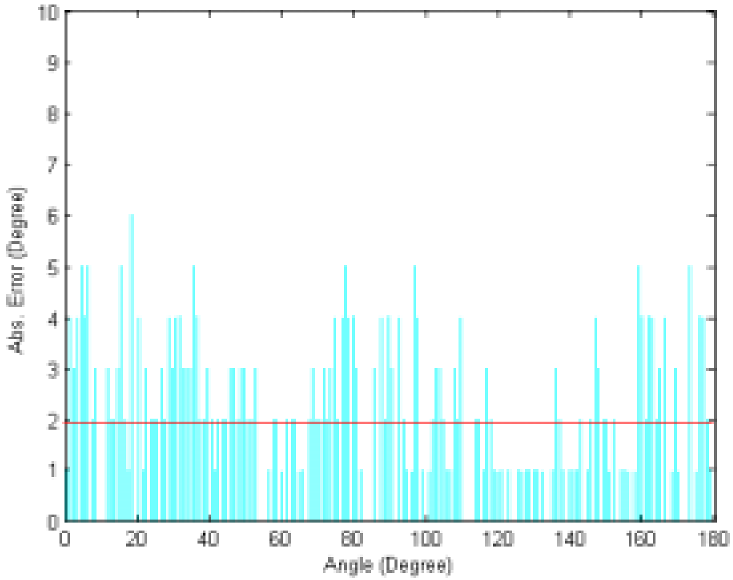


Figure 4.3(b) Angle Estimation Errors after Averaging

From the comparison, we notice some large errors are smoothed. The average absolute error drops as most error is within 4 degrees now. We conclude in most cases averaging estimations from subsections improve the result. We have a tradeoff between accuracy and computational time here. The practical solution is to use averaging estimations only when the OCR software does not produce satisfactory results.

To sum up for our angle estimation performance, the algorithm tends to return better results when the magnitude is larger and less texture is in the name cards. Besides, thresholding on image spectrum proves to increase the accuracy of the estimation.

Next, we perform the blur magnitude estimation. The blur magnitude estimation algorithm is required to identify the magnitudes in the range from 4 to 25 pixels under all blur orientations, though our algorithm works beyond 25 pixels range.

The magnitude estimation takes the result from the angle estimation as the input. The correctness of angle estimation has significant influences to our algorithm. In order to measure the performance of this magnitude estimation alone, we provide with the known angle in the first experiment. To test the algorithm, we need $90 \times 22 = 1980$ PSFs. This is unnecessary and extremely time-consuming. Therefore, we only run the experiments under four orientations, i.e. $0°, 15°, 30°$ and $45°$. The experiment has two variations. The first variation applies differential filter to the spectrum while the second not. As in the angle estimation, we record down the mean error, the absolute value of the mean error, the maximum error, the minimum error and the standard deviation. We randomly pick 10 name card images to average the errors in following experiments.

ERROR ESTIMATIONS	WITHOUT DECORRELATION				WITH DECORRELATION			
	5-9	10-14	15-19	20-24	5-9	10-14	15-19	20-24
Mean	1.06	0.88	0.22	0.20	0.22	0	0	0
|Mean|	**1.06**	**0.88**	**0.22**	**0.20**	**0.22**	**0**	**0**	**0**
Max	2	2	1	1	1	0	0	0
Min	0	0	0	0	0	0	0	0
S.Dev	0.04	0.05	0.06	0.06	0.06	-	-	-
Known Angle = 0°								

ERROR ESTIMATIONS	WITHOUT DECORRELATION				WITH DECORRELATION			
	5-9	10-14	15-19	20-24	5-9	10-14	15-19	20-24
Mean	0.90	0.64	0.10	0.28	0.24	0	0	0
|Mean|	**0.90**	**0.64**	**0.10**	**0.28**	**0.24**	**0**	**0**	**0**
Max	1	1	1	1	1	0	0	0
Min	0	0	0	0	0	0	0	0
S.Dev	0.04	0.06	0.04	0.06	0.06	-	-	-
Known Angle = 15°								

ERROR ESTIMATIONS	WITHOUT DECORRELATION				WITH DECORRELATION			
	5-9	10-14	15-19	20-24	5-9	10-14	15-19	20-24
Mean	0.90	0.62	0.12	0.30	0.20	0	0	0
|Mean|	**0.90**	**0.62**	**0.12**	**0.30**	**0.20**	**0**	**0**	**0**
Max	1	1	1	1	1	0	0	0
Min	0	0	0	0	0	0	0	0
S.Dev	0.04	0.06	0.04	0.06	0.06	-	-	-
Known Angle = 30°								

ERROR ESTIMATIONS	WITHOUT DECORRELATION				WITH DECORRELATION			
	5-9	10-14	15-19	20-24	5-9	10-14	15-19	20-24
Mean	0.86	0.62	0.16	0.32	0.18	0	0	0
|Mean|	**0.86**	**0.62**	**0.16**	**0.32**	**0.18**	**0**	**0**	**0**
Max	1	1	1	1	1	0	0	0
Min	0	0	0	0	0	0	0	0
S.Dev	0.05	0.06	0.05	0.06	0.06	-	-	-
Known Angle = 45°								

Table 4.6 Blur Extent Error Estimation for Name Card Images with Angle $0°, 15°, 30°$ and $45°$

By observing the results, we notice the algorithm with decorrelation as the first step has higher accuracy over the blurred images without decorrelation, though the algorithm has already obtained very small mean errors in all four orientations. We use the orientations in the range from $0°$ to $45°$ because the Radon transform in the range $45°$ to $90°$ is merely the reflection over the $y=x$ axis. The performance of the algorithm does not vary significantly according to the changes in orientation. Overall, the blur magnitude estimation is successful; the average mean error is 0 pixel in nearly all the cases. We notice magnitude is estimated only 1 pixel larger in all the errors and perfect estimation is recorded when the magnitude is larger than 10 pixel. Most errors occur at magnitude range 5-9 pixels. By looking at the cestrum of the collapsed signal, we notice the adjacent peak has small difference in this range and the peak searching process is error prone.

We are able to test the full algorithm on the estimation of uniform linear motion blur parameters now. To ease the experiment process, we take few combinations of blur parameters only. The orientation chosen is $0°$, $30°$, $60°$ and $90°$. The magnitude chosen is 6, 10, 15 pixel. There are total 12 combinations. The reason behind is that these blur parameter combinations are considered to be common in real world motion blurs. For every name card image, we blur with all the combinations and apply the full algorithm. We record down all the error measures as before then average the errors with 10 images.

ERRORS ESTIMATION	BLUR PARAMETER (ANGLE-DEGREE, MAGNITUDE-PIXEL)											
	0, 6	0, 10	0, 15	30, 6	30, 10	30, 15	60, 6	60, 10	60, 15	90, 6	90, 10	90, 15
Mean Angle	4.80	2.60	1.80	2.80	2.30	2.00	1.20	-2.50	-2.00	-3.20	-2.00	-1.90
|Mean| Angle	4.80	2.60	1.80	3.20	2.80	2.00	3.40	3.20	2.30	3.60	2.00	1.90
Max Angle	7	6	3	6	6	4	4	2	1	2	0	0
Min Angle	1	0	0	-2	-2	0	-4	-5	-3	-5	-4	-3
S.Dev.Angle	1.57	1.89	0.54	2.14	1.66	0.86	1.36	2.21	1.05	2.03	1.16	0.96
Mean Md.	1.40	1.00	0.60	0.40	0.40	0.50	0.40	0.80	0.60	1.60	1.00	0.60
|Mean| Md.	**2.00**	**1.20**	**0.60**	**1.50**	**0.80**	**0.50**	**1.00**	**0.80**	**0.60**	**1.80**	**1.00**	**0.60**
Max Md.	4	2	1	2	1	1	2	2	1	4	2	1
Min Md.	-2	-1	0	-2	-1	0	-2	0	0	-1	0	0
S.Dev.Md.	0.32	0.16	0.15	0.28	0.16	0.16	0.25	0.19	0.15	0.30	0.20	0.15

Table 4.7 Blur Parameter Estimation on Name Card Image with 12 Blur Parameter Combinations

By observing the results, we notice first the algorithm tends to return larger magnitude than actual when the blur orientation is estimated wrongly. Second, the algorithm has a good tolerance to the error occurred in orientation estimation. The average error in magnitude is kept below 2 pixels. When the blur magnitude is small, orientation estimation has larger errors and propagate to the magnitude estimation. An additional test shows that when absolute errors in orientation are above 10°, this algorithm fails in larger magnitudes. By looking at the cestrum, we notice that the peak diminishes when the spectrum collapses at wrong orientations. Such case usually does not happen since the extreme errors in the orientations are kept below $10°$. When angle estimation is successful in larger magnitudes, the algorithm returns the exact distance in half of the images tested.

Overall, our new algorithm on uniform linear motion blurs works well on synthetic images. The blur parameters are extracted from the blurred image with satisfying accuracy as given by the requirements in section 2.4.2. Most error occurs in the angle estimation. Synthetic blurred name cards can be restored using the Wiener filter then.

4.2 Real World Motion Blurred Images

In synthetic blurred images, we have known PSF and our algorithm recovers the PSF well. For real world blurred images, the PSF is totally unknown to us. Wiener filter is used to restore the original image with our estimated PSF. Then OCR result on the deblurred name card image is taken as a measure of the performance. Many uncertainties exist in real world blurred images as shown in Figure 4.5, one assumption we make is that the blurred image is noise free throughout our thesis.

First, we assume the motion blur existing in the name cards is uniform linear motion blur. We use the algorithm in section 3.1 to estimate the blur. The procedure of our experiment is as follows,

1. Use OCR software to recognize an original blurred name card M_1 and record down its precision p_1 and recall r_1.

2. Apply our algorithm and obtain the blur orientation θ and magnitude d.
3. Apply Wiener filter with θ and d to deblur M_1 and the resulting image is M_2.
4. Use OCR software to recognize M_2 and record down its precision p_2 and recall r_2.

We test 10 real world blurred name cards provided by Hot Card Company. A brief preview of the blurred name cards reveals that uniform linear motion blur may not be the only type of blur occurred in these images and distortions of some images are hardly to recognize. Our algorithm degrades its performance in the following cases,

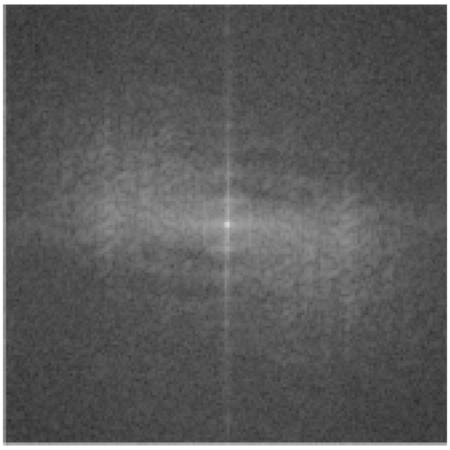
1. Additive noise is present.
2. Other type of blur is present. Example: out of focus blur, uniform accelerated motion blur.
3. Blur magnitude is small.

The OCR software used here is OmniPage Pro v14 from Scansoft.

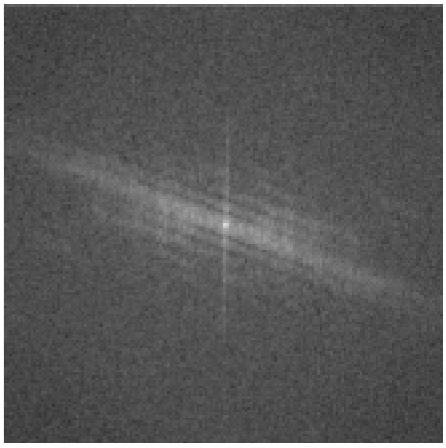
ID	BLURRED IMAGE	PRECISION 1	RECALL 1	EST. θ	EST. D	**PRECISION 2**	**RECALL 2**
1	P1120535	0%	0%	179	14	**8.43%**	**3.43%**
2	P1120541	0%	0%	114	21	**51.51%**	**34.46%**
3	P1120543	0%	0%	66	7	**0%**	**0%**
4	P1120547	0%	0%	90	16	**46.67%**	**23.3%**
5	P1120550	0%	0%	67	30	**0%**	**0%**
6	P1120564	46.08%	27.93%	175	4	**69.11%**	**55.19%**
7	P1120567	0%	0%	92	15	**50.75%**	**29.06%**
8	P1120569	0%	0%	77	11	**0%**	**0%**
9	P1120570	41.18%	11.92%	83	8	**90.08%**	**87.20%**
10	P1120574	40.28%	23.39%	96	6	**80.61%**	**64.23%**
Average *						**56.29%**	**42.14%**

Table 4.8 OCR Results for 10 Real World Blurred Images Assuming Uniform Linear Motion Blur
* Exclude images with 0% recognition.

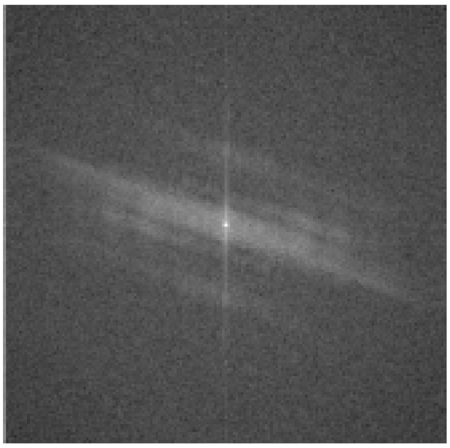
The OCR results have proved our initial guess. Our algorithm fails to improve OCR results in the 3rd, 5th and 8th images. An analysis of the Fourier spectrum of all three images shows that no clear ripples appear in the spectrum as expected by our algorithm. The 5th and 8th images are severely distorted. We argue that uniform linear motion blur is probably not the type of blur occurring in these images. Note that the estimated magnitude of 5th image is as large as 30 pixels. Furthermore possible noise has obscured the ripples in the spectrum as in Figure 4.4. Even the motion orientation is reported correctly as we can roughly estimate by eyes (excluding P1120543), the magnitude information is not enough to deblur the image.



(a)



(b)



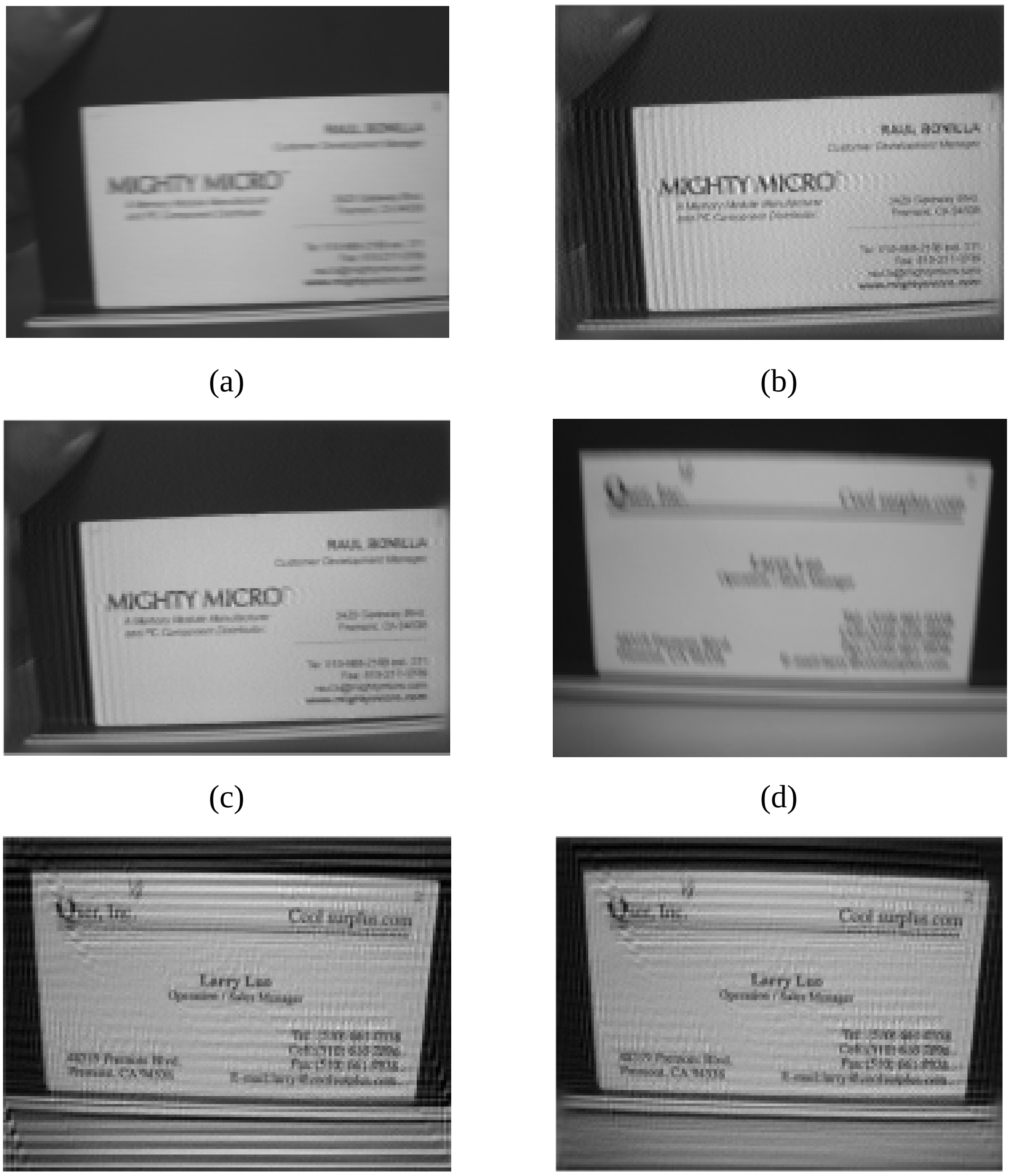
(c)

Figure 4.4(a) Fourier Spectrum of P1120543 (b) Fourier Spectrum of P1120550 (c) Fourier Spectrum of P1120569

The OCR result for the 1st image is hardly improved. One possible reason is that the name card has occupied only half of the image. The complex background has influenced the result of estimation. Besides, same problem occurring in the above three images may also present.

OCR results improve by a considerable margin in the remaining six images. In the 6th, 9th and 10th image, the distortion is smaller. Some characters can be recognized even before applying our algorithm. After deblurring, the image usually can be restored. The 6th image has relative lower recognition rate because the blurring effect is nearly negligible. The original blur magnitude maybe outside the range of estimation as the reported magnitude is 4 pixels. We believe the 2nd, 4th and 7th images are blurred mainly by uniform linear motion blur. OCR software cannot recognize anything before deblurring. After the blur parameters are successfully estimated, we can read nearly all the characters in the restored image. Unfortunately, due to the ringing effect, OCR software fails to perform well as human eyes. We believe background noise suppression or tuning of Wiener filter can be used to enhance both the precision and recall.

OCR results mainly depend on the outcome of the Wiener filter. Besides Taylor *et al.* [Taylor and Dance, 1998] has proposed a method to enhance document images after deblurring using adaptive thresholding. Their experiments show that OCR result can be obtained from a threshold calculated from the local average of the deblurred and interpolated image. As we have done an analysis in section, Wiener filter can be fine tuned to different applications. The restoration result in Figure 4.5 has used the discussed suggestion and been incorporated in the next experiment.



(a) (b)

(c) (d)

(e) (f)

Figure 4.5 (a) P1120535 (b) P1120535 Restored with Wiener Filter (c) P1120535 Restored with Edge Tapered and *C* = 0.01 (d) P1120541 (e) P1120541 Restored with Wiener Filter (f) P1120541 Restored with Edge Tapered and *C* = 0.005

Up to now, estimation has encountered difficulties in real world blurred images because of various reasons. The restoration errors are mainly the result of assuming a PSF for restoration that deviates from the true blurring PSF. Since the true blur function is totally unknown, the best we can do is to make a closer estimation.

Now the new estimation procedure in section 3 is used for the next experiment. Uniform acceleration motion blur is assumed to occur in some of the blurred name cards. We continue to carry experiments on the 10 real world blurred images prior. Our focus is on those images with poor OCR performance. We purposely keep the result from previous experiment for comparison. Note that the column ACC. TYPE indicates the type of the acceleration estimated. (B – Backward, F – Forward, N – None)

ID	BLURRED IMAGE	PRECISION 2	RECALL 2	EST. *R*	ACC. TYPE	**PRECISION 3**	**RECALL 3**
1	P1120535	8.43%	3.43%	12	B	**43.86%**	**25.64%**
2	P1120541	51.51%	34.46%	24	B	**61.47%**	**43.23%**
3	P1120543	0%	0%	-	N	**0%**	**0%**
4	P1120547	46.67%	23.3%	18	B	**62.61%**	**44.67%**
5	P1120550	0%	0%	7	B	**37.33%**	**20.74%**
6	P1120564	69.11%	55.19%	-	N	**69.11%**	**55.19%**
7	P1120567	50.75%	29.06%	18	B	**55.57%**	**43.10%**
8	P1120569	0%	0%	7	B	**34.21%**	**14.61%**
9	P1120570	90.08%	87.20%	-	N	**90.08%**	**87.20%**
10	P1120574	80.61%	64.23%	-	N	**80.61%**	**64.23%**
Average *						**66.23%**	**51.90%**
Average **						**59.43%**	**44.29%**

Table 4.9 OCR Results for 10 Real World Blurred Images Assuming Uniform Acceleration Motion Blur
* Exclude 3rd, 5th, 8th image as a comparison for Table 4.8
** Exclude 3rd image as an indicator for the overall performance on motion blurred images

By observing the results, we notice the recall for 5th and 8th images increase from 0% to 20.74% and 14.61% respectively. The low recall can be explained by the severe distortion of the texts in the blurred images. Besides possible noise has degraded the performance of OCR, as we can see obscure smear lines in the spectrum in Figure 4.4. OCR software still can not recognize anything from 3rd image. The restored image has shown clear ringing artifacts around all the texts as in Figure 4.6. We conclude that the image is blurred by other types of blurs, e.g. Gaussian blur. A further study shows that a circular PSF can be used to deblur this image, which is out of scope of our thesis.

(a) (b)

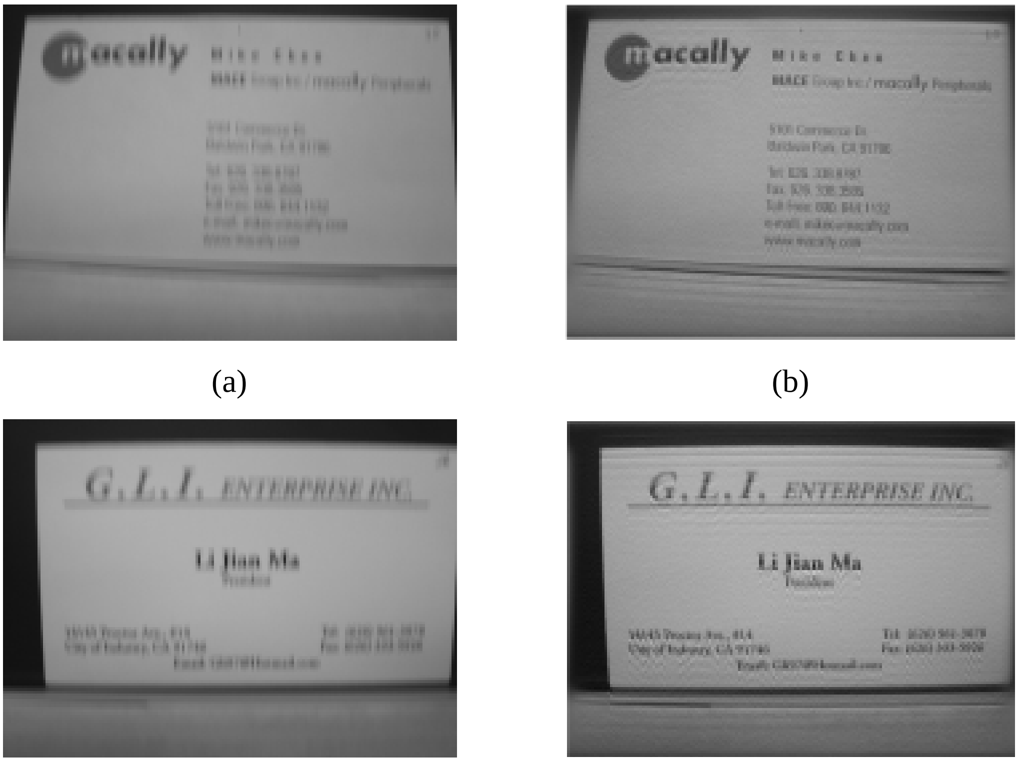
Figure 4.6(a) P1120543 (b) P1120543 Restored with Estimated Blur Function

In the 6th, 9th and 10th images, when we use either backward or forward acceleration PSF to deblur, an inverted scene is imposed on the restored image with distance of one blur extent as in Figure 4.7. We conclude that these images are best to be restored with symmetric PSF as the result is already satisfactory. The conclusion is also applicable to all motion blurred images with small blur extents. The precision and recall for the 1st image has jumped from 8.43%, 3.43% to 43.86% and 25.64%. The large improvements owe to our tuning of the Wiener filter and successful estimation of the acceleration.

(a) (b)

Figure 4.7(a) P1120570 Restored with Forward Acceleration (b) P1120570 Restored with Backward Acceleration

In the 2nd, 4th and 7th images, the blurs are all estimated with lowly backward acceleration. The recall increases from 34.46%, 23.3%, 29.06% to 43.23%, 44.67%, and 43.10% respectively. The low acceleration means the actually blur is close to uniform linear motion blur. As we look at the restored image in Figure 4.8, the ringing effect is reduced by using an accelerated PSF. However, due to the asymmetry of the PSF, the ringing artifacts appear to exist on only one side of the texts. (In horizontal blur, artifacts are on the right of the scene when using backward acceleration; left when using forward acceleration.) The artifacts lessen when we use a proper form of the Wiener filter. Most digits obscured by the blur can be read now.



(a) (b)

(c) (d)

Figure 4.8(a) P1120547 (b) P1120547 Restored with Backward Acceleration with C = 0.008 (c) P1120567 (d) P1120567 Restored with Backward Acceleration with C = 0.008

From all the results, we find that the restored images are still far from perfect. Though most characters are readable after deblurring, OCR software fails to recognize because of ringing artifacts. One observation we make is that in different regions of text, the severity of ringing also differs. We conclude the PSF usually varies over the real world blurred image, i.e. the PSF is space variant; it may be better to restore different parts of the image separately. However, to automate this process, we have to identify all the regions containing texts, which prove to be a troublesome task. To sum up, our method is not suitable for mission critical applications. Further work has to be done as we discuss in the future goals section.

5 Conclusion

In this chapter, we summarize the research work. Limitations for the applicability of this algorithm are explained and future goals are proposed.

5.1 Research Summary

The research work investigates the problem of image blurring caused by motion during image capture process of text documents. Such blurring prevents proper optical character recognition of the document text contents. One area of such applications is the name card images obtained from handheld cameras. To overcome the problem of image blurring, it is necessary to deblur the image by estimating the motion parameters, i.e. blur orientation and blur extent to restore the original image.

I.Rekleitis has developed a method to estimate the optical flow map for a single blurred image. His algorithm proves to work on both synthetic and real world blurred images. However, it reports significant errors when used to estimate the motion parameters for name card images. Such errors make his method not suitable for motion deblurring in our application.

In this thesis, a modified approach based on I.Rekleitis' method is formulated and evaluated experimentally. First we assume the only blur existing in the name card images is uniform linear motion blur. Then the following steps are used to estimate the blur. The orientation of the blur is first determined and then the blur extent in that direction is recovered. The algorithm operates in both the frequency domain and cepstral domain. The key observation is that motion blur often introduces clear ripples in the logarithm power spectrum of the blurred image. Thresholding techniques have been applied to extract the most significant ripple in the spectrum and a steerable bandpass filter is used to determine the dominant direction in the spectrum. After that, Radon transform is performed in this direction and the cestrum of the collapsed signal is analyzed to search for local peaks. The corresponding position of the peak is the blur extent. This algorithm has been implemented in MATLAB and numerous name card images have been used for experiments. It has returned very accurate results for synthetic blur images – the blur orientation is often recovered to within just a few degrees with an average less than 1 and the blur extent is estimated within 1 pixel error in most cases. When we look into real world blurred name cards, we encounter noticeable error in certain images. OCR still fails or returns poor recognition rates even both parameters are correctly obtained. This leads us to think that more severe motion has occurred during the capture process. We assume uniform acceleration motion blur is the type of the blur existing in those images with poor OCR results. The theory behind the blur is presented and the artificial creation of acceleration blur is implemented. We derive the range of the 3rd parameter (the ratio of initial velocity and acceleration) from autocorrelation functions for image pixels in the direction of the blur. A new estimation procedure on severed blurred images has been proposed. Experiments on real world blurred name cards show that the OCR results have been improved in most cases. The time spent in the whole estimation process is less than 5 seconds on Pentium VI computer.

Image restoration is an established research field with many proposed algorithms. Again we need to emphasize that the focus on this research work is the blur estimation on document images. Approaches that can be used to enhance the OCR results when the blur is known are not analyzed in depth. The applicability of this algorithm depends on how the images are blurred. Our method assumes motion blur is the most common blur occurred on document texts. When other blurs like out-of-focus blur or more severe motion blur exist, our algorithm may not return satisfying results. We will discuss it in the next section – future goals.

The contribution of our research work is summarized as follows:

1. Previous research work by J.Zhang [Zhang, 2004] is based on I.Rekleitis' algorithm whose blur distance estimation is not accurate and thus the earlier work has to resort to a process of re-estimation to recover the actual distance. The present work eliminates the need for distance re-estimation and returns accurate estimations by proposing a new modified approach for document images.
2. Previous work by J.Zhang [Zhang, 2004] assumes the relative motion between objects and camera is uniform linear motion with constant velocity, i.e. zero acceleration in the image capture process. As such, for more severe

or irregular motion, the OCR results are not too satisfactory. The present work analyzes the theory of uniform acceleration motion blur and proposes a new procedure to estimate such blur for real world blurred document images.

3. In the present work, the motion deblurring cost is less computational expensive since the re-estimation process is eliminated.
4. The theory behind the image restoration filter – Wiener filter is analyzed and an optimal form of this filter for document images is presented.

5.2 Future Work

There are number of directions that the future work can follow. The most apparent issue is to deal with additive noises. The assumption that the motion blurred images is noise free is made throughout this thesis, but future developments should take the noise factor into consideration and make the algorithm more robust in the noisy environment. Methods for removing noises can be used to pre-filter the blurred images as described in most literatures.

Currently, we assume the most severe motion occurred during the image capture process can be modeled as uniform acceleration motion blur. This assumption is not adequate in certain situations. For example, the acceleration may not be uniform or the orientation of the blur may vary, i.e. the motion path is not a straight line, during the exposure time. These blurs usually can not be modeled by a proper space-invariant PSF. One possible solution is to deblur the blurred image section by section with a different estimated PSF. For applications like name cards, we need to pay close attention to the content of the image. Name cards with large uniform backgrounds may influence the result of the algorithm since blur in a region with homogeneous brightness is undetectable. To fulfill an optimal motion deblurring for OCR, we need to further study application specific restoration algorithms for document images.

In conclusion, the algorithm developed in this thesis works for most motion blurred document images provided that the motion is not too irregular.

Reference

A.E.Savakis and H.J.Trussell, “Blur identification by Residual spectral Matching,” IEEE Transactions on Image Processing, vol.2, No.2, 141-151, April 1993.

A.M.Tekalp, H.Kaufman, and J.W.Woods, “Edge-adaptive Kalman filtering for image restoration with ringing suppression,” IEEE Trans.ASSP 37, 892-899, 1989.

D.B.Gennery, “Determination of optical transfer function by inspection of the frequency-domain plot,”J.Opt.Soc.Am.63, 1571-1577, December, 1973.

D.Kundur and D.Hatzinakos, “Blind image deconvolution,” IEEE Signal Processing Magazine, May, 1996.

I.M.Rekleitis, “Visual motion estimation based on motion blur interpretation,” MSc Thesis, School of Computer Science, McGill University, Montreal, Quebec, Canada, 1995.

J.Biemond, R.L.Lagendijk and R.M.Mersereau, “Iterative methods for image deblurring,” IEEE Proc.vol.78.No.5, May 1990.

K.C.Tan, H.Lim, and B.T.G.Tan, “Edge Errors in Inverse and Wiener Filter

Restroations of Motion-Blurred Images and Their Windowing Treatment," CVGIP: Graphic Models and Image Processing, Vol.53, No.2, 186-195, March, 1991.

K.C.Tan, H.Lim, and B.T.G.Tan, "Restoration of Real-World Motion-Blurred Images, CVGIP: Graphical Models and Image Processing," Vol.53, No.3, 291-299, May, 1991.

J.Zhang, "Name Card Image Enhancement and Restoration for Text Recognition", MSc Thesis, School of Computing, NUS, 2004.

M.Cannon, "Blind deconvolution of spatially invariance image blurs with phase," IEEE Trans.Acoust.Speech Signal Proces. 24,1976.

M.Chang, A.M.Tekalp, and T.A.Erdem. "Blur identification using the bispectrum," IEEE Transactions on Signal Processing, Vol.39, No.10, 2323-2325, October, 1991.

M.J.Taylor and C.R.Dance, "Enhancement of Document Images from Cameras", SPIE Vol.3305, 230-241, 1998.

M.Sezgin and B.Sankur, "Survey over image thresholding techniques and quantitative performance evaluation", Journal of Electronic Imaging, Vol.13, No.1, 146-165, 2004.

P.D.Welch, “The use of the fast Fourier transform for the estimation of power spectra,” IEEE Trans. Audio Electroacoust, Vol.AU-15, 70-73, June 1967.

R.C.Gonzalez and R.E.Woods, Digital Image Processing 2nd Edition, Prentice Hall, 2002.

R.Fabian and D.Malah, “Robust identification of motion and out-of-focus blur parameters from blurred and noisy images,” CVGIP: Graphical Models and Image Processing, Vol.53, No.5, 403-412, September 1991.

R.L.Lagendijk, J.Biemond, and D.E.Boekee, “Regularized iterative image restoration with ringing reduction,” IEEE Trans.ASSP 36, 1874-1887, 1988.

R.Rom, “On the cepstrum of two-dimensional functions,” IEEE Trans.Inform.Theory, vol.IT-21, 214-217, Mar.1975.

S.C.Som, “Analysis of the effect of linear smear on photographic images,” J.Opt.Soc.Am.Vol. 61, 859-864, 1971.

W.T.Freeman and E.H.Adelson. “The design and use of steerable filters,” IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol.13, No.9, 891-906, September 1991.

Y.Yitzhaky and N.S.Kopeika, "Identification of blur parameters from motion blurred images," CVGIP: Graph Models and Image Processing, Vol.59, 310-320, 1997.

Y.Yitzhaky, I.Mor, A.Lantzman, and N.S.Kopeika, "Direct method for restoration of motion-blurred images", J.Opt.Soc.Am. Vol.15, No.6, June, 1998.