RESTORATION OF ATMOSPHERIC TURBULENCE DEGRADED VIDEO USING KURTOSIS MINIMIZATION AND MOTION COMPENSATION

A Thesis
Presented to
The Academic Faculty

by

Dalong Li

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in the
School of Electrical and Computer Engineering

Georgia Institute of Technology
May 2007

RESTORATION OF ATMOSPHERIC TURBULENCE DEGRADED VIDEO USING KURTOSIS MINIMIZATION AND MOTION COMPENSATION

Approved by:

Professor Russell M. Mersereau,
Committee Chair
School of Electrical and Computer Engineering
Georgia Institute of Technology

Professor Aaron Lanterman
School of Electrical and Computer Engineering
Georgia Institute of Technology

Professor Mark J. T. Smith, Advisor
School of Electrical and Computer Engineering
Purdue University

Professor Allen R. Tannenbaum
School of Electrical and Computer Engineering
Georgia Institute of Technology

Professor Douglas B. Williams
School of Electrical and Computer Engineering
Georgia Institute of Technology

Professor May Dongmei Wang
Department of Biomedical Engineering
Georgia Institute of Technology

Date Approved: 27 November 2006

To my mom, Mei, my wife, Nan

and in memory of my grandma, Yanqing

for all of your love

ACKNOWLEDGEMENTS

I would like to thank Prof. Mark Smith and Prof. Russell Mersereau, my advisors, for invaluable supporting and guiding me throughout this research. I also thank Prof. Douglas Williams, Prof. Aaron Lanterman, Prof. Allen Tannenbaum and Prof. May Dongmei Wang for serving on my defense committee and provided invaluable comments. Their help is gratefully acknowledged. I am grateful to Dr. David Frakes for his significant help regarding the optical flow software. I am also grateful to Prof. Stanley Reeves of Auburn university who provided help with the implementation of Generalized Cross Validation (GCV). There are many helpful feedbacks from the readers of our papers. For example, Lalaké Apoyan pointed out the scintillation effects of turbulence.

I gratefully thank Hewlett Packard Labs, Eastman Kodak Research Lab, the Mathworks, Philips Research Lab for offering me the internships during my degree program. Very special thanks go to Dr. Steven Simske from HP lab for his support and encouragement throughout my program.

I thank my grandmother, Yanqing, mother, Songmei and wife, Nan, who have loved, encouraged, and motivated me through all the ups and downs in my life. They had confidence in me when I had little confidence in myself. I thank my dogs, Kelly, Vanilla and Cookie for making our lives more enjoyable.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

SUMMARY

It has been long recognized that atmospheric turbulence degrades the quality of images and video sequences. Stationary objects being observed through the atmosphere can appear blurred and waver spatially as if they are in motion. This phenomenon is well recognized, especially in astronomy. The degradation arises from the complicated, random fluctuations in the refractive index of the air, caused by fluctuations in temperature. In this thesis, the fundamental theory of turbulence such as the Kolmogorov law is reviewed. It is also shown that the point spread function (PSF) of the turbulence can be derived from the physics equations that describe atmospheric turbulence.

Atmospheric turbulence degradation is usually modeled as a linear convolution. The turbulence is dynamic and random in nature. The blurring parameter of the PSF of the turbulence is dependent on altitude, temperature, the rate of energy per mass dissipated by viscous friction, the sheer rate of the wind, and so on. Information about those turbulence conditions is often not available. Thus, the exact PSF of the turbulence blur is generally unknown in practice. Consequentially, blind image deconvolution technique is used in such a context. Blind image deconvolution is well known to be an ill-posed problem. Certain assumptions about the image and/or the blur must be made in order to find a solution. It has been observed that the kurtosis of the blurred (smoothed) image is often higher than an unblurred version. This observation is studied and justified using a frequency domain analysis where kurtosis is first represented and then interpreted. An image can be decomposed into a low frequency component and a high frequency component. It is found that

the kurtosis of an image is dominated by the interaction of the low frequency and high frequency components. Blurring alters the interaction and tends to increase the kurtosis. In addition to the theoretical analysis, experiments are conducted to verify that the smoothed image has higher kurtosis. This important observation forms the basis for the new blind deconvolution method. Kurtosis can be viewed as a metric to measure the quality of the resorted image without having the original image. In simulations, when an original image is available, one can use peak signal-to-noise ratio (PSNR), to determine the restored image that has the highest PSNR (PSNR maximization) to estimate the blurring parameter. Kurtosis minimization based blur identification works as following: given the functional form of the blur and an estimate of the parameter space, the parameter is searched by minimizing the kurtosis of the restored image. The restored image that has minimal kurtosis is used as the final estimate of the true image and the corresponding parameter is the identified blurring parameter. In many simulations, kurtosis minimization gives the same result as PSNR maximization. Kurtosis minimization is a generally applicable blur identification method. It has been tested on a variety of blurs including Gaussian blur, linear motion blur, out-of-focus blur, averaging blur and atmospheric turbulence blurs. In many experiments on standard test images, kurtosis minimization is able to give perfect estimation at different levels of noise. Moreover, it is compared with generalized cross validation (GCV) based blur identification on atmospheric turbulence blurs, which is the main application in this thesis work.

Besides blurring, turbulence also introduces geometric distortion in the video since the turbulence is time-varying. Such geometric distortion has been largely ignored in previous work in the restoration of turbulence degraded video. In this thesis, the time-varying distortion component (geometric distortion) is explicitly added into the video degradation model. The few previous researchers who have addressed the problem used a reference video, which is initially formed by time averaging of the

original video. The video frame is then warped towards the reference frame. When there is no object motion present, the approach works well in the sense that the turbulent motion is suppressed and the video is stabilized. However, the performance degrades severely when object motion does exist such as in panning or zooming or the case when the object is moving in the scene. The time averaging approach leads to a further degradation of the video in terms of the PSNR since the reference frame itself is motion blurred. A new method is introduced in this thesis that does not use a time-averaging reference video. Compensation is performed directly on the trajectories of the pixels in the video. The trajectories of the pixels are built from optical flow algorithms. An adaptive control grid interpolation (CGI) is used to compute the optical flow. This CGI algorithm can be viewed as an extension to the classic Lucas-Kanade method. The compensation along the motion trajectories effectively suppress the turbulent motion while preserving real object motion. The new method works well when there is real motion or when there is only turbulent motion in the video. Besides subjective evaluation of the result, the performance is also objectively measured by PSNR in simulated turbulence degraded video clips that include a variety of situations such as camera panning, zooming and object movement. In all the cases, the new method yields significant improvement over previous methods. The study of adaptive length of the smoothing window is also reported in this thesis. It is found that it is advantageous to adjust the smoothing window length since the turbulent motion is dynamic and approximately periodic.

Within the turbulent motion suppression framework, multiple frames of denoising are performed by warping the neighborhood frames towards the central frame followed by averaging. Not only does it effectively reduce noise, it also helps in reducing scintillation caused by atmospheric turbulence since the averaging decreases the variations in luminance. Mean squared error between consecutive frames are measured to show the reduction in luminance variations.

In addition to its importance to astronomy, atmospheric turbulence degradation can also appear in surveillance application where the video is degraded by horizontal-path atmospheric turbulence. In these applications, it is desirable to have a real time or near real time processing of the degraded video frames. A C/C++ implementation is conducted to show that real time processing is achievable.

CHAPTER I

INTRODUCTION

1.1 Statement of the Problem

Atmospheric turbulence is a well known source of distortion that can degrade the quality of images and videos acquired by cameras viewing scenes from long distances. This phenomenon is especially common in astronomy. For example, stars in outer space viewed through telescopes appear blurred since the Earth's atmosphere degrades the image quality. The physical cause of the turbulence is the fluctuations in the refractive index of air [71]. These fluctuations involve many factors including wind velocity, temperature gradients, elevation, etc. The dominant factor is usually temperature variation. In practice, it is difficult to measure these factors, and thus the parameters associated with a true physical model of atmospheric turbulence are typically unknown. Consequently the task of compensating for turbulence distortion can be viewed as a blind restoration problem. In addition to blurring, another effect associated with turbulence is geometric distortion, which arises because the turbulence is time-varying. This effect shows up in video when stationary objects appear to waver.

Taking both effects into consideration, turbulence-degraded video g may be modeled approximately as:

$$g(i,j,t) = D[x(i,j,t) * h(i,j,t), t] + \eta(i,j,t), \tag{1}$$

where * denotes two-dimensional convolution, η denotes time-varying additive noise, D denotes the turbulence induced time-varying geometric distortion, h is the dispersive distortion component of the atmospheric turbulence, g is the observed degraded video, and x is the original video. When t is fixed at a time when $D(x) = x$ (i.e.

no geometric distortion), the above model reduces to the common image degradation model:

$$g(i,j) = x(i,j) * h(i,j) + \eta(i,j), \tag{2}$$

where the observed image $g(i,j)$ is approximated as the sum of a two-dimensional convolution of the true image, $x(i,j)$, with a linear shift-invariant blur, (also known as the point-spread-function (PSF)), $h(i,j)$, and additive noise $\eta(i,j)$.

If we consider a parameterized PSF for the turbulence, equation (2) can be rewritten as:

$$g = x * h(\theta) + \eta \tag{3}$$

$$\theta \in \Omega$$

where θ is the blur parameters. The problem of recovering the true image $x(i,j)$ from the given degraded image $g(i,j)$ is called image restoration in the signal processing literature. Classical restoration methods require complete knowledge of the blurring function $h(i,j)$ prior to restoration. However, it is often impossible, or, in some cases, impractical to determine the blur *a priori*. Such is the difficulty in characterizing atmospheric turbulence in aerial and astronomy imaging. These situations typically rely on blind image deconvolution approaches.

Many blind restoration algorithms have been proposed in the past [41, 52, 84, 69]. Since blind deconvolution is an ill-posed problem, certain assumptions are typically made to make the problem tractable. For example, in the iterative blind deconvolution (IBD) method proposed by Ayers and Dainty [3] and its extensions such as the double regularization algorithm [89], images and blurs are assumed to be smooth and non-negative. Auto-regressive moving average (ARMA) models are also commonly-used models for the deblurring problem. The image is modeled as an autoregressive process and the blur is modeled as a moving average process. Both maximum likelihood [84] and generalized cross-validations [69] use this ARMA formulation.

The approach taken in the first part of this thesis may be described as follows. First assume that the form of the blurring function is known, but not the parameters. Let $\hat{x}(\theta_h)$ denote the deblurred image with the hypothetical parameter vector θ_h. Given the set $\{\Psi : \hat{x}(\theta_h|\theta_h \in \Omega)\}$, which restoration is the best approximation of the original image? The problem is easy if x is available and one can use the mean squared error or the peak signal-to-noise ratio (PSNR) as a measurement. The restored image with maximum PSNR may then be chosen as the best approximation, since, by this criterion it is the closet fit. However, x is not available in practice. In this dissertation, we introduce and explore the kurtosis, which is a measure of how outlier-prone a distribution is, as a criterion for selecting the best deblurred image. In particular, a kurtosis minimization based blur identification method is proposed and used to restore turbulent-degraded video frames.

Much of the previous work in turbulent-degraded image restoration has focused on still images and thus has only treated time invariant distortions [73, 75]. When this is done on a frame-by-frame basis, the object is not stabilized in the video processed by the method [73]. Only a few authors have considered methods based on image registration and warping techniques [18, 17] to explicitly address the geometric distortion component D. In those approaches, the time-averaged frames are typically used as reference frames and the current frames are registered towards the reference frames. The restored video has higher resolution and the video is stabilized in the sense that the geometric component of the atmospheric distortion is suppressed. However, these methods usually cannot handle situations in which both turbulence and real motion are present simultaneously. When true motion exists, the reference frame will be motion-blurred. Small, fast moving objects may even be smoothed out in the reference frames. The real motion is changed (slowed down) in the reference video because of the averaging and consequently is not preserved in the restored video.

Thus, such methods typically struggle to handle situations involving panning, zooming, or object movement. A new method is presented in this thesis to handle all of the above situations whereby the quasi-periodic nature of the geometric distortion of the turbulent motion is exploited. Motion is modeled explicitly and motion vector fields are computed. The cumulative motion vectors are computed over an estimated period of the turbulent motion. This efficiently separate the turbulent motion from the real motion. The motion modeling, estimation, and compensation are developed in greater detail in the subsequent chapters.

1.2 Scope of the Thesis

Chapter 2 provides an overview discussion of imaging through atmospheric turbulence and a discussion of the major blind image restoration methods such as generalized cross validation (GCV) and maximum likelihood. Earlier methods to address geometric distortions in turbulence degraded video are also included in this chapter. The limitation of this method is illustrated.

Chapter 3 describes the kurtosis-minimization-based blur identification. In this chapter, the statistical relationship between minimum kurtosis and optimal restoration is examined in the frequency domain using phase correlation. Then, kurtosis minimization is applied to the identification of a number of different distortions such as out-of-focus blur, Gaussian blur, atmospheric turbulence blur, and linear motion blur. Comparisons are made with Generalized Cross Validation (GCV), which is one of the highest performing methods for this kind of restoration.

Motion estimation and compensation and classic optical flow algorithms are reviewed in Chapter 4 as a precursor to the introduction of the geometric distortion suppressing algorithm.

The simulation of turbulence degradations in video is presented in Chapter 5. In order to quantitatively analyze the performance of the restoration algorithm, it is

necessary to have the ground truth video. Thus the degraded videos were generated in simulations. Experimental results on a number of both real and simulated videos are presented. There are situations where it is critical to have the restoration run in real time. Therefore, a C implementation of the algorithm was developed. It is shown that real time performance is achievable after making some modifications to the restoration algorithm.

Finally, in Chapter 6, a conclusion and further research directions are discussed.

CHAPTER II

BACKGROUND

2.1 Imaging through Turbulence

Atmospheric turbulence can cause blurring and geometrical distortion in images and videos acquired from a long distance away. Stationary objects being observed through the atmosphere can appear blurred and waver spatially as if they are in motion. This phenomenon is well recognized, especially in astronomy. The degradation arises from the complicated, random fluctuations in the refractive index of the air.

In the paragraphs that follow, we will present an overview of physical equations that describe atmospheric turbulence and some of the models used to represent the distortion effects.

The phase of a wave that has propagated through turbulence is a space- and time-varying random process. The value of a random process at one point x is a random variable, and has a probability density function $p(f[x])$, mean $\mu(x)$ and variance $\sigma^2(x)$. For a stationary process, none of these quantities depends on x, that is to say, the statistics are the same everywhere. Atmospheric turbulence is approximately a stationary process [82]. Although random processes are typically described by correlation functions or covariances, in the atmospheric science structure functions are typically used. As shown later, the optical transfer function of atmospheric turbulence is represented by a structure function. A structure function is the mean square difference between the two values of a random process at $\mathbf{x}$ and $\mathbf{x}+\mathbf{r}$ [82]:

$$D_f(\mathbf{r}) = \langle |f(\mathbf{x}) - f(\mathbf{x}+\mathbf{r})|^2 \rangle = \int_{-\infty}^{\infty} |f(\mathbf{x}) - f(\mathbf{x}+\mathbf{r})|^2 p(\mathbf{x}) d\mathbf{x} \tag{4}$$

where $\mathbf{x}, \mathbf{x}+\mathbf{r}$ are 3-d vector that represent two positions. The covariance function

(spatial correlation with the function itself) is

$$B_f(\mathbf{r}) = \langle f(\mathbf{x}) f(\mathbf{x}+\mathbf{r}) \rangle = \int_{-\infty}^{\infty} f(\mathbf{x}) f(\mathbf{x}+\mathbf{r}) p(\mathbf{x}) d\mathbf{x}. \tag{5}$$

The relationship between the structure function and the covariance function is

$$D_f(\mathbf{r}) = 2(B_f(0) - B_f(\mathbf{r})) \tag{6}$$

Within a range of separations $|\mathbf{r}| = |\mathbf{x_1} - \mathbf{x_2}|$ that is greater than the inner scale ℓ_0 (a few mm) and less than the outer scale L_0 (ranges from 10's to 100's of meters), structure function of quantities (such as refractive index and temperature) in atmospheric turbulence D can be assumed to be homogeneous (independent of position $D(\mathbf{x_1}, \mathbf{x_2}) = D(\mathbf{r})$) and isotropic (independent of directions $D(\mathbf{r}) = D(r)$ where $r = |\mathbf{r}|$). The structure function for atmospheric turbulence obeys a power law:

$$D(r) = C^2 r^{2/3} \quad \ell_0 < r < L_0 \tag{7}$$

where C is a constant. This is the well-known two-thirds power law derived by Kolmogorov [82, 20, 60]. The refractive index structure function is

$$D_n(r) = C_n^2 r^{2/3} \quad \ell_0 < \Delta r < L_0 \tag{8}$$

where C_n is the refractive index structure function constant. The refractive index fluctuation results mainly from the fluctuation of temperature in the atmosphere. The structure function of temperature in the atmosphere also follows a two-thirds law:

$$D_T(r) = C_T^2 r^{2/3} \tag{9}$$

where C_T is the temperature structure function constant. The two structure coefficients C_n and C_T are related by

$$C_n = (\frac{\partial n}{\partial T}) C_T. \tag{10}$$

The optical index of refraction for air at optical wavelengths, n, is given by

$$N = n - 1 = 80 \times 10^{-6} \frac{P}{T}, \tag{11}$$

where T is the temperature in degrees Kelvin and P is the pressure in millibars [33]. In a turbulent atmosphere, the relative fluctuations of temperature are much stronger than those of pressure so that the index fluctuations are directly related to the temperature fluctuations. From Equation (10) and Equation (11) we have

$$C_n^2 = (80 \times 10^{-6} \frac{P}{T^2})^2 C_T^2. \tag{12}$$

C_n^2 and C_T^2 have a strong dependence on altitude z and so usually they are denoted as $C_n^2(z)$ and $C_T^2(z)$. The temperature structure constant $C_T^2(z)$ is given [33] by

$$C_T^2(z) = \alpha^2 \varepsilon^{\frac{2}{3}} \gamma(z)^2 / \beta(z)^2 \tag{13}$$

where α is a constant, ε is the rate of energy per unit mass dissipated by viscous friction, $\beta(z)$ is the average shear rate of the wind, and $\gamma(z)$ is the average vertical gradient of the potential temperature.

When wavefronts pass through the atmosphere, refractive index variations in the air may perturb the wavefronts in both amplitude and phase. Of these two components, it is the phase fluctuation that is most responsible for the distortions we see. Therefore, to simplify the discussion, the wavefront is represented as $\Psi(\mathbf{x}) = e^{i\phi(\mathbf{x})}$ where $\phi(\mathbf{x})$ is the phase.

The spatial coherence function of the wavefront $\Psi(\mathbf{x})$ is defined as

$$\begin{aligned} C_\Psi(\mathbf{r}) &= \langle \Psi(\mathbf{x}) \Psi^*(\mathbf{x}+\mathbf{r}) \rangle & (14) \\ &= \langle e^{i(\phi(\mathbf{x}) - \phi(\mathbf{x}+\mathbf{r}))} \rangle & (15) \end{aligned}$$

$C_\Psi(\mathbf{r})$ is a measure of how "related" the light wave Ψ is at one position (e.g. $\mathbf{x}$) to its values at neighboring positions (say $\mathbf{x}+\mathbf{r}$). It can be interpreted as the optical transfer function (OTF) of atmospheric turbulence. If there is no phase fluctuation,

$\phi(\mathbf{x}) - \phi(\mathbf{x}+\mathbf{r}) = 0$, then $C_\Psi(\mathbf{r}) = 1$. Thus, in this case, the OTF is 1, meaning there is no turbulence distortion. The quantity $\langle e^{i(\phi(\mathbf{x})-\phi(\mathbf{x}+\mathbf{r}))}\rangle$ can be simplified through a property of characteristic function as shown below. In probability theory, the characteristic function of a random variable x is defined as the expected value of $e^{i\kappa x}$

$$M(\kappa) = \langle e^{i\kappa x}\rangle \tag{16}$$

where κ is a real number. That is, $M(\kappa)$ is the Fourier transform of the probability distribution function:

$$M(\kappa) = \int_{-\infty}^{\infty} e^{i\kappa x} p(x)\ dx. \tag{17}$$

If the random variable is a Gaussian with mean μ_x and variance σ_x^2,

$$\langle e^{i\kappa x}\rangle = e^{-\frac{\sigma_x^2\kappa^2}{2}} e^{i\kappa\mu_x}, \tag{18}$$

or when $\langle x\rangle = 0$

$$\langle e^{ix}\rangle = e^{-\frac{\langle x^2\rangle}{2}}. \tag{19}$$

This Equation (19) is to be used for the simplification of Equation (14). Since $\phi(\mathbf{x}) - \phi(\mathbf{x}+\mathbf{r})$ is a zero mean Gaussian random variable,

$$\begin{aligned} C_\Psi(\mathbf{r}) &= \langle e^{i(\phi(\mathbf{x})-\phi(\mathbf{x}+\mathbf{r}))}\rangle & (20)\\ &= e^{-\frac{\langle|\phi(\mathbf{x})-\phi(\mathbf{x}+\mathbf{r})|^2\rangle}{2}} & (21)\\ &= e^{-\frac{D_\phi(\mathbf{r})}{2}}. & (22)\end{aligned}$$

The phase shift of a wave propagating vertically (in the z direction) from height h to height $h+\delta h$ caused by refractive index fluctuations is

$$\phi(\mathbf{x}) = k\int_h^{h+\delta h} n(\mathbf{x}, z)\ dz, \tag{23}$$

where $n(x, z)$ is the index of refraction and k is the wave number.

These relationships as defined by equations (4)-(23) are important because they capture the intricate nature of atmospheric turbulence. The distortions caused by

phase variation are dependent on altitude, temperature, rate of energy per mass dissipated by viscous friction, sheer rate of the wind, and so on, working with a multiparameter function like this is problematic and thus simplified models are useful. From Equations (20) and (23), an optical transfer function (OTF) was derived by Hufnagel and Stanley [33]. In terms of discrete frequencies, their OTF of the atmosphere turbulence is given by

$$H(u,v) = e^{-\lambda(u^2+v^2)^{\frac{5}{6}}}, \tag{24}$$

where u, v, are the discrete frequency variables and λ, which controls the severity of the blur, is determined by the turbulence strength $C_N^2(z)$.

This model has wide applications. Because of its simplicity, we use this model for atmospheric turbulence degraded video, realizing that the refraction index fluctuation is a random process and that the blur is time-varying. Thus, besides the blurring effect, geometric distortion will also occur.

2.2 Previous Works in Turbulence Degraded Image Restoration

The restoration of atmospheric turbulence degraded images has been studied extensively. The Labeyrie method [42], the Knox-Thompson method [39], and the triple correlation method [53] represent speckle imaging techniques. The purpose of speckle imaging [86, 30] is to obtain an estimate of the true object from a time series of short exposure images of the object. This is accomplished by estimating both the Fourier magnitude and phase of the object, separately, and then inverse Fourier transforming. Such techniques assume that a series of short-exposure turbulence-degraded images are available.

The imaging through turbulence can be modeled with the following convolution:

$$i_n(x) = h_n(x) * o(x) \tag{25}$$

where $i_n(x)$ is the nth speckle image in an ensemble, $o(x)$ is the object that we want to recover, and $h_n(x)$ is the point-spread function. If we take Fourier transform of this relationship, the convolution becomes a multiplication. Then We can average the power spectrum over each frame and solve for the Fourier magnitude of the object as following:

$$|O(\mathbf{u})| = \left(\frac{\sum |I_n(\mathbf{u})|^2}{\sum |H_n(\mathbf{u})|^2} \right)^{1/2} \tag{26}$$

where $\mathbf{u}$ is spatial frequency vector. $|H_n(\mathbf{u})|$ is estimated from a point reference in the image. For the phase estimate of the object, the complex bispectrum is used. In statistical analysis, the bispectrum is a statistic used to search for nonlinear interactions. The bispectrum [57] is defined in spatial frequency space as:

$$I_{B,n}(\mathbf{u}, \mathbf{v}) = I_n(\mathbf{u}) I_n(\mathbf{v}) I_n^*(\mathbf{u} + \mathbf{v}) \tag{27}$$

where $\mathbf{u}$ and $\mathbf{v}$ are spatial frequency vectors. The Fourier phase of the object is shown to be recursively related to the phase of the average complex bispectrum according to a three-point integration [19]

$$\arg(O(\mathbf{u} + \mathbf{v})) = \arg(O(\mathbf{u})) + \arg(O(\mathbf{v})) - \arg(\frac{1}{N} \sum I_{B,n}(\mathbf{u}, \mathbf{v})) \tag{28}$$

where N is the number of the observed images. The phase of the object is recursively computed from this equation. After the phase and the magnitude are recovered, they are simply combined and inverse transformed to give the restored image of the object.

As shown in Equation (26), a point-shaped reference adjacent to the object is required so that $|H_n(\mathbf{u})|^2$ can be estimated. The phase of the object is estimated by Bispectral analysis as in Equation (28). Speckle imaging represents a fundamental improvement in the resolution obtainable from large ground-based telescopes. However, noise is not in the model shown in Equation (25). The performance is degraded when the image contains a certain level of noise. Generally, the computation demand is high. This is partially because there are many frames to be processed, perhaps

thousands [15], in order to achieve the desired results. There are some difficulties in applying the method to surveillance images recorded over horizontal or slant paths. For example, it is very unlikely to have a point reference for estimating $|H_n(\mathbf{u})|^2$ as required by the speckle imaging method.

Another approach to addressing the problem was the wavefront sensor technology, which is used to determine the phase perturbation in each short-exposure image. From the measured phase perturbations, deconvolution is then performed by wavefront analysis [16]. The wavefront sensing technology can also be used in adaptive optics [13] to correct the measured phase deviations by using a flexible mirror. Complicated devices are needed in such technique. Sheppard *et al* [75] presented a MAP (maximum *a posteriori*) algorithm for deconvolution with Wave Front Sensing.

The restoration of turbulence-degraded images is often recognized as a blind image deconvoltuion problem since the exact point spread function of the turbulence is usually unknown and time varying. Ayers and Dainty [3] proposed the iterative blind deconvolution method, which was applied to the restoration of turbulence-degraded images. The image-domain constraint of nonnegativity is used in the iterative algorithms to take advantage of the nonnegativity property of image intensity distributions. Promising results were obtained in simulations. However, the uniqueness and convergence properties of the deconvolution algorithm are uncertain and the effect of noise existing in the convolution data is unknown. Later, many approaches were recommended to improve the blind deconvolution method including least-squares-based approaches [46, 47, 59] and maximum likelihood estimation [73, 45]. Generally, the computation complexity is high for maximum likelihood estimation.

Other blind image deconvolution methods for the restoration of atmospheric turbulence degraded images include an image division method that was proposed by Frieden [22, 21] as a faster method for image restoration with only two short-exposure images. The atmospheric turbulence PSF is modeled as a stochastic superposition

of a series of disturbance or speckle functions. The Fourier frequency spectra of two short-exposure images are used to establish nonlinear equations for the weights and displacements of the speckle functions. The weights and displacements of the speckle functions are found by the Newton-Raphson algorithm [22] and Marquadt-Levenberg algorithm [21]. These methods are based on the assumption that the number of the speckle functions, and the forms and parameters of the disturbance functions remain unchanged. In reality, these assumptions might not hold. By assuming the PSF support is known, Zhang et al [90] proposed an approach to estimate the PSF coefficients in the frequency domain from two short-exposure turbulence-degraded images when the noise is omitted. As with other iterative restoration algorithms, the constraints of non-negativity and smoothness are used to increase the noise robustness of the approach.

In this thesis, a blind image deconvolution approach that incorporates the turbulence model is preferred since no device/sensor is needed to measure phase perturbation. The atmospheric turbulence parameter is estimated directly from the degraded images.

2.3 Blur Identification

Classical work in blur identification often relied on using spectral nulls as identification indicators. Some common image blur sources such as uniform linear motion blur and out-of-focus blur have distinct patterns of zeros in the frequency domain. Gennery [23] developed a frequency-plot inspection method that takes advantages of this phenomenon. The major limitations of the method are: 1) the PSF form must be known so that the blur parameters can be determined from the frequency zero locations; 2) the original image needs to have enough high-frequency content so that the patterns of zeros are identifiable; and 3) the noise level must be low enough so that it will not obscure the frequency zeros. Stockham et al. [81] devised two

automatic methods named homomorphic deconvolution (or cepstral averaging) and power spectrum averaging. The method is based on the assumption that the PSF is shift-invariant while the image is spatially varying. Therefore the non-stationary frequency content of the original image can be "averaged out" while the blur will survive from such averaging since it is assumed to be shift-invariant.

Recently, parametric methods have been used to identify PSFs that are more general than those accommodated by early methods. A commonly-used model for the deblurring problem is the Auto-Regressive Moving Average (ARMA) model, in which the image is modeled as an autoregressive process and the blur is modeled as a moving average process. Many algorithms use this ARMA formulation, including maximum likelihood(ML) [84] and generalized cross-validation(GCV) [69]. The ARMA model will be reviewed first, followed by the GCV and ML blind deconvolution algorithms.

2.3.1 Auto-Regressive Moving Average (ARMA) Model

The ARMA model represents an image as an autoregressive (AR) process and a blur as a moving average (MA) process. In this way, the image and blur are distinguished. This model has been found suitable for the blur identification problem. The blurring is assumed to be a linear shift-invariant convolution operation, which is a MA process. The blurred image is treated as an ARMA process. The blur identification problem is then formulated as the identification of a 2-D ARMA model [83]. The ARMA model is represented as follows:

$$x(i,j) = \sum_{(k,l)\in R_+} a_{kl} x(i-k, j-l) + \eta_1(i,j) \tag{29}$$

$$g(i,j) = \sum_{(k,l)\in R} d_{kl} x(i-k, j-l) + \eta_2(i,j) \tag{30}$$

where $x(i,j)$ is the original image with indices (i,j), $g(i,j)$ is the image degraded by shift invariant blur and additive noise, and $\eta_1(i,j)$ and $\eta_2(i,j)$ are independent,

zero-mean, white Gaussian noises with variances σ_1 and σ_2, respectively. $\eta_1(i,j)$ represents the AR model residual and $\eta_2(i,j)$ represents the random noise added to the blurred image. R_+ is the non-symmetric half-plane (NSHP) support of the AR process. The support R of the MA process given by d_{ij} is, in general, non-causal. Energy conservation of the MA process is almost always assumed in order to preserve the mean value of the image; that is:

$$\sum_{(i,j)\in R} d_{ij} = 1 \tag{31}$$

In vector-matrix form, a more compact representation is as follows:

$$x = Ax + \eta_1 \tag{32}$$

$$g = Dx + \eta_2 \tag{33}$$

Reduced to a single equation, (32) and (33) become

$$g = D(I - A)^{-1}\eta_1 + \eta_2. \tag{34}$$

With these models of image and blur, the blur identification problem becomes one of determining the parameters of an ARMA model. There are several issues involved, including: 1) the model order of both the AR and the MA models, 2) the method used to determine the parameters, 3) treatment of image boundaries, and 4) the presence of observation noise in the data.

2.3.2 Maximum Likelihood Blur Identification

The maximum likelihood (ML) method is one of the most powerful parameter estimation methods. The ML estimator determines the parameters that are most likely to have produced the blurred image [3, 12]. A number of researchers have investigated blur identification methods that apply the ML criterion, including Tekalp et al. [83], Lagendijk et al. [43], and Katsaggelos et al. [38]. All of the methods use the same basic approach, although the implementations differ somewhat.

The maximum-likelihood (ML) methods attempt to derive restoration filters by estimating the PSF, variance of the additive noise, and the AR model coefficients of the original image. Thus, the problem consists of estimating the parameter set

$$\theta = \{h(k,l), a(k,l), \sigma_1^2, \sigma_2^2\}$$

from $g(i,j)$ (σ_1^2 and σ_2^2 are the variances of η_1, and η_2, respectively). An estimate of the parameters is made such that the probability or likelihood of obtaining the observed image given the parameter set θ, is maximized. The ML estimator is given by:

$$\hat{\theta}_{ml} = \arg\{\max_{\theta\in\Theta} L(\theta)\} = \arg\{\max_{\theta\in\Theta} \log(p(g;\theta))\} \tag{35}$$

where $L(\theta)$ denotes the log-likelihood function of θ, Θ specifies the range of elements of θ, and $p(g;\theta)$ is the probability density function (pdf) of g for a given θ.

Different implementations exist to solve the nonlinear optimization problem as described by Equation (35), such as gradient-based methods and the expectation maximization (EM) algorithm [43, 38]. The EM algorithm is the most popular since it is straightforward to implement. The EM algorithm converts the nonlinear optimization problem into a linear iterative procedure. It is computationally efficient, although convergence might be slower than for a gradient-based method.

2.3.3 Generalized Cross-Validation Blur Identification

Cross-validation is a well-known model evaluation technique in statistical data analysis. It is also known as "leave-one-out" or predictive sample reuse. The basic idea is to remove some of the data before training starts. Then when training is done, the data that was removed can be used to test the performance of the learned model on "new" data.

The **holdout** method is the simplest kind of cross-validation. The data set is separated into two sets, called the training set and the testing set. The model is learned using the training set only. Then the model is used to predict the output

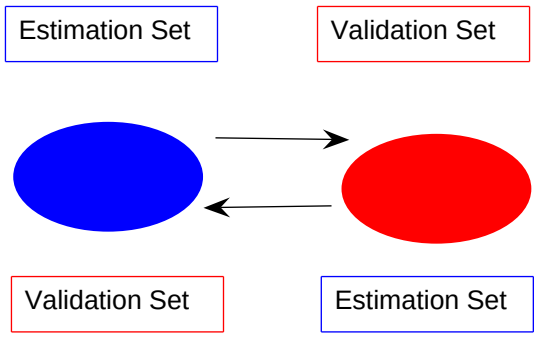


Figure 1: 2-fold Cross-validation

values for the data in the testing set (it has never seen these output values before). The errors it makes are accumulated to give the mean absolute test set error, which is used to evaluate the model. The evaluation by the holdout method can have a high variance. The evaluation may depend heavily on which data points end up in the training set and which end up in the test set, and thus the evaluation may be significantly different depending on how the division is made.

K-fold cross-validation is one way to improve over the holdout method. In the k-fold cross-validation, the data set is divided into k subsets. Each time, one of the k subsets is used as the testing set and the other $k-1$ subsets are put together to make up a training set. Then the average error across all k trials is computed to evaluate the model. The advantage of this method is that it matters less how the data gets divided. Every data point gets to be in a test set exactly once, and gets to be in a training set $k-1$ times. The variance of the resulting estimate is reduced as k is increased. The disadvantage of this method is that the training algorithm has to be rerun from scratch k times, which means it takes k times as much computation to make an evaluation. A variant of this method is to randomly divide the data into a test and training set k different times. The advantage of doing this is that you can independently choose how large each test set is and how many trials you average over. The 2-fold cross-validation is illustrated in Figure 1.

Leave-one-out cross-validation is K-fold cross-validation taken to its logical extreme, with K equal to N, the number of data points in the set. That means that N separate times, the model is trained on all the data except for one point and a prediction is made for that point. As before the average error is computed and used to evaluate the model. The evaluation given by leave-one-out cross-validation error (LOO-XVE) is good.

It had been demonstrated that cross-validation is able to estimate image parameters and regularization parameters from noisy and blurred images [68]. Reeves [69]

extended leave-one-out cross-validation to determining the blurring parameter also. The idea is quite simple. For a fixed value θ in the parameter space Θ, a restored image $f^k(\theta)$ is obtained using all but the k-th pixel values in the blurred image g. The restored image is then reblurred to predict the one that was left out. The predication error is recorded. Each time, a different pixel is left out while all the rest are used to obtain the restored image. The mean-square predication error is computed for the tested parameter. The process is performed for each candidate parameter in Θ, the parameter space. Finally, the parameter with the minimum mean-square predication error is selected as the best choice.

Let $f^k(\theta)$ be the restored image that minimizes the following criterion:

$$E^k(\hat{f}, \theta) = \frac{1}{N} \sum_{\substack{i=1 \\ i \neq k}}^{N} (g_i - [D(\theta_d)\hat{f}]_i)^2 + \alpha \|L\hat{f}\|^2 \tag{36}$$

where D is the matrix in the ARMA model as in Equation (34), θ_d is the blurring parameter, α is the regularization parameter that controls the degree of smoothing, and L is the smoothing operator.

Let D_{-k} represent the matrix D with the k-th row deleted; d_k represent the k-th row of matrix D and $g_{[-k]}$ denotes vector g with the k-th element removed. The minimizer of the estimation equation (36) is given as

$$f^k(\theta) = (D_{-k}^T D_{-k} + \alpha L^T L)^{-1} D_{-k}^T g_{[-k]}. \tag{37}$$

The restored image $f^k(\theta)$ is reblurred as in Equation (33) to predict the observation g_k that was left out when $f^k(\theta)$ is estimated. The cross-validation criterion is the mean square error over all k:

$$V_o(\theta) = \frac{1}{N} \sum_{k=1}^{N} (g_k - [D(\theta_d) f^k(\theta)])_k^2. \tag{38}$$

If θ approximates the true ARMA parameter (D,A) and the regularization parameter α well, then the above MSE is expected to be small. The θ that minimizes

$V_o(\theta)$ is selected as the estimated parameters. GCV has been compared with ML [69] and it was shown that GCV outperforms ML in real noisy blurred images.

2.4 Turbulent Motion Suppression

Few researches have considered the turbulent geometric component of the atmospheric turbulence problem. Fraser *et al.* [18] proposed an approach of first forming a prototype that is initially the temporal average of the image sequence. This prototype is motion-blurred, but is assumed to be geometrically correct. Then a hierarchically-windowed phase-correlation technique is used to register the degraded video frame to the prototype image, to sub-pixel accuracy. The prototype is updated by re-averaging of the processed image sequence. The procedure is shown in Figure 2. Another pro-

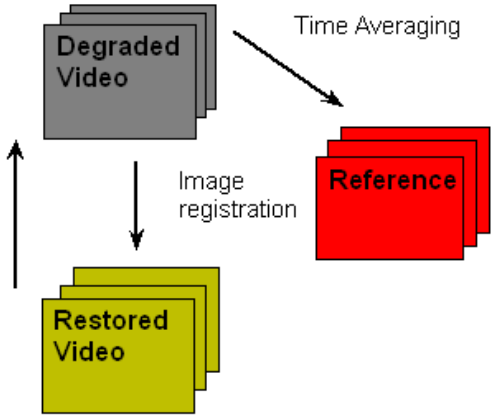


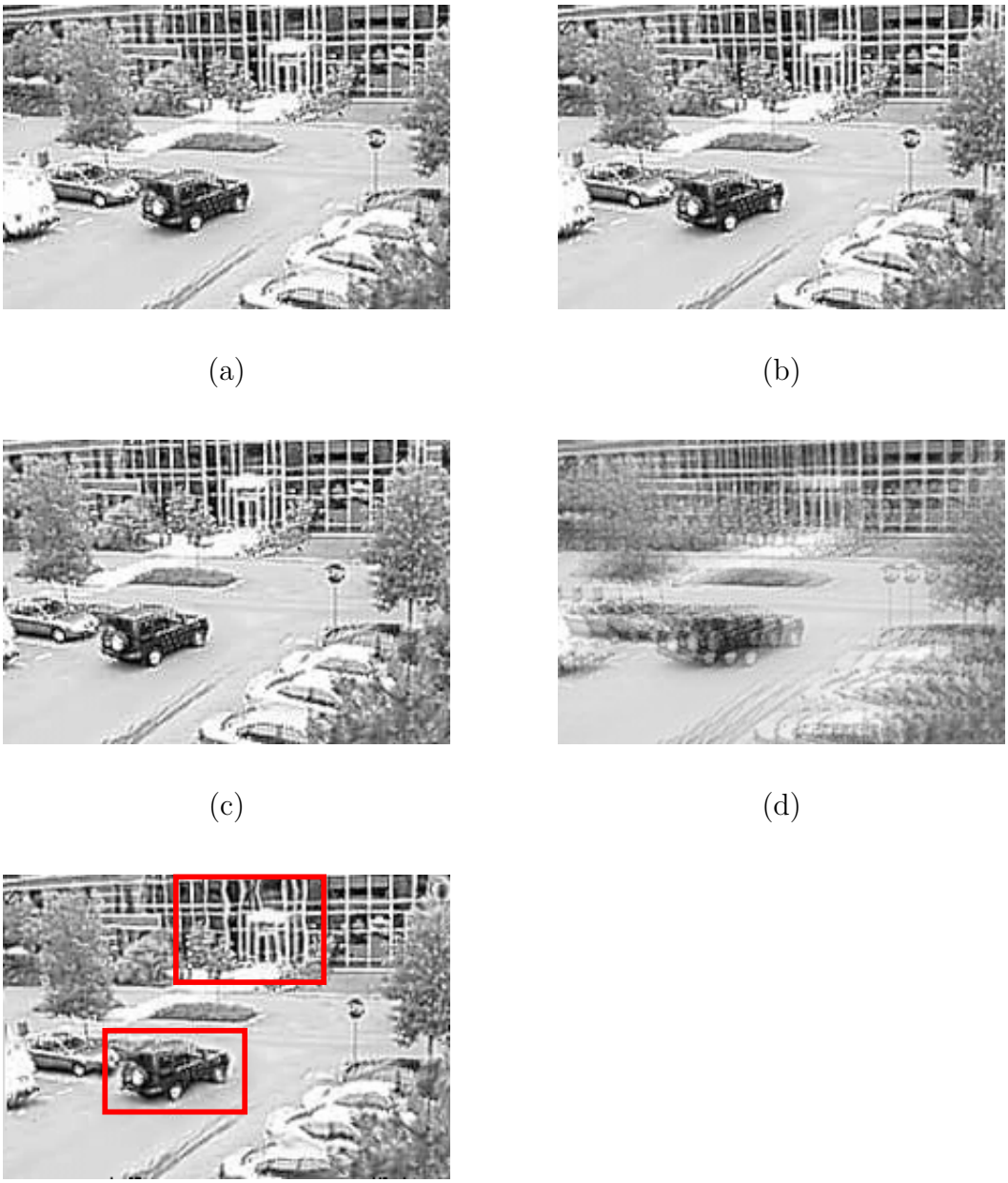
Figure 2: Time averaging reference approach for the suppressing of turbulent motion. Step 1: Use time averaging to compute reference frames; Step 2: Use image registration to warp the frame in the degraded video toward the corresponding reference frame; Step 3: Repeat the previous 2 steps by treating the processed video as the input.

cedure is the adaptive control grid interpolation used by Frakes *et al.* [17] for image registration. This method was shown to work quite well in the absence of real motion of objects in the scene. The restored video has higher resolution, and turbulence-induced distortions are suppressed. The notable shortcoming of these methods is that they are not designed to handle the situation in which both turbulence and real motion are present.

An example is illustrated in Figure 3 to show the limitation of the time-averaging reference approach. The example is a simulated panning video clip where the camera is moving from left to right. Three frames are taken to form a reference frame. Then frame 2 is warped towards the reference frame. Distortions are very obvious in the warped frame. There is no turbulent motion in the video. However, a turbulent motion effect is produced by the time-averaging approach, which is supposed to suppress the turbulent motion. These examples highlight that the time-averaging reference approach fails when real motion exists.

2.5 Summary

In this chapter, atmospheric turbulence distortion has been analyzed from the principles in physics. From these analyses, various simplified models have been considered and form the basis for practical algorithms. Much work has been done in the area of restoration and atmospheric turbulence suppression. However, the performance of many methods are challenged when noise is present in the degraded images. Most methods are focused on the blurring effect while only a subset of investigators have considered geometric distortions. In this thesis, new methods to address both forms of distortion will be introduced. Since a practical model for the turbulence degradation is available, blur identification method can be used to identify the blurring parameter. Among the previous blur identification methods, generalized cross validation (GCV) is one of the top runners and it will be used as a baseline for the new



(a) (b) (c) (d) (e)

Figure 3: Three frames are taken from a simulated panning sequence. (a) Frame 1. (b) Frame 2. (c) Frame 3. (d) The average of the three frames, which is used as the reference frame (target) for frame 2 (source). (e) The registered frame 2. There is considerable distortion in the warped image; the boxes highlight two areas of significant distortion.

blur identification method that is to be presented in the next chapter. For geometric distortion suppression, comparison against the Frakes method [17] will be made.

CHAPTER III

KURTOSIS MINIMIZATION BASED BLUR IDENTIFICATION

3.1 Kurtosis and Smoothing

In this section we discuss blur identification in the context of blind deconvolution. We show the relationship between kurtosis and smoothness and discuss how kurtosis can be used as a metric for PSF parameter prediction.

The kurtosis of a random variable is defined as its normalized fourth central moment

$$k = \frac{E((x-\mu)^4)}{\sigma^4} \tag{39}$$

where μ is the mean of x, σ is its standard deviation, and $E(x)$ represents the expectation of the variable. The kurtosis measures the peakedness of a distribution. The Gaussian distribution ($k = 3$) has a moderate tail and is called mesokurtic. A platykurtic distribution has a small tail and its kurtosis is small ($k < 3$), and a leptokurtic distribution has a long tail ($k > 3$).

Many researchers have observed the non-Gaussian statistics of natural images. More specifically, it has been shown that the histograms of filtered images typically have single modes with heavy tails, characteristic of highly kurtotic or leptokurtic distributions. This statistical regularity has been observed for derivative filters, Gabor filters, wavelets and even small random kernel filters [78, 55, 32, 76]. Gluckman [25] examined the relationship between the phase structure of images and those observed statistical regularities. Correlations in the phase angles of an image are used to explain the non-Gaussian statistics of natural images. Here the phase structures are used to analyze the relationship between the kurtosis of an image and its smoothed

version.

To begin, consider that a bandlimited signal $f(x)$ can be represented by a finite Fourier series

$$f(x) = \sum_{i=1}^{n} m_i \cos(u_i x + \phi_i), \tag{40}$$

where m_i and ϕ_i are the magnitude and phase angle associated with integer frequency u_i. For simplicity, the 1D case is considered and $f(x)$ is assumed to be zero-mean. To further simplify the analysis, we can work with symmetrically reflected signal without loss of generality. This is because symmetric reflection does not change the kurtosis. Such reflected signals have real Fourier transform, which is attractive for analysis purposes. Thus, the phase of each frequency is either 0 or π, which can be represented by the signed magnitudes $s_i \in \{-1, 1\}$. Next we will represent the image as a sum of a low-frequency component $f_l(x)$ and a high-frequency component $f_h(x)$, where

$$f_l(x) = \sum_{i=1}^{n} m_{l,i} \cos(u_{l,i} x + \phi_{l,i}) \tag{41}$$

$$f_h(x) = \sum_{i=1}^{m} m_{h,i} \cos(u_{h,i} x + \phi_{h,i}) \tag{42}$$

The k^{th} central moment of f, which has zero mean, is:

$$\mu_k(f) = \frac{1}{|\Omega|} \int_{\Omega} f^k(x) dx \tag{43}$$

where Ω is the image domain. Defining the bi-variate moments of order 4 as $\mu_{40}, \mu_{04}, \mu_{31}, \mu_{13}, \mu_{22}$, where

$$\mu_{pq}(f) = \frac{1}{|\Omega|} \int_{\Omega} f_l^p(x) f_h^q(x) dx, \tag{44}$$

the 4^{th} central moment of f is then

$$\begin{aligned}\mu_4(f(x)) &= \mu_4(f_l(x) + f_h(x)) \\ &= \frac{1}{|\Omega|}\int_\Omega (f_l(x) + f_h(x))^4 dx \\ &= \mu_{40} + 4\mu_{31} + 6\mu_{22} + 4\mu_{13} + \mu_{04}\end{aligned} \tag{45}$$

$$\mu_{31} = \frac{3}{4}\sum_{i,j,r,v}\{(m_{l,i}m_{l,j}m_{l,r}m_{h,v})(s_{l,i}s_{l,j}s_{l,r}s_{h,v})I(i,j,r,v)\} \tag{46}$$

$$\mu_{13} = \frac{3}{4}\sum_{i,j,r,v}\{(m_{l,i}m_{h,j}m_{h,r}m_{h,v})(s_{l,i}s_{h,j}s_{h,r}s_{h,v})I(i,j,r,v)\} \tag{47}$$

$$\mu_{22} = \mu_{20}\mu_{02} + \frac{1}{2}\sum_{i,j,r,v}\{(m_{l,i}m_{l,j}m_{h,r}m_{h,v})(s_{l,i}s_{l,j}s_{h,r}s_{h,v})I(i,j,r,v)\} \tag{48}$$

where $I(i,j,r,v)$ is an indicator function as introduced in [25]

$$I(i,j,r,v) = \begin{cases} 1 & \text{if } (u_i - u_j) = (u_r - u_v) \\ 0 & \text{otherwise.} \end{cases} \tag{49}$$

Intuitively, since DCT is an orthogonal transformation, $(u_i - u_j) \neq (u_r - u_v)$ means that $(u_i - u_j)$, $(u_r - u_v)$ are uncorrelated. Thus, there is no contribution to the moment and the corresponding indicator function is zero. The moments $\mu_{13} \approx 0$ and $\mu_{31} \approx 0$ because $(u_i - u_j) = (u_r - u_v)$ usually does not hold when three of the four frequencies $\{i,j,r,v\}$ lie in a single band (low or high) [25]. In the frequency plane, low frequencies are in a small inner circle (most of the signal energy is in low frequency band) and the high frequencies are outside the circle. When three points are selected inside the circle and the fourth point is selected outside the circle, the four points normally do not form a parallelogram ($(u_i - u_j) \neq (u_r - u_v)$).

The kurtosis of $f(x)$ is then

$$k_f = \frac{\mu_4(f)}{\mu_2^2(f)} \tag{50}$$

$$= \frac{\mu_{40}(f_l) + \mu_{40}(f_h) + 6(\mu_{20}\mu_{02}) + 3C}{(\mu_{20}(f_l) + \mu_{20}(f_h))^2} \tag{51}$$

Table 1: Examination of the condition in Equation (54) on the image classes.

Image Class	Total Number	Number of counterexamples	Ratio
Natural	60	6	.90
City	70	2	.97

where $C = \sum_{i,j,r,v}\{(m_{l,i}m_{l,j}m_{h,r}m_{h,v})(s_{l,i}s_{l,j}s_{h,r}s_{h,v})I(i,j,r,v)$. The kurtosis of $f_l(x)$ and $f_h(x)$ are:

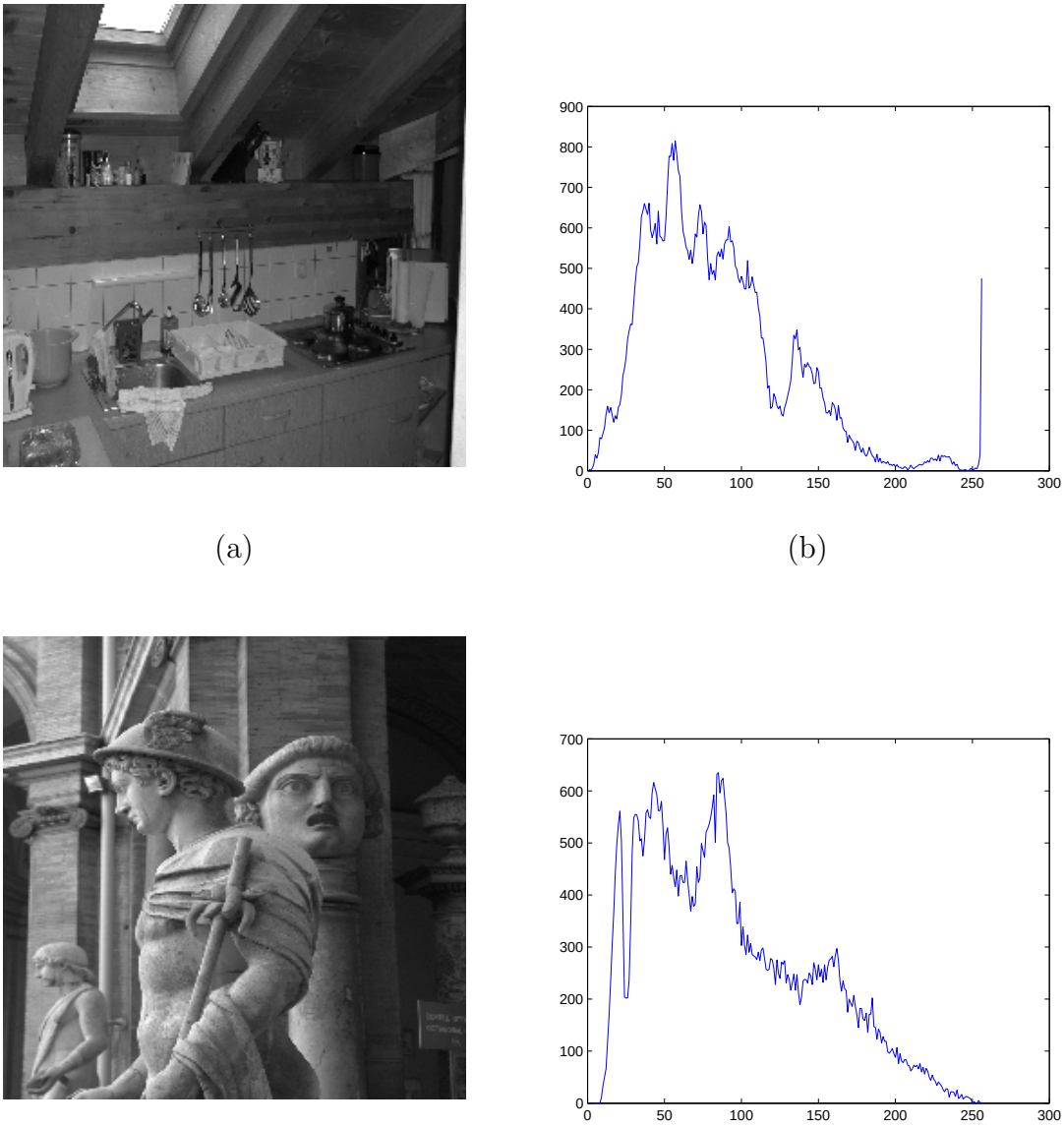
$$k_{f_l} = \frac{\mu_{40}(f_l)}{\mu_{20}^2(f_l)} \tag{52}$$

$$k_{f_h} = \frac{\mu_{40}(f_h)}{\mu_{20}^2(f_h)}. \tag{53}$$

From Equation (50) and Equation (52), the condition for $k_{f_l} > k_f$ is

$$3C < \mu_{20}(f_l)\mu_{20}(f_h)(2k_{f_l} - 6) + (k_{f_l} - k_{f_h})\mu_{20}^2(f_h) \tag{54}$$

Most natural images satisfy this condition (Equation 54). Since the point spread function (PSF) is typically a low pass filter, for natural images, blurring will increase the kurtosis. A smoothed (blurred) image has a higher kurtosis than the unblurred original image. Two groups of man-made objects (buildings) and natural (landscape) images were used to test the condition in Equation 54. Table 1 summarizes the results. It can be seen that most images meet the condition. Figure 4 shows examples of man-made objects images that do not comply with the condition. As a comparison, two images from the same group that meet the condition are shown in Figure 5. The histograms are drawn next to the images. Figure 6 and Figure 7 show both positive and negative examples from the natural image group. The histograms of those images that do not satisfy the condition are approximately unimodal (Figure 4 (b),(d) and Figure 6 (d)) or one of the peaks is bigger than the others as in Figure 6 (b). There are also some common characteristics for the images whose kurtosis increases as they are smoothed. Usually there is more than one peak and none of the histogram peaks is significantly higher than the others (Figure 5 (b),(d) and Figure 7 (b),(d)).

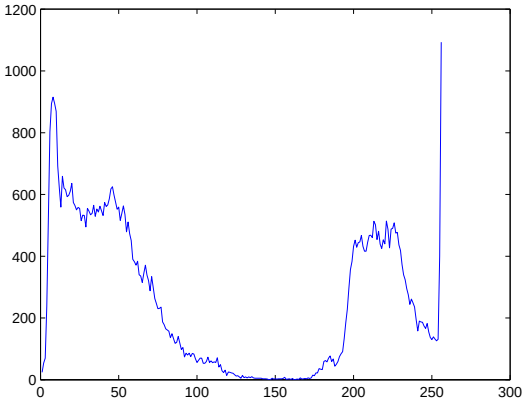


(a) (b)

(c) (d)

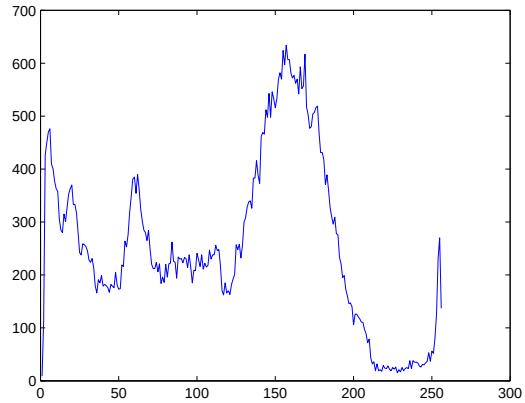
Figure 4: Counterexamples in the man-made object image data set where the kurtosis decreases as the image is blurred. The corresponding image histograms are plotted next to each image.

(a)



(b)

(c)



(d)

Figure 5: Positive examples of man-made structures where the kurtosis increases when the image is blurred. The corresponding histograms are plotted next to each image.

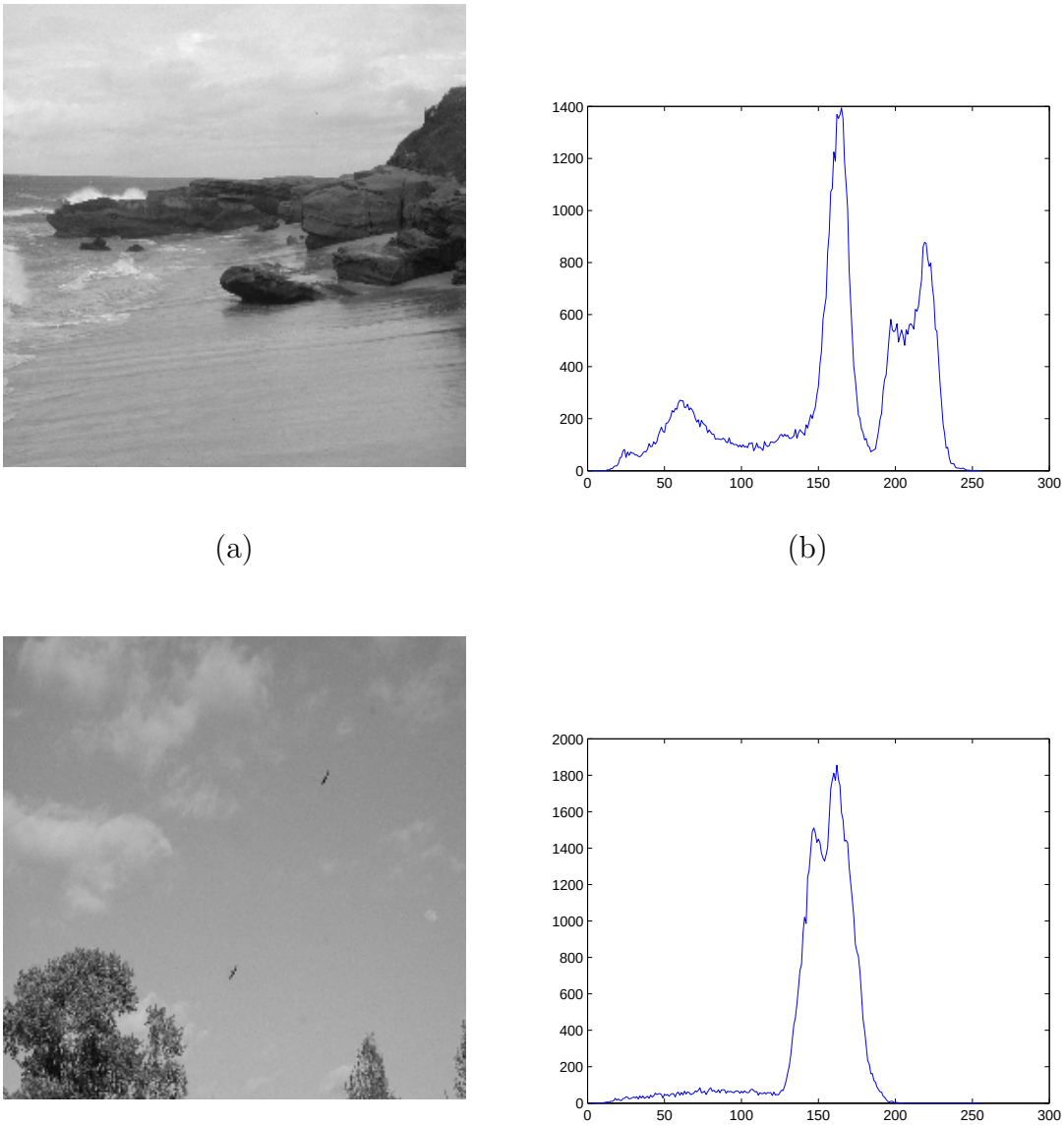


Figure 6: Counterexamples in the natural image data set where the kurtosis decreases when the image is blurred. The corresponding histograms are plotted next to each image.

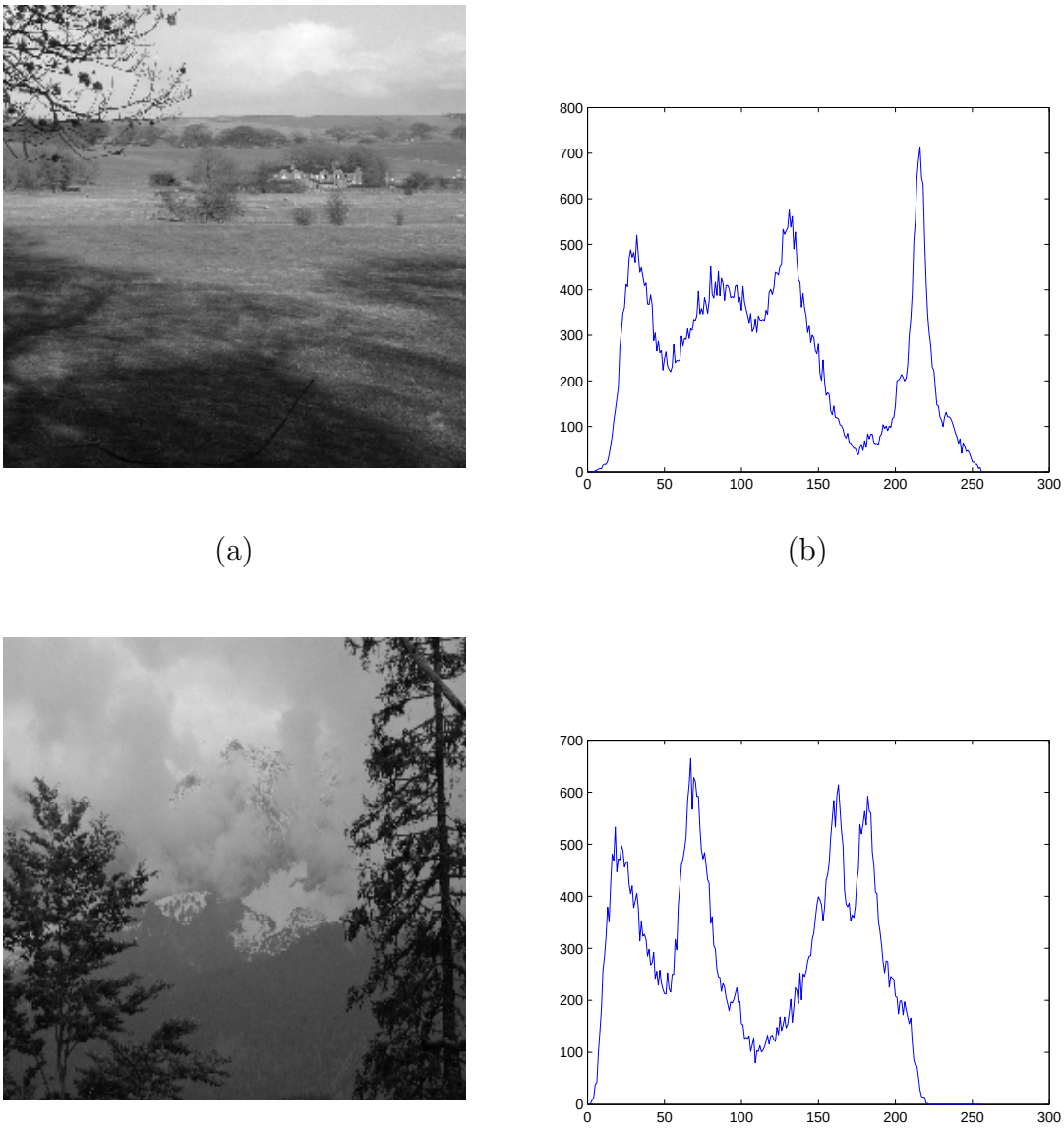


Figure 7: Positive examples of the natural images where the kurtosis increases when the image is blurred. The corresponding histograms are plotted next to each image.

3.2 Kurtosis Minimization for Blur Identification

The above analysis showed that an original unblurred image usually has a lower kurtosis than a blurred one. Based on this property, we can optimize the restoration filter parameter. The rationale is as follows. The restored image obtained using a model with the correct blurring parameter should be closer to the original image than one that uses an incorrect value. A restoration based on the incorrect parameter value should be less sharp, i.e. smoother and the corresponding kurtosis should be higher.

In this thesis, we consider the following problem statement. Given a noisy blurred image g with a known functional form for the PSF, we seek to find the best estimate of the blur parameter $\vec{\theta}$. The parameter is searched within a reasonable space Ω. At each step in the search loop, the image is deblurred using a Wiener restoration filter $G(u, v)$ or other non-blind restoration algorithm such as a regularized filter that assumes the PSF is known and the kurtosis of the deblurred image $\hat{f}(\vec{\theta})$ is computed and saved. Then the deblurred image with the smallest kurtosis is chosen as the final restored image and the corresponding parameter is regarded as the identified blurring parameter $\vec{\theta}_k$.

The Wiener restoration filter can be defined in the frequency domain as

$$G(u, v) = \frac{H^*(u, v)}{|H(u, v)|^2 + \frac{P_n(u,v)}{P_f(u,v)}}, \tag{55}$$

where P_f and P_n are the power spectral density of the signal and noise, respectively. Since practically both $P_n(u, v)$ and $P_f(u, v)$ are not known, a more realistic form of the Wiener filter is

$$G(u, v) = \frac{H^*(u, v)}{|H(u, v)|^2 + nsr}, \tag{56}$$

where the scalar nsr is the noise-to-signal ratio. The Wiener filter tends to behave as a bandpass filter. In the region where the signal is very strong relative to the noise, $P_n(u, v)/P_f(u, v) \approx 0$ and the Wiener filter approximates $H^{-1}(u, v)$ (the inverse filter for the PSF). In the region where the signal is very weak, $P_n(u, v)/P_f(u, v) \to \infty$ and

$G(u,v) \to 0$ so that noise will not be amplified in the restored image. Such bandpass characteristics of the Wiener filter are important to kurtosis minimization based blur identification. If the restoration filter amplifies noise, such as an inverse filter, the kurtosis of the corresponding restored image will be low and the minimal kurtosis will not respond to the image restored with the correct parameter. In that case, kurtosis minimization might fail because of the improper restoration algorithm.

In practice, the noise-to-signal ratio is often approximated by the noise-to-blurred-signal ratio, which can be estimated either in the spatial domain or in the frequency domain. In the spatial domain, the average of all the local estimated variances within a window such as 3×3 or 5×5 can be used as an estimate of the noise variance. Then the blurred image variance is estimated using the difference between the overall variance and the noise variance since noise and blurred image are assumed to be uncorrelated. The *nsr* can also be estimated in the frequency domain where noise is assumed to be white while the image is assumed to be band-limited. The noise energy is estimated in the high-frequency band where the image energy is very low.

An example is shown in Figure 8 where the CAMERAMAN image is blurred with a block filter of length 7, which simulates linear motion blur. Gaussian noise (variance = 0.02) is then added to the result. The *nsr* is estimated as 0.0149 in frequency domain. It can be seen that the image is successfully deblurred even when the practical form of the Wiener filter is used.

The kurtosis minimization rule can be summarized as

$$\vec{\theta}_k = \arg\{\min_{\vec{\theta}\in\Omega} k(\hat{f}(\vec{\theta}))\}. \tag{57}$$

The benchmark rule is the PSNR maximization rule,

$$\vec{\theta}_{\text{PSNR}} = \max_{\vec{\theta}\in\Omega} \text{PSNR}(\hat{f}(\vec{\theta})) \tag{58}$$

PSNR is defined as

$$\text{PSNR}(\hat{f}) = 10\log_{10} \frac{\sum_{i=1}^{M}\sum_{j=1}^{N} 255^2}{\sum_{i=1}^{M}\sum_{j=1}^{N}(f(i,j)-\hat{f}(i,j))^2} \tag{59}$$

(a)

(b)

Figure 8: (a) The noisy blurred CAMERAMAN image (horizontal linear motion blur of length 7, Gaussian noise variance is 0.02). (b) The restored image using a Wiener filter (nsr is estimated as 0.0149).

where M and N are the height and width of the image.

3.3 Reducing Ringing in Deblurred Images

In this work, the Wiener deblurring filter in Equation (55) is implemented with the discrete Fourier transform (DFT). Because the DFT is a sampled version of the DTFT, wrap-around effects exist when filtering is performed. That is to say, the DFT multiplication that implements the filtering results in circular convolution in the spatial domain, often leading to ringing effects at the boundaries. As an example, see Figure 9. The ellipses highlight the boundary discontinuities associated with circular convolution. Such boundary discontinuities contribute to the ringing effect

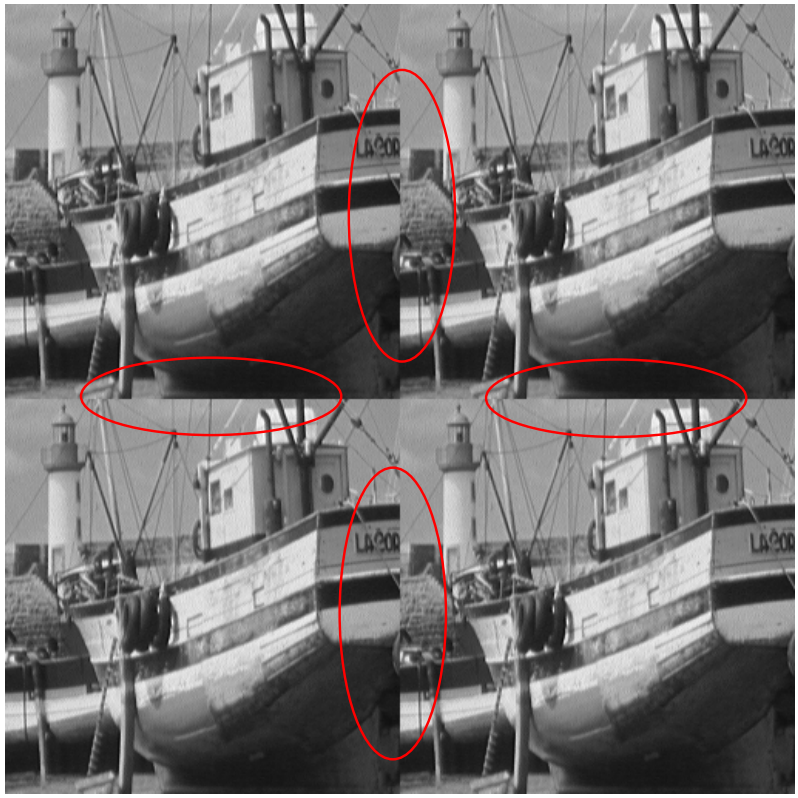
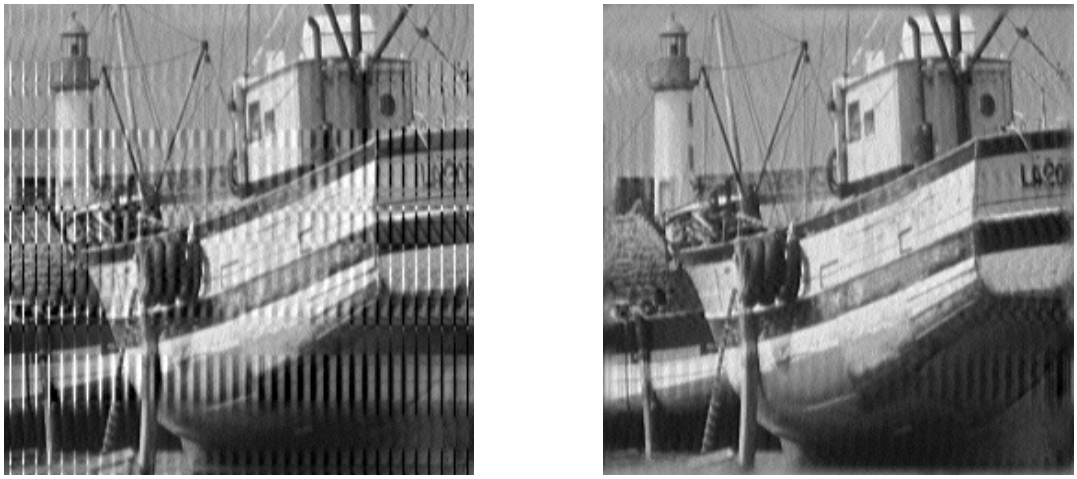


Figure 9: The periodicity implicit in the DFT representation leads to boundary discontinuities.

in deblurred images, which is illustrated in the following example. The BOAT image is blurred horizontally with a block filter of length 7. The blurred image is shown in Figure 10 and the center part of the image is cropped as the image region to be restored. No noise has been added to the blurred image. Figure 11(a) shows the result when DFT-based restoration is performed without zero padding or any form of boundary preprocessing.

Figure 10: The motion blurred CAMERAMAN image. The length of the motion blur is 7.

To reducing ringing effect, the image boundary can be preprocessed before it is



(a) (b)

Figure 11: (a) The deblurred BOAT image without boundary preprocessing, PSNR = 19.43 dB. (b) The deblurred BOAT image after the boundary smoothing, PSNR = 31.72 dB.

deblurred. Some common methods for dealing with boundary discontinuities are interpolation/smoothing at the boundaries, windowing, and symmetric extension.

Figure 11(b) shows the result when a portion of the boundary is smoothed by a Gaussian filter. In the boundary smoothing method, part of the boundary of the image is smoothed without increasing the size of the image. Symmetric extension increases the size of the image and thus increases the computational cost. The complexity of the fast Fourier transform (FFT) is $O(N^2 \log N)$ for an $N \times N$ image. On the other hand, the boundary smoothing method does not increase the size of the image, but may not perform quite as well.

3.4 Efficient Searching Method

To find the blurring parameter associated with the minimum kurtosis, simple and efficient searching methods may be employed, such as the bisection method or golden section search. The bisection method involves a division of a given interval into two

equal parts (halves). A simple bisection procedure for iteratively converging on a solution which is assumed to lie inside some interval $[a, b]$ proceeds by evaluating the function at the midpoint of the original interval $x = (a + b)/2$ and testing to see in which of the subintervals $[a, (a+b)/2]$ or $[(a+b)/2, b]$ the solution lies. The procedure is then repeated with the new interval as often as needed to locate the solution to the desired accuracy which is often measured by the range of the subintervals.

3.5 Experiments on Simulated and Real Turbulence Blurred Images

The LENA image is blurred with the turbulence OTF model ($\lambda = 0.008$) and Gaussian random noise ($\sigma = 0.01$) is added to the blurred image as in Figure 13. The blurred signal to noise ratio BSNR in this case is as low as 1dB. BSNR is defined as

$$\text{BSNR} = 10 \log \frac{\sigma_b^2}{\sigma_n^2} \tag{60}$$

where b denotes the blurred signal and n is the noise.

The parameter search space is $\Omega_\lambda = \{\lambda : 0.001i | i = 0, 1, 2, \ldots, 30\}$. Figure 12 shows the kurtosis and the PSNR as functions of the λ. Kurtosis matches very well with PSNR in this example.

Figure 13 shows the degraded and the restored LENA image with the λ estimated as 0.008, which is the value of λ that was used to create the degraded LENA image. Figure 14 shows the example on a real tropospheric turbulence degraded image. The λ is estimated as 0.0018.

For comparative purposes, the GCV blur identification method for the identification of turbulence blurring parameter λ is also implemented. A collection of 100 images was used for the experiments. The images were blurred with $\lambda = 0.001$ and random Gaussian noise was added at the level of $BSNR = 20dB$. Some examples of the degraded images are shown in Figure 15.

Since the original images are available, PSNR of the restored images from both

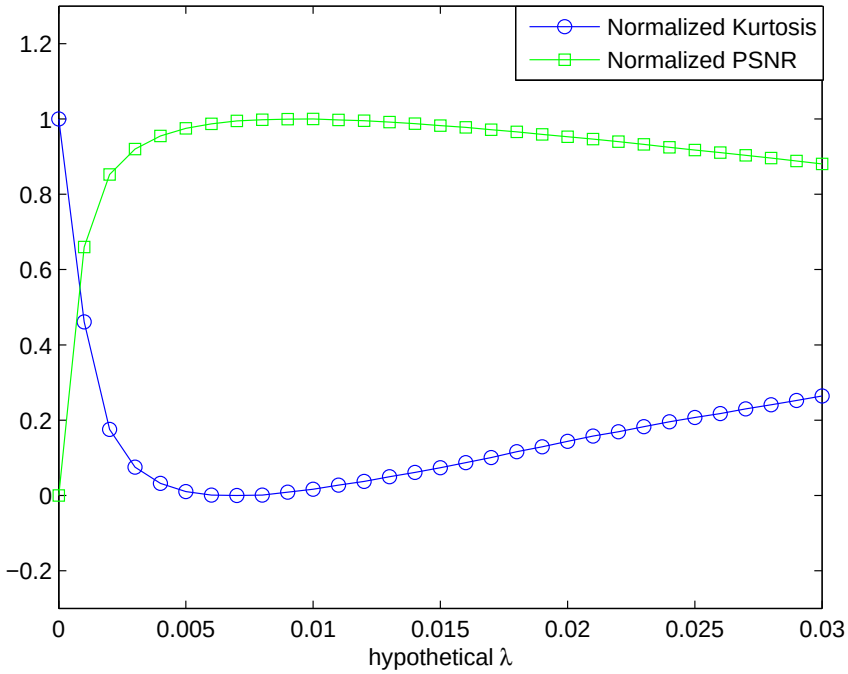
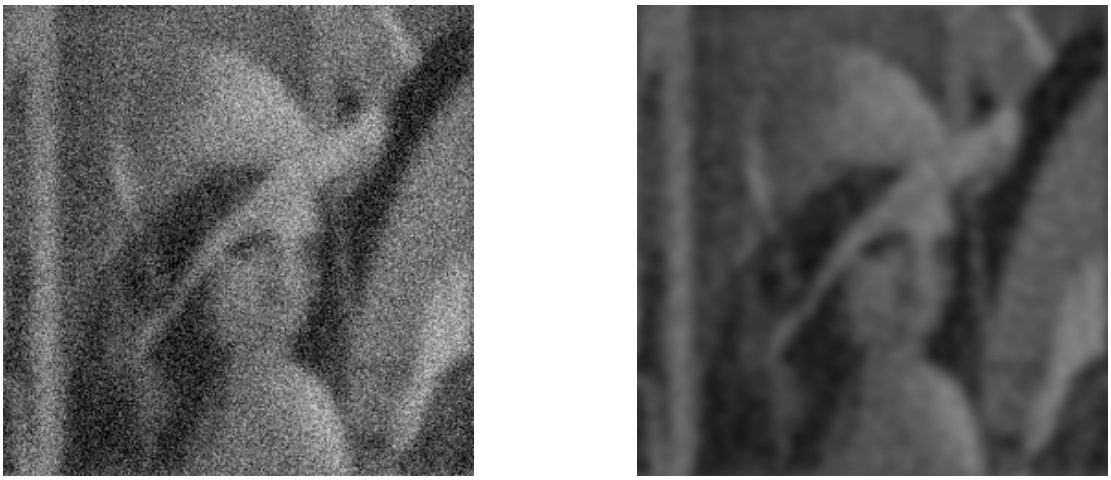
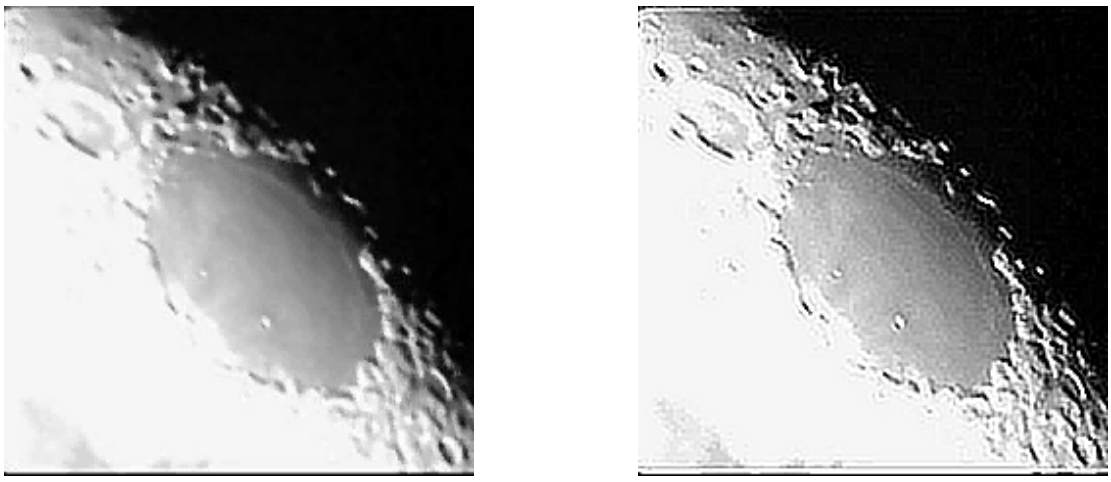


Figure 12: The Kurtosis and the PSNR of the restored LENA image as λ varies from 0 to 0.03 in steps of 0.001. The estimated λ is 0.008.



(a) (b)

Figure 13: (a) The degraded noisy blurred LENA image ($\lambda = 0.01, \sigma = 0.01$). (b) The restored image (estimated $\lambda = 0.01$).



(a) (b)

Figure 14: (a) A real turbulence degraded image. (b) The restored image (estimated $\lambda = 0.0018$).

Figure 15: Example of degraded images that are used for comparison of the kurtosis minimization method with generalized cross-validation.

methods can be computed to assess the restoration performance. Figure 16 plots the PSNR comparison of the two methods. Among the 100 images, kurtosis minimization outperforms GCV on 90 images, while GCV outperforms kurtosis minimization on 10 images. On average, Kurtosis minimization outperforms GCV by around 1dB. The algorithms run in Windows XP on a notebook computer. The CPU is a 1400MHz Pentium M processor, with 768MB of RAM. The computation time is roughly the same: on average, the GCV algorithm takes 0.20 sec to evaluate the error of a candidate λ, while kurtosis minimization takes 0.15 sec to evaluate the kurtosis of the image restored with an assumed λ. Both GCV and kurtosis minimization were computed in the DFT domain, therefore, the computation complexity is roughly the same. Because kurtosis is measured on the restored image, there is no need to perform restoration again when the parameter is identified since the image is already resorted using the identified parameter. On the other hand, GCV is not directly computed from the restored image and an additional step is required to get the final restored image.

3.6 Gaussian Blur Identification

Since an image is stored as a matrix of discrete pixels we need to create a discrete approximation to the Gaussian function before convolution can be performed. A rotationally symmetric discrete Gaussian low-pass filter of size h_{size} with standard deviation σ (positive) is used to simulate a Gaussian blur. The $(i,j)^{th}$ coefficient of the Gaussian filter is first computed as

$$h_{i,j} = e^{-\frac{i^2+j^2}{2\sigma^2}} \tag{61}$$

where $(i,j) \in \{-h_{size}, \ldots, 0, \ldots, h_{size}\} \times \{-h_{size}, \ldots, 0, \ldots, h_{size}\}$. Then $h_{i,j}$ is normalized so that $\sum_i \sum_j h_{i,j} = 1$. In this experiment, the CAMERAMAN image is blurred by the Gaussian filter ($h_{size} = 5, \sigma = 2.5$). Gaussian noise ($\sigma = 0.0025$) is added to the blurred image. For the restoration, the gaussian model is assumed but the two parameters h_{size} and σ are treated as unknown. The search space for h_{size} is

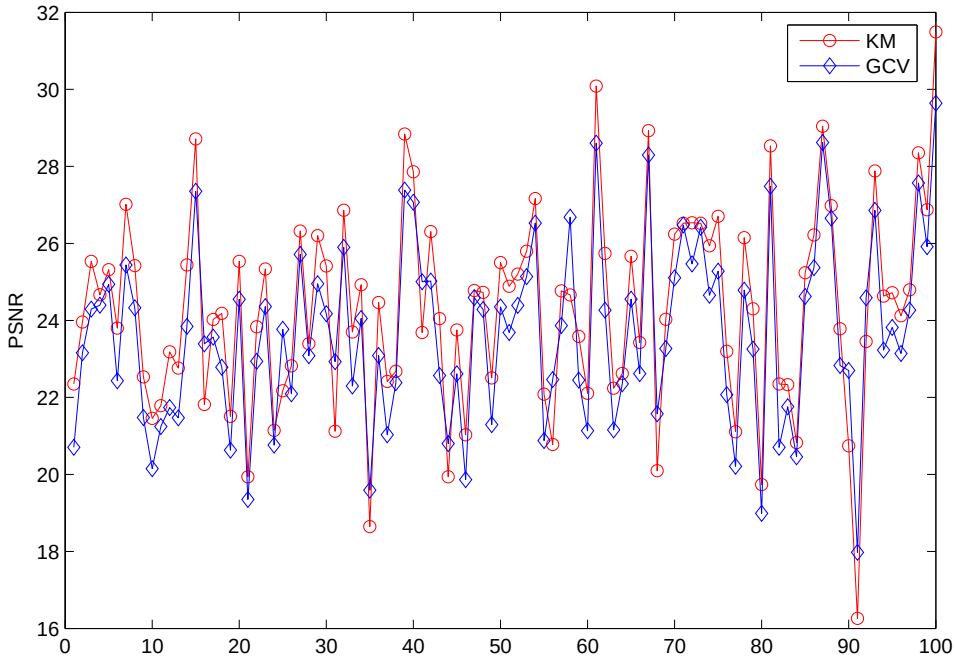


Figure 16: PSNR comparison of GCV and Kurtosis minimization on the image set. On average, Kurtosis minimization outperforms GCV by around 1dB.

Table 2: PSNR and Kurtosis of $\hat{f}(h_{size}, \sigma)$, the restored CAMERAMAN image with the hypothetical Gaussian blur size h_{size} and standard deviation σ.

h_{size}	$PSNR$				
2	16.974	16.974	16.974	16.974	16.974
3	16.192	16.105	16.078	16.064	16.06
4	15.105	15.842	16.27	16.501	16.671
5	17.163	22.964	25.09	24.273	23.248
6	15.6	16.204	14.851	13.787	13.224
7	13.071	14.858	12.107	10.891	10.29
σ	1.5	2	2.5	3	3.5
h_{size}	$Kurtosis$				
2	2.2452	2.2452	2.2452	2.2452	2.2452
3	2.2476	2.2486	2.2497	2.2525	2.2545
4	2.2323	2.2235	2.2163	2.2139	2.2128
5	2.1852	2.1518	2.1417	2.1487	2.1554
6	2.2062	2.1642	2.194	2.2115	2.2055
7	2.1948	2.1768	2.1929	2.1897	2.1701
σ	1.5	2	2.5	3	3.5

$\Omega_{h_{size}} = \{2, 3, 4, 5, 6, 7\}$ and that for σ is $\Omega_\sigma = \{1.5, 2, 2.5, 3, 3.5\}$. Table 2 shows that the minimum value of the kurtosis is achieved for $(5, 2.5)$. These are also the values where the PSNR is maximum, and these are the "correct" value in that these were the values used to generate the data.

The USC-SIPI image database [1] is a popular database for image processing research. Table 3 summarizes the Gaussian blur identification results using kurtosis minimization on some of the USC images. Different Gaussian blur PSF parameters were used and three different levels of noise were added to the blurred images. Kurtosis minimization performs very well in identifying the correct blurring parameters. Noise usually has a negative impact on the performance of identification. As shown in Table 3, the true blurring parameter for the Elaine image is $(6 \times 6, 2.0)$, but kurtosis minimization identified the parameter as $(7 \times 7, 1.5)$ at the noise level of 30 bB BSNR. Since the searching step for h_{size} and σ is 1 and .5 respectively, the identified parameter is next to the real one.

Table 3: Gaussian blur identifications on USC images using kurtosis minimization

Image	Actual Parameter(h_{size},σ)	30 dB BSNR	40 dB BSNR	60 dB BSNR
Cameraman	4×4, 2.0	4×4, 2.0	4×4, 2.0	4×4, 2.0
Boat	6×6, 3.0	6×6, 3.0	6×6, 3.0	6×6, 3.0
Baboon	5×5, 2.5	5×5, 2.5	5×5, 2.5	5×5, 2.5
Lena	5×5, 2.0	5×5, 2.0	5×5, 2.0	5×5, 2.0
Elaine	6×6, 2.0	7×7, 1.5	6×6, 2.0	6×6, 2.0

3.7 Out-of-focus Blur Identification

Out-of-focus blur is usually modeled as a circular averaging filter (pillbox) within a square $(2*r+1) \times (2*r+1)$ matrix, where r is the radius of the blur. The coefficient of an out-of-focus blur filter is:

$$h_{i,j} = \begin{cases} \frac{1}{\pi r^2} & if\ \sqrt{i^2+j^2} \leq r; \\ 0 & otherwise. \end{cases} \tag{62}$$

Two examples are shown here, whose radii are $r = 2$ and $r = 3$, respectively. The CAMERAMAN image is blurred with a pillbox filter. Gaussian noise ($\sigma = 0.0025$) is added to the blurred image. The search space for r is $\Omega_r = \{r : 1 + i * 0.1 | i = 0, 1, 2, \ldots, 30\}$.

Figure 18 shows the degraded image and the three images restored with three hypothetical radii. The kurtosis of the image that was restored with r=3.1 is the minimum as plotted in Figure 17. The kurtosises of the other two images restored with $r = 2$ and $r = 5$ are both higher.

As seen in Figure 17, although the estimated radius ($r = 3.1$) is different from the real radius ($r = 3$) the PSNR for that value is higher. The reason is that the Wiener filter is not the ideal one (Equation 55) since the original image is unknown and the practical form of the Wiener filter (Equation 56) is used instead. Moreover, the noise estimation might not be accurate enough. In Figure 17 (b), the kurtosis correctly identifies the radius as 2 but this does not correspond to the maximum PSNR.

Table 4 summarizes more out-of-focus blur identification results using kurtosis

Table 4: Out-of-focus blur identifications on USC images using kurtosis minimization

Image	Actual Radius(r)	30 dB BSNR	40 dB BSNR	60 dB BSNR
Cameraman	2.5	2.5	2.5	2.5
Boat	3	3.1	3	3
Baboon	3.2	3.2	3.2	3.2
Lena	2.4	2.4	2.4	2.4
Elaine	3.5	3.5	3.5	3.5

Table 5: Linear motion blur identifications on USC images using kurtosis minimization

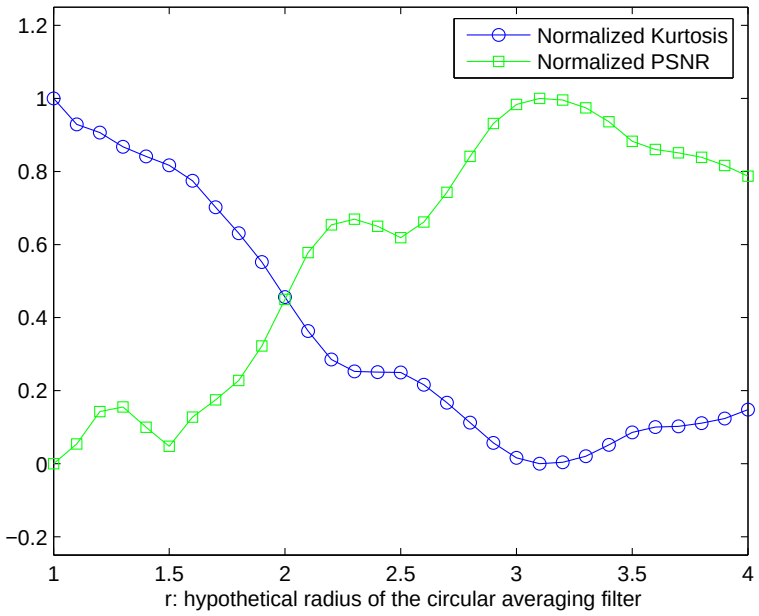
Image	Actual Length(L)	30 dB BSNR	40 dB BSNR	60 dB BSNR
Cameraman	8	8	8	8
Boat	6	6	6	6
Baboon	9	9	9	9
Lena	7	7	7	7
Elaine	5	5	5	5

minimization on the selected USC images.

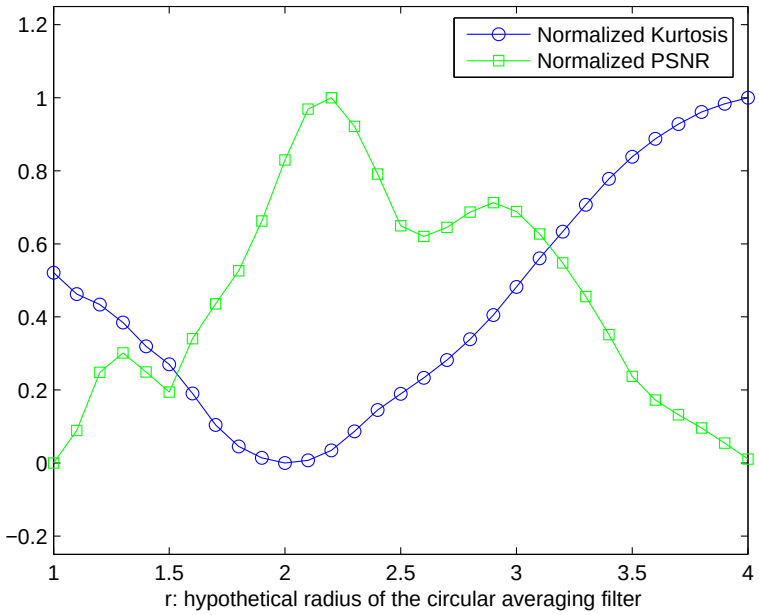
3.8 Linear Motion Blur Identification

Linear horizontal motion blur is modeled as an averaging filter within a row vector of $1 \times L$, where L is the length of the motion blur. The filter approximates the linear motion of a camera by L pixels in the horizontal direction. Two examples are shown here. The true length are 7 and 9 respectively. The cameraman image is blurred with the motion blur filters. Gaussian noise ($\sigma = 0.005$) is added to the blurred images. The search space for L is $\Omega_L = \{L : 3 + i | i = 0, 1, 2, \ldots, 10\}$. As seen in Figure 19, the estimated lengths are 7 and 9 respectively. Figure 20 shows the degraded images and three images restored with three different assumptions about the length of the motion blur. The one with the correctly identified length has the minimum kurtosis.

Table 5 summarizes more linear horizontal motion blur identification results using kurtosis minimization on those selected USC images. There is no error in the identification.

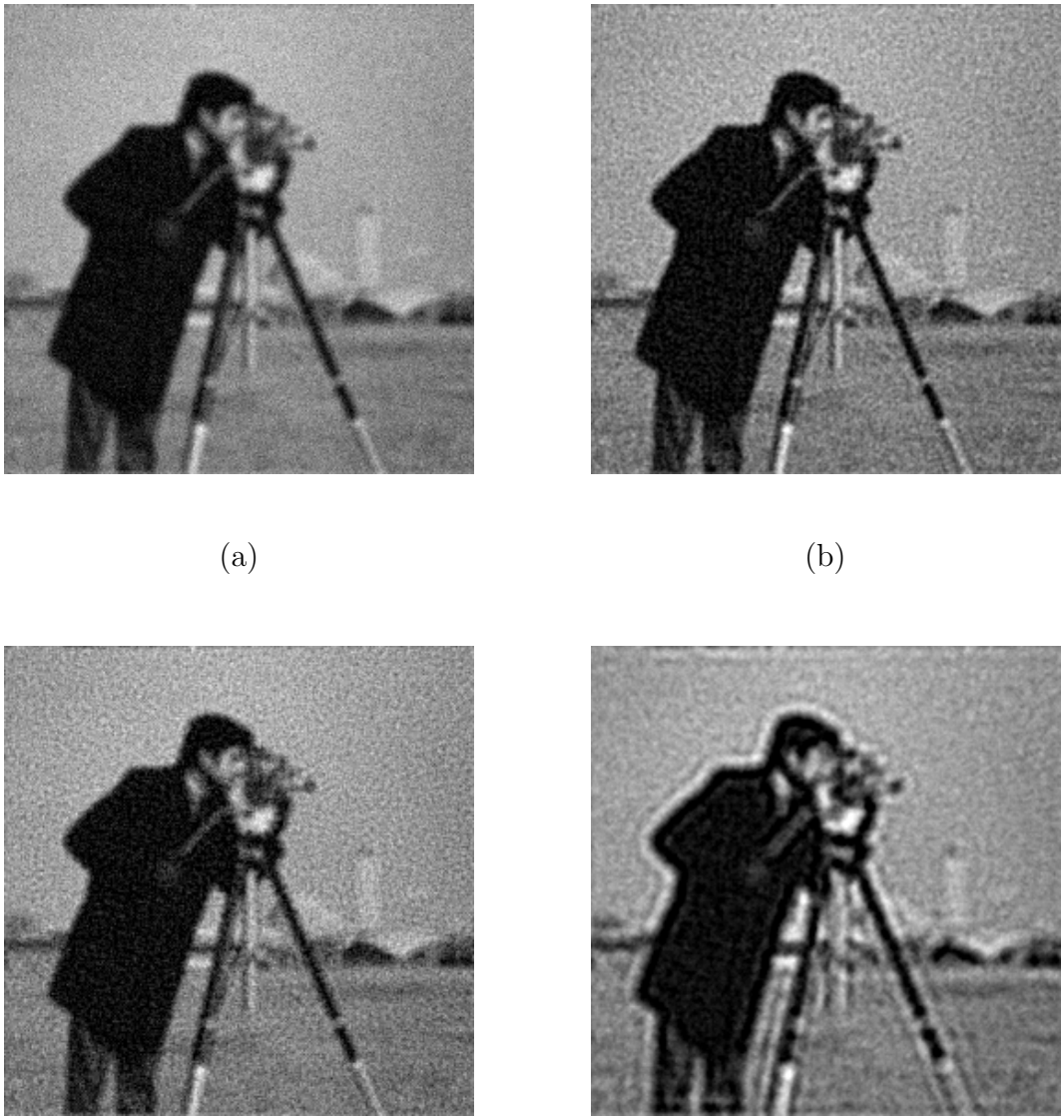


(a)



(b)

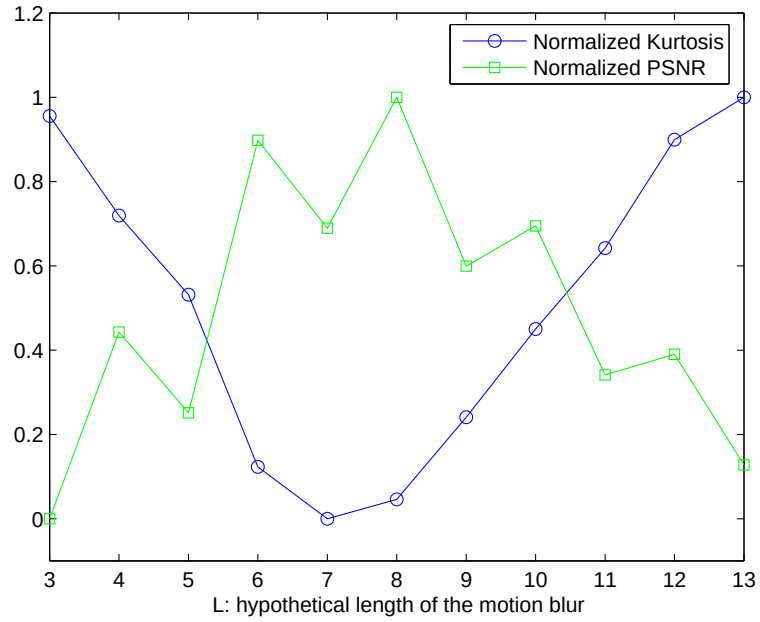
Figure 17: Out-of-focus blur identification. (a) $r = 3$, estimated radius: 3.1. (b) $r = 2$, estimated radius: 2.



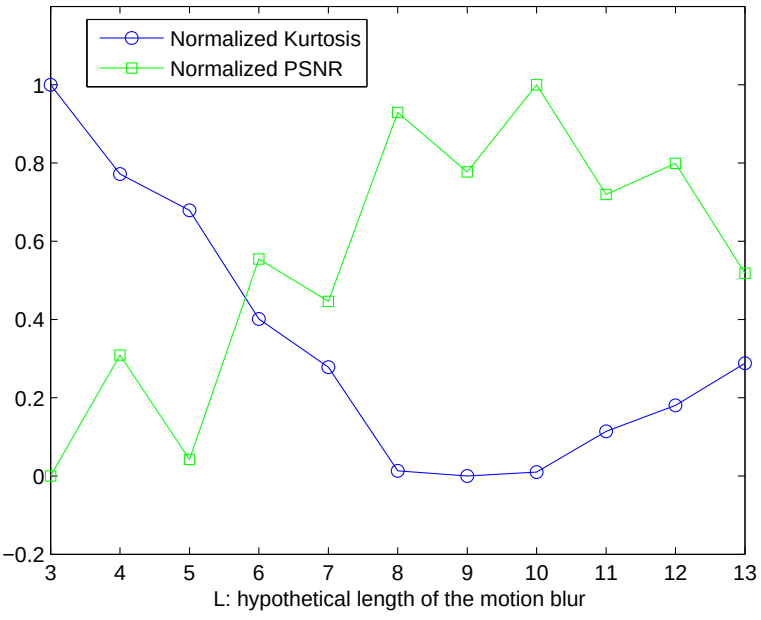
(a) (b)

(c) (d)

Figure 18: (a) The cameraman image that was degraded by out-of-focus blur ($r = 3$) and Gaussian noise ($\sigma = 0.0025$). (b) The restored image with the estimated blur radius ($r = 3.1$). (c) The restored image with $r = 2$. (d) The restored image with $r = 5$.

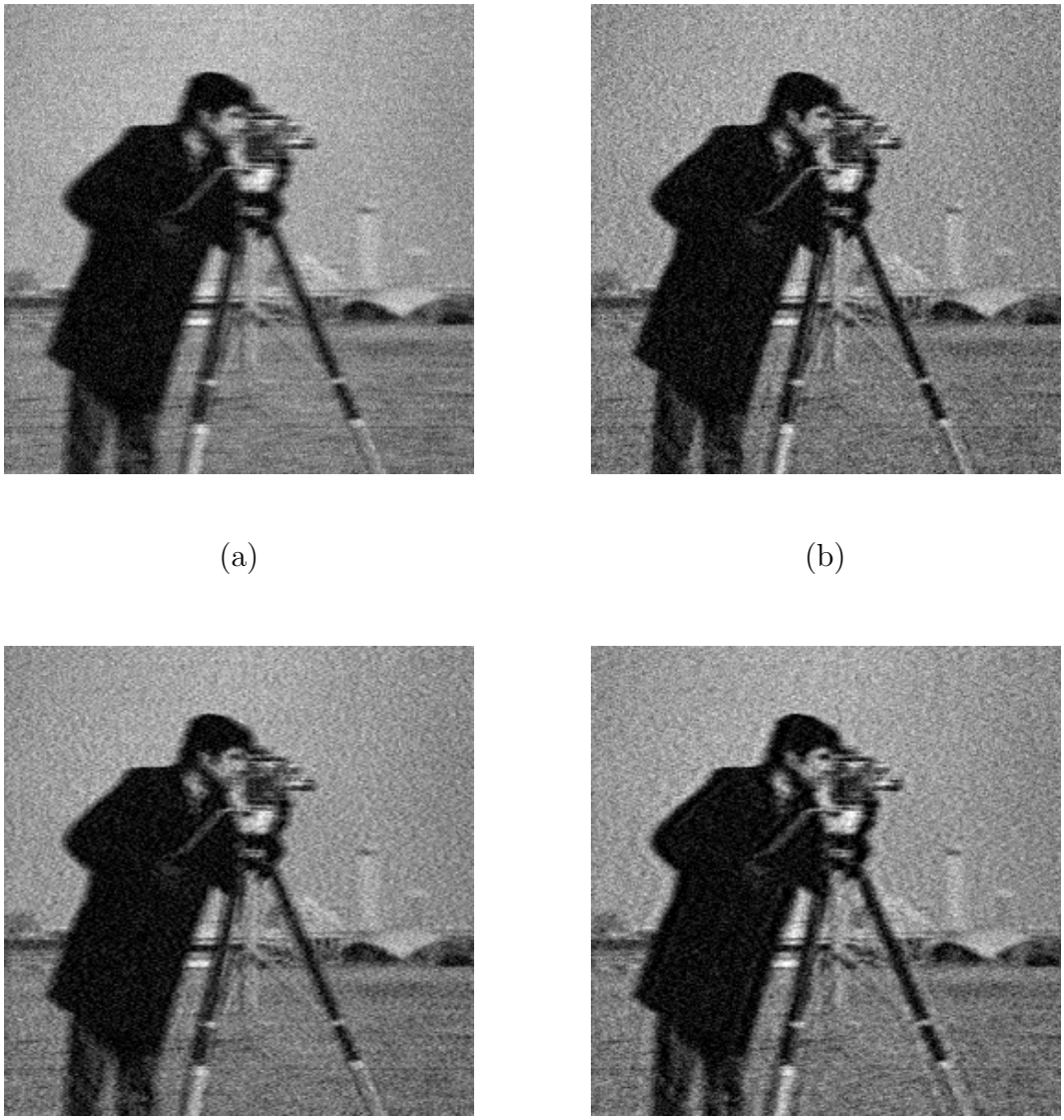


(a)



(b)

Figure 19: Linear motion blur identification. (a) $L = 7$, estimated length: 7. (b) $L = 9$, estimated length: 9.



(a) (b)

(c) (d)

Figure 20: (a) The cameraman image that was degraded by linear motion blur ($L = 7$) and Gaussian noise ($\sigma = 0.005$). (b) The restored image with the estimated length ($L = 7$). (c) The restored image with $L = 4$. (d) The restored image with $L = 9$.

3.9 Blur Identification with Incorrect Blur Model

In this experiment, the cameraman image was blurred with the turbulence model($\lambda = 0.0025$) as in Equation (24) and white noise ($\sigma = 0.002$) was added. Then the kurtosis minimization was used to identify the blur that was assumed to be an averaging filter with a square matrix. The search space of h_{size} is $\Omega_{h_{size}} = \{3, 4, 5, 6\}$. The estimated h_{size} of the blur is 5 and the PSNR of the restored image is 20.78 dB. The PSNR of the restored image using the correct model and parameter is 22.86 dB. This mismatch is not surprising since the model is not correct, but the image is still improved as shown in Figure 21. This illustrates that the technique is robust to model errors.

(a)

(b)

Figure 21: (a) Atmospheric turbulence blurred image. (b) Restored image using averaging filter blur model.

CHAPTER IV

SUPPRESSION OF ATMOSPHERIC TURBULENT MOTION

Atmospheric turbulence degradation can be viewed as having two major components: spatial blur and time varying geometric distortion. It has been shown that kurtosis minimization can be used to estimate the blurring parameter with a Wiener filter used to perform deblurring to address the spatial blur. For the time-varying geometric distortion, motion estimation can be used to compensate for turbulent motion within a certain time window.

4.1 Optical Flow

The most pervasive method of motion representation is the block based method used in the MPEG standard [4]. In this representation, vectors describe the movement of blocks of pixels. An alternative representation is optical flow [61, 9, 62, 44], which describes the movement of pixels within a video sequence. This motion is represented as a vector-one for each pixel.

Optical flow has found wide applications in areas such as pattern recognition, computer vision, and image processing.

4.1.1 Lucas-Kanade method

Optical flow methods try to calculate the motion between two image frames which are taken at times t and $t + \delta t$ at every pixel position. As a pixel at location (x, y, t) with intensity $I(x, y, t)$ will have moved by δx, δy and δt between the two frames, then under the assumption that the intensity of an object remains constant, the following

image constraint equation can be written:

$$I(x, y, t) = I(x + \delta x, y + \delta y, t + \delta t). \tag{63}$$

Assuming that the motion is small enough, the image constraint can be developed by expanding $I(x, y, t)$ in a Taylor series to get

$$I(x + \delta x, y + \delta y, t + \delta t) = I(x, y, t) + \frac{\partial I}{\partial x}\delta x + \frac{\partial I}{\partial y}\delta y + \frac{\partial I}{\partial t}\delta t + H.O.T. \tag{64}$$

where H.O.T. means higher order terms, which are usually small enough to be ignored. From these equations

$$\frac{\partial I}{\partial x}\delta x + \frac{\partial I}{\partial y}\delta y + \frac{\partial I}{\partial t}\delta t = 0 \tag{65}$$

or

$$\frac{\partial I}{\partial x}\frac{\delta x}{\delta t} + \frac{\partial I}{\partial y}\frac{\delta y}{\delta t} + \frac{\partial I}{\partial t}\frac{\delta t}{\delta t} = 0 \tag{66}$$

which results in

$$\frac{\partial I}{\partial x}V_x + \frac{\partial I}{\partial y}V_y + \frac{\partial I}{\partial t} = 0 \tag{67}$$

where Vx, Vy are the x, y components of the velocity or optical flow of $I(x, y, t)$ and $\frac{\partial I}{\partial x}, \frac{\partial I}{\partial y}$ and $\frac{\partial I}{\partial t}$ are the partial derivatives of the image at (x, y, t) with respect to the corresponding variables. Let us denote these derivatives as I_x, I_y and I_t in the following.

Thus

$$I_x V_x + I_y V_y = -I_t \tag{68}$$

or

$$\nabla I \cdot \vec{V} = -I_t \tag{69}$$

Since there are two unknowns (V_x, V_y) and one equation, they cannot be solved directly. This is known as the *aperture problem* in optical flow. It suggests that the optical flow equation is ill-posed. To find the optical flow another set of equations is needed to provide additional constraints. The additional constraint added in the Lucas and Kanade [54] solution is to assume that the optical flow is locally constant.

Assuming that the flow (V_x, V_y) is constant in a small window of size $m \times m$ with $m > 1$, which is centered at the point x, y and that the pixels are indexed as $1...n$, we can get a set of equations:

$$\begin{aligned} I_{x_1}V_x + I_{y_1}V_y &= -I_{t_1} && (70)\\ I_{x_2}V_x + I_{y_2}V_y &= -I_{t_2} && (71)\\ &\vdots && (72)\\ I_{x_n}V_x + I_{y_n}V_y &= -I_{t_n} && (73)\\ & && (74) \end{aligned}$$

Now the number of equations is larger than the number of unknowns (two); it is now an over-determined system:

$$\begin{bmatrix} I_{x_1} & I_{y_1} \\ I_{x_2} & I_{y_2} \\ \vdots & \vdots & \vdots \\ I_{x_n} & I_{y_n} \end{bmatrix} \begin{bmatrix} V_x \\ V_y \end{bmatrix} = \begin{bmatrix} -I_{t_1} \\ -I_{t_2} \\ \vdots \\ -I_{t_n} \end{bmatrix} \tag{75}$$

or

$$A\vec{V} = -b \tag{76}$$

To solve this over-determined system of equations we use the least squares method (pseudoinverse) that minimizes $\|A\vec{V} + b\|^2$:

$$A^T A\vec{V} = -A^T b \tag{77}$$

or

$$\vec{V} = -(A^T A)^{-1} A^T b \tag{78}$$

or

$$\begin{bmatrix} V_x \\ V_y \end{bmatrix} = - \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix}^{-1} (A^T I_t) \tag{79}$$

This indicates that the optical flow can be obtained by calculating the derivatives of the image with respect to all three variables (x, y, t). A weighting function $W(i, j)$, with $i, j \in [1, .., m]$ should be added to give more emphasis to the center pixel of the window. Gaussian functions are often preferred as the weighting functions. Other functions are also possible. Let W be a diagonal matrix with weights. The weighted least squares estimate is $\vec{V} = -(A^T W^2 A)^{-1} A^T W^2 b$. In addition for computing local translations, the flow model can also be extended to affine image deformations as shown in the control grid interpolation motion model, which will be discussed in section 4.2.

When applied to image registration the Lukas-Kanade method is usually applied in a coarse-to-fine iterative manner, in such a way that the spatial derivatives are first computed at a coarse scale (a smaller image) in a pyramid. The source image is warped [87] by the computed deformation, and iterative updates are then conducted at successively finer scales. An image pyramid [36, 35, 72] represents a digital image in different resolution levels. A low pass filter, e.g. a Gaussian filter, is first applied to the image. Downsampling is then used to create a reduced resolution image. The multiresolution Lucas-Kanade algorithm can be summarized as follows

- Compute Lucas-Kanade optical flow at the highest level 0.
- At level i
  - Take flow u_{i-1},v_{i-1} from level $i-1$.
  - Bilinearly interpolate it to create u_i^*,v_i^* of twice the resolution for level i.
  - Multiply u_i^*,v_i^* by 2.
  - Warp the source image based on u_i^*,v_i^*.
  - Compute the motion $u_i^{'}$, $v_i^{'}$ between the warped source image and the target image at the current level (correction in the flow) by the LK algorithm.

– Add the correction to get the flow for the current level: $u_i = u_i^* + u_i^{'}; v_i = v_i^* + v_i^{'}$.

One of the common characteristics of the Lucas-Kanade algorithm and other local optical flow algorithms, is that it does not compute a high density of flow vectors, i.e. the flow fades out quickly across motion boundaries and the inner parts of homogenous areas show little motion. Its advantage is the computation efficiency.

4.1.2 Horn-Schunck method

An alternative to the Lucas-Kanade algorithm, the Horn-Schunck method [31] of estimating optical flow is a global method that introduces a global smoothness constraint to solve the aperture problem.

A global objective (or energy) function is sought to be minimized. This function is given as

$$f = \int ((\nabla I \cdot \vec{V} + I_t)^2 + \alpha(|\nabla V_x|^2 + |\nabla V_y|^2))\mathrm{d}x\mathrm{d}y \tag{80}$$

where $\nabla I = \begin{bmatrix} I_x \\ I_y \\ I_t \end{bmatrix}$ are the derivatives of the image intensity values along the x, y and t dimensions, and $\vec{V}$ is the optical flow vector with the components Vx, Vy. The parameter α is a regularization constant. Larger values of α lead to a smoother flow. This function can be solved by calculating the Euler-Lagrange equations corresponding to the solution of the above equation. These are given by

$$\Delta V_x - \frac{1}{\alpha} I_x(I_x V_x + I_y V_y + I_t) = 0 \tag{81}$$

$$\Delta V_y - \frac{1}{\alpha} I_y(I_x V_x + I_y V_y + I_t) = 0 \tag{82}$$

where Δ denotes the Laplace operator so that $\Delta V_x = \frac{\partial}{\partial x}\frac{\partial V_x}{\partial x}; \Delta V_y = \frac{\partial}{\partial y}\frac{\partial V_y}{\partial y}$.

Solving these equations using the Gauss-Seidel iteration [37] for the flow components V_x, V_y results in an iterative scheme

$$V_x^{k+1} = \frac{\Delta V_x^k - \frac{1}{\alpha} I_x (I_y V_y^k + I_t)}{\frac{1}{\alpha} I_x^2} \tag{83}$$

$$V_y^{k+1} = \frac{\Delta V_y^k - \frac{1}{\alpha} I_y (I_x V_x^k + I_t)}{\frac{1}{\alpha} I_y^2} \tag{84}$$

where $k+1$ denotes the next iteration, which is to be calculated and k is the last calculated result. The initial velocities V_x^0, V_y^0 are usually chosen to be 0. ΔV_x can be obtained from

$$\Delta V_x = \sum_{N(p)} V_x(N(p)) - V_x(p) \tag{85}$$

where $N(p)$ are the neighbors of the pixel p. Similarly, ΔV_y is computed as

$$\Delta V_y = \sum_{N(p)} V_y(N(p)) - V_y(p). \tag{86}$$

In contrast to the Lucas-Kanade method, the Horn-Schunck algorithm produces a high density flow.

4.2 Adaptive Control Grid Interpolation Motion Model

For the computation of the optical flow/motion field, a modified control grid interpolation (CGI) representation [17] is used. This can be viewed as an extension of the Lucas-Kanade method. The optical flow is obtained by segmenting the image into small contiguous square regions. The corners of these regions form control points, which are used as the anchors from which the intermediate motion vectors are computed using bilinear interpolation. CGI allows for the representation of complex non-translational motion and in that regard is different from the conventional block matching algorithm as used in video compression such as MPEG. We use a high resolution CGI algorithm with embedded optical flow equations for calculating the motion of the control points, leading to an accurate dense motion field representation.

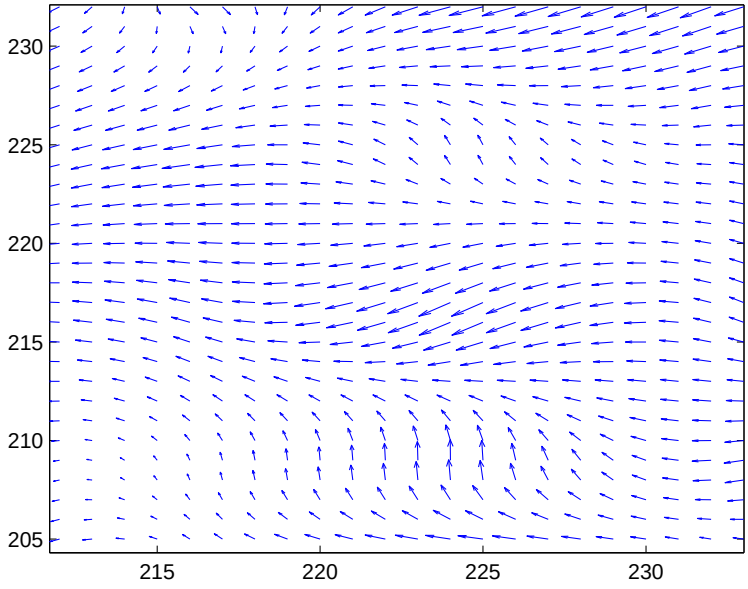


Figure 22: Example of a turbulent motion field.

Figure 22 shows an example of the motion field in a region of turbulence, where the magnitudes of the velocities have been scaled for better visualization.

Within each region, the image constraint between pixels in images I_0 and I_1 is described as:

$$\begin{aligned} I_1[i,j] &= I_0[i + d_1[i,j], j + d_2[i,j]] \\ d_1[i,j] &= \alpha_1 + \alpha_2 i + \alpha_3 j + \alpha_4 ij = \alpha^T \theta[i,j] \\ d_2[i,j] &= \beta_1 + \beta_2 i + \beta_3 j + \beta_4 ij = \beta^T \theta[i,j] \end{aligned} \tag{87}$$

where $d_1[i,j]$ is the horizontal component of the displacement vector at spatial location (i,j) and $d_2[i,j]$ is its vertical component. This is equivalent to finding the optimal motion vectors at each of the control points on the border of the region.

The bilinear parameters α, β are found in each region R by minimizing the error

$$\sum_{[i,j] \in R} (I_0[i,j] - I_1[i + \alpha^T \theta[i,j], j + \beta^T \theta[i,j]])^2. \tag{88}$$

The error function in Equation (88) is simplified by using a first-order Taylor series

expansion at (i, j)

$$\sum_{[i,j]\in R} (I_0[i,j] - I_1[i,j] - \frac{\partial I_1[i,j]}{\partial i}\alpha^T\theta[i,j] - \frac{\partial I_1[i,j]}{\partial j}\beta^T\theta[i,j])^2. \tag{89}$$

The accuracy can be further improved by changing the location of the Taylor series expansion from (i, j) to $(i + \alpha^T\theta[i,j], j + \beta^T\theta[i,j])$ and updating the parameter estimates. This process usually converges in fewer than five iterations.

For global motion such as panning, there is no need to segment the image into regions since the bilinear parameters α, β describe the motion of the entire image. On the other hand, if the motion is complicated and localized such as in turbulence (each region is moving differently), it is necessary to divide the image into regions and estimate motion parameters for each region. To divide the image into regions, a quadtree is used to decompose the image into regions. The termination criterion for subdividing each region can be either a maximum square error (Equation 88) in each block or a minimum block size.

4.3 Trajectory Estimation

Let us denote the motion between frame t and frame $t-1$ as $v_{t,t-1}(i,j)$. Once the motion between the frames has been computed, the trajectory for each pixel (i, j) can be obtained. There are two methods we propose for determining the trajectory $\mathbf{c}(i, j, k|t_0 \geq k \geq t_0 - n)$, where k is the frame number. Therefore, the trajectory $\mathbf{c}(i, j, k|t_0 \geq k \geq t_0 - n)$ can be written as the collection of the starting frame pixel $I_{t_0}(i, j)$ location in each frame k $(t_0 \geq k \geq t_0 - n)$ $\{\mathbf{c}(i, j, t_0), \mathbf{c}(i, j, t_0 - 1), \ldots, \mathbf{c}(i, j, t_0 - n)\}$.

Given the transitional matrix of motion fields $\{v_{t_0,t_0-1}(i,j), \ldots, v_{t_0-n+1,t_0-n}(i,j)\}$, the motion trajectory $\mathbf{c}(i, j, k|t_0 \geq k \geq t_0 - n)$ can be computed iteratively as follows:

$$
\begin{aligned}
\mathbf{c}(i,j,t_0) &= (i,j) \\
\mathbf{c}(i,j,t_0-1) &= \mathbf{c}(i,j,t_0) + v_{t_0,t_0-1}(\mathbf{c}(i,j,t_0)) \\
&\vdots \\
\mathbf{c}(i,j,t_0-n) &= \mathbf{c}(i,j,t_0-n+1) + v_{t_0-n+1,t_0-n}(\mathbf{c}(i,j,t_0-n+1)).
\end{aligned} \tag{90}
$$

The advantage of this approach is its computational efficiency. All of the motion fields except for the farthest one from the current frame I_{t_0}, $v_{t_0-n+1,t_0-n}(i,j)$, can be reused when the current frame moves to the next one I_{t_0+1} in the video. At each frame, only one motion field needs to be computed to construct the new motion trajectory $\mathbf{c}(i,j,t_0+1:t_0-n+1)$.

However, the above approach is subject to error propagation, i.e. any errors in $\mathbf{c}(i,j,t_0+k)$ are passed to $\mathbf{c}(i,j,t_0+k-1)$. Alternatively, we take the following approach. The transitional matrix is instead made up of motion fields $\{v_{t_0,t_0-1}(i,j), v_{t_0,t_0-2}(i,j), \ldots,$ With this choice, when the motion is computed, the source frame remains fixed, while the target frame is changed.

$\mathbf{c}(i,j,t_0:t_0-n)$ is computed by the equations

$$
\begin{aligned}
\mathbf{c}(i,j,t_0) &= (i,j) \\
\mathbf{c}(i,j,t_0-1) &= (i,j) + v_{t_0,t_0-1}(i,j) \\
&\vdots \\
\mathbf{c}(i,j,t_0-n) &= (i,j) + v_{t_0,t_0-n}(i,j).
\end{aligned} \tag{91}
$$

Though the error propagation problem is avoided, the computation load is significantly increased since the source frame shifts and each motion field needs to be computed again. To illustrate the error propagation problem, the following experiment is performed. In a turbulence video sequence, a frame i is to be registered toward frame $i+4$. Two different approaches are used to compute the motion between the

two frames: recursively as in Equation (90) where the motion between frame i and frame $i+4$ is computed from the motion fields $\{v_{i,i+1}, v_{i+1,i+2}, v_{i+2,i+3}, v_{i+3,i+4}\}$ or non-recursively as in Equation (90) where the motion between frame i and frame $i+4$ ($v_{i,i+4}$) is directly computed. After the motion is computed, the computed motion field is used to warp frame i toward frame $i+4$. The PSNR of the warped image is used to assess the accuracy of the motion estimation. Figure 23 shows the comparative result on a turbulence video clip. It can be seen that the non-recursive approach has higher PSNR since the error propagation problem is avoided. On average, the PSNR is 1 dB higher.

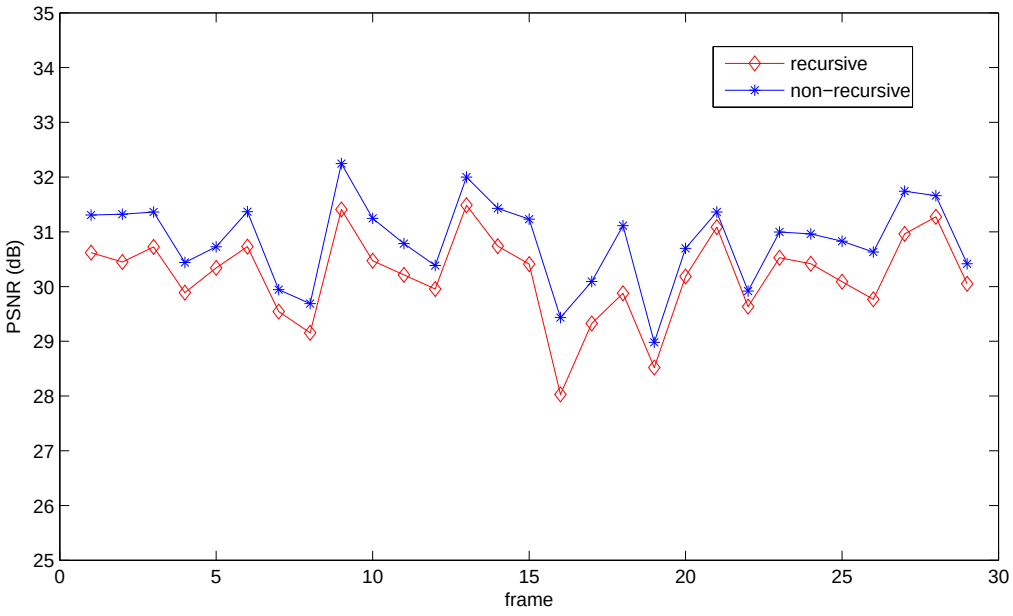


Figure 23: In a turbulence video clip, frame i is registered to frame $i+4$. Two approaches are used to estimate the motion field: Recursive (motion between consecutive frames are used) and non-recursive (directly compute the motion between the two frames). PSNR comparison of the warped frame is plotted.

4.4 Compensation of Motion Induced Distortion

Since turbulence is approximately quasi-periodic, the net displacement over the duration of a period is approximately zero. Such is not the case for real motion. Consequently, we can reduce the turbulent motion effect by using the centroid of the trajectory taken over a period $2n+1$ as the target location for a period.

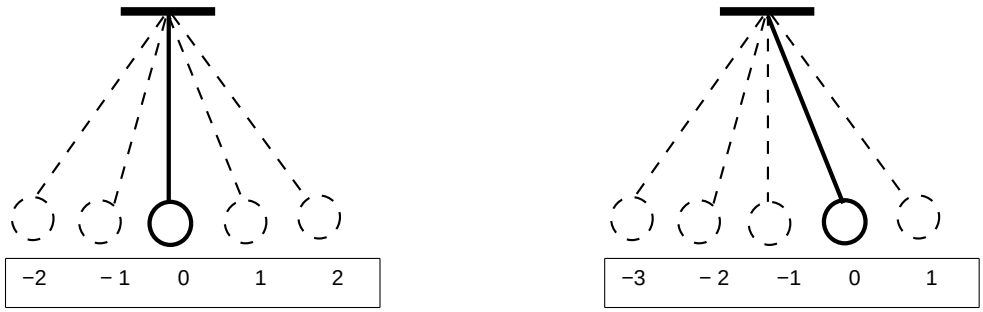
$$\hat{T}(i,j) = \frac{1}{2n+1} \sum_{t_0-n \le k \le t_0+n} \mathbf{c}(i,j,k) \tag{92}$$

$\hat{T}(i,j)$ approximates the locations of the pixels in frame t_0 without the turbulence distortion. We then warp the pixels in frame t_0 from their original locations toward their estimated locations through interpolation. The turbulent motion is suppressed by this process while other motion characteristics are preserved.

The period is initially determined experimentally and it remains fixed. Since turbulent motion is dynamic, we should be able to improve the turbulence suppression if we can find a way to adjust the length of the trajectory smoothing filter. We observed that one of the characteristics of the turbulent motion is zero mean quasi-periodicity. The pixels in the image are pushed backwards and forwards. This phenomenon is caused by the Gaussian random distribution of the phase fluctuations when optical wavefronts propagate through the atmosphere.

For simplicity, consider the example where we track the motion of a simple pendulum. If the current position is the zero displacement position as shown in Figure 24(a), the associated motion vectors might be $\{-2,-1,0,1,2\}$, which means that the pixel had been pushed backward and forward by turbulence. If we integrate the motion vectors, we get a zero. Thus there is no need to compensate for the turbulent motion at the current position since it is in the zero displacement position. For other positions, there will be an offset. The offset can be estimated from the averaged integration of the motion vectors. For example, as shown in Figure 24(b), we might have a time window with the motion vectors $\{-3,-2,-1,0,1\}$. The average of the

integration in this case results in -1, implying that the pendulum should be displaced by -1. The center point o is set as the reference (origin of the coordinate system).



(a) (b)

Figure 24: Pendulum illustration. (a) center position. (b) offset position.

$\vec{c}(t, o)$ represents the coordinate at time t. Because of the quasi-periodicy, we can expect that

$$\frac{1}{T}\int_{-T/2}^{T/2} \vec{c}(t, o) = 0 \tag{93}$$

where T is the period of the turbulent motion. Let $\vec{c}(t, p)$ represent the coordinate of the current position at time t with p as the reference point. Then

$$\vec{c}(t, p) = \vec{c}(t, o) + \vec{po} \tag{94}$$

where $\vec{po}$ is the offset. $\vec{po}$ needs to be estimated so that this offset can be compensated.

The offset is estimated by integrating over a period:

$$\frac{1}{T}\int_{-T/2}^{T/2} \vec{c}(t, p) = \frac{1}{T}\int_{-T/2}^{T/2} \vec{c}(t, o) + \vec{po}; \tag{95}$$

$$\vec{po} = \frac{1}{T}\int_{-T/2}^{T/2} \vec{c}(t, p); \tag{96}$$

$\vec{c}(t,p)$ is obtained in the trajectories through motion estimation. The experiments of the adaptive period turbulent motion compensation is reported in the next chapter, which is dedicated to implementation and experimental results.

4.5 Scintillation Effects of Atmospheric Turbulence

Scintillation or twinkling is a general terms for describing rapid variations in apparent brightness or color of a distant luminous object viewed through the atmosphere.

If the object lies outside the earth's atmosphere, as in the case of outer space objects like stars and planets, the phenomenon is called astronomical scintillation. If the object lies within the atmosphere, it is called terrestrial scintillation.

Scintillation is defined as variations in luminance only, and so it does not cause blurring of astronomical images. It is established that nearly all scintillation effects are caused by irregular refraction, which is a result of small-scale fluctuations in air density usually related to temperature gradients. Scintillation effect are illustrated in Figure 25, which shows the pixel intensity of each frame at the location (10,10) in a real turbulence degraded video clip. The pixel intensity changes dynamically in the video clip. The variance is 40.

Though scintillation effects are not modeled in the degradation model, the denoising step in the turbulent motion suppression can effectively reduce this effect. When the neighborhood frames are warped and averaged, the noise and the scintillation effects are both reduced since the intensity variations are decreased. The results are shown in the following chapter (Figure 31, Figure 32 and Figure 33).

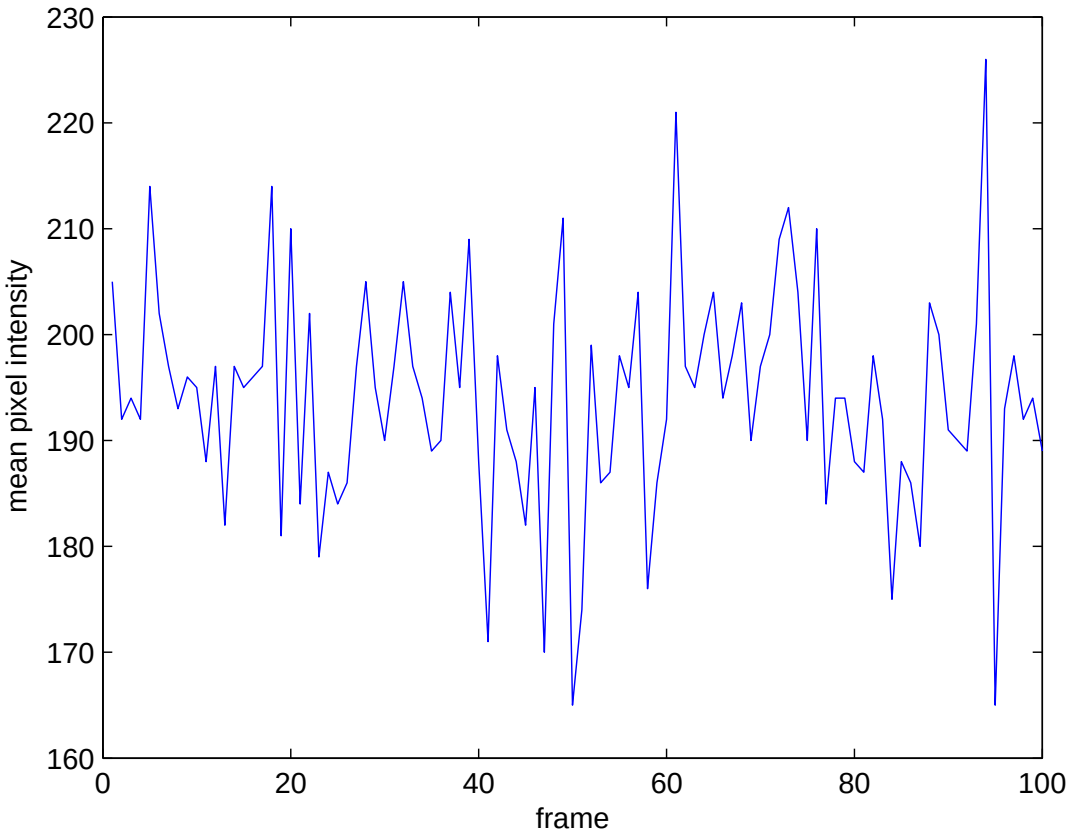


Figure 25: The pixel intensity at the location (10,10) in each frame in a real turbulence degraded video clip. The intensity variation is a result of astronomical scintillation. The variance of the intensity is 40.

CHAPTER V

IMPLEMENTATION AND EXPERIMENTAL RESULTS

This chapter is dedicated to the implementation of the algorithm and the experimental results for both simulated and real turbulence-degraded video sequences. The restoration algorithm is implemented in both Matlab and in C, the latter to illustrate that the restoration can be achieved in real-time or near real-time.

5.1 Simulation of Turbulence Degraded Video

As depicted in Equation (1), turbulence degradation is modeled in terms of a dispersive component h, a geometric distortion component D, and an additive noise component η. Although this thesis does not contain innovations for dealing with the noise component, denoising is embedded in the restoration algorithm through the restoration using a Wiener filter for the modeled blur. It is also implicit in the turbulent motion suppression. The dispersion h can be modeled by Equation (24). The geometric distortion D is modeled through spatial pixel displacements obtained typically from real turbulence video clips.

Generally, the following steps are taken to create video sequences with simulated turbulence degradation:

1. First, motion fields are computed from a real turbulence video clip and applied to the frames in a turbulence-free video as illustrated in Fig. 26(b).

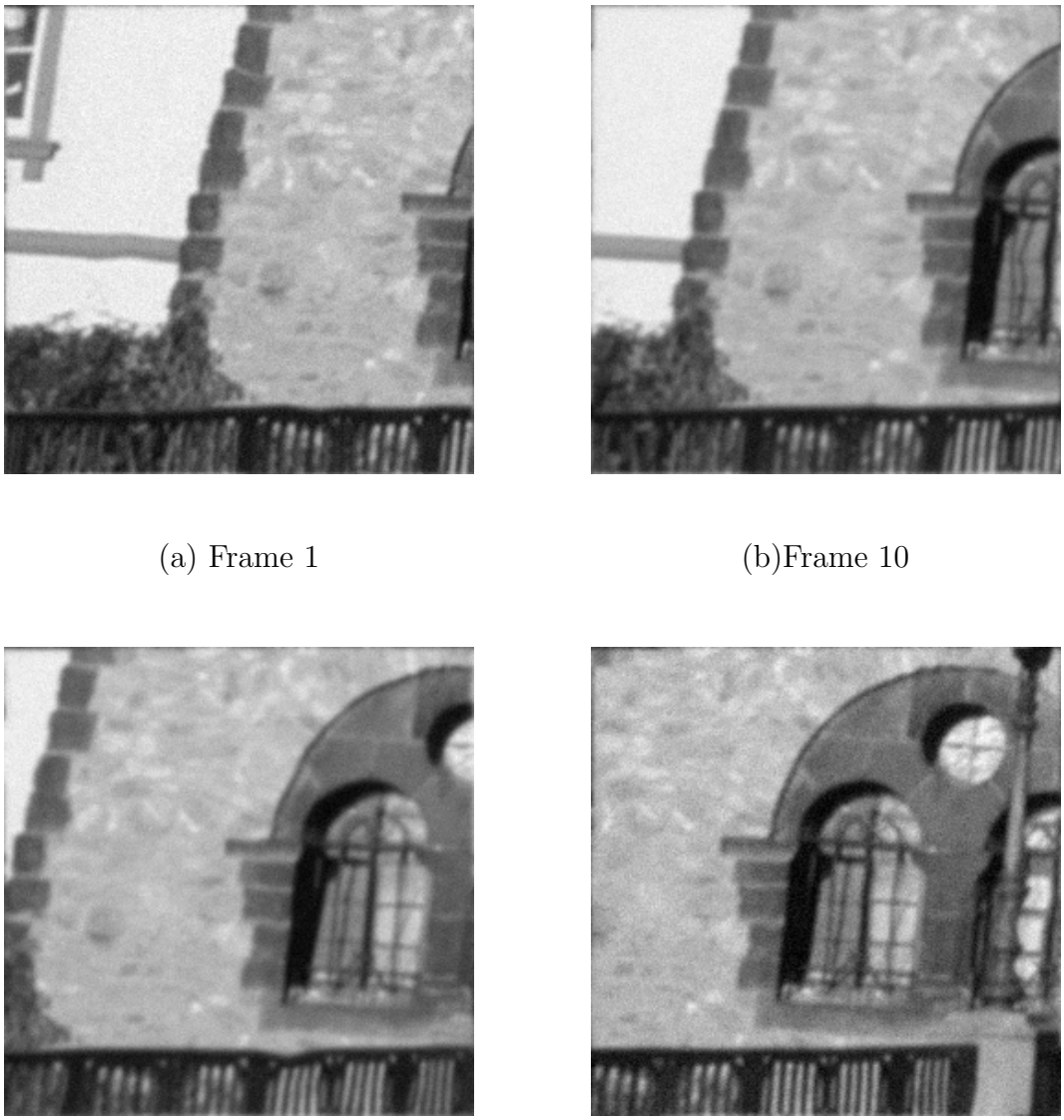
2. Then, a linear shift invariant dispersion filter is applied, the result of which is illustrated in Fig. 26(c).

3. Finally, Gaussian random noise is added to the blurred image. The noisy blurred image is shown in Fig. 26(d).

(a) (b)

(c) (d)

Figure 26: The steps to simulate turbulence degraded video. (a) An original frame from the CAR sequence. (b) The frame after the turbulent motion field was applied. (c) The distorted frame blurred by atmospheric turbulence OTF ($\lambda = 0.001$). (d) The image from (c) after random Gaussian noise ($\sigma^2 = 0.003$) was added to the distorted and blurred frame.



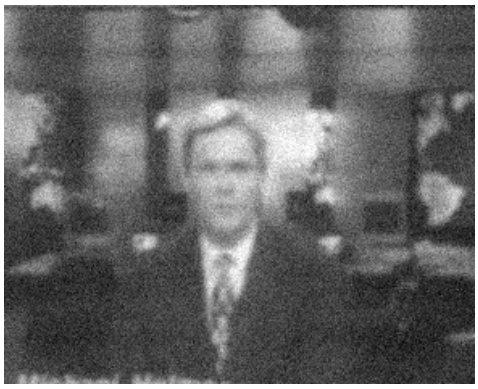
(a) Frame 1

(b)Frame 10

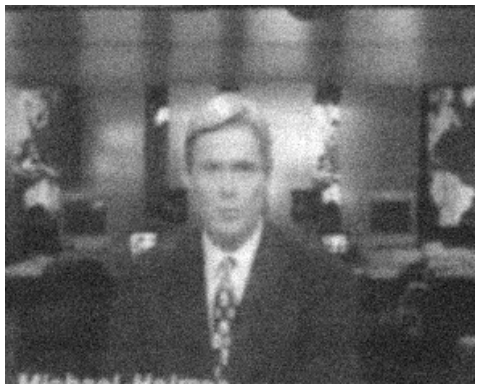
(c)Frame 20

(d)Frame 30

Figure 27: Video frames from the PANNING sequence.



(a) Frame 82



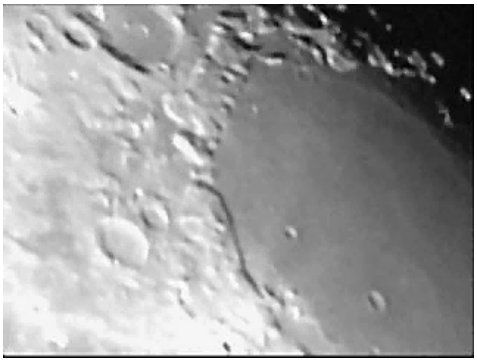
(b) Frame 92

(c) Frame 100

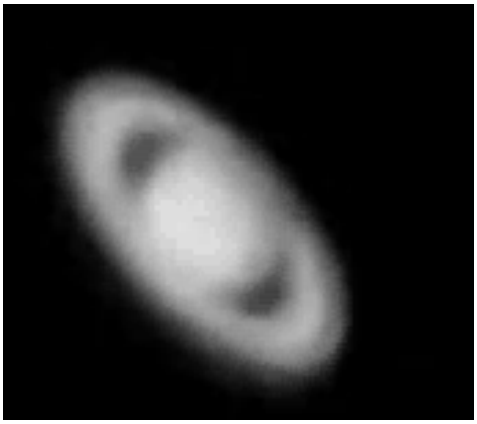
(d) Frame 110

Figure 28: Video frames from the ZOOMING sequence.

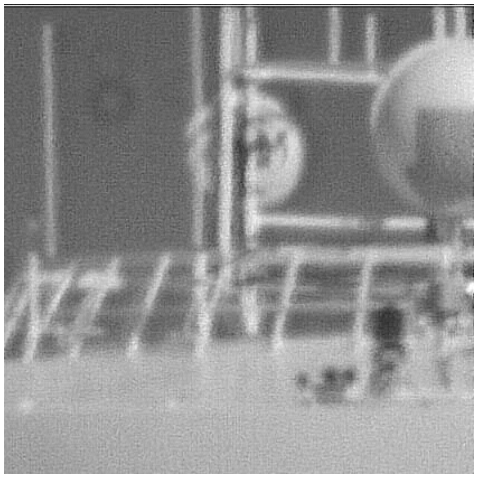
(a)



(b)

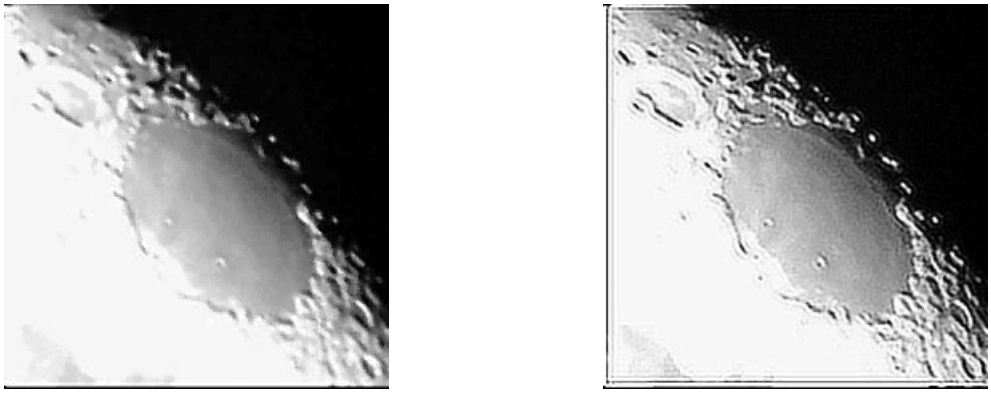


(c)



(d)

Figure 29: (a) A frame from video clip 2. (b) A frame from video clip 3. (c) A frame from video clip 4. (d) A frame from video clip 5. All are real turbulence degraded video clips.

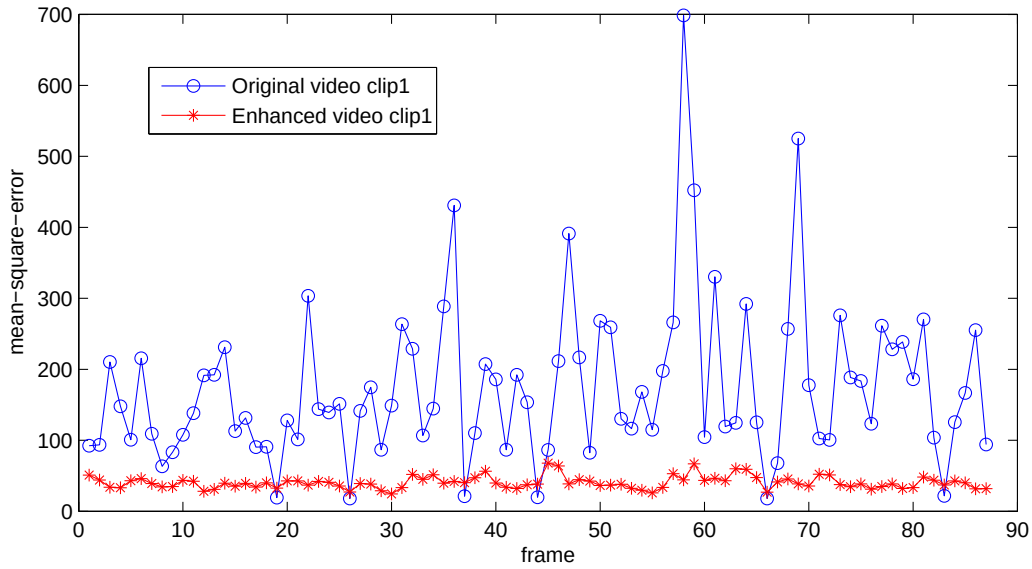


(a) (b)

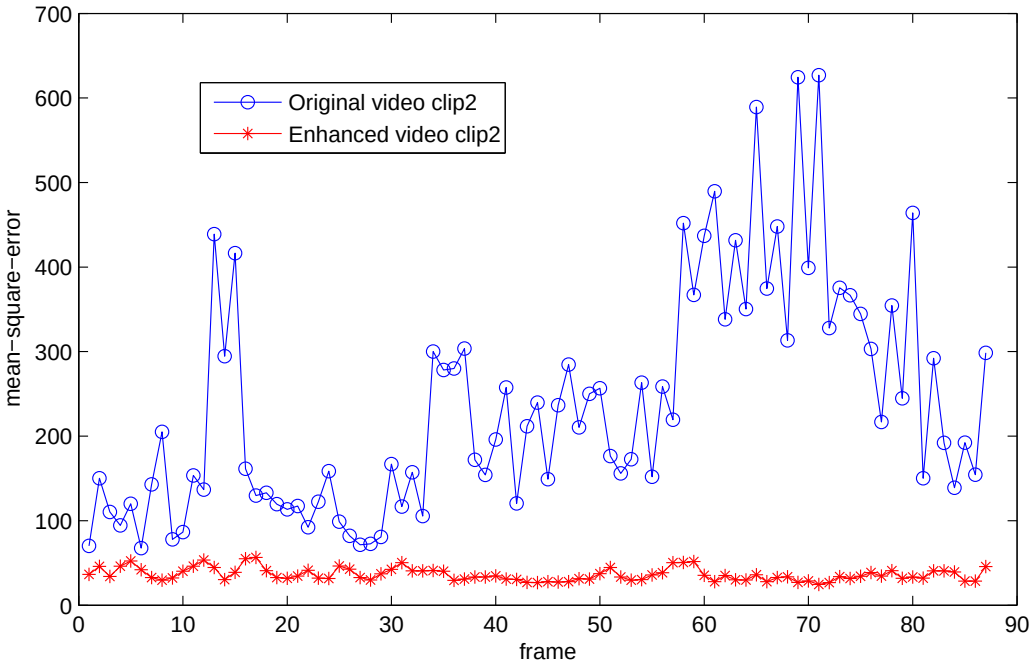
Figure 30: (a) A real atmospheric turbulence blurred frame in the video clip1. (b) The enhanced image of the moon using the new algorithm.

The λ parameters associated with the dispersion filter are extracted from real turbulence-degraded video and are then applied in step 2 of the simulation process. Noise variances are chosen from the set $\{0.0010, 0.0015, 0.0020, 0.0025, 0.0030\}$ to simulate the levels of time varying Gaussian noise encountered in real data sets.

A number of video clips were simulated by the above procedure so that we would have original and distorted video sequences for performance analysis purposes. We select those representative video clips in the experiments. The CAR sequence is a such a sequence in which the camera is fixed on a scene of a parking lot while a car is leaving the lot. The PANNING sequence is a clip in which the camera is panning a scene of a building. The ZOOMING clip shows an anchorman in a television newsroom where the camera is zooming in. Some frames from the PANNING and ZOOMING sequences are shown in Fig. 27 and Fig. 28 respectively.

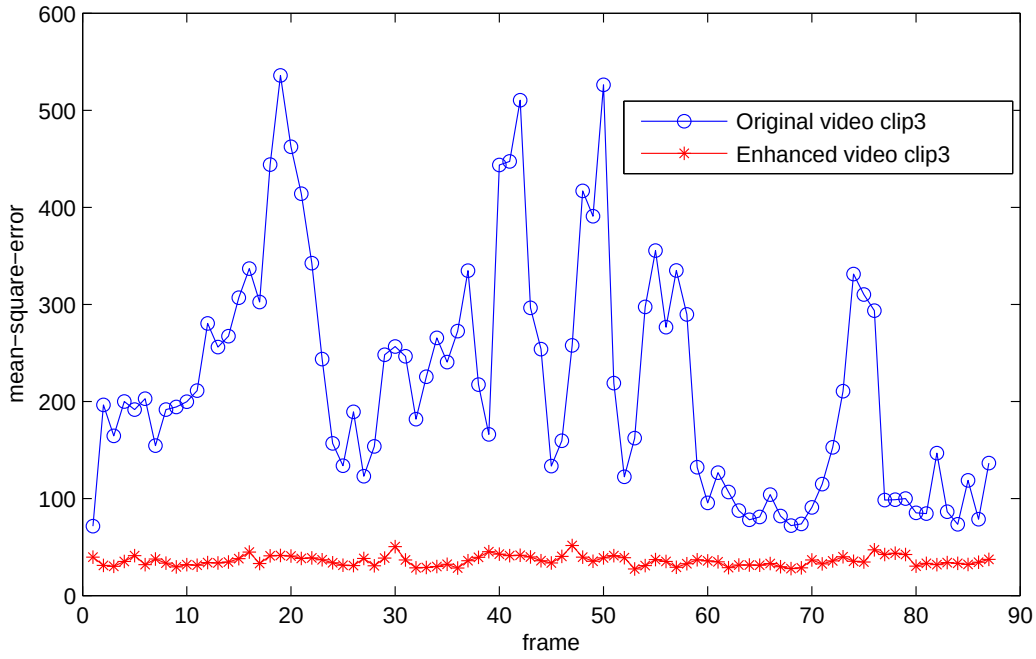


(a)

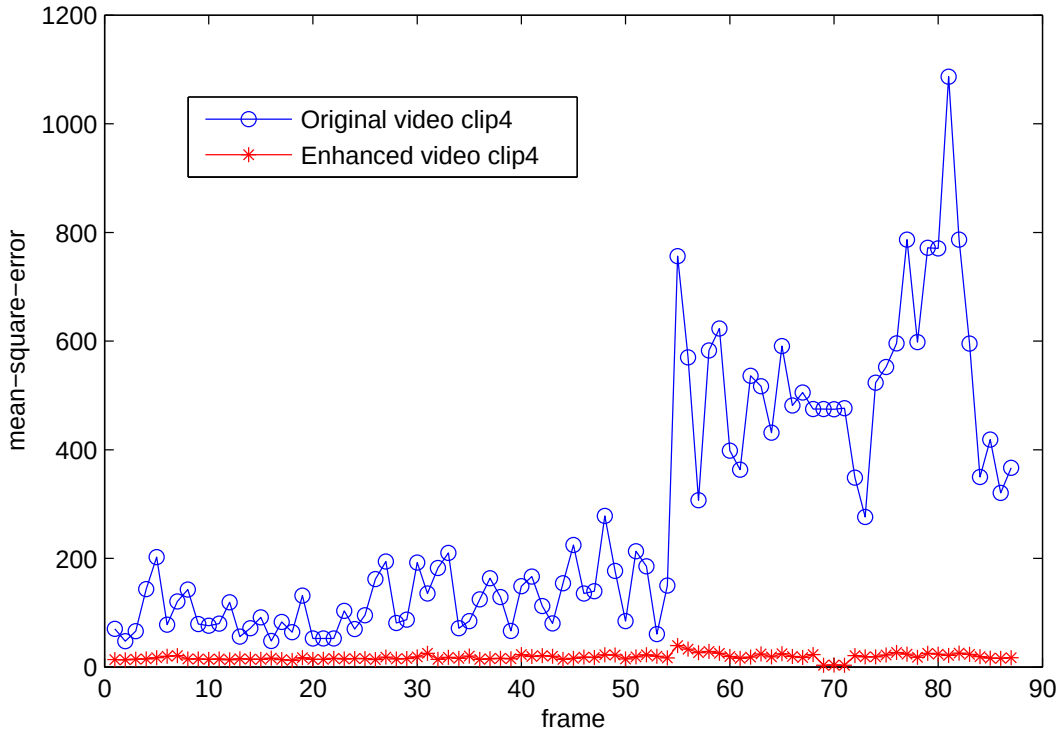


(b)

Figure 31: Mean-square-error between consecutive frames in the original video clip and enhanced video. (a) video clip 1. (b) video clip 2.



(a)



(b)

Figure 32: Mean-square-error between consecutive frames in the original video clip and enhanced video. (a) video clip 3. (b) video clip 4.

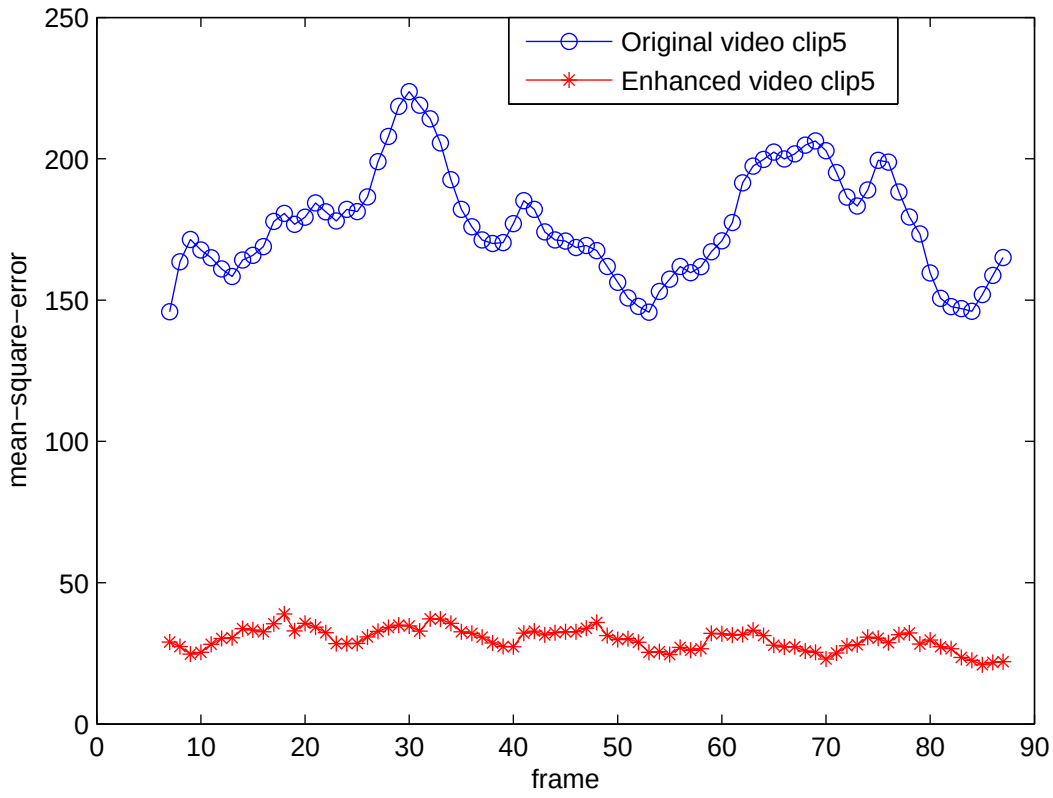


Figure 33: Mean-square-error between consecutive frames in the original video clip 5 and the enhanced video.

5.2 Results of Fixed Period Enhancement Method

5.2.1 Performance Evaluation on Real Turbulence-degraded Video

A number of real atmospheric turbulence degraded videos were also tested. Figure 30 shows an example of a frame taken from a video sequence (video clip1) of the moon acquired through a telescope. There is a fair amount of time-varying geometric distortion and spatial dispersion in the sequence. The reduction in the distortion is obvious in the processed video. The jitter and geometric distortions are no longer visible, and the image is clearly sharper. Ideally, each frame should be the same and there should be no difference between them. Because of noise and compensation errors, minute changes can be seen from frame to frame. But overall the results look very good. The mean-square-error between successive frames is shown in Figure 31(a) for both the original turbulent sequence and the enhanced sequence. While frame differences do exist in both sequences, the variations are dramatically reduced for the processed sequence. The other real turbulence degraded video clips are also tested. Example frames from those video clips are shown in Figure 29. The mean-square-error between successive frames of the video clips are shown in Figures 31, 32 and 33.

5.2.2 Performance Evaluation on Simulated Turbulence-degraded Video

To better assess the performance of our enhancement algorithms, it is useful to have the original video sequences (i.e. the ground truth). This is not available for the real turbulence degraded videos. Therefore, we rely on simulated videos for these comparisons using simulated degraded videos with ground truth. PSNRs can be computed to measure the overall restoration performance. For comparison, the time-averaging reference frame approach used in [17, 18] is also implemented. In the time-averaging reference frame approach, the video is filtered by a moving averaging filter of length 5 to create the reference sequence. Then the frames are registered to

the reference frames. Since there is a moving object in the video, the time-averaged reference frames led to poor results, as shown in Figure 34. The proposed method performs better than the time averaged reference approaches both subjectively and in terms of PSNR. A visual comparative example of the geometric distortion reduction

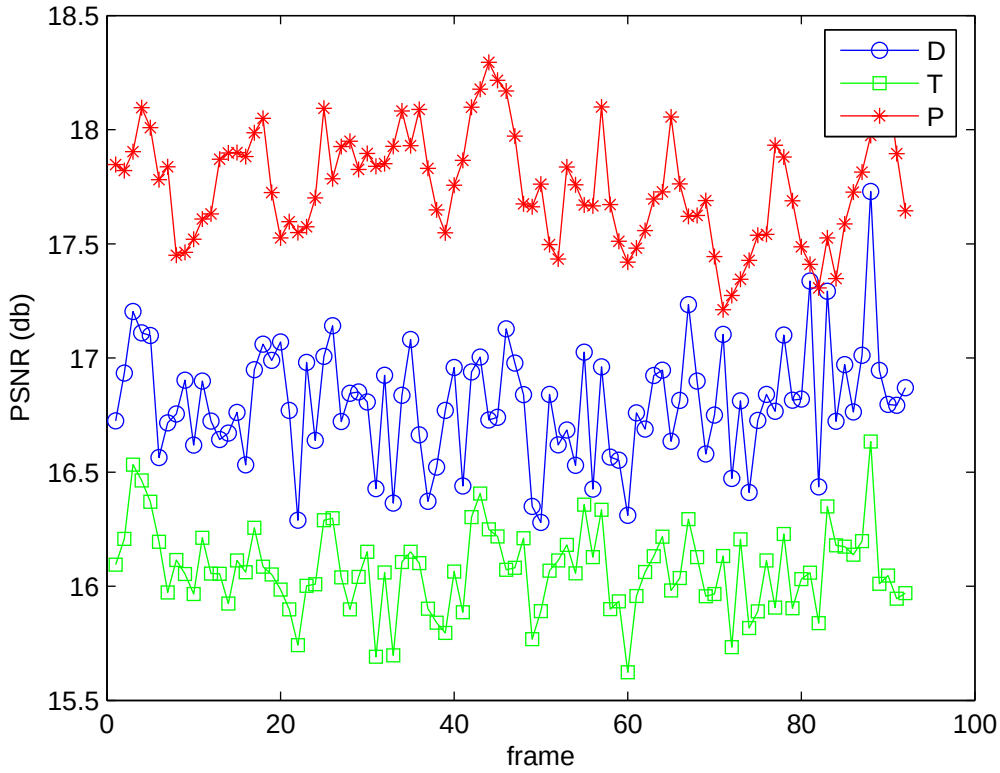


Figure 34: PSNR comparison of the restoration algorithm (fixed period) with time-averaging reference [18, 17] on the CAR video sequence. D: the degraded video; T: the enhancement method with time-averaging [18, 17]; P: the proposed algorithm with fixed period in this thesis.

is highlighted in Figure 35. The geometric distortion was not properly handled by the time-averaging reference frame approach. Moreover, the frame is blurred since there is real motions in the video.

(a) (b)

(c) (d)

Figure 35: (a) An original frame in the CAR sequence. (b) The degraded frame. (c) The restored frame by the time-averaging reference approach [18, 17]. (d) The restored frame by the new method.

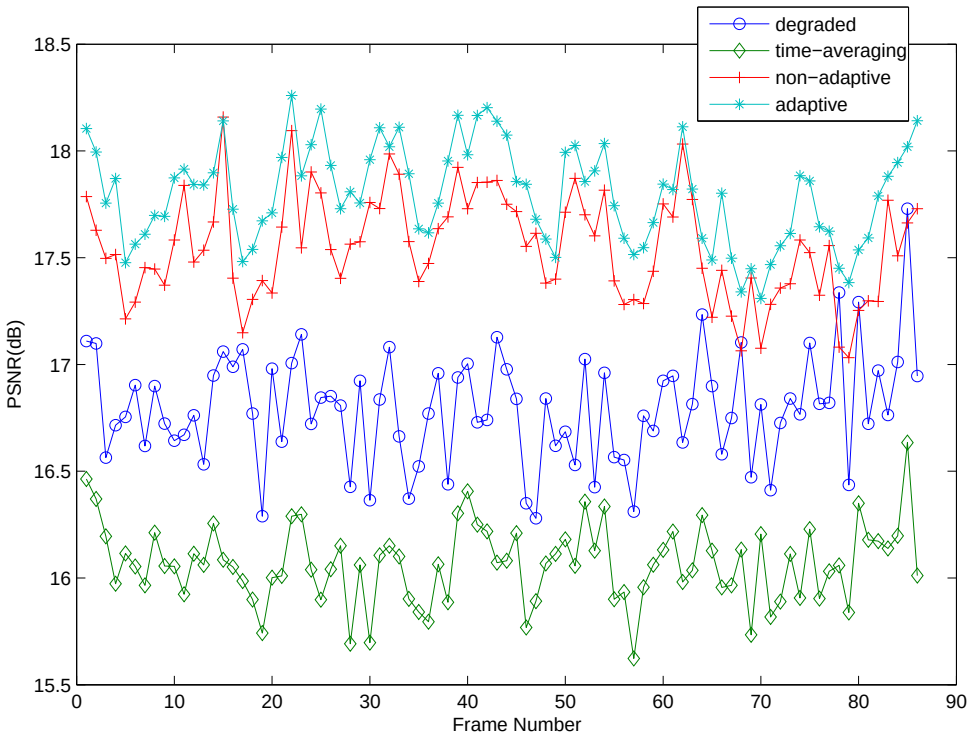


Figure 36: PSNR comparison of the time-averaged reference frame algorithm (time-averaging)[17, 18], the centroid trajectory algorithm with fixed window (non-adaptive, section 4.4), the centroid trajectory algorithm with adaptive window (adaptive, section 4.4), and the optically degraded image sequence (degraded, unprocessed) on the CAR sequence.

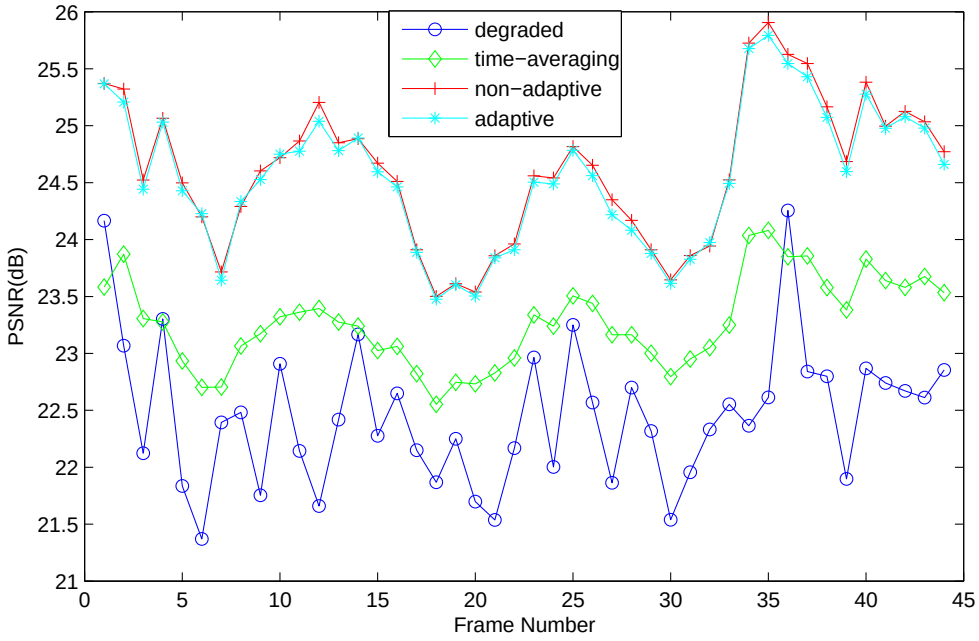


Figure 37: PSNR comparison of the time-averaged reference frame algorithm (time-averaging)[17, 18], the centroid trajectory algorithm with fixed window (non-adaptive, section 4.4), the centroid trajectory algorithm with adaptive window (adaptive, section 4.4), and the optically degraded image sequence (degraded, unprocessed) on the ZOOMING sequence.

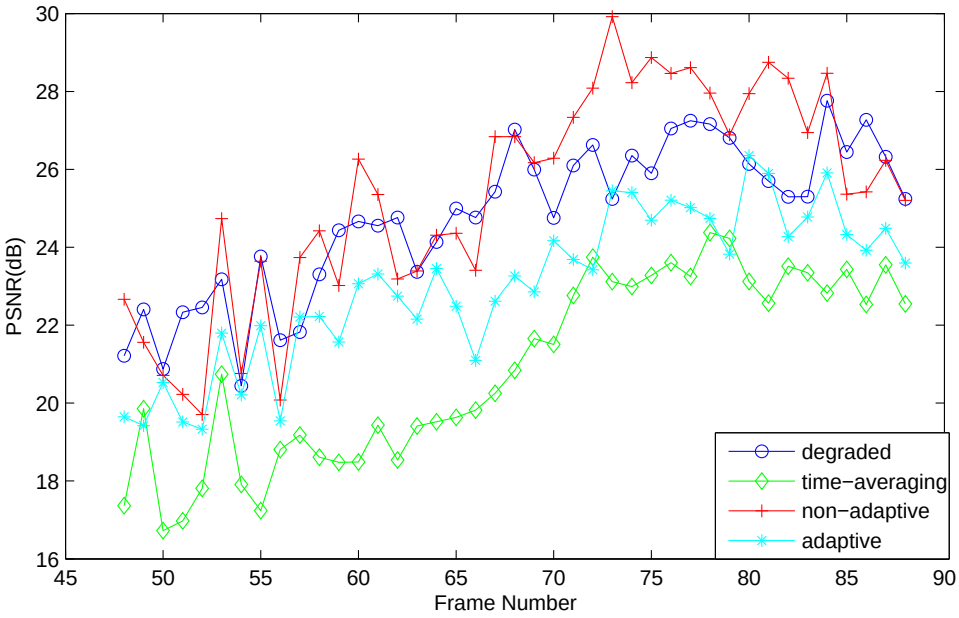


Figure 38: PSNR comparison of the time-averaged reference frame algorithm (time-averaging)[17, 18], the centroid trajectory algorithm with fixed window (non-adaptive, section 4.4), the centroid trajectory algorithm with adaptive window (adaptive, section 4.4), and the optically degraded image sequence (degraded, unprocessed) on the PANNING sequence.

5.3 Result of Adaptive Period Enhancement Method

As discussed in the previous chapter, a better restoration result might be achieved by exploring the pseudo-periodicity of each pixel in the video frame caused by the geometric distortion. The adaptive periodicity discussed in section 4.4 was implemented and the PSNRs were computed on a frame-by-frame basis to measure the overall performance for test sequences. In Figure 36, four results on the CAR sequence are compared. The first is the output of the adaptive centroid trajectory algorithm. Below it is the output of the non-adaptive centroid trajectory algorithm with a fixed time window of length 11. The PSNR of the degraded sequence is also plotted in Figure 36. At the bottom is the output from the algorithm based on time-averaging[17]. The restored sequence using this algorithm was 0.8 dB worse on average than the degraded sequence. This is because true motion exists in the degraded sequence. Therefore, the reference is significantly corrupted by time-averaging when the reference frame is made. As seen in Figure 36, the proposed adaptive method obtains improvement over the previous fixed period method in terms of PSNR.

The similar comparison of algorithms on the ZOOMING clip is shown in Figure 37. In this example, the difference between the adaptive window method and the fixed window method is not significant compared with the previous example. The reason is that the motion in this example is global. When the camera is moving toward the reporter, all the pixels in the frame are moving outward. In this case, period estimation is unreliable. In the CAR video, only the car is moving while the other parts of the scene are stationary. Thus, we can obtain good estimates of the periodicities. Figure 38 shows the result on the PANNING sequence. The motion is global and the estimates of the period are not sufficiently accurate to yield an advantage. Thus, there is no gain in using the adaptive period in this case.

5.4 Commutativity of Deblurring and Motion Suppression

In the earlier chapters, deblurring was first performed followed by turbulent motion suppression. Alternatively, motion suppression can be performed first, followed by deblurring.

For the following reasons, it is preferable for motion suppression be performed first.

- Optical flow computation involves the estimation of the gradient of the image at (x, y, t). A frame in a turbulence degraded video is generally blurred (smoothed). Since gradients can be better estimated in smoothed frames, the optical flow computation is more robust and accurate.

- Atmospheric turbulence blur is spatially varying. When motion suppression is performed, the motion between frames is computed and is used to warp the individual frames toward the central frame. The warped frames are then averaged to reduce the noise. In theory this should improve the SNR. Such an averaging process not only reduces the noise, but also makes the turbulence blurring approximately spatially invariant. This should help with the deblurring in the next step, since the blurring model assumes that the blur is spatially invariant. In this way, the spatially varying blur is converted into a spatially invariant one.

- Since motion suppression takes less time than deblurring, it should have a higher priority if there is a time constraint for the restoration algorithm. In the C implementation, the motion suppression is performed first and then deblurring takes place.

5.5 Real time implementation in C

To design a real time implementation, we need to analyze how to reduce the computational cost. Motion estimation is well known for its high computation demand. As an effort to make motion computation efficient, we might consider performing the computation of optical flow in the subband domain.

OpenCV is an Intel Open Source Computer Vision Library. It is a collection of C functions, with a few in C++, that implements popular Image Processing and Computer Vision algorithms. Several classical optical flow algorithms such as the Lucas & Kanade algorithm and the Horn & Schunck algorithm are implemented in this package. It also provides an image warping function and Video I/O functions.

The optical flow computation by the control grid interpolation (CGI) algorithm has higher accuracy than the Lucas & Kanade algorithm and the Horn & Schunck algorithm. Figure 39 shows comparative results on frames taken from a real turbulence degraded video clip. The accuracy is measured by PSNR. Given a pair of consecutive frames in the video, the PSNR between them is computed first. Then the three optical flow algorithms are used to compute the motion between the two frames and the computed motion is used to register the source frame toward the target frame. The PSNR between the registered source and the target frame is computed to reflect the accuracy of the optical flow computation. Among the three algorithms, the CGI method provides the highest accuracy in this test. On average, the PSNR before image registration is 26.09 dB, the PSNR of the image registered by CGI is 30.89 dB, that of the Lucas & Kanade algorithm is 29.11 dB, and that of the Horn & Schunck algorithm is 29.38 dB.

Qt is a GUI (graphical user interface) software toolkit. Qt simplifies the task of writing and maintaining a GUI. It is used in building a simple windows application for the restoration of atmospheric turbulence degraded video.

Our goal was to implement a real time application. Therefore, fixed period is

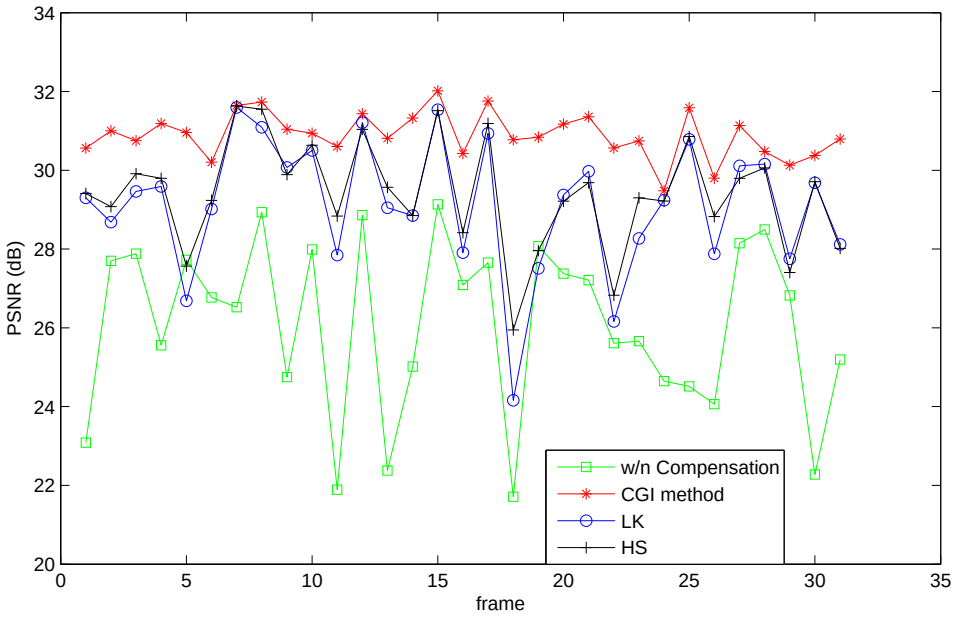


Figure 39: Comparison of the three optical flow computation algorithms on the moon video clip. The accuracy of the motion computation is evaluated by the PSNR of the registered image and the target image. The higher the PSNR, the better the optical flow estimate. On average, the PSNR before image registration is 26.09 dB, the PSNR of the image registered by CGI is 30.89 dB. That of the Lucas & Kanade algorithm is 29.11 dB, and that of the Horn & Schunck algorithm is 29.38 dB.

considered favorably since it has low computation complexity. A shorter period certainly requires less time to process a frame. The application runs in Windows XP on a notebook computer. The CPU is a 1400MHz Pentium M processor, with 768MB of RAM. It is observed that it can run at a frame rate of 30 frames per second when the period is 3 and there is no deblurring of the stabilized frame in which the turbulent motion and noise is effectively reduced. When deblurring is turned on, it will slow down the speed to around 10 frames per second. The size of the image is 256 by 256. The computational complexity is approximately proportional to the length of the period.

CHAPTER VI

CONCLUSION

6.1 Summary of Results

In this thesis work, the background of atmospheric turbulence degradation in imaging was reviewed and two aspects are highlighted: blurring and geometric distortion. Since the atmospheric turbulence blurring is unknown function, a blind image deconvolution technique was developed based on a higher order statistic (HOS). It was observed that the kurtosis generally increases as the image becomes blurred (smoothed). Such an observation was explained in the frequency domain in terms of phase correlation. It was shown that kurtosis minimization is effective in identifying the blurring parameter. Kurtosis minimization is a general method for blur identification; it is not limited to turbulence blur. To compensate for the geometric distortion, earlier work on the turbulent motion compensation was extended to deal with situations in which there is object motion. When there is no real motion, it was shown that the time-averaged frames serve as good reference frames and that the turbulent motion is reduced by registering frames towards their corresponding reference frames. When object motion is present, the approach degrades the video further since the reference frame is corrupted. Instead of using a reference frame, we smooth along the motion trajectories and real motion is preserved while the turbulence motion is effectively reduced. Though the scintillation effect is not considered separately in the model, it can be handled the same way as noise removal while the turbulence motion is suppressed.

The algorithm was initially designed and implemented in Matlab for fast prototyping. Later, a C/C++ implementation was developed to illustrate the real time

performance of the restoration algorithm.

6.2 Further Directions

The restoration of video is different from the problem of single image restoration since multiple observations are available. Though motion might be present among the frames, image registration [8] can be used to compensate for the motion. After motion compensation, the frames provide multiple observations. Thus multichannel image restoration [80, 79, 64, 58, 63, 24, 27, 67], superresolution [14] or image fusion [66] algorithms are applicable. In fact, a simple form of superresolution is used while the turbulent motion is suppressed. The neighborhood frames are warped and averaged. This is one of the simplest form of superresolution; nevertheless, it is effective and efficient. Given limited computation resources and real time demand, simpler processing is always preferable because of its low complexity. When accuracy is has a higher priority, more sophisticated methods are preferred. Further work shall investigate multichannel image restoration approaches.

Though the turbulence blurring OTF is used, it might not accurately reflect the real atmospheric turbulence blurring effect. Therefore, a "more" blind deconvolution methods that do not assume a functional form of the blur should also be explored.

REFERENCES

[1] *University of Southern California - Signal and Image Processing Institute - The USC-SIPI Image Database.* http://sipi.usc.edu/services/database/Database.html.

[2] Aime, C., Borgnino, J., Martin, F., Petrov, R., and Ricort, G., "Contribution to the Space-time Study of Stellar Speckle Patterns," *Journal of the Optical Society of America A*, vol. 3, pp. 1001–1009, July 1986.

[3] Ayers, G. R. and Dainty, J. C., "Iterative blind deconvolution method and its applications," *Optics Letters*, vol. 13, pp. 547–549, July 1988.

[4] Bergen, J. R., Anandan, P., Hanna, K. J., and Hingorani, R., "Hierarchical model-based motion estimation," in *ECCV '92: Proc. of the Second European Conference on Computer Vision*, (London, UK), pp. 237–252, Springer-Verlag, 1992.

[5] Chang, C.-C. and Lin, C.-J., *LIBSVM: a Library for Support Vector Machines*, 2001. Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`.

[6] Colavita, M. M., Shao, M., and Staelin, D. H., "Atmospheric phase measurements with the Mark III stellar interferometer," *Applied Optics*, vol. 26, pp. 4106–4112, Oct. 1987.

[7] Colavita, M. M., Swain, M. R., Akeson, R. L., Koresko, C. D., and Hill, R. J., "Effects of atmospheric water vapor on infrared interferometry," *Publications of the Astronomical Society of the Pacific*, vol. 116, p. 876C885, 2004.

[8] Cole-Rhodes, A. A., Johnson, K. L., Moigne, J. L., and Zavorin, I., "Multiresolution registration of remote sensing imagery by optimization of mutual information using a stochastic gradient.," *IEEE Transactions on Image Processing*, vol. 12, no. 12, pp. 1495–1511, 2003.

[9] Condell, J., Scotney, B., and Morrow, P., "Adaptive grid refinement procedures for efficient optical flow computation," *International Journal Computer Vision*, vol. 61, no. 1, pp. 31–54, 2005.

[10] Cortes, C. and Vapnik, V., "Support-vector network," *Machine Learning*, vol. 20, pp. 273–297, 1995.

[11] Cote, O. R., Hacker, J. M., Crawford, T. L., and Dobosy, R. J., "Clear air turbulence and refractive turbulence in the upper troposphere and lower stratosphere," in *Proc. of the American Meteorological Society's Ninth Conf. on Aviation, Range, and Aerospace Meteorol*, (Orlando, USA), 2000.

[12] Davey, B. L. K., Lane, R. G., and Bates, R. H. T., "Blind deconvolution of noisy complex-valued image," *Optics Communications*, vol. 69, pp. 353–356, Jan. 1989.

[13] Ellerbroke, B. L. and Rhoadarmer, T. A., "Adaptive wavefront control algorithms for closed loop adaptive optics," *Mathematical and Computer Modelling*, vol. 33, pp. 145–158, 2001.

[14] Farsiu, S., Robinson, M. D., Elad, M., and Milanfar, P., "Fast and robust multiframe super resolution.," *IEEE Transactions on Image Processing*, vol. 13, no. 10, pp. 1327–1344, 2004.

[15] Fontanella, J. C. and Seve, A., "Reconstruction of turbulence-degraded images using the Knox-Thompson algorithm," *Journal of the Optical Society of America A*, vol. 4, pp. 438–448, Mar. 1987.

[16] Ford, S. D., Welsh, B. M., and Roggemann, M. C., "Constrained least-squares estimation in deconvolution from wave-front sensing," *Optics Communications*, vol. 151, pp. 93–100, May 1998.

[17] Frakes, D. H., Monaco, J. W., and Smith, M. J. T., "Suppression of atmospheric turbulence in video using an adaptive control grid interpolation approach," in *Proc. of the IEEE Int. Conf. Acoustics, Speech, and Signal Processing*, (Salt Lake City, USA), pp. 1881–1884, 2001.

[18] Fraser, D., Thorpe, G., and Lambert, A., "Atmospheric turbulence visualization with wide-area motion-blur restoration," *Journal of the Optical Society of America A*, vol. 16, no. 7, pp. 1751–1758, 1999.

[19] Freeman, J. D., "Estimation of astronomical images from the bispectrum of atmospherically distorted infrared data," *Ph.D. Thesis*, 1989.

[20] Fried, D. L., "Statistics of a Geometric Representation of Wavefront Distortion," *Journal of the Optical Society of America A*, vol. 55, pp. 1427–1435, 1965.

[21] Frieden, B. R., "Turbulent image reconstruction using object power spectrum information," *Optics Communications*, vol. 109, p. 227C230, 1993.

[22] Frieden, B. R. and Oh, C., "Turbulent image reconstruction from a superposition model," *Optics Communications*, vol. 98, p. 241C244, 1993.

[23] Gennery, D. B., "Determination of Optical Transfer Function by Inspection of Frequency-domain Plot," *Journal of the Optical Society of America A*, vol. 63, pp. 1571–1577, 1973.

[24] Giannakis, G. B. and Jr., R. W. H., "Blind identification of multichannel fir blurs and perfect image restoration.," *IEEE Transactions on Image Processing*, vol. 9, no. 11, pp. 1877–1896, 2000.

[25] Gluckman, G. M., "Kurtosis and the Phase Structure of Images," in *3rd International Workshop on Statistical and Computational Theories of Vision, Nice, France, October 2003 (in conjunction with ICCV '03)*, (Nice, France), 12–15, 2003.

[26] Hamming, R. W., *Numerical Methods for Scientists and Engineers*. New York: McGraw-Hill, 1962.

[27] Harikumar, G. and Bresler, Y., "Exact image deconvolution from multiple fir blurs.," *IEEE Transactions on Image Processing*, vol. 8, no. 6, pp. 846–862, 1999.

[28] Hildebrand, F. B., *Introduction to Numerical Analysis*. New York: McGraw-Hill, 1974.

[29] Holmes, T. J., "Blind deconvolution quantum-limited incoherent imagery: Maximum-likelihood approach," *J. Optical Society of America*, vol. 9, pp. 1052 – 1061, 1992.

[30] Horch, E., "Speckle imaging in astronomy," *International J. of Imaging Systems and Technology*, vol. 6, pp. 401–417, 1995.

[31] Horn, B. and Schunk, B. G., "Determining optical flow," *Artificial Intelligence*, vol. 17, pp. 185–203, 1981.

[32] Huang, J. and Mumford, D., "Statistics of natural images and models," in *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 541–547, 1999.

[33] Hufnagel, R. E. and Stanley, N. R., "Modulation Transfer Function Associated with Image Transmission through Turbulence Media," *Journal of the Optical Society of America A*, vol. 54, pp. 52–61, 1964.

[34] Hyvärinen, A. and Oja, E., "Independent component analysis: Algorithms and applications," *Neural Networks*, vol. 13, pp. 411–430, 2000.

[35] Ichikawa, T., "A pyramidal representation of images," in *Proc. of International Conference on Pattern Recognition*, pp. 603–606, 1978.

[36] Ichikawa, T., "A pyramidal representation of images and its feature extraction facility," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 3, pp. 257–264, May 1981.

[37] Jeffreys, H. and Jeffreys, B. S., *Methods of Mathematical Physics, 3rd ed.* Cambridge, England: Cambridge University Press, 1988.

[38] Katsaggelos, A. K. and Lay, K. T., "Maximum likelihood blur identification and image restoration using the em algorithm," *IEEE Transactions on Signal Processing*, vol. 39, pp. 729–733, March 1991.

[39] Knox, K. T. and Thompson, B. J., "Recovery of images from atmospherically degraded short-exposure photographs," *Astronomical Journal*, vol. 193, pp. L45–L48, Oct. 1974.

[40] Kundur, D. and Hatzinakos, D., "Blind image deconvolution," *IEEE Signal Processing Magazine*, vol. 13, pp. 43–64, May 1996.

[41] Kundur, D. and Hatzinakos, D., "Blind image deconvolution revisited," *IEEE Signal Processing Magazine*, vol. 13, pp. 61–63, Nov 1996.

[42] Labeyrie, A., "Attainment of diffraction limited resolution in large telescopes by fourier analysing speckle patterns in star images," *Astronomy and Astrophysics*, vol. 6, pp. 85–87, May 1970.

[43] Lagendijk, R. L., Biemond, J., and Boekee, D. E., "Identification and restoration of noisy blurred images using the expectation-maximization algorithm," *IEEE Trans. Acoustics, Speech, Signal Processing*, vol. 38, pp. 1180–1191, July 1990.

[44] Lai, S.-H. and Vemuri, B. C., "Reliable and efficient computation of optical flow," *International Journal Computer Vision*, vol. 29, no. 2, pp. 87–105, 1998.

[45] Lam, E. Y. and Goodman, J. W., "Iterative statistical approach to blind image deconvolution," *Journal of the Optical Society of America A*, vol. 17, p. 1177C1184, 2000.

[46] Lane, R. G., "Blind deconvolution of speckle image," *Journal of the Optical Society of America A*, vol. 9, pp. 1508–1514, 1992.

[47] Law, N. F. and Lane, R. G., "Blind deconvolution using least squares minimization," *Optical Communication*, vol. 128, pp. 341–352, 1996.

[48] Li, D., Mersereau, R. M., Frakes, D. H., and Smith, M. J. T., "New method for suppressing optical turbulence in video," in *Proc. European Signal Processing Conference (EUSIPCO'2005)*, (Antalya, Turkey), 2005.

[49] Li, D., Mersereau, R. M., and Simske, S., "Blur identification based on kurtosis minimization," in *Proc. of the IEEE Int. Conf. Image Processing*, vol. 1, (Genoa,Italy), pp. 905–908, 2005.

[50] Li, D., Simske, S., and Mersereau, R. M., "Blind image deconvolution using constrained variance maximization," in *the Proc. of the 38th Asilomar Conf. on Signals, Systems and Computers, Pacific Grove, CA, USA*, 2004.

[51] Li, D., Simske, S., and Mersereau, R. M., "Blind image deconvolution using support vector regression," in *in Proc. of the IEEE Int. Conf. Acoustics, Speech, and Signal Processing, Philadelphia, PA, USA*, 2005.

[52] Likas, A. C. and Galatsanos, N. P., "A variational approach for bayesian blind image deconvolution," *IEEE Transactions on Signal Processing*, vol. 52, pp. 2222–2233, Aug. 2004.

[53] Lohmann, A. W., Weigelt, G., and Wirnitzer, B., "Speckle masking in astronomy - Triple correlation theory and applications," *Applied Optics*, vol. 22, pp. 4028–4037, Dec. 1983.

[54] Lucas, B. D. and Kanade, T., "An iterative image registration technique with an application to stereo vision (darpa)," in *Proc. of the 1981 DARPA Image Understanding Workshop*, pp. 121–130, April 1981.

[55] Mallat, S. G., "A theory for multiresolution signal decomposition: The wavelet representation," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 11, pp. 674–693, July 1989.

[56] McCallum, B. C., "Blind deconvolution by simulated annealing," *Optics Communications*, vol. 75, pp. 101–105, Feb 1990.

[57] Mendel, J. M., "Tutorial on higher-order statistics (spectra) in signal processing and system theory: theoretical results and some applications.," *Proc. IEEE*, vol. 79, no. 3, pp. 278–305, 1991.

[58] Molina, R., Mateos, J., Katsaggelos, A. K., and Vega, M., "Bayesian multichannel image restoration using compound gauss-markov random fields.," *IEEE Transactions on Image Processing*, vol. 12, no. 12, pp. 1642–1654, 2003.

[59] Niura, N. and Baba, N., "Extended-object reconstruction with sequential use of the iterative blind deconvolution method," *Optics Communications*, vol. 89, p. 375C379, 1992.

[60] Noll, R. J., "Zernike polynomials and atmospheric turbulence," *Journal of the Optical Society of America A*, vol. 66, pp. 207–211, Mar. 1976.

[61] Olsen, O. F. and Nielsen, M., "The generic structure of the optic flow field," *Journal of Mathematical Imaging and Vision*, vol. 24, no. 1, pp. 37–53, 2006.

[62] Ong, E. P. and Spann, M., "Robust optical flow computation based on least-median-of-squares regression," *International Journal Computer Vision*, vol. 31, no. 1, pp. 51–82, 1999.

[63] Pai, H.-T. and Bovik, A. C., "On eigenstructure-based direct multichannel blind image restoration.," *IEEE Transactions on Image Processing*, vol. 10, no. 10, pp. 1434–1446, 2001.

[64] Panci, G., Campisi, P., Colonnese, S., and Scarano, G., "Multichannel blind image deconvolution using the bussgang algorithm: spatial and multiresolution approaches.," *IEEE Transactions on Image Processing*, vol. 12, no. 11, pp. 1324–1337, 2003.

[65] Panofsky, H. A. and Dutton, J. A., *Atmospheric Turbulence : Models and Methods for Engineering Applications*. Wiley, 1984.

[66] Petrovic, V. S. and Xydeas, C. S., "Gradient-based multiresolution image fusion.," *IEEE Transactions on Image Processing*, vol. 13, no. 2, pp. 228–237, 2004.

[67] Rajagopalan, A. N. and Chaudhuri, S., "A recursive algorithm for maximum likelihood-based identification of blur from multiple observations.," *IEEE Transactions on Image Processing*, vol. 7, no. 7, pp. 1075–1079, 1998.

[68] Reeves, S. J. and Mersereau, R. M., "Optimal estimation of the regularization parameter and stabilizing functional for regularized image restoration," *Optical Engineering*, vol. 29, pp. 446–454, May 1990.

[69] Reeves, S. J. and Mersereau, R. M., "Blur identification by the method of generalized cross-validation," *IEEE Trans. Image Processing*, vol. 1, pp. 301–311, July 1992.

[70] Richardson, W. H., "Bayesian-based iterative method of image restoration," *J. Optical Society of America*, vol. 62, pp. 55–59, 1972.

[71] Roggemann, M. C. and Welsh, B., *Imaging Through Turbulence*. CRC Press, 1996.

[72] Rosenfeld, A., "Some uses of pyramids in image processing and segmentation," in *Proc. of the DARPA Image Understanding Workshop*, pp. 112–120, 1980.

[73] Schulz, T. J., "Multi-frame blind deconvolution of astronomical images," *Journal of the Optical Society of America A*, vol. 10, pp. 1064–1073, 1993.

[74] Shepp, L. A. and Vardi, Y., "Maximum likelihood reconstruction in positron emission tomography," *IEEE Trans. Medical Imaging*, vol. 1, no. 2, pp. 113–122, 1982.

[75] Sheppard, D. G., Hunt, B. R., and Marcellin, M. W., "Iterative multiframe super-resolution algorithms for atmospheric turbulence-degraded imagery," in *in Proc. of the IEEE Int. Conf. Acoustics, Speech, and Signal Processing*, (Seattle, USA), pp. 2857–2860, May 12–15, 1998.

[76] Simoncelli, E. and Adelson, E., "Noise removal via bayesian wavelet coding," in *Proc. of the IEEE International Conference on Image Processing*, (Lausanne, Switzerland), IEEE Computer Society Press, Sept. 1996.

[77] Smith, M. J. T. and Barnwell, T., "A procedure for designing exact reconstruction filter banks for tree-structured subband coders," in *Proc. of the IEEE Int. Conf. Acoustics, Speech, and Signal Processing*, pp. 27.1.1–27.1.4, Mar. 1984.

[78] Srivastava, A., Lee, A. B., Simoncelli, E. P., and Zhu, S. C., "On advances in statistical modeling of natural images," *Journal of Mathematical Imaging and Vision*, vol. 18, Jan. 2003.

[79] Sroubek, F. and Flusser, J., "Multichannel blind iterative image restoration.," *IEEE Transactions on Image Processing*, vol. 12, no. 9, pp. 1094–1106, 2003.

[80] Sroubek, F. and Flusser, J., "Multichannel blind deconvolution of spatially misaligned images.," *IEEE Transactions on Image Processing*, vol. 14, no. 7, pp. 874–883, 2005.

[81] Stockham Jr., T. G., Cannon, T. M., and Ingebresten, R. B., "Blind deconvolution through digital signal processing," *Proc. IEEE*, vol. 63, pp. 678–692, Apr. 1975.

[82] Tatarski, V. I., *Wave Propagation in a Turbulent Medium.* McGraw-Hill, 1961.

[83] Tekalp, A. M., Kaufman, H., and Woods, J. W., "Identification of image and blur parameters for the restoration of noncausal blurs," *IEEE Trans. Acoustics, Speech, Signal Processing*, vol. 34, pp. 963–972, Aug. 1986.

[84] Tekalp, R. L. L. A. M. and Biemond, J., "Maximum likelihood image and blur identification: a unifying approach," *Optical Engineering*, vol. 29, pp. 422–435, May 1990.

[85] Turk, M. and Pentland, A., "Eigenfaces for recognition," *Journal of Cognitive Neuroscience*, vol. 3, pp. 71–86, Winter 1991.

[86] Wirnitzer, B., "Bispectral analysis at low light levels and astronomical speckle masking," *Journal of the Optical Society of America A*, vol. 2, pp. 14–21, Jan. 1985.

[87] Wolberg, G., *Digital Image Warping.* Los Alamitos, CA, USA: IEEE Computer Society Press, 1994.

[88] Yap, K.-H., Guan, L., and Liu, W., "A recursive soft-decision approach to blind image deconvolution," *IEEE Transactions on Signal Processing*, vol. 51, pp. 515–526, Feb. 2003.

[89] You, Y.-L. and Kaveh, M., "A regularization approach to joint blur identification and image restoration.," *IEEE Transactions on Image Processing*, vol. 5, no. 3, pp. 416–428, 1996.

[90] Zhang, T. X., Hong, H. Y., and Shen, J., "Restoration algorithms for turbulence-degraded images based on optimized estimation of discrete values of overall point spread functions," *Optical Engineering*, vol. 44, pp. 17005:1–17, 2005.

VITA

Dalong Li was born in Luoyang, China, in 1976. He received the B. S. from University of Petroleum in 1998 and the M. S. from the Institute of Automation, Chinese Academy of Science in 2001. He received the Ph. D. degree in 2007 from the Georgia Institute of Technology. He interned at HP lab, Kodak Research, the Mathworks as well as Philips Research during his PhD program. His research interests include image restoration, blur identification, and machine learning.