Adaptive Local Statistics Filtering

by

Robert Adriaanse

A thesis

presented to the University of Waterloo

in fulfilment of the

thesis requirement for the degree of

Master of Applied Science

in

Systems Design Engineering

Waterloo, Ontario, Canada, 1997

©Robert Adriaanse 1997

National Library of Canada
Bibliothèque nationale du Canada

Acquisitions and Bibliographic Services
Acquisitions et services bibliographiques

395 Wellington Street
Ottawa ON K1A 0N4
Canada

395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file Votre référence

Our file Notre référence

The author has granted a non-exclusive licence allowing the National Library of Canada to reproduce, loan, distribute or sell copies of his/her thesis by any means and in any form or format, making this thesis available to interested persons.

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque nationale du Canada de reproduire, prêter, distribuer ou vendre des copies de sa thèse de quelque manière et sous quelque forme que ce soit pour mettre des exemplaires de cette thèse à la disposition des personnes intéressées.

The author retains ownership of the copyright in his/her thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced with the author's permission.

L'auteur conserve la propriété du droit d'auteur qui protège sa thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

0-612-21530-X

Canada

The University of Waterloo requires the signatures of all persons using or photocopying this thesis. Please sign below, and give address and date.

Abstract

One important image processing task concerns the restoration of blurred images degraded by additive noise. This thesis describes and compares locally adaptive Wiener filtering techniques in the spatial and frequency domains. The Wiener filter minimizes the Mean Squared Error (MSE).

The first technique describes an extension of locally adaptive Wiener filtering in the spatial domain. Assuming an exponentially decaying autocorrelation function, a noncausal filter is developed whose adaptive properties are dependent on the local signal autocorrelation. The development yields a recursive filter with pole positions based on local signal and noise variance and local signal autocorrelation. A two dimensional discrete implementation of this filter in the form of a recursive noncausal linear difference equation is derived. The properties of this filter are discussed and an iterative implementation is presented.

The second filtering technique is a Wiener filter based on the local power spectrum estimate. This technique makes no assumptions about the signal model and instead directly estimates the local power spectrum but requires a sufficient local neighbourhood for a reliable estimate.

Both these techniques are tested using synthetic and real images to demonstrate the adaptive nature. The results so far show that the spatial domain version provides effective smoothing due to a large region of support, reasonable edge preservation especially in noisy and low contrast conditions and smoothing along edges. The mean squared error is less than that of other noncasual Wiener filters and less than that of the Lee filter.

Acknowledgements

This project is the result of efforts from several people whom I wish to acknowledge. I wish to thank Professor M. Ed Jernigan for his guidance in defining this thesis, encouraging independent thought and valuable advice during the research. As always, my parents and Monika have supported me in every way. I am also grateful for support and criticism from VIP members past and present, who made it fun the whole way through.

Contents

1 Introduction **1**

2 Background **4**

2.1 Parametric Techniques 4

2.1.1 Lee Filter 4

2.1.2 Noncausal Symmetric Recursive Filter 6

2.2 Non-Parametric Techniques 6

3 Restoration Techniques **8**

3.1 Parametric Approaches 8

3.1.1 Introduction 8

3.1.2 Recursive Lee Model 9

3.1.3 Autocorrelation Model 10

3.1.4 Discrete Filter and LDE 15

3.1.5 Zero Frequency Filter Response 16

3.1.6 Hybrid Filter 22

3.1.7 Summary 23

3.1.8 Two Dimensional Implementation 24

3.2 Non-Parametric Approaches 26

3.2.1 Introduction 26
3.2.2 Block Spectral Estimation 28
3.2.3 Continuous Spectral Estimation 29

4 Results **30**
4.1 Introduction 30
4.2 Test Image: Edge Image 34
4.3 Test Image: Non-Stationary Image 39
4.4 Test Image: Face Image 44

5 Conclusions **49**

A Autocorrelation **52**
A.1 Introduction 52
A.2 Local Statistics 52
A.3 Autocorrelation in Images 54
A.4 Lag k Autocorrelation 55

Bibliography **58**

List of Figures

3.1 Frequency Response of the Non-DC Term as in Equation 3.15 12

3.2 Continuous Time Block Diagram of Filter as in Equation 3.16 13

3.3 Frequency Response of the Filter as in Equation 3.16 Assuming a Nearest Neighbour Average Term 14

3.4 Discrete Time Block Diagram of Filter 15

3.5 Frequency Response of Non-Iterative Mean Filter 17

3.6 Frequency Response of Nearest Neighbour Average Filter 18

3.7 Frequency Response of Recursive 3 Point Average Filter 21

4.1 Original and Original with Additive Noise 34

4.2 Non-Iterative Mean and Nearest Neighbour MSE Filters 35

4.3 3 Point Average and Hybrid MSE Filters 36

4.4 Global and Block Frequency Domain Filters 37

4.5 Continuous Frequency Domain and Lee Filters 38

4.6 Original and Original with Additive Noise 39

4.7 Non-Iterative Mean and Nearest Neighbour MSE Filters 40

4.8 3 Point Average and Hybrid MSE Filters 41

4.9 Global and Block Frequency Domain Filters 42

4.10 Continuous Frequency Domain and Lee Filters 43

4.11 Original and Original with Additive Noise 44

4.12 Non-Iterative Mean and Nearest Neighbour MSE Filters 45

4.13 3 Point Average and Hybrid MSE Filters 46

4.14 Global and Block Frequency Domain Filters 47

4.15 Continuous Frequency Domain and Lee Filters 48

A.1 Autocorrelation Cross Products:Lag 1 55

A.2 Autocorrelation Cross Products: Lags 1 to 6 56

Chapter 1

Introduction

One important task in digital image processing concerns the restoration of blurred images degraded by additive noise. This problem occurs in many practical contexts; distortion by additive noise can be due to poor quality acquisition, images observed in a noisy environment, or noise inherent in a communications channel. A benefit to researchers is that this type of noise can be easily simulated and modelled in laboratory conditions. Inherent in the restoration process is the need for criteria to measure the effectiveness of the restoration technique. The criteria may encompass resulting image quality and image characteristics or computer resources needed. The effectiveness of the image restoration algorithm depends on the basis of the algorithm and on any assumptions such as the noise and image models used.

The classical minimum Mean Squared Error (MSE) or Wiener [7] filter uses a model which assumes that the original image and noise are globally stationary. Global stationarity refers to a signal with signal statistics that remain relatively constant throughout the image. Images are in general not stationary; in fact it is exactly that non-stationarity which manifests itself in images consisting of regions of relative smoothness and regions of high edge content. The assumption of global stationarity ignores the locally changing

nature of the statistics typical in a broad class of images.

In contrast to the Global Wiener filter, the Lee filter[12] assumes a filter structure and uses local statistics from a small fixed window around a pixel of interest to estimate the non-degraded value at that point. The assumption of a filter structure parameterizes the image characteristics into a simple form.

The "middle ground" between these two approaches are filters which are locally adaptive but incorporate a greater region of support than the Lee filter. A Wiener filter which locally estimates the power spectrum at various regions in the image is interesting because it is adaptive yet does not impose an explicit image model like the Lee filter. Alternatively, a spatial domain filtering technique which parameterizes the image model may neatly avoid the practical problems of local power spectrum estimation. The goal of this thesis is to explore these two alternatives to find a better restoration technique.

The background section describes the well known Lee filter and briefly describes the classical Wiener filter. The description of these filters serves to put into context subsequent sections which focus on the problem of optimal point estimation given the local statistics and varying neighbourhood of support.

In the methods chapter, the first section explores parametric approaches. First, an optimal recursive point estimator based on the Lee filter structure is developed to motivate the idea of using recursive adaptive filters. The second section describes an extension of locally adaptive Wiener filtering in the spatial domain. Recursive filters are spatial domain filters that are appropriate for the image restoration problem and use a combination of local inputs and local outputs. This class of filters is parametric in the sense that the signal model and filter structure must be assumed.

In the next section, non-parametric approaches are used which forgo any assumptions of a signal model. Instead, the local power spectrum estimate and noise spectrum estimate directly find the optimal filter. Frequency domain filters do not require assumptions about

the signal model; they measure the signal charactersitics non-parametrically. They do, however, require a significant local neighbourhood to estimate the local power spectrum which may make them less responsive to local variations in the image.

The results section assesses the visual and MSE performance of these filters on both simulated and real images. Lastly, the conclusions summarizes the comparison of these techniques and recommends future directions.

Chapter 2

Background

This section introduces the concepts of local optimal point estimators with varying neighbourhoods of support. The section first briefly describes the local linear minimum squared error filter or Lee filter that estimates local noise and signal contributions and weights the local signal and mean accordingly. The next section describes a filter developed by Erler[4] which is a noncausal Wiener filter which assumes a global fixed autocorrelation. Finally, the classical frequency domain Wiener filter is briefly introduced.

2.1 Parametric Techniques

2.1.1 Lee Filter

The Lee filter is a simple heuristic which uses a combination of the signal and the local signal mean to produce an estimate which reduces noise in smooth areas and preserves the signal elsewhere. Lee assumes a zero mean additive independent white noise with known variance. In this model,

$$x(n) = s(n) + n(n) \tag{2.1}$$

$$\overline{x(n)} = \overline{s(n)} + \overline{n(n)} = \overline{s(n)} \tag{2.2}$$

where $x(n)$, $s(n)$ and $n(n)$ are the observed image, uncorrupted image and noise respectively. The variances are similarly related.

$$\sigma_x^2(n) = \sigma_s^2(n) + \sigma_n^2(n) \tag{2.3}$$

The form of the Lee [12] filter chosen apriori is

$$y(n) = \alpha x(n) + \beta \tag{2.4}$$

Minimizing the mean square error between $x(n)$ and $s(n)$ of this linear filter

$$MSE = E[(y_l(n) - s(n))^2] = \overline{(\alpha x(n) + \beta - s(n))^2} \tag{2.5}$$

and setting the derivatives to zero with respect to α and β yields

$$\alpha = \frac{\sigma_x^2(n) - \sigma_n^2}{\sigma_x^2(n)} \tag{2.6}$$

$$\beta = (1 - \alpha)\overline{x(n)} \tag{2.7}$$

where $\sigma_x^2(n)$ and $\overline{x(n)}$ are the estimates of local signal variance and local mean respectively and $\sigma_n^2(n) = \sigma_n^2$ since the noise is stationary. The filter response $y_l(n)$ is then

$$y_l(n) = \frac{\sigma_x^2(n) - \sigma_n^2}{\sigma_x^2(n)} x(n) + \frac{\sigma_n^2}{\sigma_x^2(n)} \overline{x(n)} \tag{2.8}$$

The Lee filter uses a noise variance estimate and a local signal variance estimate to maximize smoothing at low signal variance and acts as an all pass filter at high signal variance. The Lee filter emulates the Wiener filter in the form of a linear finite impulse response local point estimator; it provides effective smoothing and preserves edges at the expense of noisiness around edges.

2.1.2 Noncausal Symmetric Recursive Filter

Erler [5] has developed a recursive filter as a noncausal Wiener filter, where the signal autocorrelation is assumed to be a decaying exponential with constant of 0.5.

$$R_s(n) = \sigma_s^2 0.5^{|n|} + \mu_s^2 \tag{2.9}$$

The filter has an implicitly infinite region of support which allows it to powerfully smooth in a way that the Lee filter cannot. This brief explanation is simply meant to review previous work in this area and a more detailed description of this model will be explored in section 3.1.

2.2 Non-Parametric Techniques

Non-parametric techniques estimate the image charactersitics without assuming any apri-ori models. The power spectrum, or the squared magnitude of the Fourier transform, is an example. The global power spectrum $P(\Omega)$ of a function $s(n)$ is

$$P_s(\Omega) = |S(\Omega)|^2 = \left| \sum_{n=-\infty}^{\infty} s(n) e^{-2j\pi\Omega n} \right|^2 \tag{2.10}$$

If the power spectrum of the noise is known, then the Wiener filter can be calculated

$$H(\Omega) = \frac{P_s(\Omega)}{P_s(\Omega) + P_n(\Omega)} \tag{2.11}$$

where $P_n(\Omega) = \sigma_n^2$ is the global noise variance estimate for the additive white noise process. This technique is globally optimal in the minimum squared error sense. The power spectrum underestimates signal energy in the vicinity of edges which leads to excessive blurring at signal edges.

Chapter 3

Restoration Techniques

3.1 Parametric Approaches

3.1.1 Introduction

Section 3.1.2 begins by developing an alternate form of the Lee filter which puts it into the context of recursive filters. Section 3.1.3 extends this idea to develop a spatial domain Wiener filter which is based on an exponentially decaying signal model. This model produces a filter which has smoothing and edge preservation terms. These terms are described in relation to their frequency response and combined to arrive at the filter transfer function.

A two dimensional discrete implementation of this filter in the form of a noncausal linear difference equation is derived. Local mean and variance estimates are used, and the local signal autocorrelation is expressed in the form of a nearest neighbour pixel correlation coefficient. The properties of the filter are discussed and an iterative implementation is presented.

3.1.2 Recursive Lee Model

Recall the Lee filter

$$y(n) = \alpha x(n) + \beta = \frac{\sigma_x^2 - \sigma_n^2}{\sigma_x^2} x(n) + \frac{\sigma_n^2}{\sigma_x^2} \overline{x(n)} \tag{3.1}$$

Consider an alternate form of the Lee filter which has a recursive term instead of the local mean

$$y(n) = \alpha x(n) + (1 - \alpha) y(n-1) \tag{3.2}$$

The structure in equation 3.2 is the same as the Lee filter, except that the neighbouring output is used instead of the local mean. Minimizing the mean square error between $y(n)$ and $s(n)$ of this filter

$$MSE = E[(y(n) - s(n))^2] = \overline{(\alpha x(n) + (1-\alpha) y(n-1) - s(n))^2} \tag{3.3}$$

and setting the derivative with respect to α to zero yields

$$\overline{[\alpha(x(n) - y(n-1)) + y(n-1) - s(n)](x(n) - y(n-1))} = 0 \tag{3.4}$$

so

$$\alpha = \frac{\overline{(s(n) - y(n-1))(x(n) - y(n-1))}}{\overline{(x(n) - y(n-1))^2}} \tag{3.5}$$

or

$$\alpha = \frac{\overline{s(n)x(n)} - \overline{y(n-1)x(n)} - \overline{y(n-1)s(n)} + \overline{y^2(n-1)}}{\overline{x^2(n)} - 2\overline{y(n-1)x(n)} + \overline{y^2(n-1)}} \tag{3.6}$$

Note $x(n) = s(n) + n(n)$ and $n(n)$ is independent zero mean Gaussian noise, $\overline{x(n)} = \overline{s(n)}$, $\overline{s(n)x(n)} = \overline{s^2(n)}$ and $\overline{y(n-1)x(n)} = \overline{y(n-1)s(n)}$, so

$$\alpha = \frac{\overline{(s(n) - y(n-1))^2}}{\overline{n^2(n)} + \overline{(s(n) - y(n-1))^2}} = \frac{\sigma^2_{s_n y_{n-1}}}{\sigma^2_n + \sigma^2_{s_n y_{n-1}}} \tag{3.7}$$

where σ^2_n, and $\sigma^2_{s_n y_{n-1}}$ are the estimates of noise variance and covariance between $s(n)$ and $y(n-1)$ respectively. The resulting filter $y(n)$ is

$$y(n) = \frac{\sigma^2_{s_n y_{n-1}}}{\sigma^2_n + \sigma^2_{s_n y_{n-1}}} x(n) + \frac{\sigma^2_n}{\sigma^2_n + \sigma^2_{s_n y_{n-1}}} y(n-1) \tag{3.8}$$

Since $y(n-1)$ is an estimate of $s(n-1)$, the covariance $\sigma^2_{s_n y_{n-1}}$ is an estimate of $\sigma^2_{s_n s_{n-1}}$ which, when expressed in standard form, is the correlation between $s(n)$ and $s(n-1)$, or the lag 1 autocorrelation of $s(n)$. This section has served to demonstrate that autocorrelation is a useful notion for recursive MSE filters of the form in equation 3.2. Instead of relying on the ad hoc assumption of the filter structure as in equation 3.2, the idea of incorporating autocorrelation into the signal model and developing Wiener filters based on this model is explored in the next section.

3.1.3 Autocorrelation Model

A signal model [6] for a broad class of images assumes a local autocorrelation function of the form

$$R_s(n) = \sigma^2_s a^{|n|} + \mu^2_s, |a| < 1 \tag{3.9}$$

where R_s is the autocorrelation function, σ^2_s and μ^2_s are the local signal mean and variance and a is the autocorrelation coefficient. The model in equation 3.9 assumes that the correlation between pixels in an image decays exponentially with increasing distance. The autocorrelation coefficient in a discrete signal is

$$a = \frac{R_s(1) - \mu_s^2}{\sigma_s^2} = \frac{E\left[s(n)s(n+1)\right] - \mu_s^2}{\sigma_s^2} \tag{3.10}$$

where $s(n)$ is the signal and $s(n+1)$ is the nearest neighbour and $E\left[s(n)s(n+1)\right]$ is the expected value of the product $s(n)s(n+1)$. Note that the distribution of the coefficient a ranges from -1 to 1. The local signal power spectrum is

$$P_s(\Omega) = \frac{\sigma_s^2(1-a^2)}{1+a^2-2a\cos\Omega} + \mu_s^2\delta(\Omega) \tag{3.11}$$

The power spectrum depends on the autocorrelation and frequency in the first term, while the $\delta(\Omega)$ term determines the DC response. Recall the Wiener filter

$$H(\Omega) = \frac{P_s(\Omega)}{P_s(\Omega) + P_n(\Omega)} \tag{3.12}$$

where $P_n(\Omega) = \sigma_n^2$ is the global noise variance estimate. The noise variance is constant for an additive white noise process so the discrete response of the corresponding adaptive Wiener filter is

$$H(\Omega) = \frac{\frac{\sigma_s^2(1-a^2)}{1+a^2-2a\cos\Omega} + \mu_s^2\delta(\Omega)}{\frac{\sigma_s^2(1-a^2)}{1+a^2-2a\cos\Omega} + \mu_s^2\delta(\Omega) + \sigma_n^2} = \begin{cases} 1 & ,\Omega = 0 \\ \frac{(1-a^2)SNR}{(1-a^2)SNR+1+a^2-2a\cos\Omega} & ,\Omega \neq 0 \end{cases} \tag{3.13}$$

where SNR $= \sigma_s^2/\sigma_n^2$ is the signal to noise ratio. The term

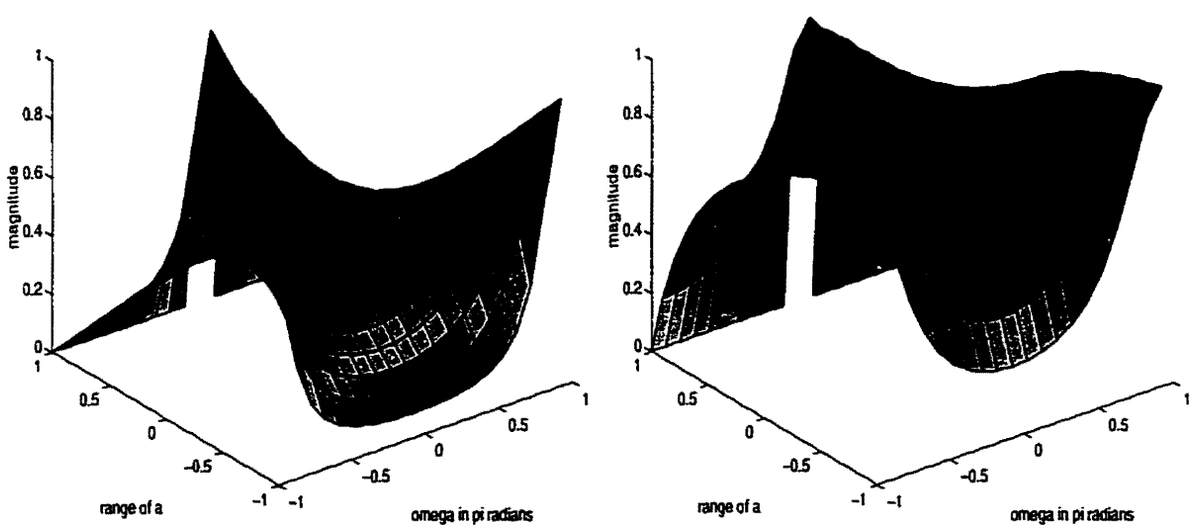
$$H(\Omega)|_{\Omega=0} = 1 \tag{3.14}$$

means that the DC gain is one, so the input mean is the same as the output mean in each local neighbourhood. The adaptive term provides the locally adaptive behavior dependent on the frequency Ω and the autocorrelation coefficient a

$$H(\Omega)|_{\Omega \neq 0} = \frac{(1-a^2)SNR}{(1-a^2)SNR + 1 + a^2 - 2a\cos\Omega} \tag{3.15}$$

The $(1-a^2)SNR$ term occurs in both the numerator and the denominator. Since $SNR > 0$ and $-1 < a < 1$, $(1-a^2)SNR > 0$. The denominator term differs from the numerator by $1 + a^2 - 2a\cos\Omega$. The term $1 + a^2 - 2a\cos\Omega$ is small when $(\Omega, a) \approx (0, 1)$ and $(\Omega, a) \approx (\pi, -1)$ so the filter passes low frequencies at $a \approx 1$ and high frequencies at $a \approx -1$. Figure 3.1 shows the smoothing effect at high autocorrelation and enhancement effect at low autocorrelation at SNR of 1 and 5 respectively.

Figure 3.1: Frequency Response of the Non-DC Term as in Equation 3.15



(a) Frequency Response at SNR = 1 (b) Frequency Response at SNR = 5

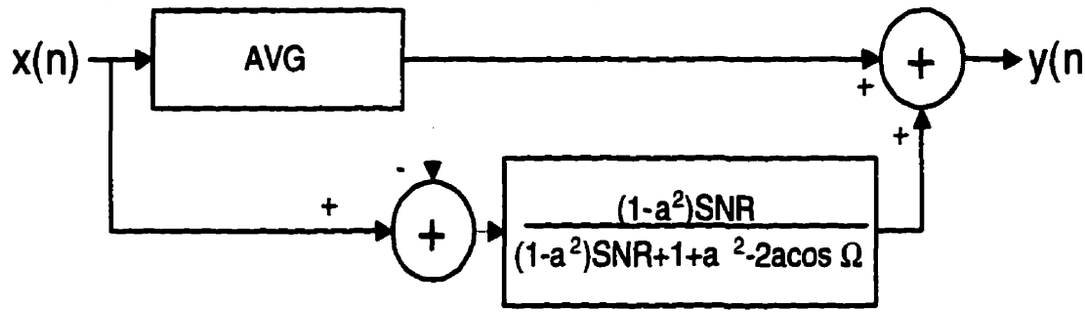
As the graph on the left of Figure 3.1 shows, the response of non-DC term is always all pass at zero autocorrelation. There is high frequency emphasis when the autocorrelation is less than zero, and low frequency emphasis when the autocorrelation is greater than

zero. Increasing the SNR tends to make the filter more all pass, as the graph on the right of Figure 3.1 shows. In summary, positive correlation corresponds to smoothing and negative autocorrelation corresponds to enhancement and high SNR correponds to a more all pass response.

Filter Response

The block diagram of the filter in figure 3.2 shows the filter terms. Note the AVG term indicates the unity DC gain referred to in equation 3.14. The representation of this term must be considered. This term is locally calculated and is meant to represent a frequency response of $H(\Omega)|_{\Omega=0} = 1$. It must approximate an impulsive response, which may be problematic since the filter uses only a local neighbourhood. The issue of finding an appropriate AVG term is crucial and various AVG models based on the criteria of equation 3.14 will be developed in section 3.1.5.

Figure 3.2: Continuous Time Block Diagram of Filter as in Equation 3.16



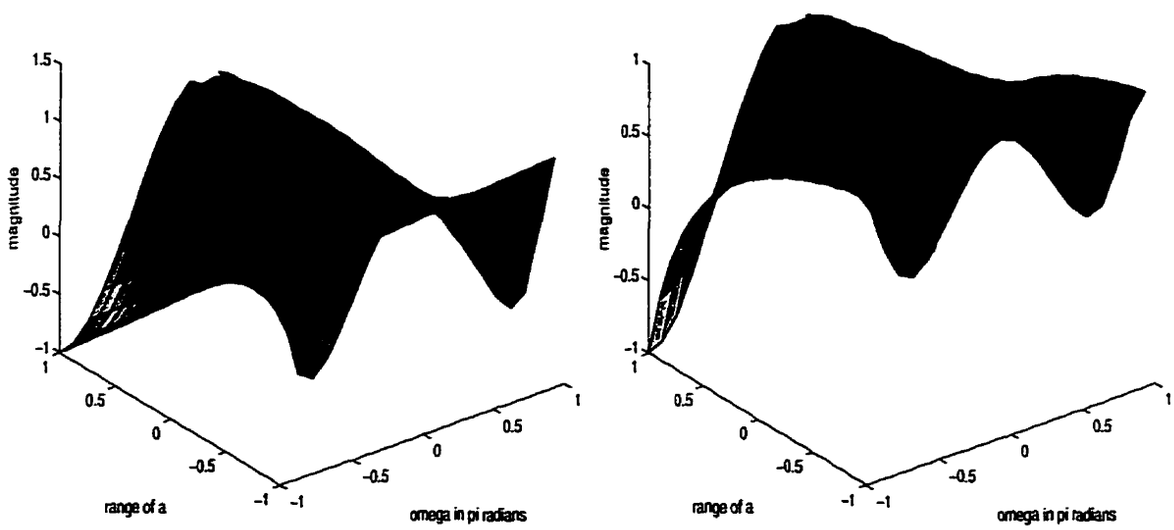
The frequency response of the filter is

$$H(\Omega) = AVG(\Omega) + (1 - AVG(\Omega))\frac{(1-a^2)SNR}{(1-a^2)SNR + 1 + a^2 - 2a\cos\Omega} \tag{3.16}$$

To see how these terms affect filter performance, it is useful to view the overall frequency

response of this filter for varying conditions of Ω and a. Assuming, for now, an AVG function which has a low pass response which consists of the average of the nearest neighbours (described in section 3.1.5) around the point of interest, figure 3.3 shows the response at $SNR = 1$ and $SNR = 5$ respectively.

Figure 3.3: Frequency Response of the Filter as in Equation 3.16 Assuming a Nearest Neighbour Average Term



(a) Frequency Response at SNR = 1 (b) Frequency Response at SNR = 5

The graph shows that at negative autocorrelation, the filter tends to be all pass, especially at high SNR. This is exactly what is desired for a Wiener filter since negative autocorrelation occurs in regions of high edge content. At positive autocorrelation the filter tends to smooth very strongly. It should be noted that the initial assumption of the autocorrelation function imposes a model on the filter and has a direct impact on the structure and smoothing behavior of the resulting filter in equation 3.16 .

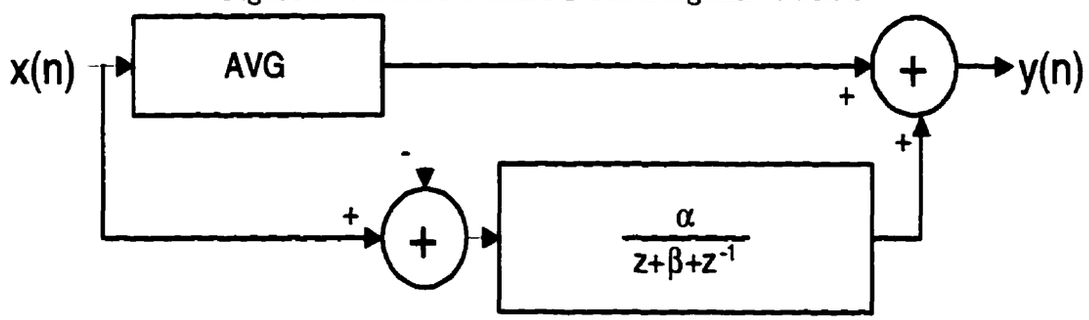
In non-edge regions of the image, autocorrelation tends to be zero since there is

random variation about a local mean. In these regions, the filter tends to smooth as well, which is desirable. Autocorrelation is positive along edges, smoothing here allows the filter to smooth along edges while edge preserving across edges. In summary, the filter in figure 3.3 has a number of desirable properties: it is able to smooth non-edge regions, preserve edges, and smooth along edges.

3.1.4 Discrete Filter and LDE

This section describes the implementation and difference equation. Figure 3.4 shows the discrete version of the filter. The transfer function using α and β is

Figure 3.4: Discrete Time Block Diagram of Filter



$$H(z) = AVG(z) + (1 - AVG(n))\left(\frac{\alpha}{z + \beta + z^{-1}}\right) \tag{3.17}$$

where

$$\alpha = \frac{-(1 - a^2)SNR}{a} \tag{3.18}$$

and

$$\beta = \frac{-[(1 - a^2)SNR + 1 + a^2]}{a} \tag{3.19}$$

and where the AVG function must have the desired unity gain at zero frequency. The best practical approximation for the AVG function is discussed in the next section.

3.1.5 Zero Frequency Filter Response

Consider the response of the filter at zero frequency. Recall the term

$$H(\Omega)|_{\Omega=0} = 1 \tag{3.20}$$

corresponds to a DC gain of unity. The question is, how best can equation 3.20 be represented in a way that fits into the filter which already has the recursive non-DC term? Since the function is supposed to represent the zero frequency component, a mean should be used. The mean function could be a simple non-recursive local average. It could also be an iterative function that incorporates neighbouring terms which could be effective in smoothing. In this case, the iterative structure implies a large region of support and hence a more effective smoothing at subsequent iterations. In the following subsections, these functions are first explained in terms of their effect on the filter and then compared in terms of their frequency response.

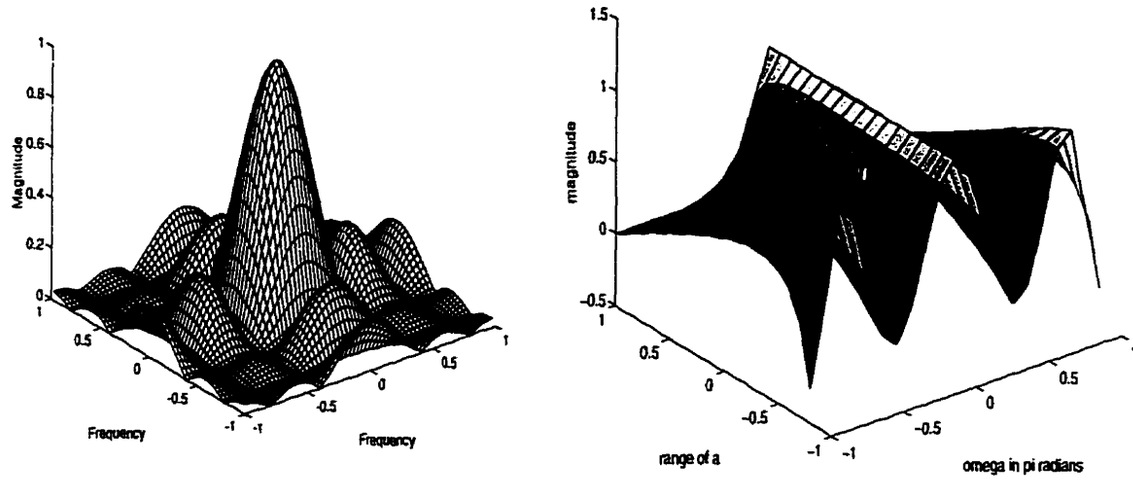
Non-Iterative Mean

The local average is taken over the same region of support as the variance and autocorrelation in order to be consistent with the model. For a seven point discrete average

$$lm = (z^3 + z^2 + z^1 + z^0 + z^{-1} + z^{-2} + z^{-3})/7 \tag{3.21}$$

which has a two dimensional frequency response shown in Figure 3.5 (a) which results in the overall frequency response in Figure 3.5 (b). The local mean provides a reasonably impulsive function to approximate equation 3.14 and the filter response provides the all-

Figure 3.5: Frequency Response of Non-Iterative Mean Filter



(a) Frequency Response of Local Mean (b) Frequency Response of Filter (SNR=1)

smoothing to all-pass behavior with increasing autocorrelation as expected. In terms of the LDE

$$\beta y(n) = \alpha x(n) + (\beta - \alpha) lm(n) + lm(n \pm 1) - y(n \pm 1) \tag{3.22}$$

where $y(n \pm 1)$ is shorthand for $y(n+1) + y(n-1)$ and lm indicates the local mean. In terms of a and the SNR

$$y(n) = \frac{((1-a^2)SNR)x(n) + (1+a^2)lm(n) - (a)lm(n \pm 1) + ay(n \pm 1)}{(1-a^2)SNR + 1 + a^2} \tag{3.23}$$

The $x(n)$ term represents the portion of the signal which is passed through. This portion is $\frac{(1-a^2)SNR}{(1-a^2)SNR+1+a^2}$ which is small when the SNR is low and nearer to unity when the SNR is large. This means the filter will smooth more at low SNR which is similar to the Lee

filter. When β is negative, autocorrelation is positive and $y(n-1)$ and $y(n+1)$ terms are added which performs smoothing. Conversely, when autocorrelation is negative, these neighbouring y terms are subtracted which provides for enhancement. The AVG term provides smoothing for a net response which ranges from all pass to maximum smoothing. The results section will show the performance of the filter in equation 3.23.

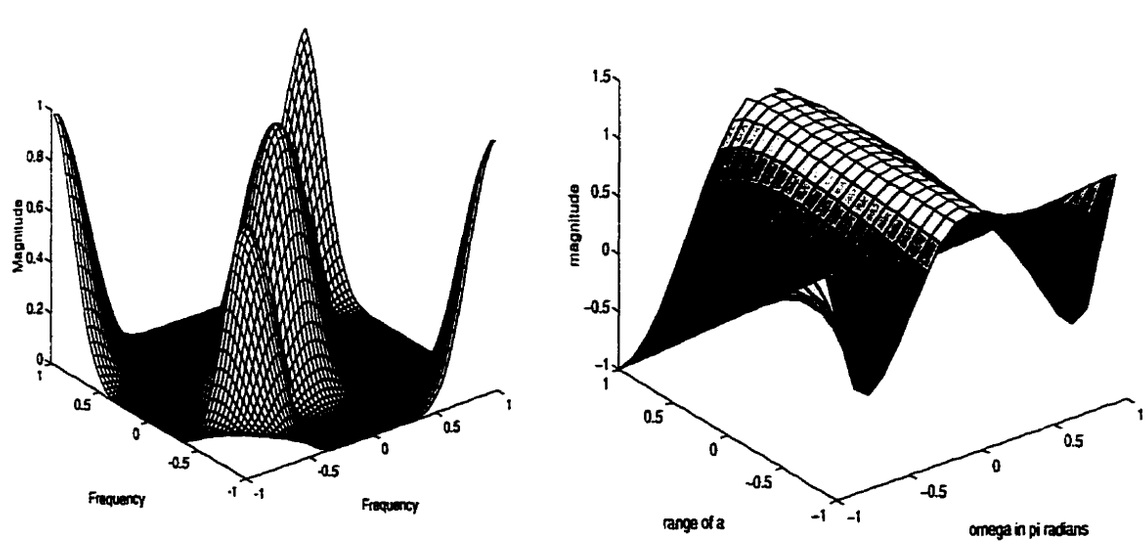
Iterative Nearest Neighbour Average

The discrete nearest neighbour average is

$$AVG_{nn}(z) = \frac{z + z^{-1}}{2} \tag{3.24}$$

which has a frequency response, after four iterations, as shown in Figure 3.6 (a). It results in the overall frequency response in Figure 3.6 (b) and the transfer function for the first

Figure 3.6: Frequency Response of Nearest Neighbour Average Filter



(a) Frequency Response of Nearest Neighbour (b) Frequency Response of Filter (SNR=1)

iteration using α and β is

$$H(z) = \left(\frac{z+z^{-1}}{2}\right) + \left(1 - \frac{z+z^{-1}}{2}\right)\frac{\alpha}{z+\beta+z^{-1}} \tag{3.25}$$

so the non constant coefficient LDE for the first iteration is

$$\beta y(n) = (\alpha+1)x(n) + \left(\frac{\beta-\alpha}{2}\right)x(n\pm 1) + \frac{1}{2}x(n\pm 2) - y(n\pm 1) \tag{3.26}$$

In terms of SNR and a for the first iteration, the filter is

$$y(n) = \frac{2((1-a^2)SNR - a)x(n) + (1+a^2)x(n\pm 1) - ax(n\pm 2) + 2ay(n\pm 1)}{2((1-a^2)SNR + 1 + a^2)} \tag{3.27}$$

Since the average function is iterative, the transfer function at iteration k is

$$H_k(z) = \left(\frac{z+z^{-1}}{2}\right)^k + \left(1 - \left(\frac{z+z^{-1}}{2}\right)^k\right)\frac{\alpha}{z+\beta+z^{-1}} \tag{3.28}$$

so the non-constant coefficient LDE is

$$\beta y_k(n) = \alpha x(n) + (\beta-\alpha)x_k(n) + x_k(n\pm 1) - y_{k-1}(n\pm 1) \tag{3.29}$$

where $x(n)$ is the original image and $x_k(n)$ is

$$x_k(n) = \sum_{m=0}^{k} \begin{pmatrix} k \\ m \end{pmatrix} \frac{x(n+k-2m)}{2^m} \tag{3.30}$$

and y is set to x for the first iteration. In terms of SNR and a, the filter is

$$y_k(n) = \frac{((1-a^2)SNR)x_1(n) + (1+a^2)x_k(n) - ax_k(n\pm 1) + ay_{k-1}(n\pm 1)}{(1-a^2)SNR + 1 + a^2} \tag{3.31}$$

The filter in equation 3.30 differs from the previous filter in that instead of using a fixed local mean, the neighbouring x terms smooth iteratively. Overall, the filter has significant smoothing since there is slightly less emphasis on the $x(n)$ term compared to the local mean filter. A summary is provided in section 3.1.7 for convenient comparison of the filters.

Iterative 3 Point Average

Another possible function is a 3 point average

$$AVG_{3p}(z) = \frac{z+1+z^{-1}}{3} \tag{3.32}$$

which has a frequency response, after 4 iterations, as shown in Figure 3.7 (a). It results in the overall frequency response in Figure 3.7 (b) so the transfer function at the first iteration using α and β is

$$H(z) = \frac{z+1+z^{-1}}{3} + \left(1 - \frac{z+1+z^{-1}}{3}\right)\frac{\alpha}{z+\beta+z^{-1}} \tag{3.33}$$

so the non constant coefficient LDE for the first iteration is

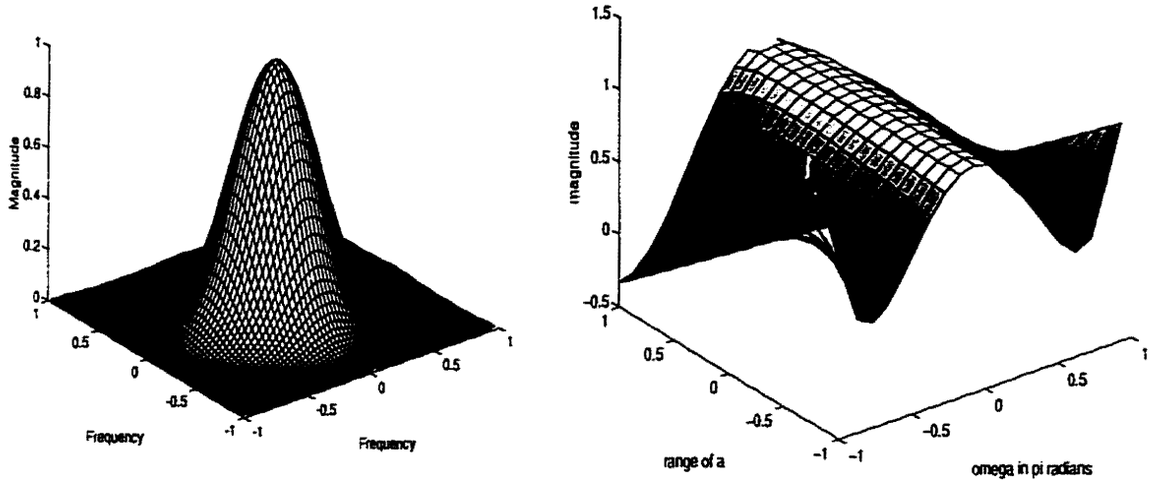
$$\beta y(n) = \frac{\beta+2\alpha+2}{3}x(n) + \frac{\beta-\alpha+1}{3}x(n\pm 1) + \frac{x}{3}(n\pm 2) - y(n\pm 1) \tag{3.34}$$

In terms of SNR and a for the first iteration, the filter is

$$y(n) = \frac{((1-a^2)SNR + \frac{1}{3}(1-a)^2)x(n) + \frac{1}{3}(1-a+a^2)x(n\pm 1) - \frac{1}{3}ax(n\pm 2) + ay(n\pm 1)}{(1-a^2)SNR + 1 + a^2} \tag{3.35}$$

This filter is similar to the nearest neighbour average filter except there is less emphasis

Figure 3.7: Frequency Response of Recursive 3 Point Average Filter



(a) Frequency Response of 3 Point Average (b) Frequency Response of Filter (SNR=1)

on the $x(n \pm 1)$ and $x(n \pm 2)$ terms, which means less smoothing. Since the average function is iterative, the transfer function at iteration k is

$$H_k(z) = \left(\frac{z+1+z^{-1}}{3}\right)^k + \left(1 - \left(\frac{z+1+z^{-1}}{3}\right)^k\right)\frac{\alpha}{z+\beta+z^{-1}} \tag{3.36}$$

so the non constant coefficient LDE is

$$\beta y_k(n) = \alpha x(n) + (\beta - \alpha)x_k(n) + x_k(n \pm 1) - y_{k-1}(n \pm 1) \tag{3.37}$$

which as the same structure as the Nearest Neighbour case except $x_k(n)$ is

$$x_k(n) = \sum_{m=0}^{k}\sum_{l=0}^{k-m}\begin{pmatrix} k \\ m \end{pmatrix}\begin{pmatrix} k-m \\ l \end{pmatrix}\frac{x(n+k-2m-l)}{3^m} \tag{3.38}$$

The results section will show the performance of the filter in equation 3.38.

3.1.6 Hybrid Filter

The approaches so far have considered a number of averaging functions to address the issue of a zero frequency filter response in order match the Wiener response in a local context. In contrast to the methods presented thus far, another alternative to satisfying the unity gain requirement is to augment the filter gain, which ranges from 0 to 1, with the input signal instead of an averaging function so that the total gain is unity. This approach follows the Lee filter in that the response changes when the local statistics support it, but is otherwise all-pass. Consider the requirement

$$|H(\Omega)|\,|_{\Omega=0} = 1 \tag{3.39}$$

and the uncompensated filter as in equation 3.15 is

$$H(\Omega)|_{\Omega\neq 0} = \frac{(1-a^2)SNR}{(1-a^2)SNR + 1 + a^2 - 2a\cos\Omega} \tag{3.40}$$

so the gain, without compensation, is

$$0 < |H(\Omega)|_{\Omega\neq 0}| < 1 \tag{3.41}$$

since

$$(1-a^2)SNR > 0 \tag{3.42}$$

and

$$1 + a^2 - 2a\cos\Omega > 0 \tag{3.43}$$

Now if the structure is assumed to be

$$H(\Omega) = \frac{(1-a^2)SNR + K}{(1-a^2)SNR + 1 + a^2 - 2a\cos\Omega} \tag{3.44}$$

then for this structure to meet the constraint of equation 3.39,

$$K = |1 + a^2 - 2a\cos\Omega||_{\Omega=0} = 1 + a^2 - 2a = (1-a)^2 \tag{3.45}$$

This approach is similar in spirit to the Lee filter in that it smoothes, but also enhances, when the local statistics supports it, and is otherwise all-pass. Pursuing this strategy may be useful in exploring a hybrid filter which has the filter richness of the autocorrelation model and local response similar to the Lee filter. The discrete response of the filter is

$$H(z) = \frac{(1-a^2)SNR + (1-a)^2}{(1-a^2)SNR + 1 + a^2 - az - az^{-1})} \tag{3.46}$$

and the non-constant coefficient difference equation is

$$y(n) = \frac{((1-a^2)SNR + (1-a)^2)x(n) + ay(n \pm 1)}{(1-a^2)SNR + 1 + a^2} \tag{3.47}$$

The hybrid filter in equation 3.47 uses only the neighbouring output terms for either smoothing or enhancement. It bridges the ideas behind the autocorrelation model and the Lee filter. The response is similar to the Wiener filters described, except that the response of the neighbouring x terms, which provide additional smoothing, are removed and the filter gain is compensated with the signal input.

3.1.7 Summary

This section compares the filter in terms of the LDE coefficients at the first and k^{th} iterations. Table 3.1 shows the coefficients at the first iteration which gives an idea of

the weightings on $x(n)$, $x(n \pm 1)$ and $x(n \pm 2)$ from the original image, and $y(n \pm 1)$, the neighbouring outputs. Table 3.2 shows the coefficients for the kth iteration on the original observed image $x(n)$, and on the AVG terms and neighbouring outputs $y(n \pm 1)$.

Table 3.1: Comparison of LDE Coefficients in terms of a and SNR for the first iteration

Filter Name	$x(n)$	$x(n \pm 1)$	$x(n \pm 2)$	$y(n \pm 1)$
Non-Iterative Mean	$(1-a^2)SNR + (1+a^2)\text{lm}$	$-(a)\text{lm}$		a
Nearest Neighbour	$(1-a^2)SNR - a$	$\frac{1}{2}(1+a^2)$	$-\frac{1}{2}a$	a
3 Point Average	$(1-a^2)SNR + \frac{1}{3}(1-a)^2$	$\frac{1}{3}(1-a+a^2)$	$-\frac{1}{3}a$	a
Hybrid	$(1-a^2)SNR + (1-a)^2$	-	-	a

Note: All terms are over $(1-a^2)SNR + 1 + a^2$.

Table 3.2: Comparison of LDE Coefficients in terms of a and SNR for the kth iteration

Filter Name	AVG type	$x(n)$	$AVG(n)$	$AVG(n \pm 1)$	$y_k(n \pm 1)$
Non-Iter.	local mean	$(1-a^2)SNR$	$1+a^2$	$-a$	a
Near. Neigh.	$\frac{x(n\pm1)}{2}$	$(1-a^2)SNR$	$1+a^2$	$-a$	a
3 Point Avg	$\frac{x(n\pm1)+x(n)}{3}$	$(1-a^2)SNR$	$1+a^2$	$-a$	a
Hybrid	none	$(1-a^2)SNR + (1-a)^2$	-	-	a

Note: All terms are over $(1-a^2)SNR + 1 + a^2$.

3.1.8 Two Dimensional Implementation

The filters developed in the previous sections have been described in one dimension only. In extending these filtering techniques to two dimensions, the goal is to find a reasonable implementation given that the images are represented in cartesian coordinates but that the signal model is anisotrophic. Anisotrophic filters have identical behavior in any direction and the two dimensional frequency response is radially symmetric about the origin. The nearest neighbour autocorrelation is calculated in x and y directions. Since the filters developed so far depend on the nearest neighbour autocorrelation estimate, it follows that the filtering occurs independently in x and y directions, and the results are summed and divided by 2 to get the result. This allows the filter to react independently

to either edge preserve or smooth in the vertical and horizontal directions. The trade-off is that the filters are not a perfect extension from one to two dimensions due to the representation of the images.

Since these filters are noncausal, they have been implemented iteratively. This implies the filter begins on the observed, degraded image, and subsequent iterations restore the image until a stopping criteria has been met. The criteria used in the filters presented so far has been to stop iterating when the change in the next iteration becomes sufficiently small. The image from the previous iteration is therefore saved and compared with the current iteration until the filter converges.

Appendix A describes exactly how the autocorrelation was estimated. The noise variance was estimated by finding the mode of the variance histogram as in [1].

3.2 Non-Parametric Approaches

3.2.1 Introduction

In this section, a non-parametric approach is used which forgoes any assumption of a signal model. Instead, the local power spectrum estimate is used to directly find the optimal filter. If the image model is known, then the problem of image restoration would be limited to finding the optimal estimates of the parameters in the model. Since only the noise model is known, it is worth exploring non-parametric techniques. The classical Wiener filter assumes the image is stationary and finds the power spectrum of the entire image. Although this technique is optimal in the global MSE sense, it ignores the locally changing nature of many images, so an adaptive version is considered.

A local frequency domain MSE filter method assumes that the image can be filtered in sections of sufficient extent to allow reasonable estimation of the power at each frequency. This presents a practical problem since a signal must be infinitely long to exactly estimate the power spectrum, but the goal is to obtain a good estimate of the power spectrum in a finite neighbourhood. In fact, the neighbourhood of samples must be small enough so that the assumption of local stationarity is reasonable but large enough so that the power spectrum can be estimated with some accuracy. The goal is to find a good estimate using as few data points as possible.

The optimal power spectrum estimate cannot be known since the exact image model is unknown, so there is no clear definition of what is meant by "optimal" in terms of how closely a power spectrum estimate will match the true power spectrum. The optimal solution balances the biasing inherent in finite neighbourhoods while preserving enough resolution for a close estimate of the true power spectrum. The two dimensional Fourier Transform is used. Recall the Wiener filter

$$H(\Omega) = \frac{P_s(\Omega)}{P_s(\Omega) + P_n(\Omega)} \tag{3.48}$$

where $P_n(\Omega) = \sigma_n^2$ is the global noise variance estimate for an additive white noise process. The local version of this filter considers $P_s(\Omega)$ as the local estimate of the signal power spectrum. The power spectrum $P(\Omega)$ of a function $x(n)$ is the square of the magnitude of the Fourier Transform which is the Fourier Transform of the autocorrelation function according to the Wiener-Knichin theorem [9]

$$P(\Omega) = \sum_{x=-\infty}^{\infty} R(k)e^{-2j\pi k\Omega} \tag{3.49}$$

where

$$R(k) = E[x(n)x(k+n)] \tag{3.50}$$

is the lag k autocorrelation function. The power spectrum therefore just takes all the autocorrelation lags of a function and changes the basis to the frequency domain which facilitates the direct implementation of the Wiener filter.

In considering a window of small spatial extent, it becomes important to consider the type of windowing function used to create the Fourier transform. The importance of windowing in spectral estimation stems from the fact that the signal is short duration while the desired frequency response $H(\Omega)$ has an infinite duration impulse response. What is required is an impulse response of finite duration whose transform adequately approximates the true impulse response. Using a rectangular windowing function introduces undesirable ripples in the transform, known as the Gibbs phenomenon. For this reason, other windowing functions were considered. These window functions are tapered so as to reduce the abrupt transition of the rectangular window. The trade-off is that some

distortion occurs which smoothes the transform. A triangular or Bartlett window[10]

$$w(n) = 1 - \frac{|n|}{(M+1)}, -M \leq n \leq M \tag{3.51}$$

and Hanning window[9]

$$w(n) = 1/2 + 1/2\cos(\frac{\pi n}{M}), -M \leq n \leq M \tag{3.52}$$

were used with real and simulated images. Tests on a variety of images indicated that for images whose power spectra have more energy concentrated at lower frequencies, the windowing functions did improve the MSE performance. In images with significant energy at higher frequencies the result was worse. In natural scene images, there was no windowing function which was consistently better in terms of MSE performance. In summary, a rectangular windowing function was used; the choice of which windowing function is "optimal" is difficult since the exact image model is unknown and so there are no clear criteria to measure how closely the power spectrum estimate matches the true power spectrum.

3.2.2 Block Spectral Estimation

The first technique estimates $P_s(\Omega)$ in a 16 by 16 neighbourhood in blocks which do not overlap. Only images whose size are a multiple of 16 can be used. Since the noise power spectrum is constant, the filter estimates the 16 by 16 region of $P_n(\Omega)$ to be $P_n(\Omega) = \sigma_n^2$ for each block. The power spectrum, $P_s(\Omega)$, is

$$P_s(\Omega) = |DTFT(x_i(n))|^2 - \sigma_n^2 \tag{3.53}$$

where $x_i(n)$ is the ith 16 by 16 neighbourhood for the image. $P_s(\Omega)$ is truncated to zero where necessary to avoid negative power spectrum values. The advantage of this technique is that the filter is fast and gives good results if the image is stationary within each block. Since the blocks do not overlap, there can be block artifacts associated with not estimating the power spectrum at every point.

3.2.3 Continuous Spectral Estimation

In contrast to non-overlapping blocks, the filter can instead calculate the estimate for each point. In this case, $P_s(\Omega)$ is

$$P_s(\Omega) = |DTFT(x(n))|^2 - \sigma_n^2 \tag{3.54}$$

where $x(n)$ is the 16 by 16 neighbourhood centred at n. After filtering the block centred at n, the central point of the result is the output $y(n)$. This process is done for every point in the image, resulting in a filter that is more computationally expensive but void of block artifacts.

Chapter 4

Results

4.1 Introduction

The previous chapters of this thesis have examined adaptive recursive and frequency domain Wiener filters for digital image filtering. The emphasis has been on creating restoration filters to remove additive Gaussian noise from natural scene images. Spatial domain Wiener filters based on the autocorrelation model and frequency domain models have been developed. In this section, the results of applying these filters are presented.

The criteria for filtering effectiveness are the MSE and the subjective appearance of the image. The images considered are 0 to 255 gray scale images. In each case, a variety of noise variances were used. The filters developed so far are

- "Lee", the Lee filter as in equation 2.8
- "Non-Iterative Mean", the spatial MSE filter using local mean as in equation 3.23
- "Nearest Neighbour", the spatial MSE filter using iterative nearest neighbour average as in equations 3.29 and 3.30

- "3 Point Average", the spatial MSE using iterative 3 point average as in equations 3.37 and 3.38
- "Hybrid", the hybrid filter which uses the autocorrelation model in a Lee structure as in equation 3.47
- "Global", the frequency domain Global Wiener Filter as in equation 2.11
- "Block", the frequency domain Block Wiener Filter as in equation 3.53
- "Continuous", the frequency domain Continuous Wiener Filter as in equation 3.54

For the filtering techniques described, three images were chosen. The first is a simple edge with flat sections on either side. It is meant to give a clear idea of the filter behavior in edge and non-edge regions. The second image is a mix of text, checkerboard pattern, and edges at various orientations which are generally highly non-stationary. This is an extreme case for image filters which depend on a local stationarity. The final image is a face with some high and low contrast edges in the background which gives a better idea of general performance.

Since the filter behaviour was similar, and because of the large number of filters and images involved, a complete set of reproductions is presented only for the noise variance of 200 case. The reader should be aware that there may be additional distortion of the resulting images due to transferring the images to this printed material. Table 4.1 summarizes the results for three images.

In general, the spatial MSE filters tended to have similar results, with the 3 Point Average filter performing better than the Non-Iterative mean and Nearest Neighbour filters. The Hybrid filter was competitive, with generally less effective smoothing but a higher tolerance for local non-stationarity, as can be seen in the non-stationary image results. The frequency domain filters were generally mediocre; they smoothed to some

Table 4.1: Summary of Results

MSE	Noise	Non-Iter.	Near. Neigh.	3 Point Average	Hybrid	Global	Block	Cont	Lee
Edge	192.1	16.18	11.38	10.98	12.90	73.99	73.21	73.91	17.44
Non-Stat	203.1	197	457	483	155	181.2	166.3	169.9	136.9
Face	200.3	63.55	64.24	51.24	59.6	105.6	106.0	102.3	58.92

extent but tended to oversmooth edges. The problem of choosing a suitable window size was made difficult by the conflicting objectives of a precise local power spectrum estimate and a small enough window over which to assume local stationarity.

In terms of computational expense, the Non-Iterative mean filter tended to converge rapidly since the local mean was calculated non-iteratively. The spatial MSE filters had the heaviest burden, and the frequency domain methods were the quickest since no local statistics were needed. Depending on the size of stationary regions, the number of iterations could be high in images with large smooth regions; typically 3-4 iterations were required for most images.

Edge Image

In this image, the recursive smoothing effects of the spatial MSE filters allowed larger implicit regions of support and hence a much lower MSE than the frequency domain methods. The Hybrid filter had a slightly higher MSE but also gave a visually more pleasing result because it did not smooth over the edge as much as the MSE filters. The frequency domain methods were not as effective although they did some overall smoothing. The Lee result was similar to the Hybrid filter, although it did not smooth as well since it lacks a recursive smoothing strusture.

Non-Stationary Image

This image was interesting because it is an extreme case of a non-stationary image to test filters which rely on stationarity assumptions. The non-stationary image contains a number of sharp, thin, diagonal edges which were smoothed by the spatial MSE filters, greatly increasing the MSE for Nearest Neighbour and 3 Point Average cases. The Hybrid filter tended to be more conservative. It smoothed only when the SNR and autocorrelation was high enough to justify it. This effect was positive where the assumption of local stationarity was bad. The frequency domain methods performed on the non-stationary image produced results similar to the edge image case; slight smoothing reduced the MSE compared to the spatial MSE methods.

Face Image

The face image is a more typical natural image which has regions of relative smoothness and some areas of high edge content. The spatial MSE filters were able to smooth to some extent, although this was limited by the presence of background texture. The frequency domain methods were not as effective but did do some overall smoothing.

4.2 Test Image: Edge Image

Figure 4.1 shows the original and the original plus noise. As in all the images in the results section, the noise variance is 200 and the images consist of 256 gray levels. For the edge image, the edge height is 20.

Figure 4.1: Original and Original with Additive Noise

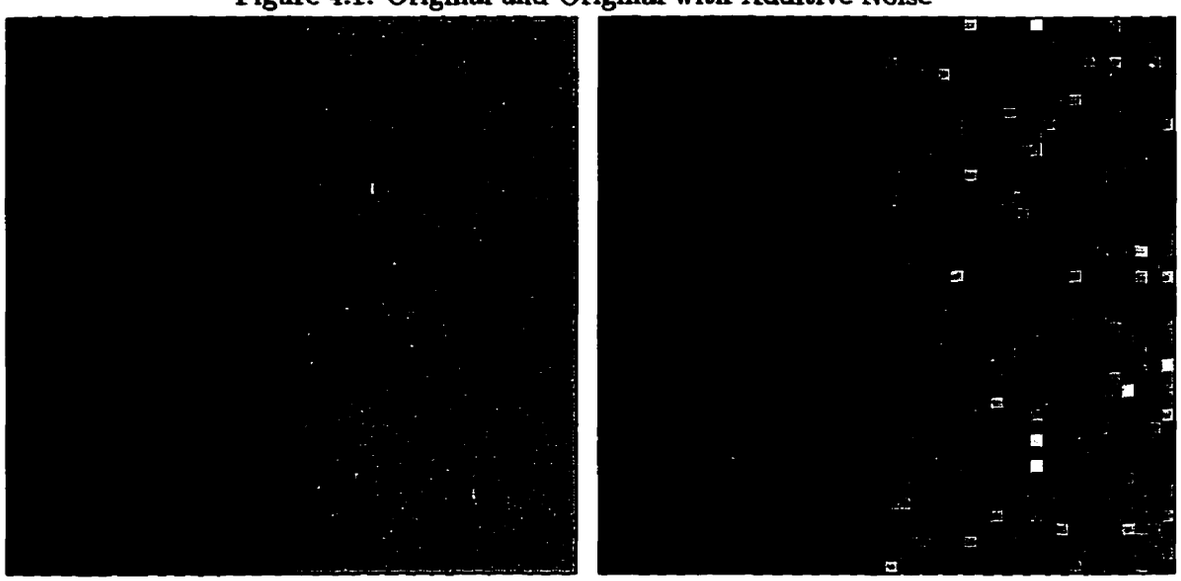


Figure 4.2 shows the Non-Iterative Mean and Nearest Neighbour average filters. The Non-Iterative Mean filter shows some patchiness in the smooth regions because the non-recursive smoothing has a limited spatial extent. The area in the vicinity of the edge has more noise since smoothing is not as strong. The Nearest Neighbour filter shows effective smoothing, with some erosion of the edge due to some residual smoothing.

Figure 4.2: Non-Iterative Mean and Nearest Neighbour MSE Filters

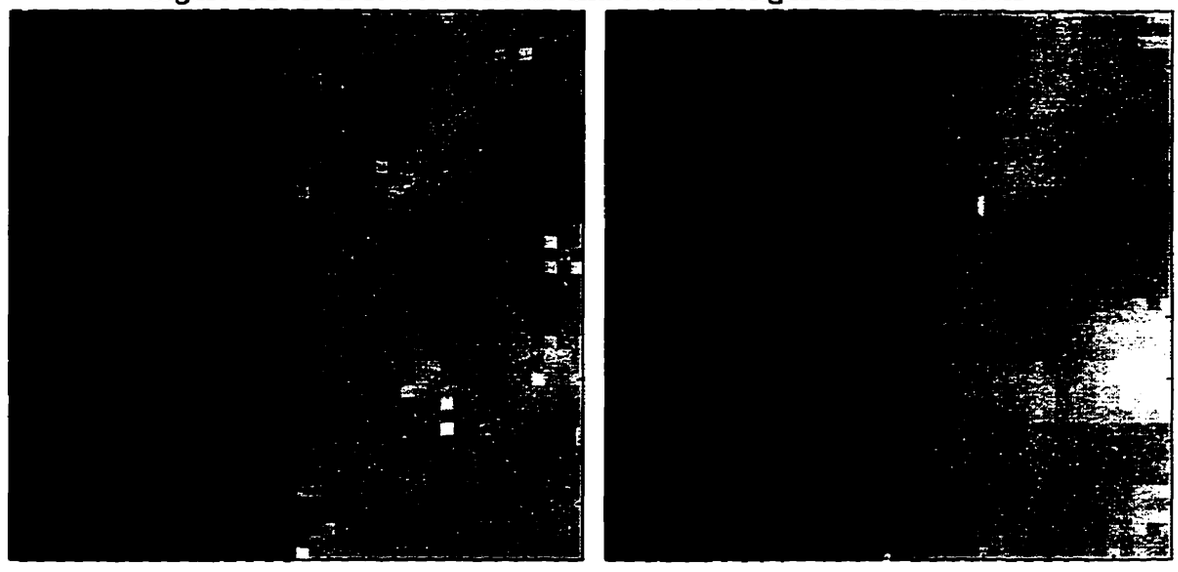


Figure 4.3 shows the 3 point average and hybrid filters. The 3 Point Average filter is very similar to the Nearest Neighbour result in this case. Once again, there is effective smoothing and some edge erosion. The Hybrid filter has better edge preservation since it is not an aggressive a smoother. A side-effect of this is that there is residual patchiness. The Hybrid filter favours preservation of features rather than smoothing.

Figure 4.3: 3 Point Average and Hybrid MSE Filters

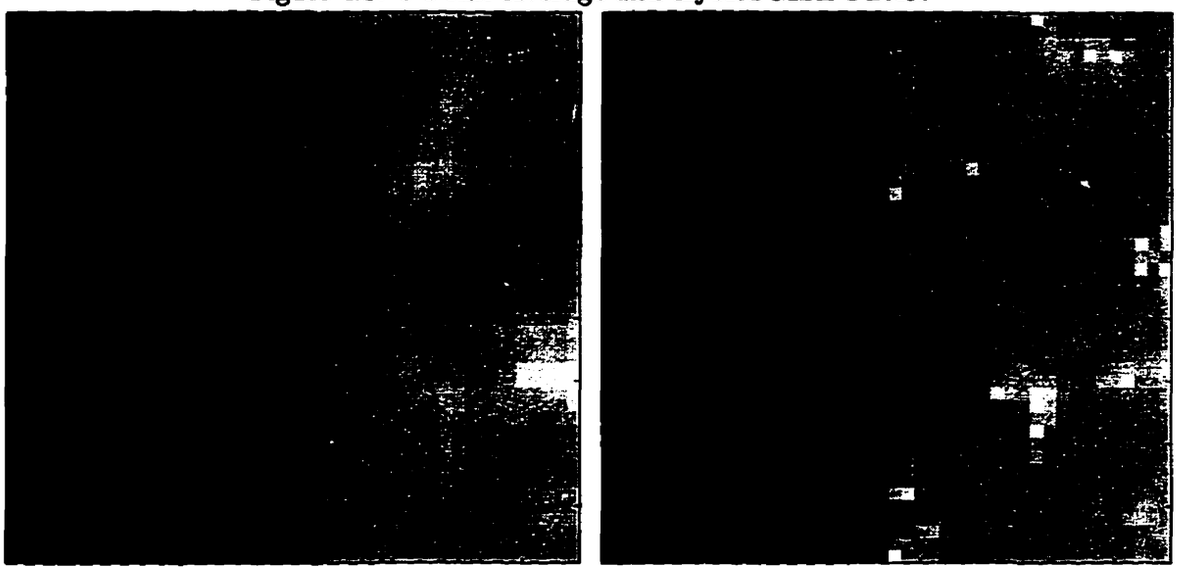


Figure 4.4 shows the Global and Block frequency domain filters. The Global and Block frequency domain filters smooth to a lesser degree then the spatial filters.

Figure 4.4: Global and Block Frequency Domain Filters

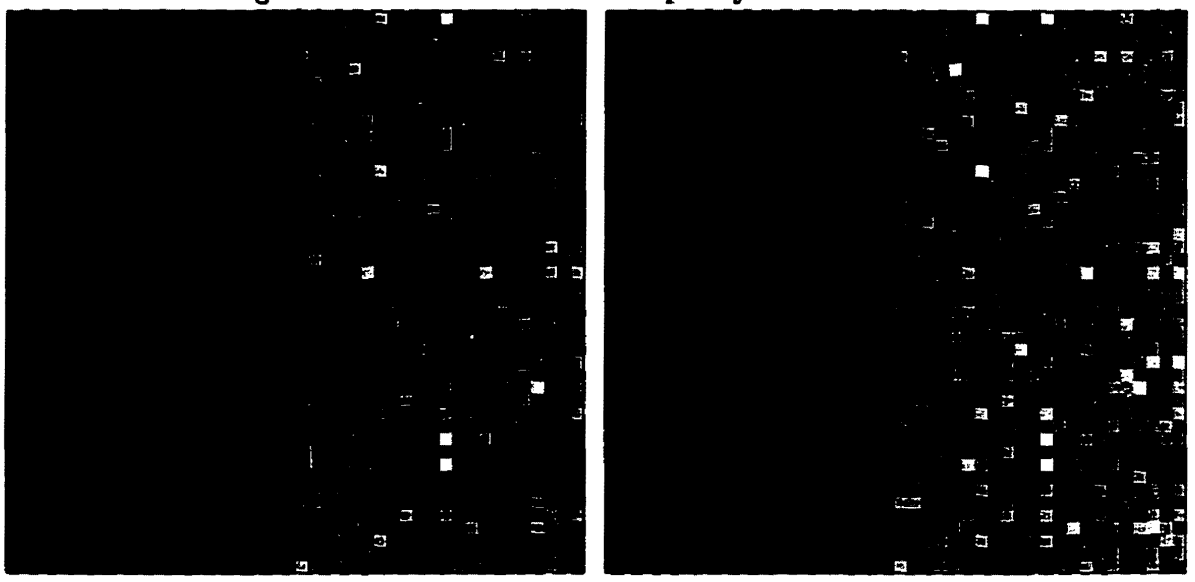
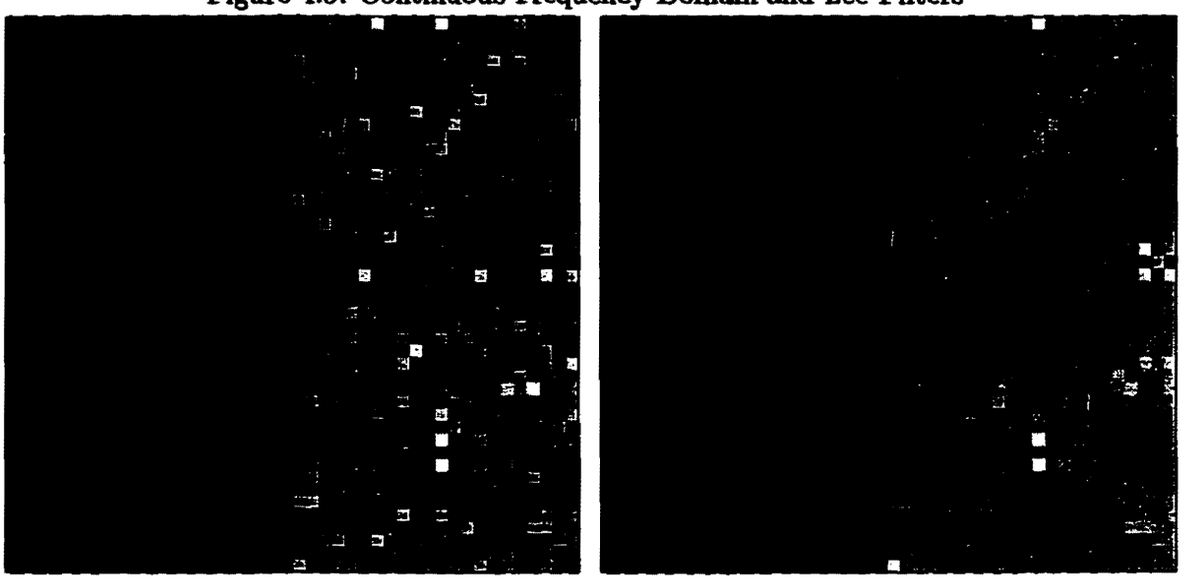


Figure 4.5 shows the continuous Block frequency domain and Lee filters. The Continuous frequency domain filter has a slightly better smoothing effect, but is otherwise very similar to the other frequency domain filters. The Lee filter is similar to the Hybrid filter; it preserves the edge and smoothes elsewhere. The key difference is that the Hybrid filter smoothes more effectively because its recursive structure gives it a large implicit region of support.

Figure 4.5: Continuous Frequency Domain and Lee Filters



4.3 Test Image: Non-Stationary Image

Figure 4.6 shows the original and the original plus noise.

Figure 4.6: Original and Original with Additive Noise

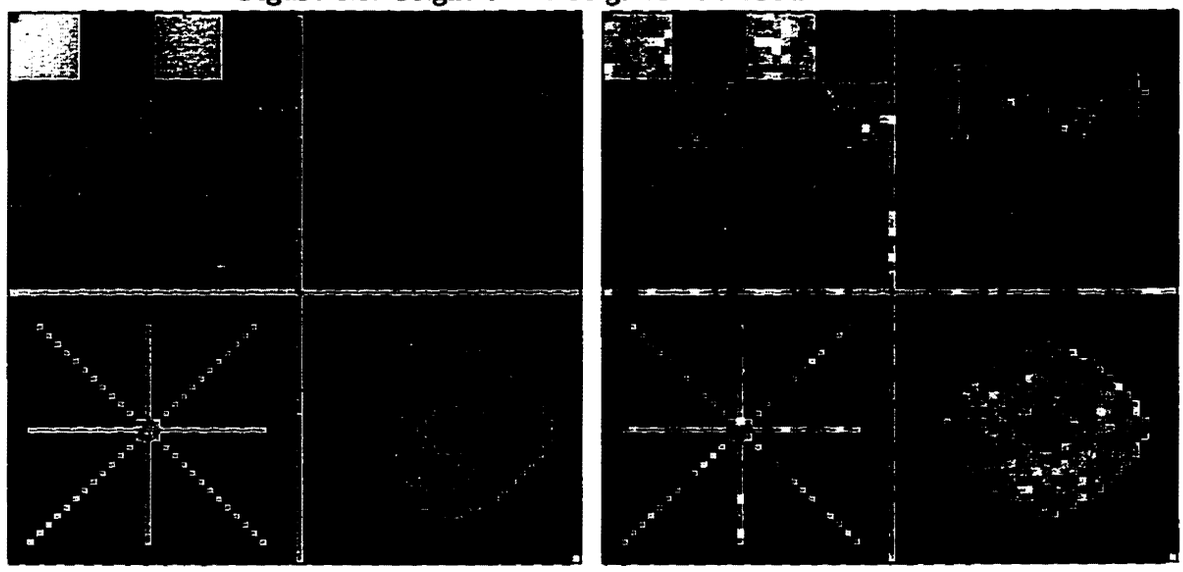


Figure 4.7 shows the Non-Iterative Mean and Nearest Neighbour average filters. The Non-Iterative Mean result shows some smoothing, especially in the dark areas around the bagel shape. This is not surprising since the neighbourhood of support for local statistics is 7 by 7 ans since most smooth areas in this image are smaller than this size, there is little opportunity for smoothing to occur. The Nearest neighbour filter shows the negative aspect of using only the nearest neighbours, the filter rings in the diagonal line areas, and around the lettering. This is a drawback for the nearest neighbour averaging approach.

Figure 4.7: Non-Iterative Mean and Nearest Neighbour MSE Filters

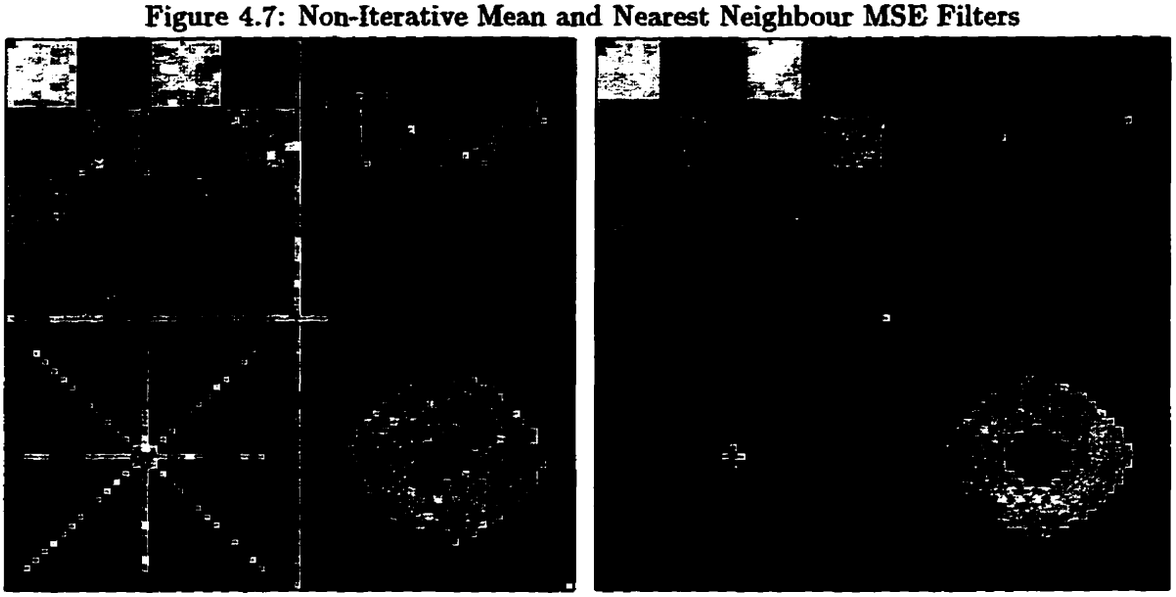


Figure 4.8 shows the 3 point average and hybrid filters. The 3 Point average filter, like the Nearest Neighbour filter, has edge erosion near the lines and lettering as well, but has less ringing. The Hybrid filter is more like the Lee filter in that it smoothes in stationary areas, although the smoothing effect requires a larger region of stationarity than the Lee filter. An example of this is in the checkerboard squares; the Lee filter turns on smoothing (see Figure 4.10), while the Hybrid filter requires a larger neighbourhood switch from all-pass to maximum smoothing.

Figure 4.8: 3 Point Average and Hybrid MSE Filters

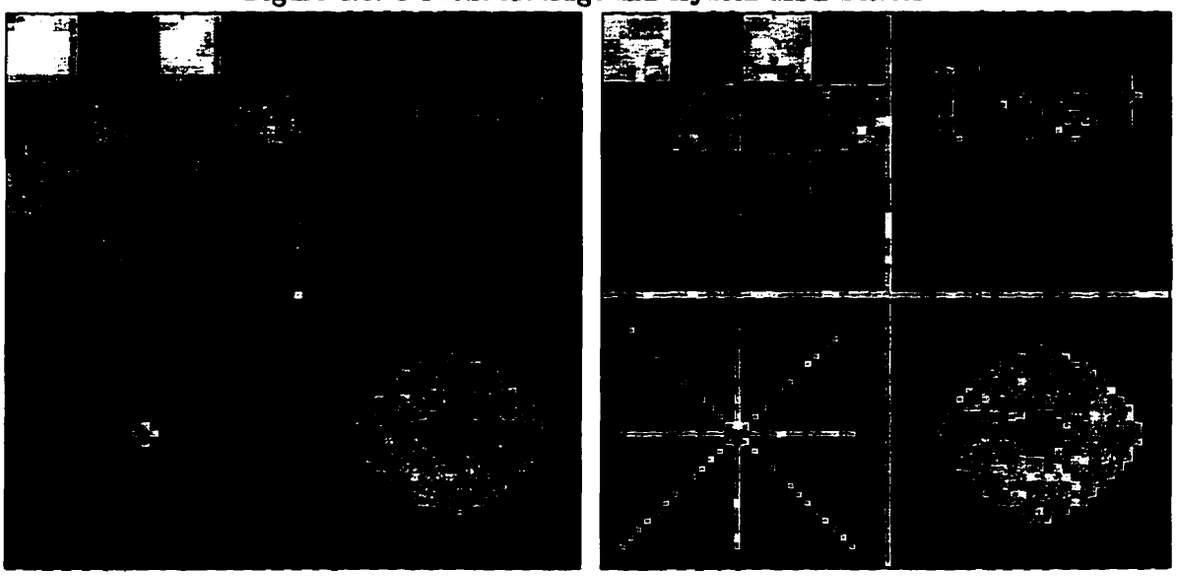


Figure 4.9 shows the Global and Block frequency domain filters. These filters do some overall smoothing and cause less edge erosion than the parametric filters.

Figure 4.9: Global and Block Frequency Domain Filters

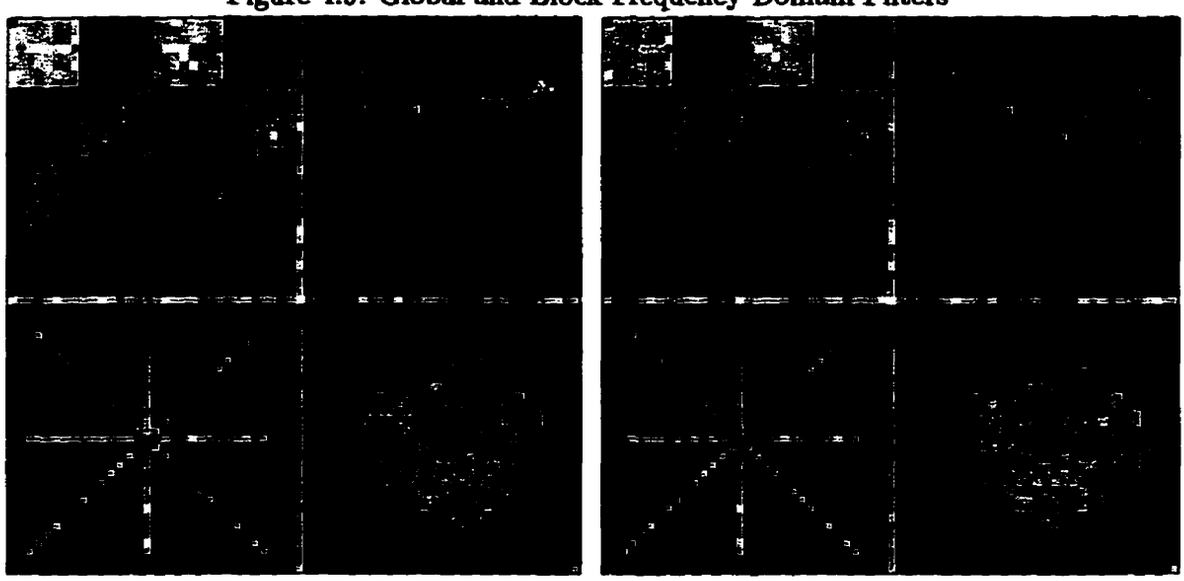
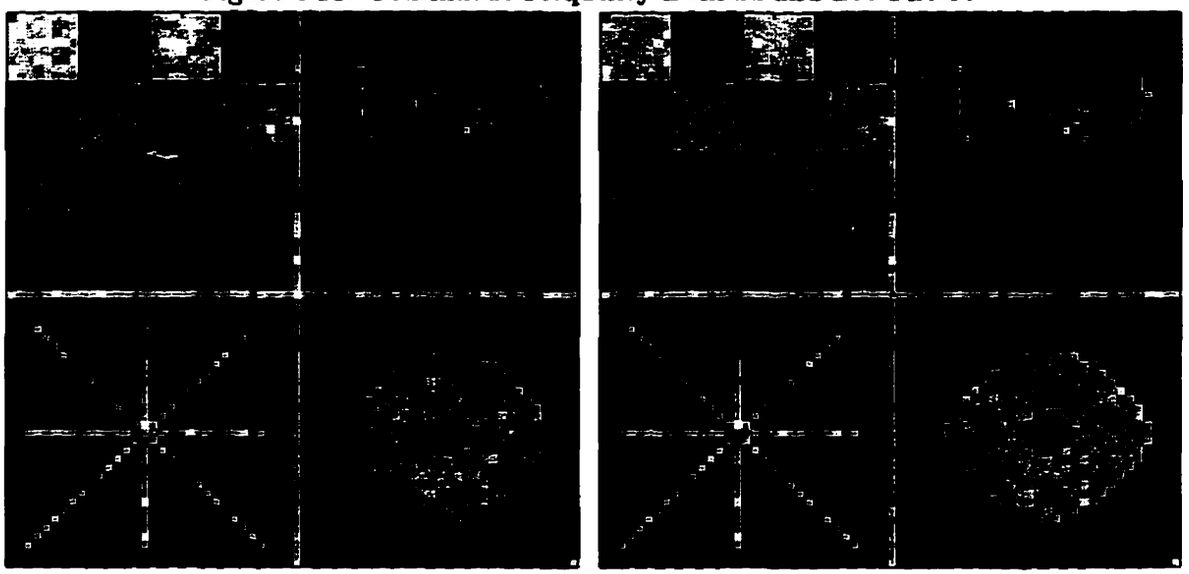


Figure 4.10 shows the Continuous frequency domain and Lee filters. The Continuous filter is similar to the other frequency domain filters, and the Lee filter is able to smooth in stationary areas such as the dark area around the bagel, and on the checkerboard squares.

Figure 4.10: Continuous Frequency Domain and Lee Filters



4.4 Test Image: Face Image

Figure 4.11 shows the original and the original plus noise.

Figure 4.11: Original and Original with Additive Noise

Figure 4.12 shows the Non-Iterative Mean and Nearest Neighbour average filters. The Non-iterative Mean filter is an effective smoother and the edges are preserved as in the area around the shoulder. The Nearest Neighbour filter also preserves edges and gives an overall smoother appearance.

Figure 4.12: Non-Iterative Mean and Nearest Neighbour MSE Filters

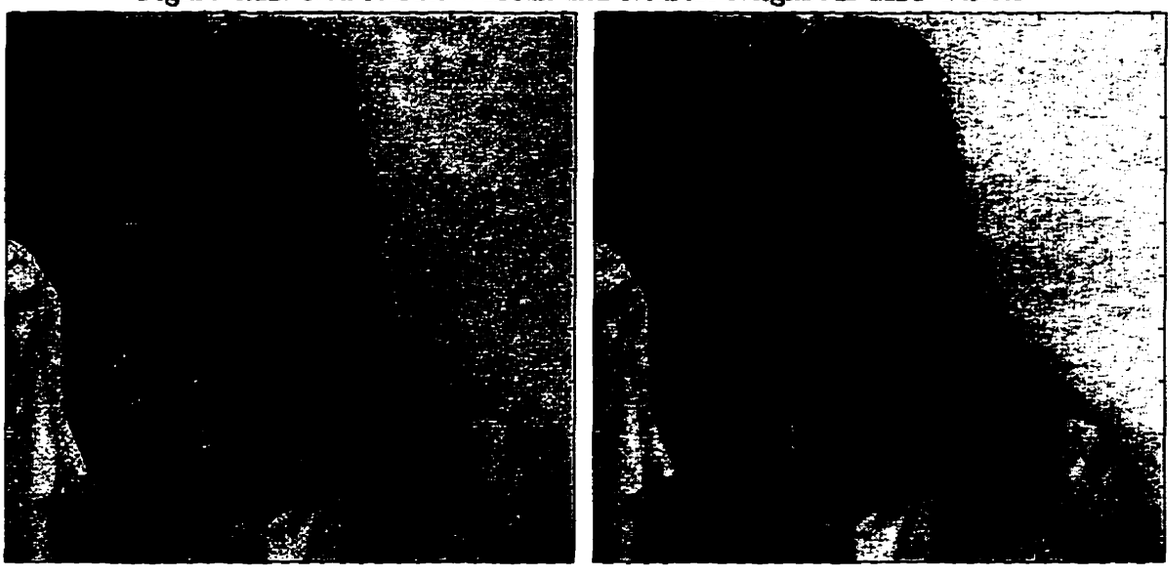


Figure 4.13 shows the 3 point average and hybrid filters. The 3 Point average filter is similar to the Nearest Neighbour filter, while the Hybrid filter has somewhat better edge preservation.

Figure 4.13: 3 Point Average and Hybrid MSE Filters

Figure 4.14 shows the global and block frequency domain filters. The Global and Tiled filters have less smoothing than the parametric filters.

Figure 4.14: Global and Block Frequency Domain Filters

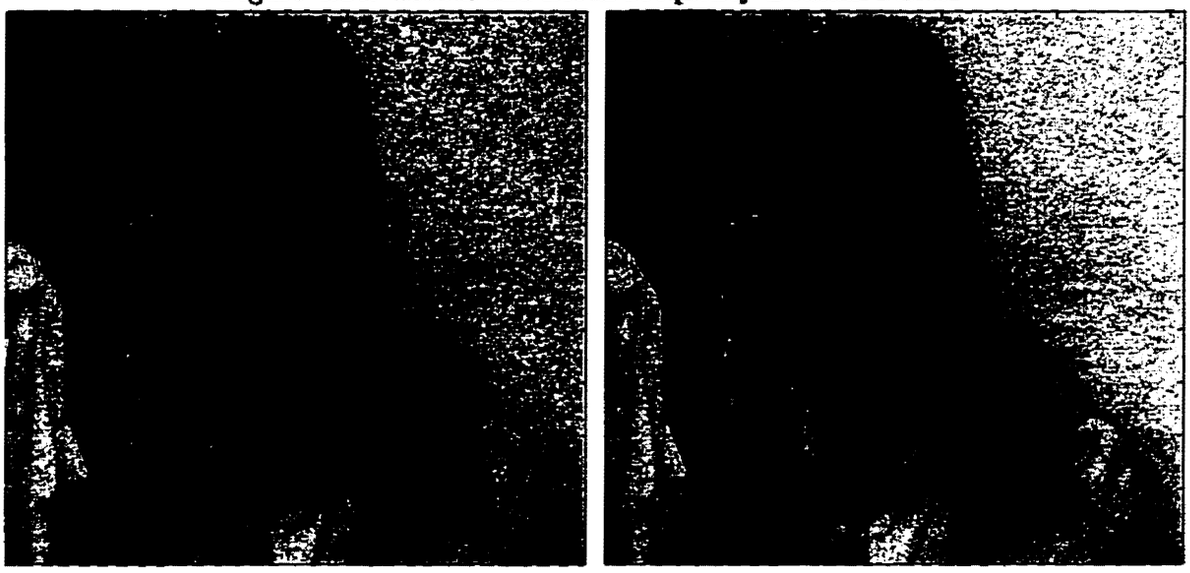
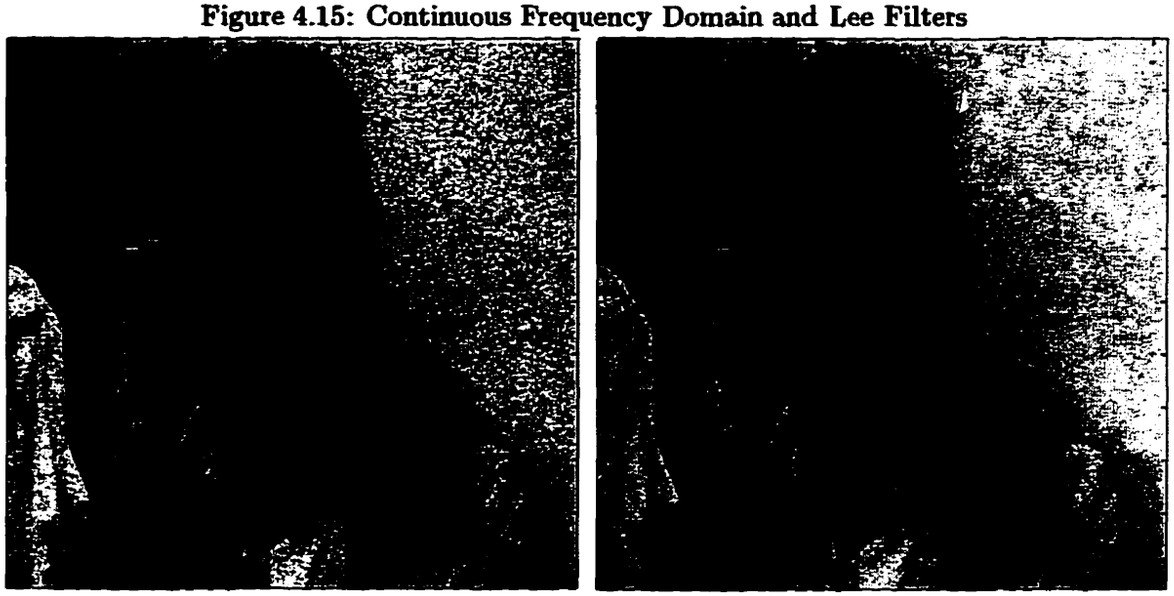


Figure 4.15 shows the continuous block frequency domain and Lee filters. The Continuous filter is similar to the other frequency domain filters, and the Lee is similar to the Hybrid filter.

Figure 4.15: Continuous Frequency Domain and Lee Filters



Chapter 5

Conclusions

The first objective in this thesis is to develop parametric point estimators which use local statistics and a varying neighbourhood size to optimally recover an image corrupted by additive Gaussian noise. The second objective is then to compare these techniques with locally adaptive frequency domain methods.

The structure of parametric filters developed has been inferred from an autocorrelation model in which the signal autcorrelation is assumed to decay exponentially with increasing distance, this has led to the concept of a nearest neighbour autocorrelation estimate. Three filters have been derived using three averaging functions in order to meet the requirement of local unity DC gain. A hybrid filter has also been developed which takes advantage of the autocorrelation model but uses a Lee structure which smoothes or enhances when the local statistics supports it, and is otherwise all-pass.

The second set of filters uses local frequency domain methods which do not require apriori signal model assumptions. Instead, local power spectrum estimates are used to directly get the Wiener filter result.

The Spatial MSE filters were much more effective at smoothing, and gave results better than the Lee filter in many cases, although the choice of the averaging function

had a significant effect on the overall smoothing behavior. In the case of the Non-Iterative Mean, it performs better when the image is highly non-stationary, although the Hybrid filter did even better when the assumption of local stationarity was bad. The Nearest Neighbour and 3 Point Average filters are effective smoothers, but require more iterations. The frequency domain filters did not perform as well because of the problem of accurately estimating the power spectrum in a fixed local neighbourhood. In summary, the 3 Point Average filter was the best filter since it performed better than the other Spatial MSE filters; the iterative mean allowed a greater region of support than the Non-Iterative filter and did not suffer from artifacts as in the Nearest Neighbour filter case. The autocorrelation model approach is therefore a valid one since it achieves an effective compromise between global non-parametric filters such as the Global Wiener filter yet improves upon the simple structure and fixed neighbourhood of support of the Lee filter. The Hybrid filter was also interesting from the point of view that it still manages a reasonable MSE result yet is more tolerant to local non-stationarity.

In terms of performance, the overhead of calculating the autocorrelation is similar to calculating any second order local statistic. Appendix A describes the way the two dimensional nearest neighbour autocorrelation estimate is calculated, as well as describing how it is used to detect edges and smooth areas. Once it has been calculated, the spatial MSE filters are reasonably quick since only a few operations per pixel are required. Convergence was assumed when the difference in the image from one iteration to the next was sufficiently small that no significant activity was occurring in the filter.

For future work, a few possibilities exist. In terms of computational efficiency, there may be ways of reaching convergence faster by having a more efficient algorithm or incorporating iterative techniques such as those suggested in [2] and [20]. Also, the use of image models other than the exponentially decaying exponential could be used. Lastly, instead of using a Wiener filter framework, some other framework, such as one which

enables image enhancement may be worth investigating.

Appendix A

Autocorrelation

A.1 Introduction

This appendix first shows that calculated autocorrelation coefficients are reliable only if the region of support is adequate. It then shows that using only lag 1 autocorrelation coefficients falsely predicts positive autocorrelation across edges. After identifying and explaining this behavior, a notion of autocorrelation is presented that uses a combination of lags that reliably estimates autocorrelation in smooth areas, along edges and across edges.

A.2 Local Statistics

The implementation of the estimators discussed requires knowledge of local image statistics. Filter performance depends on the extent to which local mean, variance and nearest neighbour correlation coefficient can be accurately estimated. The local neighbourhood must include a sufficient number of samples to achieve an accurate local estimate.

At the same time, the signal model of the adaptive Wiener filter assumes that the

signal is stationary over the local window. The extent to which this assumption is valid is dependent on the size of the window. A relatively large local neighbourhood may be stationary in a smooth region of the image, but the larger the neighbourhood, the greater the likelihood that the neighbourhood may include both smooth and edge regions. A small window is more likely to be stationary, but there must be sufficient samples for an accurate estimate.

The reliability of variance estimation can be approached formally by finding a confidence interval [3] over the variance estimate. One technique for finding the confidence interval is to consider an unbiased variance estimate of a local neighbourhood Φ containing n samples, so

$$\sigma^2 = \frac{1}{n-1}\sum_{i\epsilon\Phi}(x_i - \overline{x})^2 \tag{A.1}$$

If the image elements can be modelled by a normally distributed process $y \sim N(\mu, \sigma^2)$, then the variance, or second central moment varies according to a Chi squared distribution, $y^2 \sim X^2$. For an n sample estimate, X^2 will have n-1 degrees of freedom. The 95% confidence interval, that is, the bound within which the sample variance is likely to be inside with a probability of .95 is

$$\frac{(n-1)\sigma^2}{X^2_{0.025,n-1}} \leq \sigma^2 \leq \frac{(n-1)\sigma^2}{X^2_{0.975,n-1}} \tag{A.2}$$

For a 49 sample variance estimate, $X^2_{0.025,n-1} \approx 71$, $X^2_{0.975,n-1} \approx 36$. The size of the 95 % confidence interval is $.67\sigma^2 \leq \sigma^2 \leq 1.5\sigma^2$ which is quite a large bound. The situation only improves to some extent when the number of samples increases to 100 or more. The point here is that for second order statistical estimates, a very large number of samples is required. It is also important that variance and autocorrelation window sizes be identical, otherwise artifacts relating to non- synchronous transition of filter parameters invariably

lead to a poorer overall result.

A.3 Autocorrelation in Images

If the original, uncorrupted signal is not available, the observed signal f must be used or we must estimate a from f so

$$a = \frac{\overline{f(n)f(n+1)} - \overline{f(n)}^2}{\overline{f(n)^2} - \overline{f(n)}^2} \tag{A.3}$$

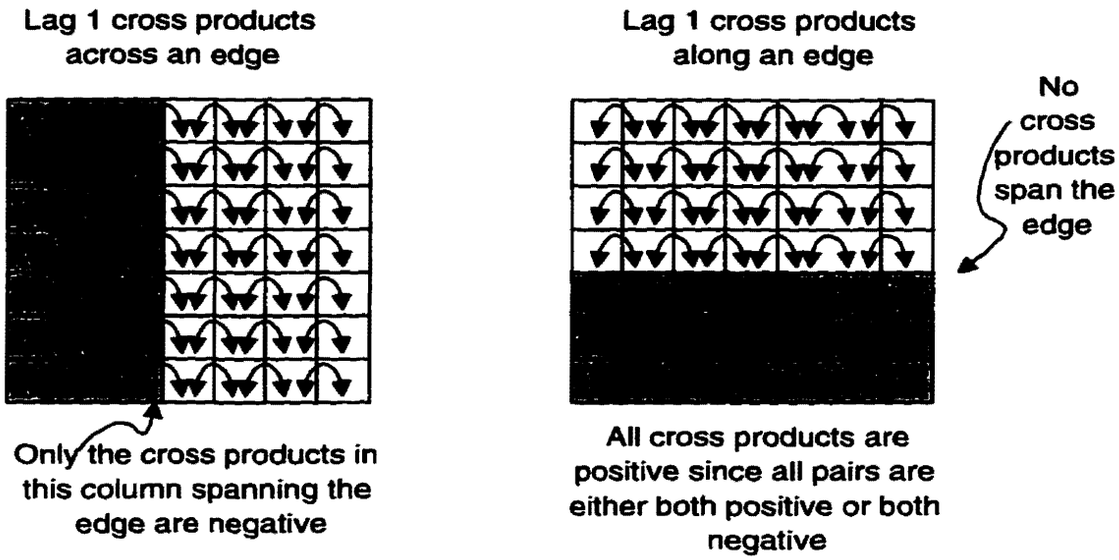
Consider calculating a_i in a 7 by 7 neighbourhood where the $f(n)f(n+1)$ products consist of the 42 cross products

$$\text{cross products of } \mathrm{a_i} = \sum_{\mathrm{i=1}}^{6}\sum_{\mathrm{j=1}}^{7} \mathrm{f(i,j)f(i+1,j)} \tag{A.4}$$

The spanning cross products as in figure A.1 will be negative since they consist of one element greater than the average and one element less than the average. The other cross products are positive since they consist of elements either both greater than or both less than the average. In a 7 by 7 region, this corresponds to 7 spanning cross products and 35 other cross products as in the left side of figure A.1. The right side of figure A.1 shows positive cross products along an edge.

The negative contribution to autocorrelation from the spanning cross products are overwhelmed by the positive contribution of the non spanning cross products. The question now is, what can we do to have negative autocorrelation across edges?

Figure A.1: Autocorrelation Cross Products:Lag 1



A.4 Lag k Autocorrelation

The lag is just the distance between observations, so nearest neighbour autocorrelation is just lag 1 autocorrelation. The cross products of the lag k autocorrelation are

$$\text{lag k cross products of } a_i = \sum_{i=1}^{7-k}\sum_{j=1}^{7} f(i,j)f(i+k,j) \tag{A.5}$$

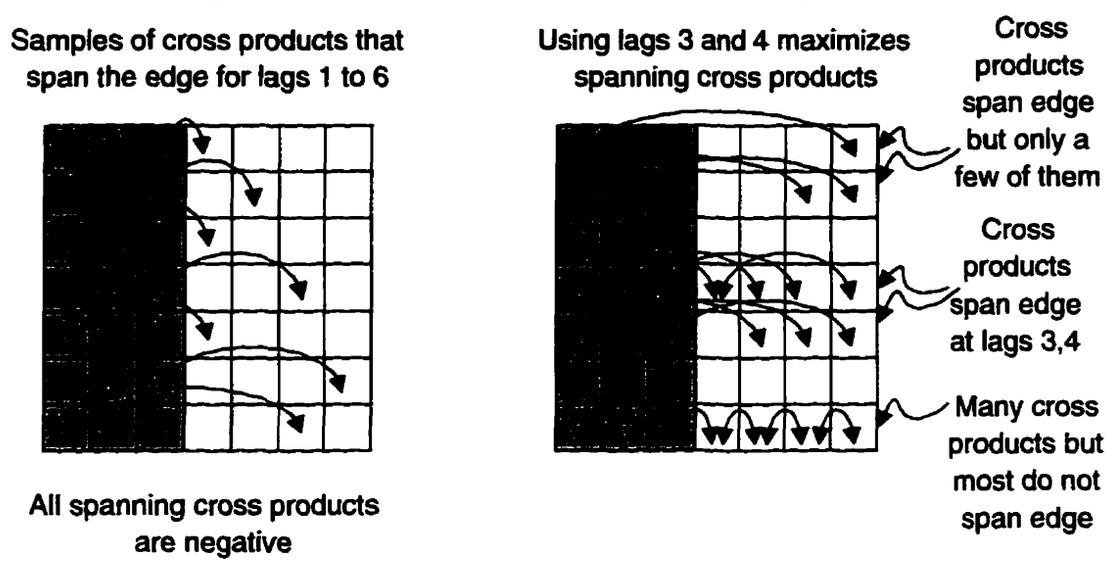
$$\text{lag k cross products of } a_j = \sum_{i=1}^{7}\sum_{j=1}^{7-k} f(i,j)f(i,j+k) \tag{A.6}$$

For a 7 by 7 neighbourhood the possible lag autocorrelations range from $k = 1$ to 6, the number of products is $7 * (7 - k)$. Introducing lag autocorrelations for $k > 1$ may help the problem of positive autocorrelation across edges.

Using lags raises the question of how the region of support is being used. In the case of lag 2, the first and last columns are being used once in the cross products and the

others are used twice. In lag 4, the middle elements contribute to two cross products and the remaining elements are used in one cross product.

Figure A.2: Autocorrelation Cross Products: Lags 1 to 6



The lag pairs of 1,6 and 2,5 and 3,4 all use the region of support evenly and all have 49 cross products, enough for a reasonably robust estimate that also fits the -1 to $+1$ theoretical autocorrelation bound. Figure A.2 gives the results for using the autocorrelation pairs 3,4. Trials so far have shown that the lag 3, 4 autocorrelations seem to be appropriately responsive to high and low contrast edges in both high and low noise conditions.

Using lags other than lag 1 suggests that the model should be adjusted appropriately. For example, the model could be

$$R_s(3) = \sigma_s^2 a^{|3|} + \mu_s^2 \tag{A.7}$$

so that the autocorrelation coefficient would be

$$a = \left(\frac{R_s(3) - \mu_s^2}{\sigma_s^2} \right)^{\frac{1}{3}} \tag{A.8}$$

The effect of this change is that the distribution of a tends to be more extreme, the distribution of values of a favours either a strongly positive or strongly negative response. The effect on performance was that the filters were either strong smoothers or all-pass, with little intermediate behavior, producing an overall poorer result . This can be explained in part due to the relatively rough estimate of a, and subsequent raising of this estimate to the third power. In the filters developed, autocorrelation estimates using lags 3 and 4 were used. This tradeoff enabled the filters to detect edges yet avoid the inaccuracies associated with raising local statistics measures to high order.

Bibliography

[1] H. C. Andrews and B. R. Hunt, Digital Image Restoration, Prentice-Hall, Englewood Cliffs, New Jersey, 1977.

[2] Richard Barrett & al., "Templates for the Solution of Linear Systems: Building Blocks for Iterative Methods", SIAM, Philedelphia, 1994.

[3] G. E. P. Box, W. G. Hunter, and J. S. Hunter, Statistics for Experimenters. Toronto: John Wiley & Sons, Inc., 1978.

[4] K. Erler, "Adaptive Recursive Filtering," M. A. Sc. thesis, University of Waterloo, Waterloo, 1991.

[5] K. Erler and M. E. Jernigan, "Adaptive image restoration using recursive image filters," IEEE Trans. Sig. Proc., vol. 42, no. 7, pp.1877-1881, 1994.

[6] V. S. Frost et al., "A Model for Radar Images and Its Application to Adaptive Digital Filtering of Multiplicative Noise", IEEE PAMI, Vol. PAMI-4, No. 2, March 1982, pp. 157-166.

[7] R. C. Gonzalez and R. E. Woods, Digital Image Processing, Addison-Wesley, New York, 1992.

[8] M. E. Jernigan and K. Erler, "Adaptive processing of images," IEEE Pacific Rim Conf. Comm. Sig. Proc., May 9-10, 1991.

[9] Steven. M. Kay, "Modern Spectral Estimation" Prentice-Hall, Englewood Cliffs, New Jersey, 1988.

[10] Murat Kunt, "Digital Signal Processing" Artech House, Norwood, MA, 1986.

[11] M. J. Lahart, "Local image restoration by a least squares method," J. Opt. Soc. Am., vol. 10, pp. 1333-1339, 1979.

[12] J. S. Lee, "Digital image enhancement and noise filtering by use of local statistics," IEEE PAMI,

vol. 2, no. 2, pp. 165-168, 1980.

[13] B. Mahesh, W. J. Song and W. A. Pearlman, "Adaptive estimators for filtering noisy images," Opt. Eng., vol. 29, pp. 488-494, 1990.

[14] V. J. Mathews, "Adaptive polynomial filters," IEEE Signal Processing Magazine, pp 10-26. July 1991.

[15] Matlab Image Processing Toolbox Manual, Prentice-Hall, 1995.

[16] A. V. Oppenhiem and J. S. Lim, "The importance of Phase in Signals," Proc. IEEE,vol. 69, no. 5, pp. 529-541, 1981.

[17] M. A. K. Paradis, "An examination of multiplicative lateral inhibition filtering for image processing," M. A. Sc. thesis, University of Waterloo, Waterloo, 1994.

[18] W. J. Song and W. A. Pearlman, "Edge-preserving noise filtering based on adaptive windowing," IEEE Trans. Circ. and Syst., vol. 35, pp. 1046-1055, 1988.

[19] C. M. Thompson and L. Shure, Image Processing Toolbox User's Guide (For Use with MATLAB), The MathWorks Inc., Natick, MA, 1993.

[20] C. D. Yoo, M. M. Bace, J. S. Lim. "An iterative method for designing separable Wiener filters," IEEE Trans. Sig. Proc., vol. 43, no. 1, pp.262-270, 1995.