Adaptive Image Restoration:

Perception Based Neural Network

Models and Algorithms

Author: Stuart William Perry (9108283)

Submitted in fulfilment of the requirements for

a Ph.D. in Electrical and Information Engineering

School of Electrical and Information Engineering,

University of Sydney,

NSW, 2006

AUSTRALIA

The University of Sydney

AUSTRALIA

Table of Contents

Content Page

List of Figures

List of Tables

Acknowledgements

Abstract

Chapter 1: Introduction 1

1.1 Overview of This Chapter 1

1.2 Image Degradations 1

1.3 Classical Restoration Techniques 5

1.3.1 Transform Related Techniques 6

1.3.2 Algebraic Restoration Techniques 8

1.4 Emergent Restoration Techniques 11

1.4.1 Neural Network Approach 11

1.4.2 Space Variant Methods 12

1.4.3 Adaptive Methods 13

1.4.4 Blind Image Restoration 14

1.4.5 Colour Image Restoration 16

1.5 Perception Motivated Restoration 16

1.6 Overview of This Thesis 18

1.7 Summary 19

Chapter 2: Contributions of the Thesis 20

2.1 Introduction 20

2.2 Image Restoration using a Neural Network 21

2.2.1 Problem Formulation 22

2.2.2 Neural Network Restoration 24

2.3 Improvement on the Basic Neural Network Algorithm 25

2.4 Space Variant Processing 26

2.5 Adaptive Regularisation Methodologies 27

2.6 New Error Measure 28

2.7 LSMSE-based Cost Functions for Image Restoration 33

2.7.1 LVMSE Modified Cost Function 34

2.7.2 Log LVR Modified Cost Function 35

2.8 Summary 36

Chapter 3: Basic Neural Network Algorithm 37

3.1 Introduction 37

3.2 Neural Network Restoration Algorithms in the Literature 38

3.3 The Proposed Algorithm 43

3.4 Analysis 47

3.5 Dealing with Spatially Variant Distortion 51

3.6 Implementation Considerations 53

3.7 Experimental Results 54

3.7.1 Experimental Setup 54
3.7.2 Efficiency 55
3.7.3 Spatially Variant Distortion 55
3.7.4 An Application Example 57
3.8 Summary 59

Chapter 4: Weight Assignment in Adaptive Image Restoration 61
4.1 Introduction 61
4.2 Adaptive Constraint 62
4.2.1 Motivation 62
4.2.2 The Gradient Based Method 66
4.2.3 Local Statistics Analysis 72
4.3 Dealing with Spatially Variant Distortion 76
4.4 Semi-Blind Deconvolution 78
4.5 Implementation Considerations 81
4.6 Experimental Results 82
4.6.1 Experimental Setup 83
4.6.2 Effects of noise 84
4.6.3 Spatially Variant Distortion 89
4.6.4 Efficiency 92
4.6.5 An Application Example 93
4.7 Summary 96

Chapter 5: Perception Based Cost Functions for Image Restoration 97
5.1 Introduction 97

5.2 Motivation 98
5.3 A LVMSE-Based Cost Function 99
5.3.1 The Extended Algorithm for the LVMSE Modified Cost Function 101
5.3.2 Analysis 109
5.4 A log LVMSE-Based Cost Function 116
5.4.1 The Extended Algorithm for the Log LVR Modified Cost Function 118
5.4.2 Analysis 121
5.5 Implementation Considerations 126
5.6 Experimental Results 127
5.6.1 Colour Image Restoration 127
5.6.2 Grayscale Image Restoration 131
5.6.3 LSMSE of Different Algorithms 133
5.6.4 Robustness Evaluation 135
5.7 Summary 138

Chapter 6: Conclusion 139
6.1 Introduction 139
6.2 Overview of Thesis Objectives 139
6.3 Background 140
6.4 Perception Motivated Error Measures 141
6.5 Advances on the Basic Neural Network Algorithm 142
6.6 Perception-Based Algorithm Modifications 143
6.7 Future Work 144
6.8 Summary 145

Appendix A: Calculation of Weights and Bias Inputs 146

Appendix B: Analysis of the Adaptive Constraint Approximation 149

Appendix C: Analysis of the Factors in the Gradient Descent Method 152

References 155

Publication List 169

Papers Published 169

Papers in Review 171

Glossary 172

List of Figures

Figure	Page
Figure 2.1: Effects of noise on regions of varying texture.	33
Figure 3.1: Original, degraded and restored images suffering spatially variant distortion.	57
Figure 3.2: Degraded and restored wing images.	59
Figure 4.1: Graph of maximum energy reduction versus constraint value.	69
Figure 4.2: The λ values selected for each pixel during the initial iteration of the energy minimisation based algorithm for a typical image.	71
Figure 4.3: Images with varying levels of additive noise.	75
Figure 4.4: Degraded image restored with various PSF estimates.	79
Figure 4.5: Original images.	83
Figure 4.6: Constraint values chosen versus variance.	85
Figure 4.7: Degraded and restored images for noise of variance 4.22.	86
Figure 4.8: Degraded and restored images for noise of variance 18.52.	87
Figure 4.9: Degraded and restored images for noise of variance 18.30.	88
Figure 4.10: Results of experiments on an image degraded by spatially variant distortion.	91

Figure 4.11: Degraded and restored chromosome images. 95

Figure 5.1: Flow-chart representing Algorithm 5.1. 108

Figure 5.2: Graph of function for Case 1. 112

Figure 5.3: Graph of function for Case 2. 114

Figure 5.4: Graph of function for Case 3. 116

Figure 5.5: Colour images restored using various algorithms. 130-131

Figure 5.6: Grayscale images restored using various algorithms. 132-133

Figure 5.7: Graph of LSMSE for various algorithms and levels of noise. 134

Figure 5.8: Images restored using correct and faulty networks. 137

List of Tables

Table	Page
Table 2.1: Comparison of the SNR, LSMSE and the LVMSE for various image regions.	32
Table 3.1: Degrading PSFs.	55
Table 4.1: Statistics of degraded and restored images for various levels of noise.	89
Table 4.2: Degrading PSFs.	90
Table 4.3: Results of spatially variant experiment.	92
Table 4.4: Algorithm run times.	92

Acknowledgements

The author would like to acknowledge the help obtained from the following persons during the writing of this thesis:

In particular, the author would like to thank Dr. Ling Guan of the School of Electrical and Information Engineering, University of Sydney, who was always available to help, advise and direct the author during his Ph.D. studies and the writing of this thesis. Without Dr. Guan's help, this thesis would not be possible.

The author would like to thank the staff at VISLAB, University of Sydney, for help creating the colour images in this thesis, and his wife, Ada Chan, for her help in proof-reading this thesis. Any errors that remain are the responsibility of the author alone.

In addition, the author would like to thank his colleagues, Hausan Wong, Songyang Yu, and Matthew Kyan for their help during the course of this thesis, and Miroslav Trajkovic for advice regarding matrix algebra which proved very helpful when writing this thesis.

Abstract

This thesis describes research into the field of image restoration. Restoration is a process by which an image suffering some form of distortion or degradation can be recovered to its original form. Two primary concepts within this field have been investigated.

The first concept is the use of a Hopfield neural network to implement the constrained least square error method of image restoration. In this thesis, the author reviews previous neural network restoration algorithms in the literature and builds on these algorithms to develop a new faster version of the Hopfield neural network algorithm for image restoration. The versatility of the neural network approach is then extended by the author to deal with the cases of spatially variant distortion and adaptive regularisation. It is found that using the Hopfield-based neural network approach, an image suffering spatially variant degradation can be accurately restored without a substantial penalty in restoration time. In addition, the adaptive regularisation restoration technique presented in this thesis is shown to produce superior results when compared to non-adaptive techniques and is particularly effective when applied to the difficult, yet important, problem of semi-blind deconvolution.

The second concept investigated in this thesis, is the difficult problem of incorporating concepts involved in human visual perception into image restoration techniques. In this thesis, the author develops a novel image error measure which compares two images based on the differences between local regional statistics rather than pixel level differences. This measure more closely corresponds to the way humans perceive the differences between two images. Two restoration algorithms are developed by the author based on versions of the novel image error measure. It is shown that the algorithms which utilise this error measure have improved per-

formance and produce visually more pleasing images in the cases of colour and grayscale images under high noise conditions. Most importantly, the perception based algorithms are shown to be extremely tolerant of faults in the restoration algorithm and hence are very robust.

A number of experiments have been performed to demonstrate the performance of the various algorithms presented.

Chapter 1: Introduction

1.1 Overview of This Chapter

This chapter provides a background for the reader regarding the subject of image restoration. Understanding the fundamental theory, methods and concepts, and the current state of the art in the field of image restoration is vitally important to the understanding of the material presented in the subsequent chapters of this thesis.

This chapter begins with a short discussion on image degradations and their mathematical models in Section 1.2. Section 1.3 details the basic theory of image restoration and the classical techniques used to restore images, while Section 1.4 details new emerging techniques for adaptively restoring images and handling colour images and images degraded by spatially variant distortion. Section 1.5 considers the way humans perceive visual information and how this information may be incorporated into restoration techniques. Section 1.6 provides an overview of the subsequent chapters of this thesis and Section 1.7 summarises this chapter.

1.2 Image Degradations

All lifeforms require methods for sensing the environment. Being able to sense one's surroundings is of such vital importance in the natural world that ever since life has appeared upon the planet, there has been a constant race to develop more sensory methods and increased sensory accuracy to out-compete other creatures. As a consequence of this "sensory arms race", advanced life forms have at their disposal an array of highly accurate senses. Some unu-

sual sensory abilities are present in the natural world, such as the ability to detect magnetic fields and electric fields, or the use of sound waves to determine the internal structure of objects. Despite this, one of the most prized and universal senses utilised in the natural world is vision.

Advanced animals living above ground rely heavily on vision. Birds and lizards maximise their fields of view with eyes on each side of their skulls, while other animals direct their eyes forward to observe the world in three dimensions. Nocturnal animals often have large eyes to maximise light intake, while predators such as eagles have very high resolution eyesight to identify prey while flying. The natural world is full of animals of almost every colour imaginable. Some animals blend in with surroundings to escape visual detection, while others are brightly coloured to attract mates or warn aggressors. Everywhere in the natural world, animals and plants make use of vision for their daily survival. The reason for the heavy reliance on eyesight in the animal world is simple, vision is by far the best tactical sense available. To survive in the wild, animals must be able to move rapidly. Hearing and smell provide warning regarding the presence of other animals, yet only a small number of animals such as bats have developed these senses sufficiently to be used to run away from predators, or chase down prey. For the majority of animals, only vision enables efficient rapid movements.

Humans rely on vision to a much greater extent than most other animals. Unlike the majority of creatures we see in three dimensions with high resolution and colour. In humans the senses of smell and hearing have taken second place to vision. Humans have more facial muscles then any other animal, because in our society facial expression is used by each of us as the primary indicator of the emotional states of other humans, rather than the scent signals used by many mammals. For this reason it is not surprising that we naturally desire to give our

machines and computers the ability to see. Since the human world revolves around visual stimuli, machines will never be able to fully interact with humans without being able to process visual information.

However the task is not simple, the human visual system is the product of billions of years of evolution and is naturally highly complex. To give machines some of the abilities that we take for granted is the subject of intensive on-going research.

This thesis is concerned with one of the primary visual information processing problems, handling degraded images. Images are often recorded under a wide variety of circumstances. As imaging technology is rapidly advancing, our interest in recording unusual or irreproducible phenomena is increasing as well. We often push imaging technology to its very limits. For this reason we will always have to handle images suffering from some form of degradation.

Since our imaging technology is not perfect, every recorded image is a degraded image in some sense. Every imaging system has a limit to its available resolution and the speed at which images can be recorded. Often the problems of finite resolution and speed are not crucial to the applications of the images produced, but there are always cases where this is not so. There exists a large number of possible degradations that an image can suffer. Common degradations are blurring, motion and noise. Blurring can be caused when an object in the image is outside the cameras depth of field sometime during the exposure. For example, a foreground tree might be blurred when we have set up a camera with a telephoto lens to take a photograph of a mountain. A blurred object loses some small scale detail and the blurring process can be modelled as if high frequency components have been attenuated in some manner in the image [1,2]. If an imaging system internally attenuates the high frequency components in the image, the result

will again appear blurry, despite the fact that all objects in the image were in the camera's field of view. Another commonly encountered image degradation is motion blur. Motion blur can be caused when a object moves relative to the camera during an exposure. Such as a car driving along a highway in an image. In the resultant image, the object appears to be smeared in one direction. Motion blur can also result when the camera moves during the exposure. Noise is generally a distortion due to the imaging system rather than the scene recorded. Noise results in random variations to pixel values in the image. This could be caused by the imaging system itself, or the recording or transmission medium. Sometimes the definitions are not clear such as in the case where an image is distorted by atmospheric turbulence, such as "heat haze". In this case, the image appears blurry due to the fact that the atmospheric distortion has caused sections of the object to be imaged to move about randomly. This distortion could be described as "random motion blur", but can often be modelled as a standard blurring process. Some types of image distortions, such as certain types of atmospheric degradations, [3-7], can be best described as distortions in the phase of the signal. Whatever the degrading process, image distortions can fall into two categories [1, 2].

- Some distortions may be described as spatially invariant or space invariant. In a space invariant distortion all pixels have suffered the same form of distortion. This is generally caused by problems with the imaging system such as distortions in optical system, global lack of focus, or camera motion.

- General distortions are what is called spatially variant or space variant. In a space variant distortion, the degradation suffered by a pixel in the image depends upon its location in the image. This can be caused by internal factors, such as distortions in the optical system, or by external factors, such as object motion.

In addition, image degradations can be described as linear or non-linear [1]. In this thesis, we consider only those distortions which may be described by a linear model. For these distortions, a suitable mathematical model is given in Chapter 2.

1.3 Classical Restoration Techniques

The previous section discussed the various types of distortion that an image may suffer. In some cases, the image degradations encountered in everyday photography are either considered not severe enough to merit correction, or the scene imaged is not considered important enough to have the distortions corrected. However there are many cases where this is not so. Photographs of scientific interest taken of rare phenomena can suffer from many distortions. It may be too expensive or impossible to duplicate the phenomena. Consider a space probe sent to take photographs of a distant planet. If the images have become blurred or degraded by noise during the transmission back to Earth, it may not be possible to instruct the spacecraft to take a second image of the planet. If a security camera photographs a crime in progress, a blurry picture may be the best hope for identifying the offender. In addition, some imaging technologies may have inherent problems. There will always be cases when the image is considered important enough to attempt to correct the distortion.

The act of attempting to obtain the original image given the degraded image and some knowledge of the degrading factors is known as “image restoration”. The problem of restoring an original image, when given the degraded image, with or without knowledge of the degrading point spread function (PSF) or degree and type of noise present is an ill-posed problem [1,8 - 10] and can be approached a number of ways such as those given in [1, 11 - 13]. For all useful

cases a set of simultaneous equations is produced which is too large to be solved analytically. Common approaches to this problem can be divided into two categories, inverse filtering or transform related techniques, and algebraic techniques. An excellent review of image restoration techniques is given by [1]. The following references also contain surveys of restoration techniques, Katsaggelos [14], Sondhi [15], Andrews [16], Hunt [17], and Frieden [18].

1.3.1 Transform Related Restoration Techniques

Transform related restoration techniques involve analysing the degraded image after an appropriate transform has been applied. By acting directly on the transformed image before applying an inverse transform, or using the transformed image information to develop an inverse filter, an image may be partially restored. A number of transform related techniques exist and all of the following techniques are described in [1].

A. Inverse Filter

In the fourier domain the transfer function of this filter is the inverse of the transfer function of the distortion applied to the image. This produces a perfect restoration in the absence of noise, but the presence of noise has very bad effects. Some ad-hoc solutions modify the filter transfer function so that it approaches zero in regions where the noise power is greater than the signal power.

B. Wiener Filter

This filter is better than the inverse filter in the presence of noise because it uses *a priori* statistical knowledge of the noise field. The transfer function is chosen to minimise the mean square restoration error using statistical information on both the image and noise fields. Ozkan, et al. recently examined a method of accounting for spatial and temporal correlations when using multiframe Wiener filters to restore image sequences [19].

C. Parametric Estimation Filters

These filters are variations on the Wiener filter and are described by E.R. Cole [20]. Some examples are:

- POWER SPECTRUM FILTER: This filter matches the power spectrum of the reconstructed image to the original image. However, the power spectrum, unlike the fourier spectrum, is not unique to an image. Hence this filter may result in a large mean square error, unlike the traditional Wiener filter. A small variation in the transfer function of the power spectrum filter produces a very similar filter called the Geometrical mean filter.

- CONSTRAINED LEAST-SQUARES FILTER: This Wiener filter variation adds an extra term to the Wiener filter transfer function in the form of a design spectral variable. This variable may be used to minimise higher order derivatives of the estimate [21].

Some work has been done recently in the field of adaptive image restoration using recursive image filters such as that presented by Erler and Jernigan [22]. Mallikarjuna and Chaparro

investigated using a composite model of an image, where the gross information had been separated from the textural information [23]. After this had been achieved, the nonstationarities in the mean and autocorrelation could be used to help restore the image.

D. Kalman Filter

The Kalman filter is a recursive filter for image restoration which has been examined a great deal recently. Wu and Kundu examined a faster simpler version of this filter and modified the filter to take account of non-gaussian noise [24]. Citrin and Azimi-Sadjadi also investigated image restoration with the Kalman filter [25].

E. Homomorphic Filter Restoration

Another class of filters which work on the principle of transforming the degraded image into another representation space. In theory, the new representation space may be such that the restoration operations are more easily performed. The concept of “power spectrum equalisation” is used as a restoration criteria.

1.3.2 Algebraic Restoration Techniques

Algebraic techniques involve attempting to find a direct solution to the distortion by matrix inversion techniques, or techniques involving an iterative method to minimise a degradation measure.

A. Pseudoinverse Spatial Image Restoration

This set of image restoration techniques attempt to restore an image by considering the vector space model of the image degradation and attempting to restore the image in this vector-space domain. This involves finding an approximation to the inverse of the matrix blurring operator which is multiplied with the column scanned image vector to produce the degraded observed image. These blur matrices are invariably very large and it is not computationally feasible to invert them. However it is often possible to find the inverses of the row and column operators separately if the blur matrix has a special form and apply them sequentially to the rows and columns of the image. This technique is known as pseudoinverse image restoration. This technique does not take into account the effects of noise in the calculations of the pseudoinverse and so is sensitive to noise in the image.

B. SVD Pseudoinverse Spatial Image Restoration

Using the technique of singular value decomposition (SVD), any matrix can be decomposed into a series of eigenmatrices. By applying this technique to decompose the blur matrix, the resultant eigenmatrices can be used to develop an estimation technique where successive estimations of the reconstructed image are based on the previous estimate. This method is effective with problems of ill-conditioning, since it is possible to interactively terminate the expansion before numerical problems result. However the presence of noise in the observed image can often lead to numerical instability.

C. Wiener Estimation

In this method the noise field is modelled again as a two-dimensional random process with a known mean and co-variance function. In addition, the ideal image is assumed to also be a sample of a two-dimensional random process with known first and second moments. The restoration matrix and a bias vector are found to minimise the mean square error formula.

D. Constrained Image Restoration

Previous techniques considered images only as arrays of numbers, however a restored image is spatially smooth and strictly positive in amplitude. Often constrained restoration techniques are based on Wiener estimation and regressional techniques. Reeves and Mesereau have developed a method of assessing the validity of sets of constraints using cross-validation [26]. There are a great number of possible constraints which may be applied by the following methods:

- specification of individual pixel values.
- ratios of values of some pixels.
- amplitude limits.

Often *a priori* information is available for use as constraints, e.g. amplitude limits to fit the image into the dynamic range of the display. It is usually better to incorporate the constraints in the restoration process rather than to apply them to the final results. Many different constraint parameters exist:

- Positivity: By taking the log of an image before restoration, the resultant estimate is guaranteed to be strictly positive. Recently Snyder, et al. examined the restoration of images using a positivity constraint, and proposed a method of regularising the solution [27].

- Probability Density approaches: The techniques of maximum entropy, maximum likelihood, and maximum a posteriori are constraints developed to estimate a probability density from observation of its moments. J.P. Burg's technique yielded a closed form solution to this problem, however it is unstable in the presence of noise [28]. J.A. Edward and M.M. Fitelson developed a technique based on the maximum entropy principle with a positivity constraint [29]. P.A. Jansson, et al. and T.S. Huang, et al. have developed methods based on amplitude constraints [30, 31]. Gokmen and Li tackled the problem of edge detection and surface reconstruction using a smoothness constraint that varied spatially [32]. Zervakis and Venetsanopoulos have recently examined the problem of restoring images suffering non-linear distortion using a least squares restoration measurement [33].

1.4 Emergent Restoration Techniques

A number of new restoration methodologies have developed in recent years as has interest in new aspects of image restoration problems. Researchers are now becoming very interested in the subjects of spatially variant restoration, adaptive restoration, blind restoration, and colour restoration.

1.4.1 Neural Network Approach

In this approach the image is restored by the minimisation of a cost function. However, this

cost function optimisation is performed by using a neural network approach. In this way neural networks can be used to implement a range of filters to an image. The iterative solution of the minimisation approach to image restoration is very amendable to neural network implementation. The main reason is that the learning concept associated with neural networks brings truly adaptive processing to restoration so that such iterative techniques are not just efficient, but produce high quality results. Additional benefits offered by neural networks are their extremely parallel nature [34]. Zhou, et al. proposed a Hopfield neural network approach to constrained mean square error restoration [35]. The network was further optimised by Paik and Katsaggelos [36]. Iiguni, Sakai and Tokumaru developed several new learning algorithms for a multilayered neural network based on the Extended Kalman filter [37]. Steriti and Fiddy compared the Hopfield network energy minimisation approach to the singular value decomposition method of finding a matrix inverse and found that the Hopfield method was superior [38]. The neural network approach to image restoration will be examined more closely in Chapter 3.

1.4.2 Space Variant Methods

Most of the algorithms in Section 1.3 do not adapt easily to the case of images degraded by spatially variant distortion. Since most degraded images in the real world suffer some form of spatially variant degradation, recent years have seen increasing interest in this field. A number of interesting algorithms have been developed.

Perry and Guan showed that the neural network-based algorithms could be extended to the spatially variant case, [39], the development of which is detailed in Chapters 3 and 4. Koch, et al. used a multiple model Extended Kalman filters (EKF) procedure to restore images with variations in the blur parameters [40]. Ozkan, et al. developed a method to handle spatially variant

blurs using the method of projection onto convex sets (POCS) [41]. Recently Sung and Choi presented a method for restoring images suffering spatially variant blur using a self-organising neural network (SONN) to learn the characteristics of the spatial variance [42]. Patti, et al. proposed a new approach for restoring spatially variant restoration of progressive and interlaced video [43]. Their method is based on Kalman filtering.

1.4.3 Adaptive Methods

Many image restoration methods assume that the image can be modelled by a stationary process. This is however not a valid assumption. The local regional statistics of different regions in the image are in general not the same. For this reason if a global regularisation is chosen for all pixels in the image, edges will invariably be smoothed in order to increase noise suppression. The restoration quality is sensitive to the value of the regularisation term. If the regularisation term is weighted too weak in the error measure, the resultant restored image will contain noise artifacts. On the other hand, if the regularisation term is weighted too strongly in the error measure, the resultant restored image will be blurred. Various methods have been employed to chose the optimal value of regularisation parameter for image restoration [44-46].

In [47], the authors used fuzzy logic control to adaptively vary the regularisation parameter to achieve the optimal balance between removing edge ringing effects and suppressing noise amplification. Their method was based on gray level range within the neighbourhood of each pixel and was particularly tailored for gamma camera images where Poisson noise dominates. In [48], the authors used a model-based neural network approach to adaptively vary the restoration parameters. In [49], a regularisation functional was developed to vary the regularisation parameter from iteration to iteration in a restoration algorithm. Reeves used a generalised

cross-validation (GCV) criterion restore images using a spatially variant regularisation term [50]. The use of the mulitresolution nature of the wavelet transform as a basis for spatially adaptive regularisation for restoration was also proposed [51, 52].

1.4.4 Blind Image Restoration

In this field of study, the blur degradation is not known in advance. Some surveys of blind deconvolution methods are given by [53] and [54]. There are a number of approaches to this problem.

One approach is to examine the image to determine the blur impulse response and noise power. This is usually done by isolating the image of a suspected known object in the image and observing how the object was degraded. A point source or edge in the image is often the most useful. Noise variance can be estimated by isolating a region of relatively constant background. Savakis and Trussell have investigated identifying the blurring function by computing the best match between the restoration residual power spectrum and its expected value [55]. Reeves and Mersereau used generalised cross-validation in their investigation in order to identify a blurring function [56]. Galatsanos and Katsaggelos studied the problem of choosing the regularisation parameter and estimating the noise variance in image restoration [57]. Pavlovic and Tekalp recently developed a method of identifying blur based on parametric modelling of the blur in the continuous spatial domain [58]. Trussell and Fogel proposed a method to estimate and restore spatially variant motion blurs in image sequences by examining the relative motions of different parts of the image [59]. Anarim, et al. presented a technique for the identification of blur parameters using expectation maximisation (EM) [60].

Another approach is to indirectly determine the degrading function using multiple images or similar images:

- Temporal redundancy: in some cases such as television, we might be presented with many frames containing the same image, in this case noise can be removed by averaging the images. If each frame contains the same subject but different blur point spread functions, the image can be restored by conversion to the fourier domain, taking the log of the image, and temporal averaging [1].

- Single frame images: these images can be restored by examining ideal images with similar content to the suspected contents of the degraded image. Stockham, et al. developed a method whereby some of the parameters of the restoration are discovered by sectioning the image, taking it into the Fourier domain, take the image log and averaging the results [61]. Some research has been done using statistical information about the change in blur to develop better Wiener filtering techniques to handle this problem such as those shown in Slepian [62], Guan and Ward [63], and Ward [64].

Sivakuma and Desai examined the problem of unknown blur functions and proposed a multilevel perceptron approach [65]. Mesarovic, et al. proposed a two step algorithm to simultaneously restore the image and estimate the parameters of a restoration filter by using a hierarchical Bayesian approach [66]. Pai, Havlicek and Bovik showed that given three blurred versions of the same image and some restrictions on the size of the original image, sufficient conditions exist for exact multichannel blind image restoration [67]. May, et al. tackled the problem of blind image restoration by incorporating local image characteristics as the basis for bound constraints on individual pixels [68]. You and Kaveh considered a regularisation

approach to joint blur identification and restoration [69]. Ayers and Dainty and Davey, et al. have developed an iterative technique based on phase retrieval algorithms using simple inverse filtering [70-72]. Fish, et al. adapted the Richardson-Lucy algorithm for deconvolution while Holmes developed an algorithm based on the expectation maximisation algorithm [73-77]. Thiebaut and Conan used the error function minimisation method of Lane with the implementation of strict constraints using a reparameterisation of the problem [78,79].

1.4.5 Colour Image Restoration

The restoration of colour images is a difficult task but a logical next step for research in the field of image restoration. When restoring a colour image consideration must be made in regard to the correlation between different colour planes. [80] provides a good review of current technology and research in the field of digital colour imaging. The authors of [81] approach this problem by decorrelating the colour planes using the Karhunen-Loeve transformation, while Galatsanos, et al. have developed a spatially adaptive filter for colour images which utilises local within and between channel image properties [82].

1.5 Perception Motivated Restoration

In Section 1.4.3, we mentioned the problems caused by considering the image to be an ensemble of stationary processes. Any restoration process based on this concept can only ever produce sub-optimal results. However there is another consideration. When a restoration algorithm is designed with the aim of creating an image which will be more pleasing to the human eye, we must incorporate some kind of model of the human visual system. At the basis of most restoration algorithms is some form of image error measure which is being minimised.

The most common method to compare the similarity of two images is to compute their mean square error (MSE). However the MSE relates to the power of the error signal and has little relationship to human visual perception. An important drawback to the MSE and any cost function which attempts to use the MSE to restore a degraded image is that the MSE treats the image as a stationary process. All pixels are given equal priority regardless of their relevance to human perception. This suggests that information is ignored. When restoring images for the purpose of better clarity as perceived by humans the problem becomes acute. When humans observe the differences between two images, they do not give much consideration to the differences in individual pixel level values. Instead humans are concerned with matching edges, regions and textures between the two images. This is contrary to the concepts involved in the MSE. From this it can be seen that any cost function which treats an image as a stationary process can only produce a sub-optimal result.

Considerations regarding human perception have been examined in the past [83 - 93]. A great deal of work has been done toward developing linear filters for the removal of noise which incorporate some model of human perception [94 - 96]. In these works it is found that edges have a great importance to the way humans perceive images. Ran and Farvardin considered psychovisual properties of the human visual system in order to develop a technique to decompose an image into smooth regions, textured regions and regions containing what are described as “strong edges” [97]. This was done with a view primarily toward image compression. Similarly Bellini, Leone and Rovatti developed a fuzzy perceptual classifier to create what they described as a “pixel relevance map” to aid in image compression [98]. Hontsch and Karam developed a perceptual model for image compression which decomposed the image into components with varying frequency and orientation [99]. A perceptual distortion measure

was then described which used a number of experimentally derived constants. Huang and Coyle considered the use of stack filters for image restoration [100]. They used the concept of a weighted mean absolute error (WMAE), where the weights were determined by the perceptually motivated visible differences predictor (VDP) described in [101]. In the past, research has for the most part been concerned with the preservation of edges and the reduction of ringing effects caused by low-pass filters and the models presented to take account of human perception are often complicated. In this work we contend that the manner in which smooth regions are restored is of equal importance to the manner in which edges are restored. This is due to the fact that noise in an image is much more noticable in smooth regions, which has a important part to play in whether an image is perceived as “good” or “bad”. We will show in this thesis that complicated models of human visual perception may not be needed to produce improved restoration algorithms. Simple functions which incorporate some psychovisual properties of the human visual system can be easily incorporated into existing algorithms and can provide improvements over current techniques.

1.6 Overview of This Thesis

This thesis consists of 6 chapters. The first chapter provides material of an introductory nature to describe the basic concepts and current state of the art in the field of image restoration. Chapter 2 describes briefly the contributions of this thesis to the field of image restoration, gives a mathematical description of the restoration problem from the Hopfield neural network perspective, and presents a novel image error measure which is computationally inexpensive and yet powerful. Chapter 3 describes current algorithms based on this method. In addition a new algorithm is presented which is an improvement on the previous algorithms and the algorithm is extended to handle spatially variant degradations. A number of experiments

are performed to show the improvement in performance resulting from the use of the new algorithm. Chapter 4 extends the algorithm presented in Chapter 3 to implement adaptive constraint restoration methods for both spatially invariant and spatially variant degradations. A method of constraint selection is presented using concepts of human perception and experiments are performed which compare the algorithms performance to other algorithms in terms of restoration quality, efficiency, and semi-blind deconvolution performance. Chapter 5 utilises the image error measure presented in Chapter 2 to introduce novel restoration algorithms. Experiments are performed on the new algorithms to show an improvement in performance when compared to previous algorithms. Chapter 6 summarises this thesis.

1.7 Summary

This chapter has presented some introductory material which is important for understanding the subsequent chapters of this thesis. Firstly, the importance of image processing and the nature of image degradations was described. Then the concepts involved in image restoration are detailed and a short survey of classical image restoration methods is presented. These methods are classified into non-iterative transform based techniques and algebraic techniques. Following this, emergent image restoration research is described. This includes research into the use of neural networks for cost function optimisation and current investigations in the problems of spatially variant, adaptive, semi-blind and colour restoration. Section 1.5 described the importance of incorporating concepts from human visual perception into the field of image restoration, and detailed current research into the perception based image processing and compression techniques.

Chapter 2: Contributions of the Thesis

2.1 Introduction

In this thesis, the author has made a number of developments in the field of image restoration which, after extensive literary search, are believed to be novel. The author investigated the concept of using a Hopfield-based neural network to restore images which have suffered some form of degradation and additive noise. Previous algorithms described in the literature have been enhanced and their functionality expanded. In particular, the author has made three contributions to image restoration:

1. A fast algorithm has been developed which is an improvement on previous algorithms to decrease restoration time markedly without any reduction in restoration quality. The algorithm is extended to restore images suffering spatially variant distortion. The author shows that using the method described in this thesis, spatially variant images can be restored with only a little increase in restoration time.

2. The concept of adaptive constraint parameter (ACP) restoration is introduced in order to cope with different statistical properties in different parts of an image. The author investigates two methods of allocating each pixel in the image a value of constraint parameter. Gradient descent based methods are shown to be unacceptable in this case and the author has developed a method to assign each pixel a constraint parameter value using simple principles of human visual perception.

3. Continuing on the subject of using human visual criteria to judge images, the author has proposed a novel, simple image error measure which uses concepts of human perception to judge the difference between two images, without being computationally expensive. This image error measure is then used as the basis for a family of new restoration cost functions. The author has developed two new restoration algorithms which are designed to be fast and computationally inexpensive, and yet be an improvement on traditional algorithms.

This chapter briefly details the contributions of this thesis to the field of image restoration. Subsequent chapters will elaborate upon these contributions and provide experimental and mathematical justifications. In order to put things into perspective, Section 2.2 briefly describes linear image restoration and the basic neural network formulation of the problem. Section 2.3 details the improvement the author has made to increase the speed of the basic neural network algorithm, while Section 2.4 details the expansion of the restoration algorithm to the case of spatially variant distortion. Section 2.5 describes the algorithm developed by the author to implement adaptive constraint parameter restoration. Section 2.6 presents the novel image error measure developed by the author and shows how this error measure is a closer match to human concepts of image quality when compared to the MSE. Section 2.7 details new algorithms developed by the author, which use the Hopfield neural network model as a base for the minimisation of new non-linear restoration cost functions incorporating the error measure developed in Section 2.6. Section 2.8 summarises this chapter.

2.2 Image Restoration using a Neural Network

To restore an image degraded by a linear distortion, a restoration cost function is developed. The cost function is created using knowledge about the degraded image and an estimate of the

degradation, and possibly noise, suffered by the original image to produce the degraded image. The free variable in the cost function is an image, often denoted by $\hat{\boldsymbol{f}}$, and the cost function is designed such that the $\hat{\boldsymbol{f}}$ which minimises the cost function is an estimate of the original image. A common class of cost functions is based on the mean square error (MSE) between the original image and the estimate image. Cost functions based on the MSE often have a quadratic nature.

2.2.1 Problem Formulation

All linear image degradations can be described by their impulse response. Consider a PSF of size *P* by *P* acting on an image of size *N* by *M.* In the case of a two-dimensional image, the PSF may be written as $h(x, y)$. When noise is also present in the degraded image, as is often the case in real world applications, the image degradation model in the discrete case becomes [2]:

$$g(x, y) = \sum_{\alpha}^{N} \sum_{\beta}^{M} f(\alpha, \beta) h(x, y; \alpha, \beta) + n(x, y) \tag{2.1}$$

where $f(x, y)$ and $g(x, y)$ are the original and degraded images respectively, and $n(x, y)$ is the additive noise component of the degraded image. If $h(x, y)$ is a linear function then by lexicographically ordering $g(x, y)$, $f(x, y)$ and $n(x, y)$ into column vectors of size *NM*, we may restate (2.1) as a matrix operation [1], [2]:

$$\boldsymbol{g} = \boldsymbol{H}\boldsymbol{f} + \boldsymbol{n} \tag{2.2}$$

where $\boldsymbol{g}$ and $\boldsymbol{f}$ are the lexicographically organised degraded and original image vectors, $\boldsymbol{n}$ is the additive noise component and $\boldsymbol{H}$ is a matrix operator whose elements are an arrangement of the elements of $h(x, y)$ such that the matrix multiplication of $\boldsymbol{f}$ with $\boldsymbol{H}$ performs the same operation as convolving $f(x, y)$ with $h(x, y)$. In general, $\boldsymbol{H}$ may take any form. However if $h(x, y)$ is spatially invariant with $P \ll \min(N, M)$ then $h(x, y;\alpha, \beta)$ becomes $h(x-\alpha, y-\beta)$ in (2.1) and $\boldsymbol{H}$ takes the form of a block-Toeplitz matrix. If $h(x, y)$ has a simple form of space variance then $\boldsymbol{H}$ may have a simple form, resembling a block-Toeplitz matrix.

In this work, the degradation measure we consider minimising starts with the constrained least square error measure [2]:

$$E = \frac{1}{2}\|\boldsymbol{g} - \boldsymbol{H}\hat{\boldsymbol{f}}\|^2 + \frac{1}{2}\lambda\|\boldsymbol{D}\hat{\boldsymbol{f}}\|^2 \qquad (2.3)$$

where $\hat{\boldsymbol{f}}$ is the restored image estimate, λ is a constant, and $\boldsymbol{D}$ is a smoothness constraint operator. Since $\boldsymbol{H}$ is often a low pass distortion, $\boldsymbol{D}$ will be chosen to be a high pass filter. The second term in (2.3) is the regularisation term. The more noise that exists in an image, the greater the second term in (2.3) should be, hence minimising the second term will involve reducing the noise in the image at the expense of restoration sharpness.

Choosing λ becomes an important consideration when restoring an image. Too great a value of λ will oversmooth the restored image, whereas too small a value of λ will not properly suppress noise.

2.2.2 Neural Network Restoration

Neural network image restoration approaches are designed to minimise a quadratic programming problem [35 - 39]. The general form of a quadratic programming problem can be stated as:

Minimise the energy function associated with a neural network given by:

$$E = -\frac{1}{2}\hat{\boldsymbol{f}}^T \boldsymbol{W} \hat{\boldsymbol{f}} - \boldsymbol{b}^T \hat{\boldsymbol{f}} + c \tag{2.4}$$

comparing this with (2.3), $\boldsymbol{W}$, $\boldsymbol{b}$, and c are functions of $\boldsymbol{H}$, $\boldsymbol{D}$, λ, and $\boldsymbol{n}$, and other problem related constraints. In terms of a neural network energy function, the (i,j)th element of $\boldsymbol{W}$ corresponds to the interconnection strength between neurons (pixels) i and j in the network. Similarly, vector $\boldsymbol{b}$ corresponds to the bias input to each neuron.

Equating the formula for the energy of a neural network with equation (2.3), the bias inputs and interconnection strengths can be found such that as the neural network minimises its energy function, the image will be restored. From [35], setting $L = MN$, the interconnection strengths and bias inputs were shown to be:

$$w_{ij} = -\sum_{p=1}^{L} h_{pi} h_{pj} - \lambda \sum_{p=1}^{L} d_{pi} d_{pj} \tag{2.5}$$

$$b_i = \sum_{p=1}^{L} g_p h_{pi} \tag{2.6}$$

where w_{ij} is the interconnection strength between pixels i and j, and b_i is the bias input to neuron (pixel) i. In addition, h_{ij} is the (i,j)th element of matrix $\boldsymbol{H}$ from equation (2.2) and d_{ij} is the (i,j)th element of matrix $\boldsymbol{D}$ from equation (2.3). The full derivation of (2.5) and (2.6) is given in Appendix A. In this thesis, we consider a sequential neural network algorithm in which each pixel's energy contribution is reduced individually during each iteration.

2.3 Improvement on the Basic Neural Network Algorithm

In Chapter 3, the basic neural network algorithms for the restoration of images are examined. The author then builds on these previously described algorithms to create a new faster algorithm. This algorithm produces identical results to the previous algorithms, yet takes much less time to restore an image. Briefly, this contribution may be summarised as follows:

Equating the formula for the energy of a neural network with quadratic cost function the bias inputs and interconnection strengths can be found such that as the neural network minimises its energy function, the image will be restored. In previous algorithms the image was restored by calculating the input to the neuron, and then updating the neuron value in unit steps, $\Delta \hat{f}_i = \frac{u_i}{|u_i|}$, where u_i is the input to the current neuron, until the energy contribution of the current neuron to the energy function is minimised.

The author found that a much faster way to minimise the energy contribution of the current

neuron was to allow the neuron to take a single larger step of $\Delta \hat{f}_i' = \frac{-u_i}{w_{ii}}$. This allows the time taken during each iteration to be greatly reduced without any reduction in restoration quality.

The algorithm may be further enhanced based on the fact any neuron in the network is connected by non-zero weights to only a finite neighbourhood of pixels centred on it. The size of this neighbourhood and the weights between the centre pixel and its neighbours do not vary with the current neuron's position in the network, (as long as that neuron is sufficiently far from the edges of the image and the degradation is space invariant). This means that, as long as boundary regions of the image are avoided, the entire $\boldsymbol{W}$ matrix can be stored as a single small "weighting mask". This greatly reduces computational complexity and storage space required by the algorithm on a computer.

2.4 Space Variant Processing

In the previous section, the analysis of the restoration procedure did not assume that the degradation was space invariant. In fact the degradation may be space variant and the network will still converge to the correct result. Many image restoration methods have difficulties being extended to the space variant case, however in Chapter 3 the author shows that the fast algorithm presented in that chapter can easily be extended to the space variant case. A space variant distortion will not allow one weighting mask to be used for the entire image. In the general case, each pixel would require an individual weighting mask. However such an array of weighting masks would not contain any zero elements and hence require much less storage space than the $\boldsymbol{W}$ matrix.

The author investigated a simple case of space variant distortion. The case examined in Chapter 3 is one where each row of the image is degraded by a different PSF, but the PSFs cycle though a closed set of available PSFs. In that chapter, the author showed that for V such distortions, only $2V$-2 weighting masks were required. This method requires substantially less storage than is needed for computing the entire $\boldsymbol{W}$ matrix. The author shows that an image degraded by this process can be restored exactly with a restoration time comparable to the space invariant case.

2.5 Adaptive Regularisation Methodologies

In Chapter 4, the author proposed a restoration algorithm, which implements adaptive constraint processing (ACP). To utilise an adaptive regularisation parameter in the restoration method means that each pixel may have a different value of regularisation parameter. The intention is to suppress noise more strongly in regions where errors are most noticable.

The author considered methods for assigning each pixel a constraint value and showed that a gradient descent based method did not work in this case. Since the author is particularly concerned with restoring images for the purpose of human visual evaluation, the use of concepts of human perception to select constraint values for each pixel was considered. By considering how simple local regional statistics effect the human ability to detect noise in a restored image, the author presents a technique for selecting regularisation parameter values by relating the value of the constraint to a pixel's local variance by a log-linear relationship. This technique is shown to produce visually superior restored images under a range of conditions. Computationally, the author shows that by creating a set of weighting masks where each mask implements a different constraint parameter value, adaptive constraint restoration can be achieved with very

little additional restoration time.

In addition, the author shows in Chapter 4 how the adaptive constraint methodology can prove very useful in the difficult field of semi-blind image restoration. Semi-blind image restoration is the problem of restoring an image when some aspects of the degrading function are unknown. The author shows that the statistics-based adaptive regularisation technique can compensate for uncertainty in the degrading function. In particular, if the function is space variant in an unknown manner, the adaptive technique the author presents in Chapter 4 is shown to produce a superior restoration compared to a non-adaptive method even when the parameters of the degradation are known.

2.6 New Error Measure

Another major contribution of this thesis is a novel image error measure which is both simple and powerful. Often an image is filtered for the purpose of greater visual quality or clarity as perceived by humans, such as an old photograph or a television transmission [1]. In these cases an enhancement algorithm is attempting to produce an image which human beings will find visually pleasing. For an enhancement algorithm to produce an image which humans find pleasant it should possess a way of improving the image's quality which takes into account human visual preferences. This section will briefly describe a perception based error measure for this purpose.

Many classical image restoration cost functions are based on the mean square error (MSE):

$$MSE = \frac{\sum_{x=0}^{M-1}\sum_{y=0}^{N-1}[\hat{f}(x,y) - f(x,y)]^2}{MN} \qquad (2.7)$$

where $\hat{f}(x,y)$ is the restored image of interest.

This error measure compares images on a pixel to pixel basis, and in effect makes a statement about the power of the noise signal created by the subtraction of the two images to be compared. This kind of information is mathematically useful. However cost functions based on the MSE favour slow variations in the image and bear very little relationship to the manner in which humans view the differences between two images. Humans tend to pay more attention to sharp differences in intensity within an image [92-94], for example edges or noise in background regions. Hence an error measure should take into account the concept that low variance regions in the original image should remain low variance regions in the enhanced image, and high variance regions should likewise remain high variance regions. This implies that noise should be kept at a minimum in background regions, where it is most noticable, but noise suppression should not be as important in highly textured regions where image sharpness should be the dominate consideration. These considerations are especially important in the field of colour image restoration. Humans appear to be much more sensitive to slight colour variations than they are to variations in brightness.

In view of the above problems with classical error measures such as the MSE, the author investigated a different error measure which is based on the comparison of local statistics, local standard deviation mean square error (LSMSE). The LSMSE is calculated by comparing the local standard deviations in the neighbourhood of each pixel in the images we wish to com-

pare. The mean square error between the two standard deviations gives an indication of the degree of similarity between the two images. This error measure requires matching between the high and low variance regions of the image, which is more intuitive in terms of human visual perception.

The local standard deviation in the A by A neighbourhood of pixel (x, y) in image $\boldsymbol{f}$ is given by:

$$\sigma_A(\boldsymbol{f}(x,y)) = \sqrt{\sum_{i=x-\frac{A}{2}}^{x+\frac{A}{2}} \sum_{j=y-\frac{A}{2}}^{y+\frac{A}{2}} \frac{(\boldsymbol{f}(i,j) - M_A(\boldsymbol{f}(x,y)))^2}{A^2}} \tag{2.8}$$

where the local mean of the A by A neighbourhood of pixel (x, y) in image $\boldsymbol{f}$ is given by:

$$M_A(\boldsymbol{f}(x,y)) = \sum_{i=x-\frac{A}{2}}^{x+\frac{A}{2}} \sum_{j=y-\frac{A}{2}}^{y+\frac{A}{2}} \frac{\boldsymbol{f}(i,j)}{A^2} \tag{2.9}$$

Using the above conventions, we can define the LSMSE between two NxM images $\boldsymbol{f}$ and $\boldsymbol{g}$ as:

$$\mathrm{LSMSE}_A(\boldsymbol{f}, \boldsymbol{g}) = \sum_{x=0}^{N-1} \sum_{y=0}^{M-1} \frac{(\sigma_A(\boldsymbol{f}(x,y)) - \sigma_A(\boldsymbol{g}(x,y)))^2}{NM} \tag{2.10}$$

The LSMSE in effect requires the matching of homogeneous statistical regions between the

two images to be compared. Hence background regions should remain as noise free as possible and highly textured regions should not be smoothed by the restoration procedure. An alternate definition of this error measure is given by the comparisons of local variances rather than local standard deviations:

The local variance in an A by A neighbourhood of pixel (x, y) in image $\boldsymbol{f}$ is given by:

$$\sigma_A^2(\boldsymbol{f}(x,y)) = \sum_{i=x-\frac{A}{2}}^{x+\frac{A}{2}} \sum_{j=y-\frac{A}{2}}^{y+\frac{A}{2}} \frac{(\boldsymbol{f}(i,j) - M_A(\boldsymbol{f}(x,y)))^2}{A^2} \tag{2.11}$$

where $M_A(\boldsymbol{f}(x,y))$ is given by (2.9).

Using the above conventions, we define the LVMSE between two NxM images $\boldsymbol{f}$ and $\boldsymbol{g}$ as:

$$\text{LVMSE}_A(\boldsymbol{f},\boldsymbol{g}) = \sum_{x=0}^{N-1} \sum_{y=0}^{M-1} \frac{(\sigma_A^2(\boldsymbol{f}(x,y)) - \sigma_A^2(\boldsymbol{g}(x,y)))^2}{NM} \tag{2.12}$$

LVMSE stands for local variance mean square error (LVMSE). Although the LSMSE and the LVMSE are related, we may wish to use one or the other depending on the circumstances. The LSMSE is best for measuring the error between two images since its range of possible values is not as great as the LVMSE. The LVMSE on the other hand is easier and faster to calculate since the square root calculation needed to compute the standard deviations is absent. The utilisation of the LVMSE and the LSMSE should produce similar results.

Figure 2.1 illustrates the connection between the LSMSE and human perception. Figures 2.1a, 2.1b and 2.1c show three regions of an image. Figure 2.1a shows a smooth region, Figure 2.1b shows an image edge and Figure 2.1c shows a high texture region within the image. Figures 2.1d, 2.1e and 2.1f show, respectively, the smooth, edge and texture regions degraded by Gaussian noise of standard deviation 17.12. It is clear from these figures that noise is most noticable in Figure 2.1d, where image detail is low. In Figure 2.1e, noise is again noticable in the smooth regions of the image, but less noticable around the edge. In Figure 2.1f, noise is the least noticable due to the presence of texture. This corresponds with the arguments above regarding the manner in which human perceive noise in images. For each pair of images, the SNR, the LSMSE, and the LVMSE were computed by comparing the original and noisy images. The following table illustrates the results.

Table 2.1: Comparison of the SNR, LSMSE and the LVMSE for various image regions.

Image Region	SNR (dB)	LSMSE	LVMSE
Smooth Region	2.54	114.66	79162
Edge Region	10.57	87.49	72650
Textured Region	7.98	18.25	70775

From Table 2.1 we can see that the SNR fails to correspond to human perception, stating that the image least corrupted is the edge image. The LSMSE on the other hand clearly shows that noise has the greatest effect on the smooth image and is least noticable on the highly textured image. This matches human observation. The LVMSE also follows the same trend as the LSMSE, however the larger values of the LVMSE are less desirable for comparing images than

the LSMSE.

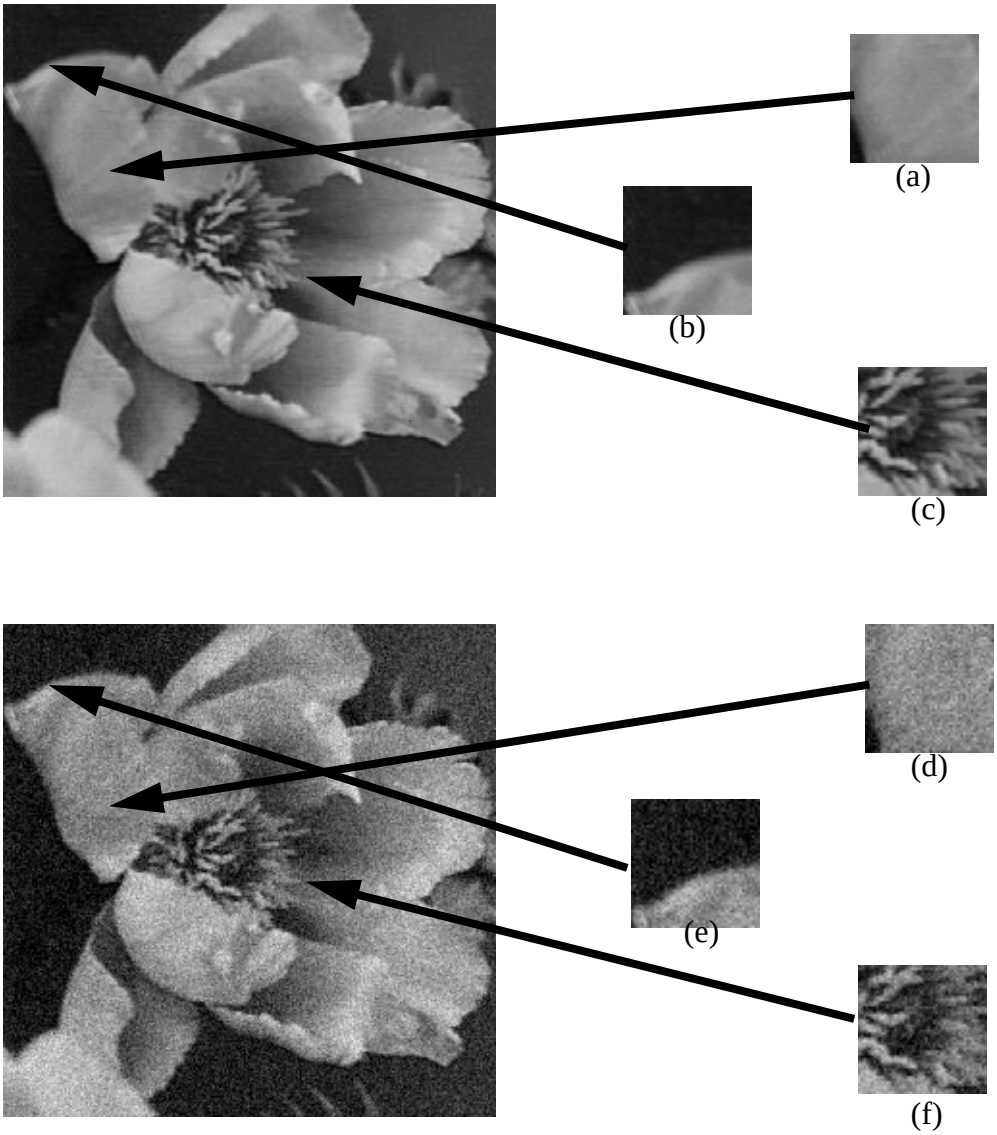


Figure 2.1: (a) Smooth region of flower image. (b) Edge region of flower image. (c) Textured region of flower image. (d) Image 2.1a with noise added. (e) Image 2.1b with noise added. (f) Image 2.1c with noise added.

2.7 LSMSE-based Cost Functions for Image Restoration

In Chapter 5, the author uses the image comparison measure introduced in the previous section to develope new restoration algorithms. Two image restoration algorithms are introduced.

These algorithms originated from the constrained least square error cost function, but move away from the Hopfield neural network model even further than the adaptive algorithm presented in Chapter 4.

2.7.1 LVMSE Modified Cost Function

The first novel algorithm in Chapter 5 is based on enhancing the basic constrained least square error cost function to include a term which minimises LVMSE. The modified cost function is given by:

$$E = \frac{1}{2}\|\boldsymbol{g} - \boldsymbol{H}\hat{\boldsymbol{f}}\|^2 + \frac{\lambda}{2}\|\boldsymbol{D}\hat{\boldsymbol{f}}\|^2 + \theta \sum_{x=0}^{N-1}\sum_{y=0}^{M-1} \frac{(\sigma_A^2(\hat{\boldsymbol{f}}(x,y)) - \sigma_A^2(\boldsymbol{g}(x,y))^*)^2}{NM} \qquad (2.13)$$

where the first two terms are the standard constrained least square error cost function, which was explained in Section 2.2. $\sigma_A^2(\hat{\boldsymbol{f}}(x,y))$ is the variance of the local region surrounding pixel (*x*,*y*) in the image estimate and $\sigma_A^2(\boldsymbol{g}(x,y))^*$ is the variance of the local region surrounding pixel (*x*,*y*) in the degraded image scaled to predict the variance in the original image. In Chapter 5, the author also presents a method for estimating $\sigma_A^2(\boldsymbol{g}(x,y))^*$. In general, if we consider an image degraded by a linear process, then we find that a useful approximation is:

$$\sigma_A^2(\boldsymbol{g}(x,y))^* = K(x,y)(\sigma_A^2(\boldsymbol{g}(x,y)) - J(x,y)) \qquad (2.14)$$

where $J(x, y)$ is a function of the noise added to the degraded image at point (x, y) and $K(x, y)$

is a function of the degrading point spread function at point (x, y). The author has developed an algorithm to minimise (2.13) by using the components which are similar to the previous neural network model to extend the algorithm presented in Chapter 3.

2.7.2 Log LVR Modified Cost Function

After analysing the nature of (2.13), the author presents an alternative version of the new algorithm in Chapter 5. This new version performs similarly to the algorithm based on (2.13), yet in some respects is an improvement.

The second version of the LSMSE-based cost function suggested by the author is:

$$E = \frac{1}{2}\|\boldsymbol{g} - \boldsymbol{H}\hat{\boldsymbol{f}}\|^2 + \frac{\lambda}{2}\|\boldsymbol{D}\hat{\boldsymbol{f}}\|^2 + \theta \sum_{x=0}^{N-1}\sum_{y=0}^{M-1} \frac{\left(\ln\left(\frac{\sigma_A^2(\hat{\boldsymbol{f}}(x,y))}{\sigma_A^2(\boldsymbol{g}(x,y))^*}\right)\right)^2}{NM} \tag{2.15}$$

where $\sigma_A^2(\hat{\boldsymbol{f}}(x,y))$ and $\sigma_A^2(\boldsymbol{g}(x,y))^*$ are as defined in the previous section. As with (2.13), an algorithm is presented in Chapter 5 to minimise (2.15).

The new algorithms have some of the adaptive properties of the algorithm in Chapter 4, with greater ease of use and robustness. The performances of the new algorithms are shown to be superior to previous algorithms when applied to grayscale and especially colour images in highly noisy environments. In addition the author shows that the LSMSE-based algorithms are much more fault tolerant than previous algorithms in the presence of network errors.

2.8 Summary

This chapter was written to clearly and briefly describe the contributions to the field of image restoration presented in this thesis. The contributions include faster restoration algorithms using neural networks with respect to previous neural network restoration algorithms, adaptive restoration techniques, and perception-based error measurement and applications in restoration. The contributions mentioned in this chapter will be more clearly described and justified (experimentally and mathematically) in subsequent chapters. In subsequent chapters, the above contributions of this thesis will be examined against both classical approaches to image processing and previous neural network approaches described in the literature.

The author trusts that these contributions are novel and non-trivial.

Chapter 3: Basic Neural Network Algorithm

3.1 Introduction

In this chapter, we investigate the basic neural network approach to the problem of restoring an image degraded by some function in the presence of noise. In Chapter 2, the image restoration problem was stated and the least mean square error cost function for image restoration was described. In addition, the basic neural network model was given and it was shown how the weights and bias inputs of the neurons in the network may be assigned to solve the image restoration problem. This chapter builds on the material presented in Chapter 2 to examine the image restoration algorithms which result from these fundamental concepts. In Section 3.2, we analyse some previous neural network algorithms developed in the literature to solve this problem and detail the advantages and disadvantages of each. In Section 3.3, a new algorithm is presented with greatly improved performance when compared to the previous algorithms. This algorithm will form the basis for all subsequent algorithms in this thesis. It will then be shown that the introduced algorithm will converge on a solution in a finite number of iterations and we will analyse some important properties of the introduced algorithm in Section 3.4. Section 3.5 expands the algorithm to handle space variant degradations and Section 3.6 details implementation considerations. Some experimental results are presented in Section 3.7. Section 3.8 summarises this chapter.

3.2 Neural Network Restoration Algorithms in the Literature

In the network described by Zhou, et al. for an image with $S + 1$ gray levels, each pixel is represented by $S + 1$ neurons [35]. Each neuron can have a value of 0 or 1. The value of the *i*th pixel is then given by:

$$\hat{f}_i = \sum_{k=0}^{S} v_{i,k} \tag{3.1}$$

where $v_{i,k}$ is the state of the *k*th neuron of the *i*th pixel. Each neuron is visited sequentially and has its input calculated according to:

$$u_i = b_i + \sum_{j=1}^{L} w_{ij} \hat{f}_j \tag{3.2}$$

where u_i is the input to neuron *i*, and $\hat{f}_j$ is the state of the *j*th neuron. Based on u_i, the neuron's state is updated according to the following rule:

$$\Delta \hat{f}_i = G(u_i)$$

where

$$G(u) = \begin{cases} 1, u > 0 \\ 0, u = 0 \\ -1, u < 0 \end{cases} \tag{3.3}$$

The change in energy resulting from a change in neuron state of $\Delta \hat{f}_i$ is given by:

$$\Delta E = -\frac{1}{2} w_{ii} (\Delta \hat{f}_i)^2 - u_i \Delta \hat{f}_i \qquad (3.4)$$

If $\Delta E < 0$, then the neuron's state is updated. This algorithm may be summarised as:

Algorithm 3.1:

repeat

{

For $i = 1, ..., L$ do

{

For $k = 0, ... , S$ do

{

$$u_i = b_i + \sum_{j=1}^{L} w_{ij} \hat{f}_j$$

$$\Delta \hat{f}_i = G(u_i)$$

where $$G(u) = \begin{cases} -1, u < 0 \\ 0, u = 0 \\ 1, u > 0 \end{cases}$$

$$\Delta E = -\frac{1}{2} w_{ii} (\Delta \hat{f}_i)^2 - u_i \Delta \hat{f}_i$$

if $\Delta E < 0$, then $v_{i,k} = v_{i,k} + \Delta \hat{f}_i$

$$\hat{f}_i = \sum_{k=0}^{S} v_{i,k}$$

}

}

$t = t + 1$

}

until ($\hat{f}_i(t) = \hat{f}_i(t-1)\ \forall i = 1, ..., L$)

In the paper by Paik and Katsaggelos, Algorithm 3.1 was altered to remove the step where the energy reduction is checked following the calculation of $\Delta\hat{f}_i$ [36]. Paik and Katsaggelos presented an algorithm which made use of a more complicated neuron. In their model, each pixel was represented by a single neuron which takes discrete values between 0 and *S*, and is capable of updating its value by ± 1, or keeping the same value during a single step. A new method for calculating $\Delta\hat{f}_i$ was also presented:

$$\Delta\hat{f}_i = G'_i(u_i)$$

where

$$G'_i(u) = \begin{cases} -1, & u < -\theta_i \\ 0, & -\theta_i \le u \le \theta_i \\ 1, & u > \theta_i \end{cases} \qquad (3.5)$$

where $\theta_i = -\frac{1}{2}w_{ii} > 0$

This algorithm may be presented as:

Algorithm 3.2:

repeat

{

For $i = 1, ..., L$ do

{

$$u_i = b_i + \sum_{j=1}^{L} w_{ij} \hat{f}_j$$

$$\Delta \hat{f}_i = G'_i(u_i)$$

where

$$G'_i(u) = \begin{cases} -1, & u < -\theta_i \\ 0, & -\theta_i \le u \le \theta_i \\ 1, & u > \theta_i \end{cases}$$

where $\theta_i = -\frac{1}{2} w_{ii} > 0$

$$\hat{f}_i(t+1) = K(\hat{f}_i(t) + \Delta \hat{f}_i)$$

where

$$K(u) = \begin{cases} 0, u < 0 \\ u, 0 \le u \le S \\ S, u \ge S \end{cases}$$

}

$t = t + 1$

}

until ($\hat{f}_i(t) = \hat{f}_i(t - 1)$ $\forall i = 1, ..., L$)

Algorithm 3.2 makes no specific check that energy has decreased during each iteration and

so in [36] they proved that Algorithm 3.2 would result in a decrease of the energy function at each iteration. Note that in Algorithm 3.2, each pixel only changes its value by ± 1 during an iteration. In Algorithm 3.1, the pixel's value would change by any amount between 0 and S during an iteration since each pixel was represented by $S+1$ neurons. Although Algorithm 3.2 is much more efficient in terms of the number of neurons used, it may take many more iterations than Algorithm 3.1 to converge to a solution (although the time taken may still be faster than Algorithm 3.1). If we consider that the value of each pixel represents a dimension of the L dimensional energy function to be minimised, then we can see that Algorithms 3.1 and 3.2 have slightly different approaches to finding a local minimum. In Algorithm 3.1, the energy function is minimised along each dimension in turn. The image can be considered to represent a single point in the solution space. In Algorithm 3.1, this point moves to the function minimum along each of the L axes of the problem until it eventually reaches a local minimum of the energy function. In Algorithm 3.2, for each pixel, the point takes a unit step in a direction that reduces the network energy along that dimension. If the weight matrix is negative definite ($-\boldsymbol{W}$ is positive definite), however, regardless of how these algorithms work, the end results must be similar (if each algorithm ends at a minimum). The reason for this is that when the weight matrix is negative definite, there are no local minima, except for the global minimum. That is, the function has only one minimum. In this case the matrix $\boldsymbol{W}$ is invertible and taking (2.4) we see that:

$$\frac{\partial E}{\partial \hat{\boldsymbol{f}}} = -\boldsymbol{W}\hat{\boldsymbol{f}} - \boldsymbol{b} \tag{3.6}$$

Hence the solution is given by:

$$\hat{\boldsymbol{f}}^{*} = -\boldsymbol{W}^{-1}\boldsymbol{b} \tag{3.7}$$

(assuming that $\boldsymbol{W}^{-1}$ exists).

The $\hat{\boldsymbol{f}}^{*}$ is the only minimum and the only stationary point of this cost function, so we can state that if $\boldsymbol{W}$ is negative definite and Algorithm 3.1 and Algorithm 3.2 both terminate at a local minimum, the resultant image must be close to $\hat{\boldsymbol{f}}^{*}$ for both algorithms. Algorithm 3.1 approaches the minimum in a zig-zag fashion, whereas Algorithm 3.2 approaches the minimum with a smooth curve. If $\boldsymbol{W}$ is not negative definite then local minima may exist and Algorithms 3.1 and 3.2 may not produce the same results. If Algorithm 3.2 is altered so that instead of changing each neuron's value by ± 1 before going to the next neuron, the current neuron is iterated until the input to that neuron is zero, then Algorithms 3.1 and 3.2 will produce identical results. Each algorithm will terminate in the same local minimum.

3.3 The Proposed Algorithm

Although Algorithm 3.2 is an improvement on Algorithm 3.1, it is not optimal. From iteration to iteration, neurons often oscillate about their final value, and during the initial iterations of Algorithm 3.1 a neuron may require 100 or more state changes in order to minimise its energy contribution. A faster method to minimise the energy contribution of each neuron being considered is suggested by examination of the mathematics involved. For an image where each pixel is able to take on any discrete integer intensity between 0 and *S*, we assign each pixel in the image to a single neuron able to take any discrete value between 0 and *S*. Since the formula for the energy reduction resulting from a change in neuron state $\Delta \hat{f}_i$ is a simple quadratic, it is possible to solve for the $\Delta \hat{f}_i$ which produces the maximum energy reduction. Theorem 3.1 states that this approach will result in the same energy minimum as Algorithm 3.1 and hence

the same final state of each neuron after it is updated.

Theorem 3.1: *For each neuron i in the network during each iteration, there exists a state change* $\Delta\hat{f}_i'$ *such that the energy contribution of neuron i is minimised.*

Proof:

Let u_i be the input to neuron i which is calculated by:

$$u_i = b_i + \sum_{j=1}^{L} w_{ij}\hat{f}_j$$

Let ΔE be the resulting energy change due to $\Delta\hat{f}_i$.

$$\Delta E = -\left(\frac{1}{2}\right)w_{ii}(\Delta\hat{f}_i)^2 - u_i\Delta\hat{f}_i \qquad (3.8)$$

Differentiating ΔE with respect to $\Delta\hat{f}_i$ gives us:

$$\frac{\partial}{\partial\hat{f}_i}\Delta E = -w_{ii}\Delta\hat{f}_i - u_i$$

The value of $\Delta\hat{f}_i$ which minimises (3.8) is given by:

$$0 = -w_{ii}\Delta\hat{f}_i{}' - u_i$$

Therefore

$$\Delta\hat{f}_i{}' = \frac{-u_i}{w_{ii}} \tag{3.9}$$

QED.

Based on theorem 3.1, the proposed algorithm is presented below.

Algorithm 3.3.

repeat

{

for $i = 1, \ldots, L$ do

{

$$u_i = b_i + \sum_{j=1}^{L} w_{ij}\hat{f}_j$$

$$\Delta\hat{f}_i = G(u_i)$$

where $$G(u) = \begin{cases} -1, u < 0 \\ 0, u = 0 \\ 1, u > 0 \end{cases}$$

$$\Delta E_{ss} = -\left(\frac{1}{2}\right)w_{ii}(\Delta\hat{f}_i)^2 - u_i\Delta\hat{f}_i \tag{3.10}$$

if $\Delta E_{ss} < 0$ then $\Delta \hat{f}_i' = \frac{-u_i}{w_{ii}}$

$\hat{f}_i(t+1) = K(\hat{f}_i(t) + \Delta \hat{f}_i')$

where

$$K(u) = \begin{cases} 0, u < 0 \\ u, 0 \le u \le S \\ S, u \ge S \end{cases}$$

}

t = t+1

}

until ($\hat{f}_i(t) = \hat{f}_i(t-1) \; \forall i = 1, ..., L$)

Each neuron is visited sequentially and has its input calculated. Using the input value, the state change needed to minimise the neuron's energy contribution to the network is calculated. Note that since $\Delta \hat{f}_i \in \{-1, 0, 1\}$ and $\Delta \hat{f}_i$ and $\Delta \hat{f}_i'$ must be the same sign as u_i, step (3.10) is equivalent to checking that at least a unit step can be taken which will reduce the energy of the network. If $\Delta E_{ss} < 0$, then

$$-\frac{1}{2} w_{ii} - u_i \Delta \hat{f}_i < 0$$

$$-\frac{1}{2} w_{ii} - |u_i| < 0$$

$$\therefore -w_{ii} < 2|u_i|$$

Substituting this result into the formula for $\Delta \hat{f}_i'$ we get:

$$\Delta \hat{f}_i' = \frac{-u_i}{w_{ii}} > \frac{u_i}{2|u_i|} = \frac{1}{2}\Delta \hat{f}_i$$

Since $\Delta \hat{f}_i'$ and $\Delta \hat{f}_i$ have the same sign and $\Delta \hat{f}_i = \pm 1$ we obtain:

$$\left|\Delta \hat{f}_i'\right| > \frac{1}{2} \tag{3.11}$$

In this way $\Delta \hat{f}_i'$ will always be large enough to alter the neuron's discrete value.

Algorithm 3.3 makes use of concepts from both Algorithm 3.1 and Algorithm 3.2. Like Algorithm 3.1 the energy function is minimised in solution space along each dimension in turn until a local minimum is reached. In addition, the efficient use of space by Algorithm 3.2 is utilised. Note that the above algorithm is much faster than either Algorithm 3.1 or 3.2 due to the fact that this algorithm minimises the current neuron's energy contribution in one step rather than through numerous iterations as did Algorithms 3.1 and 3.2.

3.4 Analysis

In the paper by Paik and Katsaggelos, it was shown that Algorithm 3.2 would converge to a fixed point after a finite number of iterations and that the fixed point would be a local minimum of *E* in (2.3) in the case of a sequential algorithm [36]. Here we will show that Algorithm 3.3 will also converge to a fixed point which is a local minimum of *E* in (2.3).

Algorithm 3.2 makes no specific check that energy has decreased during each iteration and so in [36] they proved that Algorithm 3.2 would result in a decrease of the energy function at

each iteration. Algorithm 3.3 however changes the current neurons state if and only if an energy reduction will occur and $|\Delta \hat{f}_i| = 1$. For this reason Algorithm 3.3 can only reduce the energy function and never increase it. From this we can observe that each iteration of Algorithm 3.3 brings the network closer to a local minimum of the function. The next question is "Does Algorithm 3.3 ever reach a local minimum and terminate?". Note that the gradient of the function is given by:

$$\frac{\partial E}{\partial \hat{\boldsymbol{f}}} = -\boldsymbol{W}\hat{\boldsymbol{f}} - \boldsymbol{b} = -\boldsymbol{u} \tag{3.12}$$

where $\boldsymbol{u}$ is a vector whose ith element contains the current input to neuron i. Note that during any iteration, $\boldsymbol{u}$ will always point in a direction that reduces the energy function. If $\hat{\boldsymbol{f}} \neq \hat{\boldsymbol{f}}^*$ then for at least one neuron a change in state must be possible which would reduce the energy function. For this neuron, $u_i \neq 0$. The algorithm will then compute the change in state for this neuron to move closer to the solution. If $|\Delta \hat{f}_i'| > \frac{1}{2}$ the neuron's state will be changed. In this case we assume that no boundary conditions have been activated to stop neuron i from changing value. Due to the discrete nature of the neuron states we see that the step size taken by the network is never less than 1.

To re-state the facts obtained so far:

- During each iteration Algorithm 3.3 will reduce the energy of the network.

- A reduction in the energy of the network implies that the network has moved closer to a

local minimum of the energy function.

- There is a lower bound to the step size taken by the network and a finite range of neuron states. Since the network is restricted to changing state only when an energy reduction is possible, the network cannot iterate forever.

From these observations we can conclude that the network reaches a local minimum in a finite number of iterations, and that the solution given by Algorithm 3.3 will be close to the solution given by Algorithm 3.1 for the same problem. The reason why algorithms 3.1 and 3.3 must approach the same local minimum is the fact that they operate on the pixel in an identical manner. In Algorithm 3.1 each of the $S + 1$ neurons associated with pixel i is adjusted to reduce its contribution to the energy function. The sum of the contributions of the $S + 1$ neurons associated with pixel i in Algorithm 3.1 equals the final grayscale value of that neuron. Hence during any iteration of Algorithm 3.1 the current pixel can change to any allowable value. There are $S + 1$ possible output values of pixel i and only one of these values results when the algorithm minimises the contribution of that pixel. Hence whether the pixel is represented by $S + 1$ neurons or just a single neuron, the output grayscale value that occurs when the energy contribution of that pixel is minimised during the current iteration remains the same. Algorithms 3.1 and 3.3 both minimise the current pixels energy contribution, hence they must both produce the same results. In practice the author has found that all three algorithms generally produce identical results, which suggests that for reasonable values of the parameter λ, only a single global minimum is present.

Note that in this investigation we have not made any assumptions regarding the nature of the weighting matrix, $\boldsymbol{W}$, or the bias vector, $\boldsymbol{b}$. $\boldsymbol{W}$ and $\boldsymbol{b}$ determine where the solution is in the

solution space, but as long as they are constant during the restoration procedure the algorithm will still terminate after a finite number of iterations. This is an important result, and implies that even if the degradation suffered by the image is space variant or if we assign a different value of λ to each pixel in the image, the algorithm will still converge to a result. Even if $\boldsymbol{W}$ and $\boldsymbol{b}$ are such that the solution lies outside of the bounds on the values of the neurons, we would still expect that there exists a point or points which minimise E within the bounds. In practice we would not expect the solution to lie entirely out of the range of neuron values. If we assume that Algorithm 3.3 has terminated at a position where no boundary conditions have been activated. Then the condition:

$$\left|\Delta \hat{f}_i'\right| = \left|\frac{-u_i}{w_{ii}}\right| < \frac{1}{2}, \forall i \in \{0, 1, \ldots, L\}$$

must have been met. This implies that:

$$\left|u_i\right| < \frac{1}{2}\left|w_{ii}\right|, \forall i \in \{0, 1, \ldots, L\} \tag{3.13}$$

In [36], Paik and Katsaggelos noticed this feature as well, since the same termination conditions apply to Algorithm 3.2. The self-connection weight, w_{ii}, controls how close to the ideal solution the algorithm will approach before terminating. Since increasing the value of λ increases the value of w_{ii}, we would expect also that the algorithm would terminate more quickly and yet be less accurate for larger values of λ. This is found to occur in practice. When λ increased, the number of iterations before termination drops rapidly.

3.5 Dealing with Spatially Variant Distortion

In the above analysis, no conditions were placed upon the form of the matrix $\boldsymbol{H}$. In fact the neural network cost function only assumes that the weighting matrix, $\boldsymbol{W}$, is symmetric. In the case of a space invariant degradation, the matrix $\boldsymbol{H}$ will be symmetric. However in the real world this is not often the case. As discussed in Chapter 1, the general form of an image degradation will be space variant. However many of these distortions will still be linear and so the model given by equation (2.2) still holds. In the case of a linear space variant distortion, $\boldsymbol{H}$ will not be symmetric, but the weighting matrix, $\boldsymbol{W}$, will still be symmetric. This can be clearly shown by examining equation (2.5). By converting (2.5) into matrix notation we see that the weighting matrix is given by:

$$\boldsymbol{W} = -\boldsymbol{H}^T\boldsymbol{H} - \lambda\boldsymbol{D}^T\boldsymbol{D} \tag{3.14}$$

Assume that $\boldsymbol{H}$ is not symmetric, such that $\boldsymbol{H} \neq \boldsymbol{H}^T$, then:

$$\boldsymbol{W}^T = -(\boldsymbol{H}^T\boldsymbol{H} + \lambda\boldsymbol{D}^T\boldsymbol{D})^T$$

$$= -(\boldsymbol{H}^T\boldsymbol{H})^T - \lambda(\boldsymbol{D}^T\boldsymbol{D})^T$$

$$= -\boldsymbol{H}^T(\boldsymbol{H}^T)^T - \lambda\boldsymbol{D}^T(\boldsymbol{D}^T)^T$$

$$= -\boldsymbol{H}^T\boldsymbol{H} - \lambda\boldsymbol{D}^T\boldsymbol{D} = \boldsymbol{W}$$

Hence the weighting matrix, $\boldsymbol{W}$, is symmetric. This is a powerful feature of the neural network approach. If the image suffers a known form of space variance then the weights of the neurons in the network can be adjusted to restore the image exactly with very little additional

computational overhead.

If the degrading PSFs follow some regular or cyclic pattern then further optimisations can be made to the algorithm. Instead of developing the full non-symmetric form version of $\boldsymbol{H}$ for the entire image, certain regular patterns of space variance allow us to compute a set number of simpler alternative versions of the $\boldsymbol{H}$ matrix which describe each different type of degrading PSF the image has suffered. From this we can compute a number of alternative versions of the $\boldsymbol{W}$ matrix. Hence we have a number of different sets of weights to select from when restoring the image. In the worst case, completely random spatial variance, there are NM unique sets of weights to restore the image. However, if patterns of spatial variance can be discovered, the number of unique sets of weights can be significantly reduced. We will analyse the cyclic spatially variant case in detail. Consider a cyclic spatially variant distortion obtained by the use of V PSFs $h_0(x, y), ..., h_{V-1}(x, y)$. The pixels in any one row of the image are acted upon by the same PSF, however the PSF applied to each row is varied cyclically through the sequence:

$$S_H = \{h_0(x, y), h_1(x, y), ..., h_{V-1}(x, y), h_{V-2}(x, y), h_{V-3}(x, y), ..., h_1(x, y)\} \quad (3.15)$$

The sequence S_H has a period of $2V$-2, and hence $2V$-2 unique sets of weights are required to restore the image. This type of distortion is similar to the degradations involved with side-scan sonar images. The method employed in [39] to handle spatially variant distortion was to precompute a number of sets of weights to handle the various PSFs degrading the image. Since the spatial variation of the PSFs was known, the correct set of weights could be chosen to restore the image accurately. Section 3.7 details some experiments performed to restore an image degraded by a cyclic distortion.

3.6 Implementation Considerations

Despite the increase in efficiency and speed of Algorithm 3.3 when compared to Algorithms 3.1 and 3.2, there are still a number of ways that the algorithm can be made more efficient. The *i*th row of the weighting matrix describes the interconnection strengths between neuron *i* and every other neuron in the network from the "viewpoint" of neuron *i*. The weighting matrix is *NM* by *NM*, which is clearly a prohibitively large amount of data which requires storage. However the mathematical discussion in the previous sections implies a short cut.

By examining (2.5) we observe that in the case of $P \ll \min(M, N)$, it can be seen that when calculating the input to each neuron, only pixels within a certain rectangular neighbourhood of the current pixel contribute non-zero components to the neuron input. In addition it can be seen that the non-zero interconnection strengths between any given pixel and a neighbouring pixel depend only on the position of the pixels relative to each other in the case of spatially invariant distortion. Using the above observations, the input to any neuron (pixel) in the image can be calculated by applying a mask to the image centred on the pixel being examined. The case of spatially variant distortion requires more weighting masks to be created, however in the analysis above each unique set of weights requires only one weighting mask to describe it. For a *P* by *P* distortion, each weighting mask contains only $(2P - 1)^2$ terms. A 5 by 5 degrading PSF acting on a 250 by 250 image requires a weight matrix containing 3.9×10^9 elements, yet a weighting mask of only 81 elements. In addition, by considering the finite regions of support of the degrading and filtering impulse functions represented by $\boldsymbol{H}$ and $\boldsymbol{D}$, the weighting masks and bias inputs to each neuron may be calculated without storing matrices $\boldsymbol{H}$ and $\boldsymbol{D}$ at all. They may be calculated using only the impulse responses of the above matrices.

Even in the case of an image suffering spatially variant distortion we can precompute all the relevant weighting masks for the problem and assign each pixel an optimal mask before restoration is commenced. An image suffering spatially variant distortion can in this way be restored using an adaptive approach with very little additional overhead compared to a spatially invariant approach.

3.7 Experimental Results

To test Algorithm 3.3, we structured a series of experiments. In the first experiment, the efficiency of Algorithms 3.1, 3.2, and 3.3 were compared to one another. The second experiment implements Algorithm 3.3 with an image degraded by space variant distortion. In the third experiment, a practical example of the use of this method will be given. In this section the images will be compared by measuring their Signal to Noise Ratios (SNR) and LSMSEs (where LSMSE is as defined in Chapter 2).

3.7.1 Experimental Setup

In experiments one and two, the images were blurred using a gaussian PSF with the impulse response:

$$h(x, y) = \frac{1}{2\pi\sigma_x\sigma_y}\exp\left[-\left(\frac{x^2}{2\sigma_x^2} + \frac{y^2}{2\sigma_y^2}\right)\right] \tag{3.16}$$

where σ_x and σ_y are the standard deviations of the PSF in the x and y directions respectively.

3.7.2 Efficiency

The time taken to restore an image was compared among Algorithms 3.1, 3.2, and 3.3. A degraded image was created by blurring a 256 by 256 image with a Gaussian blur of size 5 by 5 and standard deviation 2.0. Noise of variance 4.22 was added to the blurred image. Each algorithm was run until at least 85% of the pixels in the image no longer changed value or many iterations had passed with the same number of pixels changing value during each iteration. Algorithm 3.1 was stopped after the sixth iteration when no further improvement was possible, and took 6067 seconds to run on a SUN Ultra SPARC 1. Algorithm 3.2 was stopped after the 30th iteration with 89% of pixels having converged to their stable states and took 126 seconds to run. Algorithm 3.3 was stopped after the 18th iteration with 90% of pixels stable and took 78 seconds to run. Algorithm 3.3 is much faster than Algorithms 3.1 and 3.2, despite the fact that algorithms 3.1 and 3.3 approach the same local minimum and hence give the same results. The computation time of Algorithm 3.3 can be expected to increase linearily with the number of pixels in the image, as can the computation times of Algorithms 3.1 and 3.2. The single step neuron energy minimisation technique of Algorithm 3.3 provides its superior speed and this trend does hold for any size image. Various types of distortions and noise would not be expected to change the speed relationship between Algorithms 3.1, 3.2 and 3.3 for any given image. This is due to the fact that each algorithm was shown to converge to similar points in solution space and instead use different methods to reach this point.

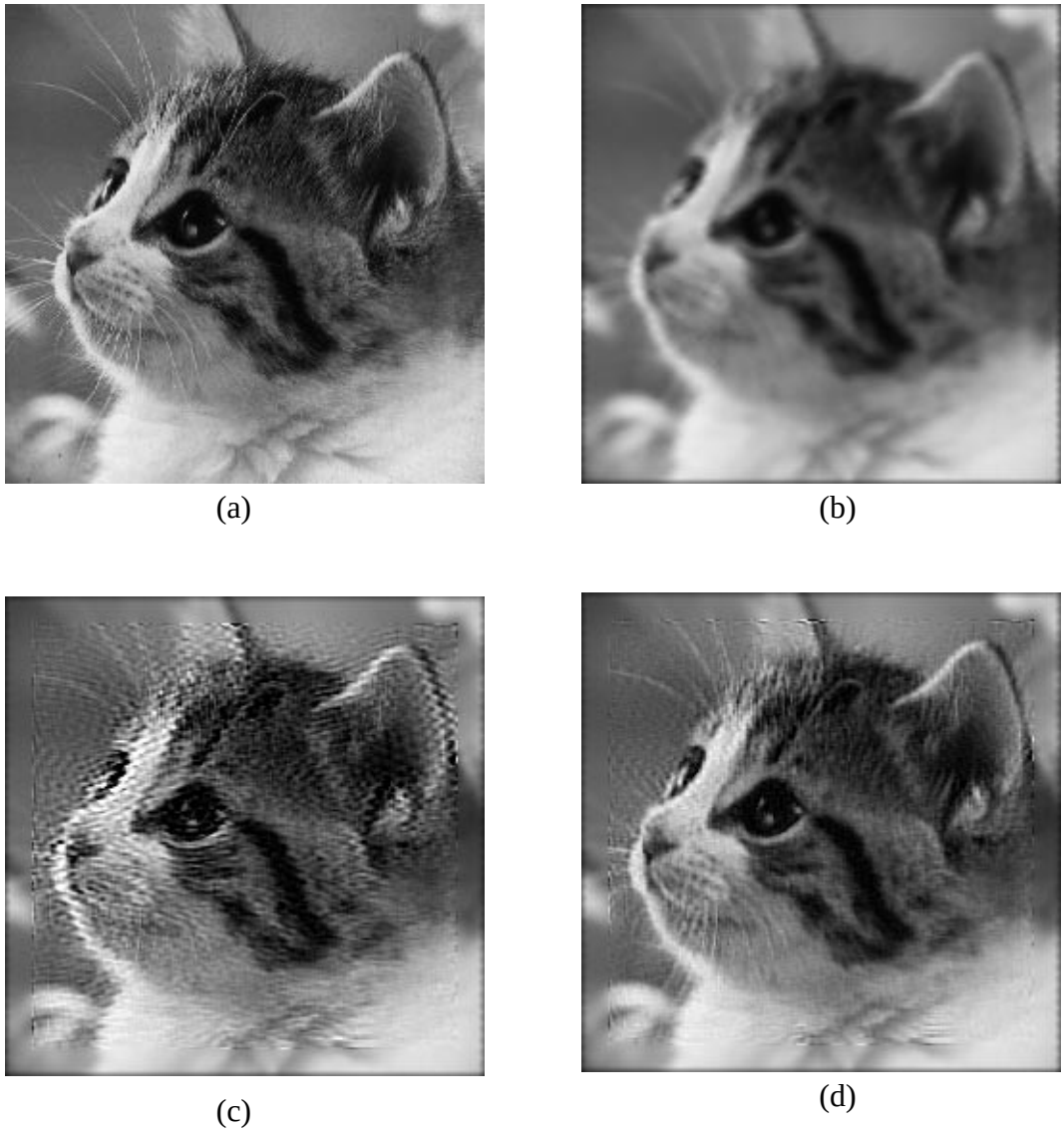
3.7.3 Spatially Variant Distortion

An image was created using a cyclic variation of 7 by 7 Gaussian PSFs. Using the analysis in Section 3.5, V was set to be 4. Table 3.1 details the degrading PSFs used to blur the image as per equation (3.15):

Table 3.1: Degrading PSFs.

Standard Deviation	1.5	2.0	3.0	4.0
PSF	$h_0(x, y)$	$h_1(x, y)$	$h_2(x, y)$	$h_3(x, y)$

The original image is shown in Figure 3.1a and the degraded image is shown in Figure 3.1b. This image was restored using two techniques. The first technique was a spatially invariant approach. The spatially variant distortion was approximated as a spatially invariant distortion by using a 7 by 7 gaussian PSF of standard deviation 2.55. That is, all pixels were acted upon by one weighting mask, whose components were calculated by approximating the space variant distortion as a space invariant distortion. Since the space variant distortion was very severe, the approximation caused instability in the restored image. The time to restore this image using the spatially invariant approximation was 230 seconds on a SUN Ultra SPARC 1. This image is shown in Figure 3.1c. The second technique was the proposed spatially variant approach with the six correctly calculated weighting masks. This restoration not only is better than Figure 3.1c, but recovers almost all the fine details lost in distortion. The time to restore this image using the spatially variant method was 314 seconds on a SUN Ultra SPARC 1. This image is shown in Figure 3.1d. Note that using Algorithm 3.3, the image could be restored using the correct spatially variant weights with only an extra 36% time penalty when compared to the spatially invariant approximation.



(a) (b)

(c) (d)

Figure 3.1: Original, degraded and restored images suffering spatially variant distortion.

3.7.4 An Application Example

Algorithm 3.3 was applied to the problem of restoring an image with an unknown level of motion blur. Restoring an image distorted by motion blur is a very common problem.

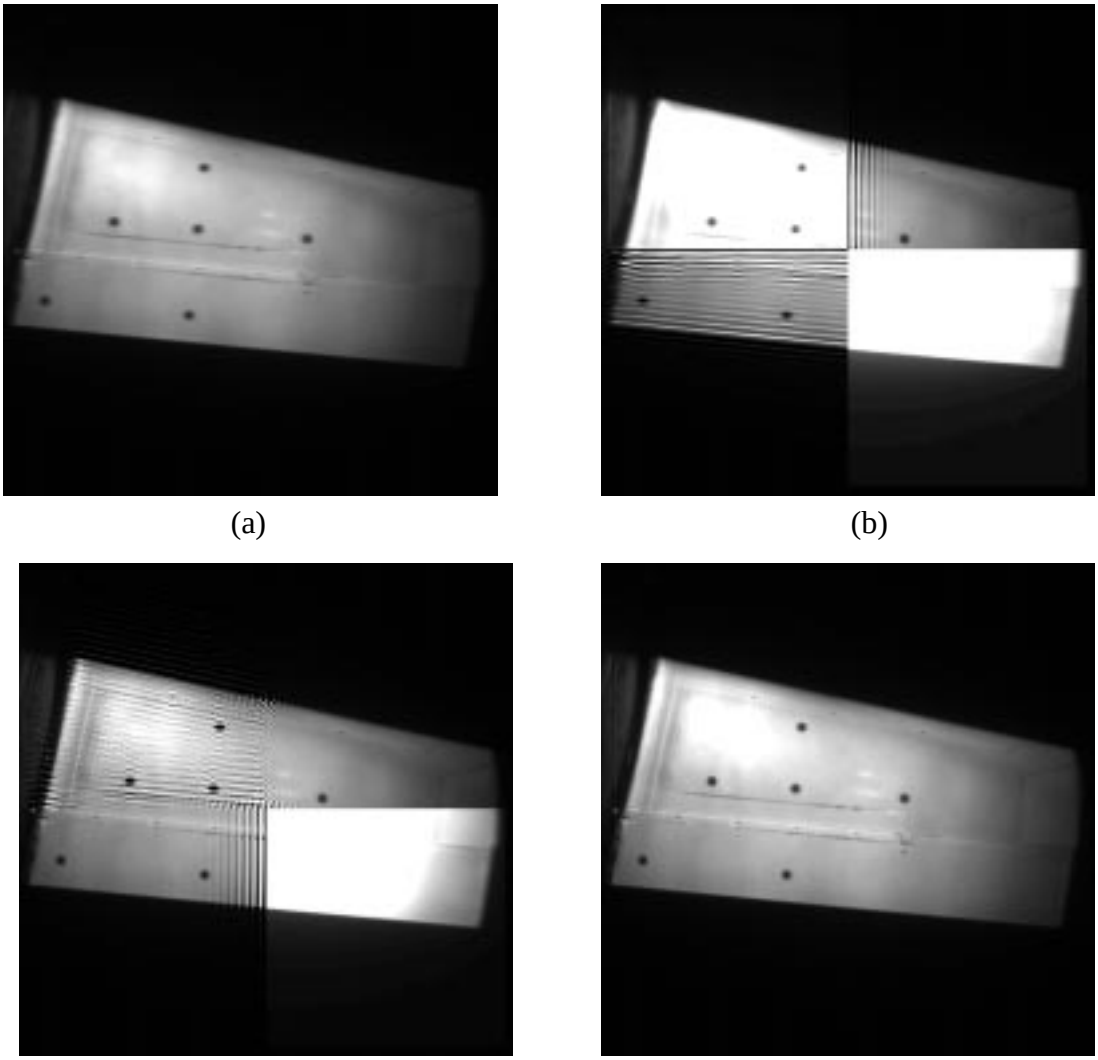
An image was supplied to us showing an aircraft wing in a state of assembly. The camera

had shifted with an unknown degree of movement during some of the exposures thereby blurring the images. Figure 3.2a shows an image of the wing with camera movement.

Although the image appears to be blurred by motion from left to right, it was desired to confirm this by restoring the image using the network parameters derived from motion in horizontal, vertical and both diagonal directions. The neural network algorithm is perfectly suited to rapid investigations due to the fact that different regions of the image can be set to be restored using the parameters of different point spread functions. Figure 3.2b shows Figure 3.2a restored using both diagonal point spread functions (top left and bottom right of the image), a vertical motion blur point spread function (bottom left of the image), and a horizontal motion blur point spread function (top right of the image). Using the image partitioning ability of this algorithm, four exploratory restorations were performed simultaneously. As a result of this, Figure 3.2b demonstrates that the best matching point spread function was that associated with horizontal camera movement, since the top right section of the image is clearly the most well restored.

The second part of the investigation was to determine the whether the horizontal motion was from left to right, right to left or due to camera shake rather than motion blur (in which case the motion will be in both directions). Figure 3.2c shows Figure 3.2a restored using the parameters of four different point spread functions. The top-left of the image was restored using the parameters of a uniform point spread function (as a control). The bottom-left of the image was restored with a point spread function associated with camera shake. The top-right and bottom right regions were restored using the parameters of horizontal motion blur to the right and left respectively. Figure 3.2c shows that the image was degraded with horizontal motion blur to the right as this set of network parameters produce the sharpest restoration. Figure 3.2d shows the

final restored image using the optimal point spread function determined previously.



(a) (b)

(c) (d)

Figure 3.2: Degraded and restored wing images.

3.8 Summary

In this chapter, we have examined the basic neural network optimisation algorithm. We first looked at the problem of restoring an image acted upon by a degrading PSF in the presence of noise. The problem of reversing the degrading process and approximating the original image can be formulated as a minimisation process on a constrained least square error measure func-

tion. This error measure can be minimised by relating the terms of the error measure with the formula for the energy of a Hopfield-style neural network. By selecting suitable interconnection strengths (weights) and bias inputs for the neurons, we can equate the neural network energy function with the constrained least square error cost function. The problem then reduces to selecting the optimal algorithm to minimise the network energy function. We examined two such algorithms presented previously in the literature and considered the advantages and disadvantages of both. We then presented a novel algorithm which brings together features from both the previous algorithms. The new algorithm was shown to converge to a solution in a finite number of steps with much faster speed than either of the previous algorithms. We examined some interesting properties of the algorithm including the fact that the convergence of the algorithm was irrespective of whether the degradation was spatially invariant or spatially variant. The algorithm was expanded to the space variant case and some implementation considerations involved with using the algorithm in practice were mentioned. A number of experiments were performed. The experiments showed that the proposed algorithm is much faster than previous algorithms and in particular is able to accurately restore images degraded by space variant degradations without a substantial time penalty. Finally, a practical example of the use of the algorithm was given.

Chapter 4: Weight Assignment in Adaptive Image Restoration

4.1 Introduction

In the previous chapter, it was seen that as long as the weights of the network remain constant from iteration to iteration, Algorithm 3.3 will converge. This result was shown to be irrespective of the spatial variance of the weights and opens the door for adaptive restoration methods.

In adaptive restoration, the weights are varied to implement different values of regularisation parameter for different regions of the image. By doing this, we can adjust restoration parameters to suppress noise more greatly in regions where it is most noticable, and less so in regions where image sharpness is the dominant consideration.

There are many ways to vary the regularisation parameter spatially across the image. We first show that the proposed method based on gradient descent can only find sub-optimal solutions in this adaptive approach, and mathematically verify a fact, which has been observed in restoration practice, that this method would generally use a small value of regularisation parameter for textured regions, and a large value for smooth regions. Using this observation, we then introduce a regional processing approach based on local statistics. This approach also has some relationship to biological vision systems in that it emphases on edges. The algorithm which varies the neural weights to take account of a spatially variant PSF, as described in

Chapter 3, is expanded to combine the adaptive constraint concept developed. We then look at how adaptive constraint restoration can be used to compensate for incomplete knowledge of the degrading PSF or incomplete knowledge of the spatial variance of the degradation.

Section 4.2 introduces the adaptive constraint algorithms. Section 4.3 describes the problem of adaptive constraint restoration for spatially variant distortions. Section 4.4 considers the problem of semi-blind deconvolution. Section 4.5 details some implementation considerations, while Section 4.6 describes experiments performed to test the concepts presented in this chapter. Section 4.7 summarises this chapter.

4.2 Adaptive Constraint

The first adaptive weight methodology we will consider is the adaptation of the constraint factor to take account of the non-stationary nature of the image.

4.2.1 Motivation

Images are, by nature, ensembles of non-stationary processes. For this reason, solutions based on any stationary model can only produce a suboptimal restoration. An optimal restoration may be achieved by treating statistically dissimilar regions of an image with different restoration strategies or parameters. One method to achieve this is by using an adaptive regularisation scheme. When implementing an adaptive regularisation parameter scheme, an important consideration is on what basis should the regularisation value be determined. In this section, we study the space invariant case. The result will be generalised to the space variant case in the next section.

To substantiate this issue, we first generalise the quadratic model in (2.3) to

$$E = \frac{1}{2}\left\|\boldsymbol{g} - \boldsymbol{H}\hat{\boldsymbol{f}}\right\|^2 + \frac{1}{2}\left\|\sqrt{\Lambda}\boldsymbol{D}\hat{\boldsymbol{f}}\right\|^2 \tag{4.1}$$

where

$$\sqrt{\Lambda} = \begin{bmatrix} \sqrt{\lambda_1} & 0 & 0 \\ 0 & \dots & 0 \\ 0 & 0 & \sqrt{\lambda_{NM}} \end{bmatrix} \tag{4.2}$$

is a diagonal matrix to reflect the adaptive processing nature. When $\lambda_1 = \lambda_2 = \dots = \lambda_{NM}$, Equation (4.1) is reduced to the conventional constrained least squares formula given in (2.3).

By relating (4.1) to the formula for the energy of a neural network, the neural weight between neurons *i* and *j* in the network is given by:

$$w_{ij} = -\sum_{p=1}^{L} h_{pi}h_{pj} - \sum_{p=1}^{L} \lambda_p d_{pi}d_{pj} \tag{4.3}$$

Equation (4.3) is overly complicated and can be approximated as:

$$w_{ij} = -\sum_{p=1}^{L} h_{pi}h_{pj} - \lambda_i \sum_{p=1}^{L} d_{pi}d_{pj} \tag{4.4}$$

Note that the matrix $\boldsymbol{W}$, whose (i, j)th element is given by (4.4), is not symmetric. As long as a check is made on whether the energy will be decreased before updating a pixel value, the lack of symmetry will not cause the algorithm to fail to converge. This is due to the fact that $\boldsymbol{W}$ may be replaced by a symmetric equivalent version, $\boldsymbol{W}^*$, without changing the energy function.

Define $\boldsymbol{W}^* = \frac{(\boldsymbol{W} + \boldsymbol{W}^T)}{2}$, where $\boldsymbol{W}$ is the non-symmetric weighting matrix whose (i, j) th element is given by (4.4). Note $\boldsymbol{W}^*$ is symmetric since:

$$(\boldsymbol{W}^*)^T = \frac{(\boldsymbol{W} + \boldsymbol{W}^T)^T}{2} = \frac{(\boldsymbol{W}^T + \boldsymbol{W})}{2} = \boldsymbol{W}^*$$

Taking the formula for the energy of the neural network, and noting that $a^T = a$ when a is a scalar, then we obtain:

$$E = -\frac{1}{2}\hat{\boldsymbol{f}}^T \boldsymbol{W}^* \hat{\boldsymbol{f}} - \boldsymbol{b}^T \hat{\boldsymbol{f}} + c$$

$$= -\frac{1}{2}\hat{\boldsymbol{f}}^T \left(\frac{\boldsymbol{W} + \boldsymbol{W}^T}{2}\right) \hat{\boldsymbol{f}} - \boldsymbol{b}^T \hat{\boldsymbol{f}} + c$$

$$= -\frac{1}{4}\hat{\boldsymbol{f}}^T \boldsymbol{W} \hat{\boldsymbol{f}} - \frac{1}{4}\hat{\boldsymbol{f}}^T \boldsymbol{W}^T \hat{\boldsymbol{f}} - \boldsymbol{b}^T \hat{\boldsymbol{f}} + c$$

$$= -\frac{1}{4}\hat{\boldsymbol{f}}^{T}\boldsymbol{W}\hat{\boldsymbol{f}} - \left(\frac{1}{4}\hat{\boldsymbol{f}}^{T}\boldsymbol{W}\hat{\boldsymbol{f}}\right)^{T} - \boldsymbol{b}^{T}\hat{\boldsymbol{f}} + c = -\frac{1}{2}\hat{\boldsymbol{f}}^{T}\boldsymbol{W}\hat{\boldsymbol{f}} - \boldsymbol{b}^{T}\hat{\boldsymbol{f}} + c \tag{4.5}$$

From (4.5) we can see that the form of the energy functions given by using $\boldsymbol{W}$ and $\boldsymbol{W}^{*}$ are identical. This means that if we assign neuron weights according to (4.4), the algorithm will still converge.

The elements of $\boldsymbol{W}^{*}$ can be considered an alternative approximation of (4.3) and are given by:

$$w_{ij} = -\sum_{p=1}^{L} h_{pi}h_{pj} - \left(\frac{\lambda_i + \lambda_j}{2}\right)\sum_{p=1}^{L} d_{pi}d_{pj} \tag{4.6}$$

However since we have shown that both approximations produce identical energy functions, we will hence use the approximation given by (4.4). In Appendix B we show that approximating (4.3) as (4.4) does not greatly degrade the quality of the restored images. Since approximation (4.4) is much easier to implement than (4.3), this approximation will be used in the subsequent sections. The next question is how to determine the values of λ_i in (4.4) for each pixel in the image.

We investigate two methods for determining the regularisation parameter. The first is an extension of the gradient based method considered in the last chapter. The second is based on local statistics in the framework of an intelligent neural network.

4.2.2 The Gradient Based Method

First we will show that although the gradient descent algorithm given in the previous chapter reduces the energy level, it by no means provides an optimal solution in adaptive restoration.

Equations (2.5) and (2.6) indicate that differences in the regularisation parameter effect only the weights of the neural network, not the initial values or the bias inputs.

Equation (3.4) gives the energy change resulting from a change of neuron state Δf_i. Substituting (3.2) into (3.4) yields:

$$\Delta E_i = -\left[\sum_{j=1}^{L} w_{ij}\hat{f}_j + b_i\right]\Delta\hat{f}_i - \frac{1}{2}w_{ii}(\Delta\hat{f}_i)^2 \qquad (4.7)$$

It should be noted that $\sum_{j=1}^{L} w_{ij}\hat{f}_j + b_i$ is the input to each neuron and has the same sign as $\Delta\hat{f}_i$ if ΔE is to be negative. It is important to note that we are only considering the cases where ΔE is negative since when $\Delta E \geq 0$, there will be no pixel value update at pixel $\hat{f}_i$.

Definition 4.1: *The greatest energy minimisation (GEM) method for selecting the constraint value for each pixel is defined as choosing for each pixel in each iteration the value of* λ *from a range of allowable* λ *values,* $\lambda_a \leq \lambda \leq \lambda_b$, $\langle \lambda_a, \lambda_b \geq 0 \rangle \in \Re$, *which best minimises (4.7).*

Based on the definitions and other aforementioned assumptions, we present two theorems.

Theorem 4.1: *If the GEM method is used to select a suitable* λ *value from a range of possible* λ *values, where* $\lambda_a = 0$ *and* $\lambda_b = \infty$ *in Definition 4.1, then* λ_b *will always be chosen.*

Theorem 4.2: *If the GEM method is used to select a suitable* λ *value, satisfying* $\lambda_a \leq \lambda \leq \lambda_b$ *where both* λ_a *and* λ_b *are finite, then either* λ_a *or* λ_b *will be chosen unless all available* λ *values would produce the same resultant decrease in energy.*

To prove the above theorems, we must rearrange (4.7). Expanding (4.7) using (2.5) we get:

$$\begin{aligned}\Delta E_i &= \left[\sum_{j=1}^{L}\sum_{p=1}^{L} h_{pi}h_{pj}\hat{f}_j - b_i\right]\Delta\hat{f}_i \\ &+ \lambda_i\Delta\hat{f}_i\sum_{j=1}^{L}\sum_{p=1}^{L} d_{pi}d_{pj}\hat{f}_j \\ &+ \frac{1}{2}(\Delta\hat{f}_i)^2\sum_{p=1}^{L} h_{pi}^2 + \frac{1}{2}\lambda_i(\Delta\hat{f}_i)^2\sum_{p=1}^{L} d_{pi}^2\end{aligned}$$

$$= A\Delta\hat{f}_i + \lambda_i B\Delta\hat{f}_i + C(\Delta\hat{f}_i)^2 + D\lambda_i(\Delta\hat{f}_i)^2 \qquad (4.8)$$

where $A = \sum_{j=1}^{L}\sum_{p=1}^{L} h_{pi}h_{pj}\hat{f}_j - b_i$, $B = \sum_{j=1}^{L}\sum_{p=1}^{L} d_{pi}d_{pj}\hat{f}_j$, $C = \frac{1}{2}\sum_{p=1}^{L} h_{pi}^2$, and

$D = \frac{1}{2}\sum_{p=1}^{L} d_{pi}^2$.

It should be noted that $u_i = -A - \lambda B$ and $C + \lambda D = -\frac{w_{ii}}{2}$ (4.9)

Consider the $\Delta \hat{f}_i$ which maximises the energy reduction for pixel $\hat{f}_i$. Let this factor be defined as $\Delta \hat{f}_i'$. To compute $\Delta \hat{f}_i'$, we differentiate (4.8) relative to $\Delta \hat{f}_i$:

$$\frac{\partial}{\partial \Delta f_i} \Delta E_i = A + \lambda_i B + 2C\Delta \hat{f}_i + 2D\lambda_i \Delta \hat{f} \tag{4.10}$$

and set (4.10) to zero to obtain the optimal $\Delta \hat{f}_i'$.

$$\Delta \hat{f}_i' = \frac{-(A + \lambda_i B)}{2(C + \lambda_i D)} \tag{4.11}$$

Rearrangement of (4.8) yields:

$$\Delta E_i = (A + \lambda_i B)\Delta \hat{f}_i + (C + \lambda_i D)(\Delta \hat{f}_i)^2 \tag{4.12}$$

The substitution of (4.11) into (4.12) gives us:

$$\Delta E_i' = \frac{-(A + \lambda_i B)^2}{2(C + \lambda_i D)} + \frac{(A + \lambda_i B)^2}{4(C + \lambda_i D)} = \frac{-(A + \lambda_i B)^2}{4(C + \lambda_i D)} \tag{4.13}$$

We can confirm from (4.11) that $\Delta \hat{f}_i'$ always has the same sign as u_i and that $\Delta E_i'$ is always

negative for positive values of λ, as expected. The graph of $\Delta E_i'$ versus λ_i is sketched in Fig 4.1.

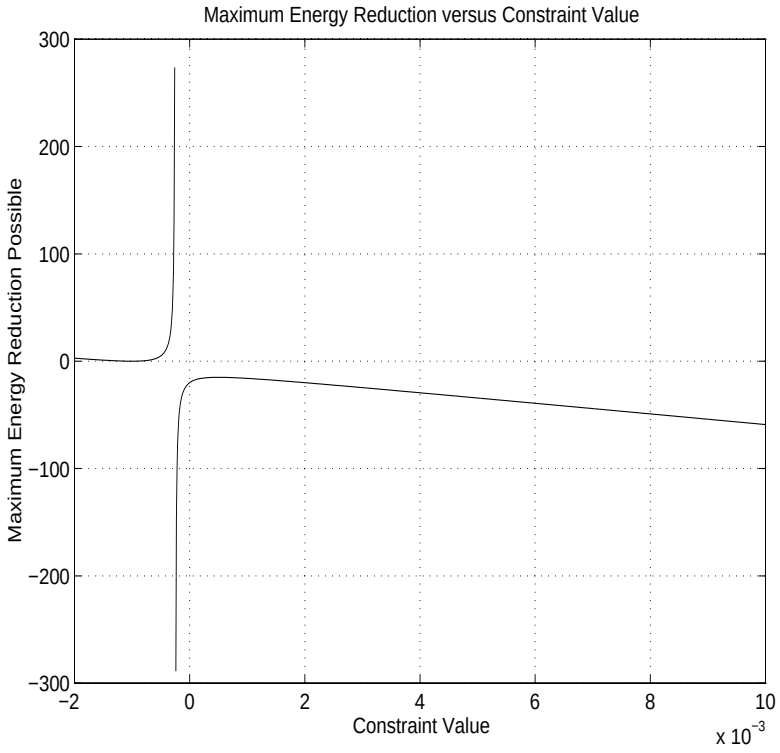


Figure 4.1: Graph of maximum energy reduction versus constraint value.

We can see from (4.13) that for all acceptable values of ΔE_i, $\lim_{\lambda_i \to \infty} \Delta E_i' = -\infty$. This proves Theorem 4.1 and implies that a great energy reduction can always be obtained by choosing a very large value of λ_i. If we allow energy considerations alone to dictate the choice of λ_i, the results would be poor. An acceptable range of λ values must be set. Since *C* and *D* are always positive, and $\Delta E_i'$ is always negative, (4.13) has a maximum at $\lambda_i = -\frac{A}{B}$.

Since (4.13) is a convex function when $\Delta E_i' < 0$, then we observe that if the set of acceptable λ values is finite and constrained between λ_a and λ_b ($\lambda_a < \lambda_b$), then (4.13) indicates that under most circumstances, the best energy reduction will be obtained by choosing either λ_a or

λ_b, and not any intermediate values of λ. Only under the conditions of $\frac{-A^2}{4C} \approx \frac{-B^2}{4D} \approx 0$ will numerical errors possibly enable the choice of intermediate values of λ. This proves Theorem 4.2.

The next question is which λ values will be chosen in low and high variance regions of the image to maximise the energy reduction. To clearly answer this question, it is necessary to examine how the factors *A, B, C,* and *D* vary as a function of image statistics. This is examined in Appendix C. From this analysis, a number of conclusions can be drawn. When *A* is low and *B* is high, the largest λ_i value gives the best energy minimisation, however when *A* is high and *B* is low, the smallest value of λ_i produces the best energy minimisation.

Observation 1: In high texture regions of the image or near edges, both *A* and *B* can be expected to be large. However the contribution from factor *B* will be less significant for two reasons.

- The factor *B* in (4.13) is multiplied by λ_i which is always << 1.

- The factor *B* corresponding to a double application of a high-pass filter suggests that in the presence of gradual edges, *B* will not be very large. Since noise is added to the image after blurring, the presence of noise will contribute most to the value of *B*.

Therefore we can expect that $\frac{-A^2}{4C} < \frac{-\lambda_b B^2}{4D}$, and so the best energy reduction would be obtained by choosing the lowest value of λ available.

Observation 2: In low texture regions of the image, *A* can be expected to be small. Factor *B* may also be small. However the presence of noise in the initial estimate should make *B*'s effect on (4.13) override that of *A*. Hence in low texture regions of the image, choosing the largest available λ_i produces the best energy minimisation.

This analysis has been verified in practice. Figure 4.2 shows the selection of λ values during the first iteration for an image degraded by a 5 by 5 Gaussian PSF of standard deviation 2.0, with additive noise of variance 18.49. The darker regions in Figure 4.2 denote the selection of larger λ values, and the lighter regions denote the selection of smaller λ values.

Figure 4.2: The λ values selected for each pixel during the initial iteration of the energy minimisation based algorithm for a typical image.

From Figure 4.2 we can clearly see that to a great extent, only the lowest or the highest value of λ is chosen by the gradient descent method.

4.2.3 Local Statistics Analysis

The above analysis shows us that gradient descent is not an optimal approach for choosing λ_i in adaptive regularisation. However, it leads us to the observation that in low texture regions, a high value of λ_i results in a visually pleasing result, while in high texture regions, a low λ_i value works best. These observations encourage us to use local statistics, since these are a good measure of the image roughness to locally determine the λ_i value. Mathematically:

$$\lambda_i = Y(S_i) \tag{4.14}$$

where $Y(S_i)$ is a function of the local image statistics S_i at $\hat{f}_i$.

Since S_i has almost exactly the same value for all pixels, $\hat{f}_i$, in a statistically homogenous area, the λ_i value is also almost exactly the same. Therefore the structure of the processing model is further modified such that, instead of assigning each pixel a different λ_i, we assign each statistically homogenous area a λ_k.

Assume that there are K homogenous areas. By first properly rearranging the pixels in such a way that the pixels in a homogenous area are consecutively indexed in $\hat{\boldsymbol{f}}$ to form a new vector $\hat{\boldsymbol{f}}^*$, equation (4.1) can be rewritten as:

$$E = \frac{1}{2}\{\left\|\boldsymbol{g}^* - \boldsymbol{H}^* \hat{\boldsymbol{f}}^*\right\|^2 + \hat{\boldsymbol{f}}^* (\boldsymbol{D}^{*T} \Lambda^* \boldsymbol{D}^*) \hat{\boldsymbol{f}}^*\} \tag{4.15}$$

where $\hat{\boldsymbol{f}}^{*T} = \left[\boldsymbol{f}_1^T \ \ldots \ \boldsymbol{f}_K^T\right]$ with $\boldsymbol{f}_k$ being the vector consisting of the pixels in the kth homogenous area,

$$\boldsymbol{H}^* = \begin{bmatrix} \boldsymbol{H}_1 \\ \ldots \\ \boldsymbol{H}_K \end{bmatrix}, \Lambda^* = \begin{bmatrix} \Lambda_1 & & \\ & \ldots & \\ & & \Lambda_K \end{bmatrix}, \boldsymbol{D}^* = \begin{bmatrix} \boldsymbol{D}_1 \\ \ldots \\ \boldsymbol{D}_K \end{bmatrix}, \boldsymbol{g}^* = \begin{bmatrix} \boldsymbol{g}_1 \\ \ldots \\ \boldsymbol{g}_K \end{bmatrix} \tag{4.16}$$

with $\boldsymbol{H}_k$, Λ_k, $\boldsymbol{D}_k$ and $\boldsymbol{g}_k$ being the submatrices (vectors) of $\boldsymbol{H}^*$, Λ^*, $\boldsymbol{D}^*$ and $\boldsymbol{g}^*$, corresponding to $\boldsymbol{f}_k$, and $\Lambda_k = \lambda_k \boldsymbol{I}$, with $\boldsymbol{I}$ being the identity matrix. Define $\boldsymbol{H}_k = \left[\boldsymbol{H}_{k1} \ \ldots \ \boldsymbol{H}_{kK}\right]$ and $\boldsymbol{D}_k = \left[\boldsymbol{D}_{k1} \ \ldots \ \boldsymbol{D}_{kK}\right]$, $k = 1, 2, \ldots, K$, then mathematical manipulation of (4.15) leads to:

$$E = -\frac{1}{2}\sum_{k=1}^{K} \{[\hat{\boldsymbol{f}}^{*T}(\boldsymbol{H}_k^T\boldsymbol{H}_k + \lambda_k\boldsymbol{D}_k^T\boldsymbol{D}_k)\hat{\boldsymbol{f}}^* + 2\boldsymbol{g}_k^T\boldsymbol{H}_k\hat{\boldsymbol{f}}^*]\} + \|\boldsymbol{g}^*\|^2$$

$$= \frac{1}{2}\sum_{k=1}^{K}\left\{\boldsymbol{f}_k^T(\boldsymbol{H}_{kk}^T\boldsymbol{H}_{kk} + \lambda_k\boldsymbol{D}_{kk}^T\boldsymbol{D}_{kk})\boldsymbol{f}_k + \sum_{\substack{l=1 \\ l\neq k}}^{K}\sum_{\substack{m=1 \\ m\neq k}}^{K}\boldsymbol{f}_l^T(\boldsymbol{H}_{kl}^T\boldsymbol{H}_{km} + \lambda_k\boldsymbol{D}_{kl}^T\boldsymbol{D}_{km})\boldsymbol{f}_m + 2\boldsymbol{g}_k^T\boldsymbol{H}_k\hat{\boldsymbol{f}}^*\right\} + \|g^*\|^2 \tag{4.17}$$

Apparently, $\boldsymbol{H}_{kk}^T\boldsymbol{H}_{kk} + \lambda_k\boldsymbol{D}_{kk}^T\boldsymbol{D}_{kk}$ represents the intra-connections within area k, and $\boldsymbol{H}_{kl}^T\boldsymbol{H}_{km} + \lambda_k\boldsymbol{D}_{kl}^T\boldsymbol{D}_{km}$ represents the inter-area contributions from areas l and m to area k. Equation (4.17) is the extension of a biologically motivated neural network: the network of networks (NoN) [102]. The significance of this mapping is as follows:

1. The human visual system pays little attention to individual pixels in an image. Instead it looks for areas of similar pixels in a statistically homogenous sense, and it is more sensitive to edges [83].

2. The NoN is a computational model imitating a simplified representation of the human cortex, or part of it, the biological visual processing machine. The neurons in the same cluster are similar to one another.

3. By representing a pixel with a neuron, a homogenous image area is mapped to a cluster of neurons which are similar to one another. Local statistics is a good criterion to measure the level of similarity.

Therefore, using a NoN with a statistical criterion to adaptively determine the λ value and in turn the processing architecture may potentially simulate some aspects of human perception in recovering genuine information lost in recording.

Now the important issue is selecting the λ_i for each homogenous area. In this thesis, a curve fitting approach is proposed. In this approach, the largest and the smallest values of S_i, S_{imax} and S_{imin}, are identified and the corresponding λ_{imax} and λ_{imin} are determined.

Experiments quickly led to the conclusion that the explicit form of (4.14) is a log-linear function:

$$\lambda_i = a\log(S_i) + b \tag{4.18}$$

It was found that if the variance in a region gradually increases by equal steps, then the change in variance level is much more noticable when the overall variance levels are low rather than when the overall variance levels are high. Humans are less able to discern an increase in noise levels in high variance regions than they are for low variance regions. A log-linear relationship is therefore suggested. Figure 4.3 shows this effect.

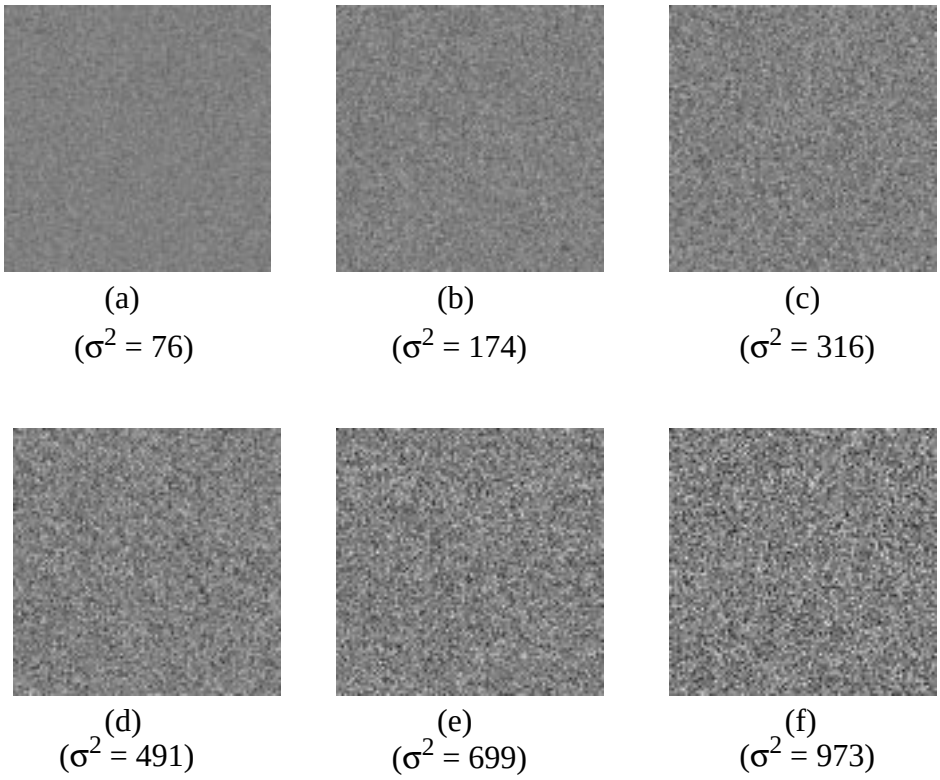


Figure 4.3: Images with varying levels of additive noise.

In Figure 4.3, six images with a constant level of pixel intensity are superimposed by noise of increasing variance. The increasing level of noise between each image and its neighbour on the right is readily noticable for the first three images when variance levels are low, compared to the last three when variance levels are high. The difference in apparent noise level between the last two images is the weakest of the entire set of images, despite the fact that the increase in variance between these two images is the greatest of the set.

The above observations lend weight to the use of constant λ values for each iteration based on those during the first iteration. Since the λ selected during the first iteration can often produce visually pleasing results (large λ in the background and small λ on the edges), these can be held constant throughout the restoration procedure. The λ values may not act as favourably for later iterations in the algorithm. For example, the local variance levels in the areas surrounding edges in the image may increase during the restoration due to ripple effects from nearby edges such that they become large enough to cause a large value of λ to be applied to edges of the image, hence producing the opposite effect to that which we are attempting to achieve.

4.3 Dealing with Spatially Variant Distortion

In the previous section, the weights of the neural network were varied spatially in order to implement the adaptive regularisation parameter. A similar concept was considered in Chapter 3 to handle a simple form of spatially variant distortion. In Chapter 3, the author used multiple sets of weights to successfully restore an image degraded by a cyclic variation of gaussian PSFs, of different standard deviations. In this section, we will examine the integration of the spatially variant restoration technique of Chapter 3 with the adaptive constraint technique.

The method employed in Chapter 3 to handle spatially variant distortion was to precompute a number of sets of weights to handle the various PSFs degrading the image. Since the spatial variation of the PSFs was known, the correct set of weights could be chosen to restore the image accurately. However a fixed λ was used in Chapter 3. To implement a spatially variant restoration technique with an adaptive constraint parameter, we produce as many sets of weights as needed to handle the space variant degradation and every possible choice of regular-

isation parameter.

Re-stating the discussion in Chapter 3, the cyclic spatially variant distortion we consider is that obtained by the use of V PSFs $h_0(x, y)$, ..., $h_{V-1}(x, y)$. The pixels in any one row of the image are acted upon by the same PSF, however the PSF applied to each row is varied cyclically through the sequence:

$$S_H = \{h_0(x, y), h_1(x, y), ..., h_{V-1}(x, y), h_{V-2}(x, y), h_{V-3}(x, y), ..., h_1(x, y)\} \quad (4.19)$$

The sequence S_H has a period of $2V$-2, and hence $2V$-2 unique sets of weights are required to restore the image when the same regularisation parameter is chosen for every pixel. When additional sets of weights are created to implement an adaptive regularisation parameter scheme, the following analysis applies:

Taking R as the number of choices of regularisation parameter to be used in the restoration, then the sets of weights to be used to restore a row blurred by the ith element of sequence S_H form the set:

$$\boldsymbol{WSM_i} = \{\boldsymbol{Wm}_0^i, \ldots, \boldsymbol{Wm}_{R-1}^i\} \quad (4.20)$$

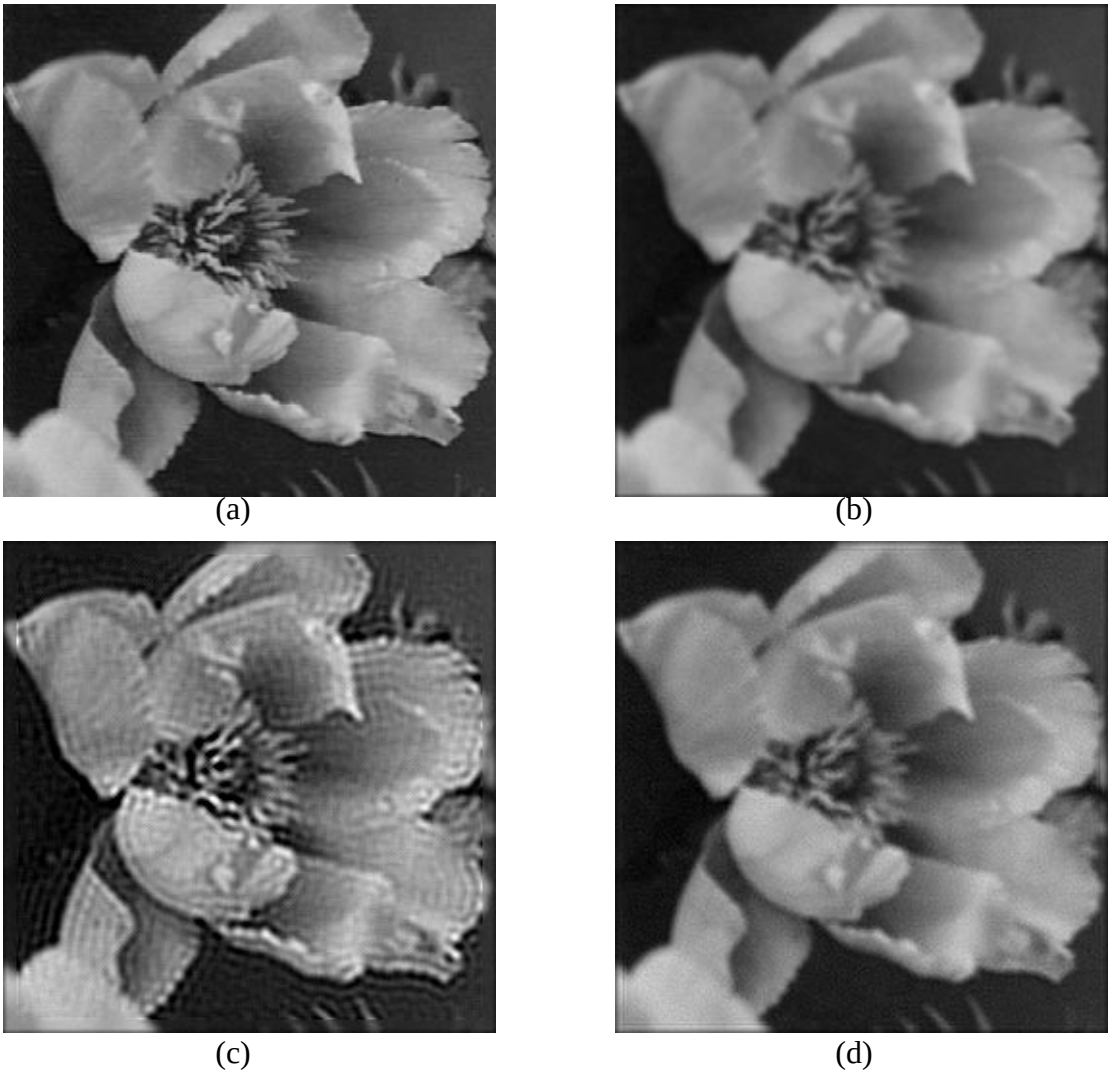
where $\boldsymbol{Wm}_j^i$ is the set of weights required to restore a row degraded by PSF $[S_H]_i$ using the jth choice of regularisation parameter being considered. The restoration problem becomes a problem of selecting the correct set of weights from the super set of:

$$\boldsymbol{WMA} = \{\boldsymbol{WSM}_0, \ldots, \boldsymbol{WSM}_{2V-3}\} \tag{4.21}$$

Restoration of the image is accomplished by selecting the relevant set $\boldsymbol{WSM}_i$, based on the row coordinate of the current pixel being examined, then selecting the optimal $\boldsymbol{Wm}_j^i$ within that set based on the selection schemes described in Section 4.2. Section 4.6 describes an experiment preformed to implement the spatially variant restoration scheme with an adaptive regularisation parameter as described above.

4.4 Semi-Blind Deconvolution

The previous sections have assumed that the nature of the degrading PSF is known. In practical circumstances however, this may not be the case. We may have little or no knowledge of the degrading PSF in the case of a spatially invariant distortion or may not know the exact nature of the space variance in a case of spatially variant distortion. In addition, our estimate of the degrading PSF may be corrupted by noise. Restoring an image with incomplete knowledge of the nature of the degradation is called *semi-blind deconvolution*. Figure 4.4 shows the effects of using an incorrect PSF estimate during the restoration procedure.



(a) (b)

(c) (d)

Figure 4.4: Degraded image restored with various PSF estimates.

Figure 4.4a is the original flower image. Figure 4.4b is the flower image degraded by a 5 by 5 Gaussian blur of standard deviation 2.0 with additive noise of variance 4.22. Figure 4.4c shows the image restored using the parameters of a 7 by 7 Gaussian blur of standard deviation 3.0 and Figure 4.4d shows the image restored using the parameters of a 3 by 3 Gaussian blur of standard deviation 1.0. In Figure 4.4c, the degrading PSF estimate is too strong and so the image can be said to have been “over restored”. Although the edges and high variance regions in Figure 4.4c appear sharp, ringing effects are apparent in the smooth (low variance) regions of the

image. In Figure 4.4d, on the other hand, the degrading PSF estimate is too weak and so the image can be said to have been “under restored”. In the smooth (low variance) regions of Figure 4.4d, we see no artifacts, however the edges and high variance regions of Figure 4.4d appear blurry.

This would indicate that a too strong PSF estimate produces problems in low variance regions whereas a too weak PSF estimate produces problems in high variance regions. Figure 4.4d appears similar to the result obtained when a large value of λ is used during the restoration process. In fact, since λ controls the degree of smoothness in the solution, the effect of increasing λ is similar to performing the restoration with a weaker PSF estimate. In the adaptive λ image restoration method described in the previous sections, a small value of λ is used in high variance regions and a larger value of λ is used in low variance regions. This implies that the adaptive constraint restoration method may have an application in semi-blind image restoration. An example should illustrate this.

Assume that an image has suffered a spatially variant distortion and only an estimate of the average PSF is available. By using the adaptive constraint restoration method we can consider that each value of λ approximates a different strength PSF estimate and hence we effectively have a range of available PSF estimates to chose from. In the high variance regions of the image, the λ value is small, this is similar to selecting the strongest PSF estimate. This is the preferable choice since over-restoring the high variance regions of the image is less visually disturbing than under-restoring them. To reduce the effects of ripples in the low variance regions of the image a weak PSF estimate is preferable. This is automatically achieved since the adaptive constraint restoration method selects the largest value of λ for low variance regions. As mentioned above, larger values of λ produce results similar to using weaker PSF

estimates. From this we can see that the adaptive λ restoration method can be used to compensate for inaccuracies in the estimate of degrading PSF. This concept will be further investigated in Section 4.6.

4.5 Implementation Considerations

A major practical consideration is the number of choices of regularisation value. One would expect that a large value of R will give a better restoration quality, however large values of R would slow down the constraint precomputing stage of the algorithm and waste memory. It is desirable to have as low a value of R as possible. In practice setting $R = 3$ is usually sufficient, giving the algorithm the choice of doing nothing or selecting one of three constraint values for regions for high, medium or low texture levels.

In this investigation, we associated each set of weights with a range of variance values computed in a certain neighbourhood of size A by A of the current pixel. A variance threshold, κ, was set, below which the pixel being examined would not be updated. This is due to the fact that for extremely low variance regions of an image blurring may not be noticable, in this case restoration can only serve to enhance noise and waste time. This technique can yield improved results, however the level of variance threshold and whether or not a variance threshold is used at all depend on the image being restored and the degrading function. The variance method has the advantage that it does not require the current pixel to be acted upon by each possible set of weights to compute the regularisation parameter required. Another advantage to the variance method is the ability to fine tune the variance thresholds, κ, or A, the area size, to suit a particular type of image being examined.

When we consider using variance to determine the regularisation parameter, we expect that precomputation of the parameter, based on the degraded image statistics would produce similar results to computing the parameter in each iteration. High variance regions in the degraded image should remain high variance in the restored image, and low variance regions should likewise remain low variance, hence by this argument the chosen values of regularisation parameter in the first iteration should remain approximately the same throughout the restoration procedure. This also has the added advantage that the algorithm is guaranteed to converge according to the analysis in Chapter 3.

In practice, precalculating the optimal set of weights for each pixel has further advantages: During restoration, the image estimate may converge on the solution smoothly or with varying degrees of oscillation around the final value. During an oscillatory restoration, the image statistics may change in unpredictable ways, causing the regularisation parameter chosen from iteration to iteration to vary also. This may not result in an optimal choice of regularisation value during any one iteration or an optimal average value of regularisation parameter.

Precomputation of the regularisation parameter results in a faster restoration than that obtained by computation of the regularisation parameter during each iteration. In fact, by precomputing the regularisation values for each pixel, the adaptive constraint algorithm takes only slightly more time than a non-adaptive algorithm for large images, as will be shown in the next section.

4.6 Experimental Results

To test the adaptive regularisation parameter algorithms presented in this chapter, we struc-

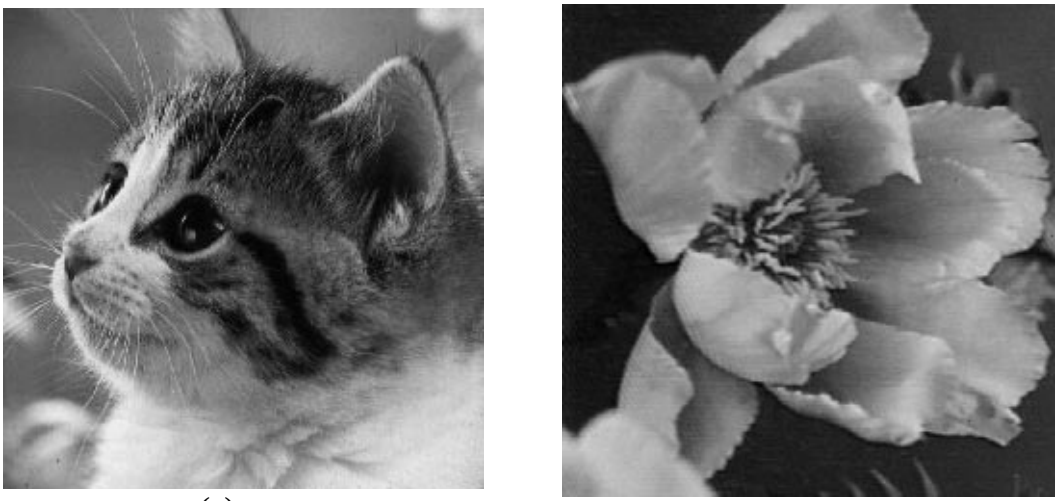
tured a series of experiments. The first experiment examined the effects of noise on the performance of the algorithm. The second experiment implemented the adaptive constraint algorithm with an image degraded by space variant distortion. The second experiment also examines the problem of semi-blind deconvolution. The third experiment examines the processing efficiency. In the fourth experiment a practical example of the use of this method will be given. In this section the images will be compared by measuring their Signal to Noise Ratios (SNR) and LSMSEs (LSMSE is as defined in Chapter 2).

4.6.1 Experimental Setup

In experiment one, the images in Figure 4.5 were blurred using a gaussian PSF with the impulse response:

$$h(x, y) = \frac{1}{2\pi\sigma_x\sigma_y}\exp\left[-\left(\frac{x^2}{2\sigma_x^2} + \frac{y^2}{2\sigma_y^2}\right)\right] \tag{4.22}$$

where σ_x and σ_y are the standard deviations of the PSF in the x and y directions respectively.



(a) (b)

Figure 4.5: Original images.

The number of choices of regularisation parameter, R, was set to 5. The following constraint matrix, $d(x, y)$, was used:

$$\begin{bmatrix} 1.0/6.0 & 1.0/6.0 & 1.0/6.0 & 1.0/6.0 & 1.0/6.0 \\ 1.0/6.0 & 4.0/6.0 & -3.0/6.0 & 4.0/6.0 & 1.0/6.0 \\ 1.0/6.0 & -3.0/6.0 & -20.0/6.0 & -3.0/6.0 & 1.0/6.0 \\ 1.0/6.0 & 4.0/6.0 & -3.0/6.0 & 4.0/6.0 & 1.0/6.0 \\ 1.0/6.0 & 1.0/6.0 & 1.0/6.0 & 1.0/6.0 & 1.0/6.0 \end{bmatrix}$$

4.6.2 Effects of Noise

Two different images were used to compare the various algorithms. Each image was degraded by the same PSF. For this experiment, a PSF of size $P = 5$ was used with $\sigma_x = \sigma_y = 2.0$. Various levels of white noise, with variances approximately equal to 4 and 18, were added to each degraded image. This resulted in 8 images to analyse.

Table 4.1 shows the SNR and LSMSE between the original and the degraded images, and images restored with a constant value of λ, the greatest energy reduction technique for selecting λ, the adaptive constraint technique of Kang and Katsaggelos [49], and with an adaptive λ value based on local image variance and (4.18). Figure 4.6 shows regularisation parameters used in the experiment plotted against the level of local variance. The values of λ were associated with variance thresholds in a way consistent with (4.18).

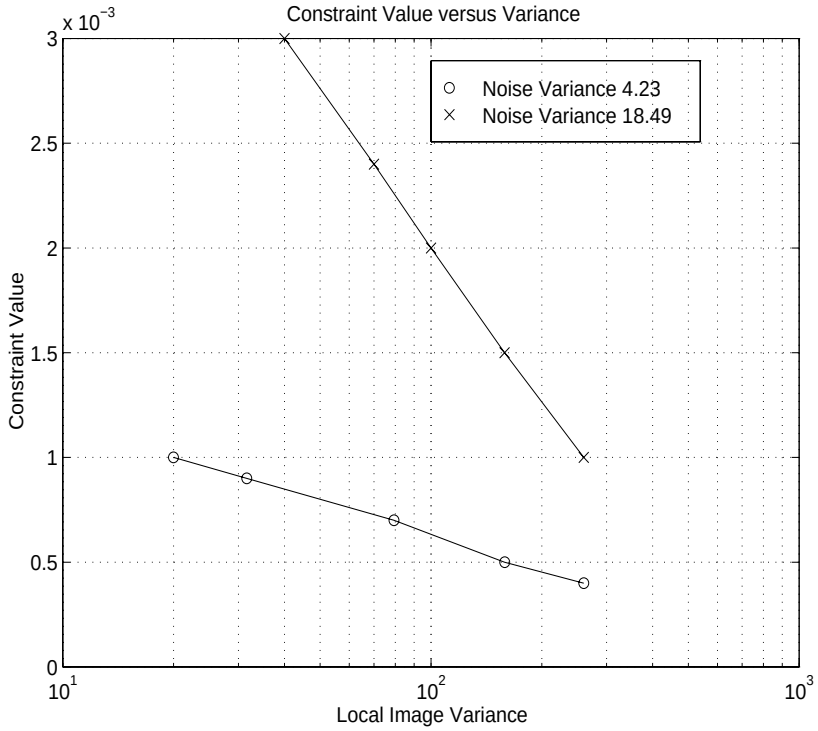


Figure 4.6: Constraint values chosen versus variance.

Figures 4.7 and 4.8 show the degraded and restored images of the flower image, while Figure 4.9 shows the degraded and restored images of the cat image in the case of a noise variance of 18.30.

(a)

(b)

(c)

(d)

(e)

Figure 4.7: Degraded and restored images for noise of variance 4.22.

(a)

(b)

(c)

(d)

(e)

Figure 4.8: Degraded and restored images for noise of variance 18.52.

(a)

(b)

(c)

(d)

(e)

Figure 4.9: Degraded and restored images for noise of variance 18.30.

Table 4.1: Statistics of degraded and restored images for various levels of noise.

Noise Variance	SNR (dB)					LSMSE				
	DI	NA	GER	KK	VB	DI	NA	GER	KK	VB
Cat Image										
4.24	12.61	13.30	12.55	11.38	13.19	42.98	14.58	14.98	30.57	13.51
18.30	12.25	12.81	11.02	10.83	12.58	38.65	17.45	30.43	30.46	15.79
Flower Image										
4.22	15.67	16.65	15.13	15.74	16.95	20.53	4.35	9.74	6.56	2.53
18.52	14.94	15.93	12.61	14.32	15.82	19.35	5.94	33.54	10.88	3.74

Legend: DI = Degraded Image; NA = Non-Adaptively restored image; GER = Image restored using energy minimisation method for constraint selection; KK = Image restored using the Kang and Katsaggelos algorithm; VB = image restored using Variance Based constraint selection.

From Figures 4.7, 4.8 and 4.9, the images restored using the adaptive image restoration methods appear clearer and more visually pleasing despite a slight decrease in SNR when compared to the images produced by the non-adaptive algorithm. Table 4.1 indicates that using variance as a criteria to chose the value of regularisation parameter produces images with an improved LSMSE as noise levels increase.

4.6.3 Spatially Variant Distortion

An image was created using a cyclic variation of 7 by 7 Gaussian PSFs. Using the analysis in Section 4.3, V was set to be 4. Table 4.2 details the degrading PSFs used to blur the image as per equation (4.19):

Table 4.2: Degrading PSFs.

Standard Deviation	1.5	2.0	3.0	4.0
PSF	$h_0(x, y)$	$h_1(x, y)$	$h_2(x, y)$	$h_3(x, y)$

Noise of variance 4.28 was added to the blurred image. The degraded image is shown in Figure 4.10a. This image was restored using four techniques. The first technique was a non-adaptive spatially invariant approach. A regularisation parameter value of 0.0007 was used and the spatially variant distortion was approximated as a spatially invariant distortion by using a 7 by 7 gaussian PSF of standard deviation 2.55. This image is shown in Figure 4.10b. The second technique was an adaptive spatially invariant approach. The same approximation of the PSF was in the previous experiment was used, however the regularisation parameter was varied for each pixel using the same local variance levels and λ values as in the low noise example in the previous experiment. This image is shown in Figure 4.10c.

The image was then restored using a non-adaptive spatially variant restoration method, with a regularization parameter value of 0.0007. This image is shown in Figure 4.10d. In the final approach, the degraded image was restored using a spatially variant, adaptive constraint method, with the variance based λ value selection algorithm. The regularisation value was selected in the same manner as image 4.10d. This image is shown in Figure 4.10e. The statistics are summarised in Table 4.3.

(a)

(b)

(c)

(d)

(e)

Figure 4.10: Results of experiments on an image degraded by spatially variant distortion.

Table 4.3: Results of spatially variant experiment.

Image	SNR (dB)	LSMSE
Degraded	13.94	36.01
Space Invariant Non-adaptive	15.03	6.381
Space Invariant Adaptive	14.92	5.45
Space Variant Non-adaptive	15.87	6.041
Space Variant Adaptive	16.01	4.84

It is interesting to note that by using an adaptive approach we can compensate for a lack of knowledge regarding the degrading PSF. In the cases where the degraded images are restored by the space invariant approach, we can see that using the adaptive technique produces a much clearer image with a lower LSMSE although the cyclic blurring effect is marginally visible. In all cases, using the adaptive approach produces a clearer image with a lower LSMSE.

4.6.4 Efficiency

In this experiment, the time taken to restore an image was compared among the four different cases from the space variant experiment. Each algorithm was run three times on a SUN Ultra 1 workstation, the average results for each algorithm are tabulated below.

Table 4.4: Algorithm run times.

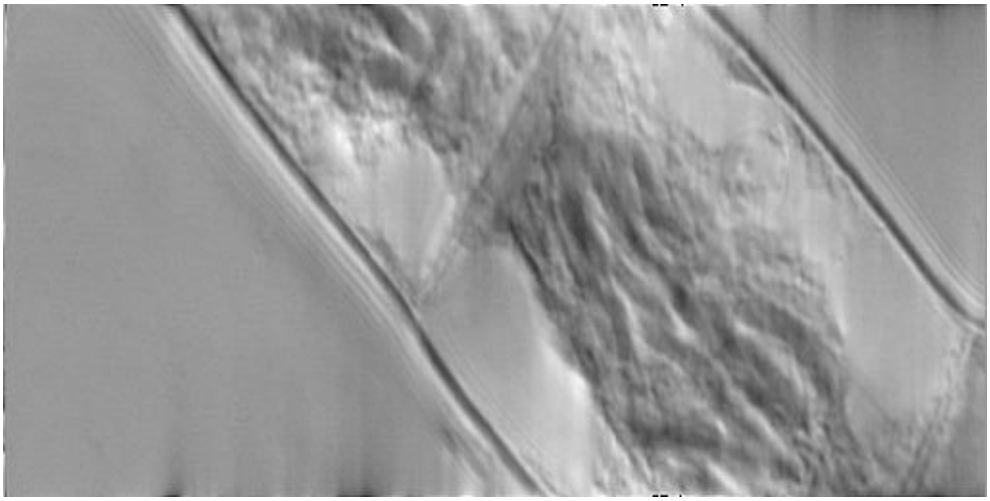
Algorithm	Time (CPU seconds)
Non-Adaptive Spatially Invariant	259
Non-Adaptive Spatially Variant	370
Adaptive Spatially Invariant	206
Adaptive Spatially Variant	667

It is worth noting that the execution times of the adaptive regularisation parameter algorithms are similar to the execution times of the non-adaptive algorithms. The adaptive algorithms require the calculation of five times the number of weighting masks as the non-adaptive algorithms, and also require pre-restoration calculations of local variance to precompute the regularisation parameter for each pixel. The reason for the similar execution times is that the time lost through setting up the adaptive parameters of the network is offset by the time gained through the non-adjustment of pixels in low variance regions of the image. The most important fact that we can observe from Table 4.4 is that the time difference required for a fully adaptive, spatially variant restoration is only double the time required for the much simpler non-adaptive spatially invariant restoration. This is much faster than any previously reported methods. In fact as the size of the image increases we would expect the time required for adaptive spatially variant restoration to approach the time required for non-adaptive, spatially invariant restoration. This is due to the fact that the extra time required for the adaptive spatially variant restoration is primarily taken up by the initial extra weighting mask creation. The quality and speed of the adaptive spatially invariant approximation method offers a promising alternative to the traditional semiblind deconvolution methods, especially when the PSF is space variant.

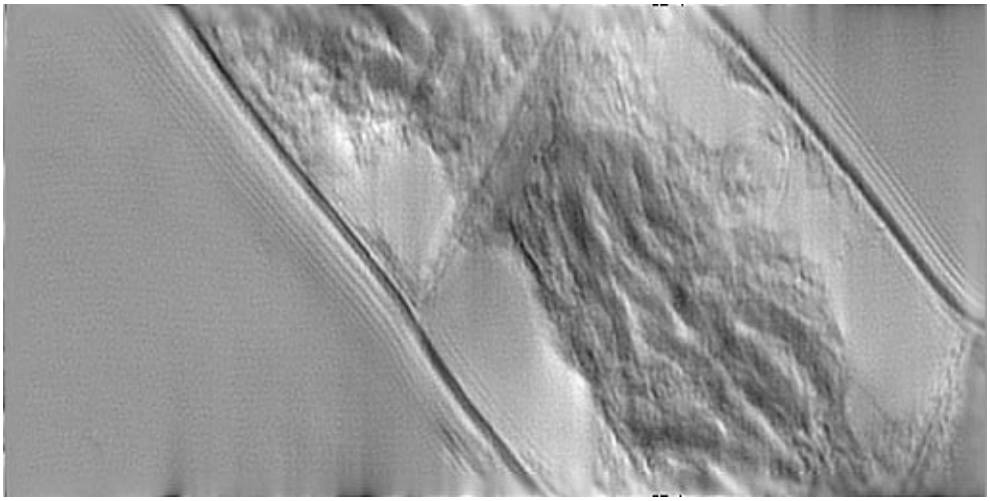
4.6.5 An Application Example

The above neural network algorithm was applied to the problem of restoring images with an unknown level of blur. Images were supplied to us showing chromosomes in a solution imaged by a high powered optical microscope. Limitations of the optical system had blurred the images of the chromosomes. To extract further information about the chromosomes in the images, the above neural network algorithm was used to enhance the images. Figure 4.11a shows one of the

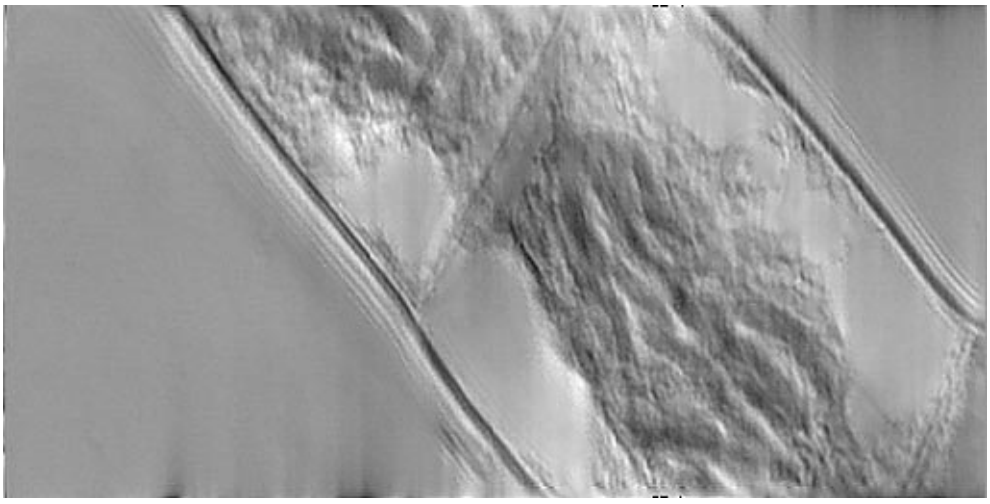
original degraded images. Figure 4.11b shows the image 4.11a restored using the non-adaptive algorithm. Note that the restored image is sharper than the original image, but some noise has also been amplified as well as some ringing effects. Figure 4.11c shows the image 4.11a restored using the adaptive algorithm. The level of sharpness is comparable to the results of the non-adaptive approach, however the level of background noise and ringing effects have been greatly reduced. In this case the adaptive restoration algorithm has been successful at enhancing the detail present in the image.



(a)



(b)



(c)

Figure 4.11: Degraded and restored chromosome images.

4.7 Summary

In this chapter the use of adaptive weights in constrained deconvolution methods was considered. Since the human visual system favours the presense of edges and boundaries, rather than more subtle differences in intensity in homogenous areas, [83], noise artifacts may be less disturbing in high contrast regions than in low contrast regions. It is then advantageous to use a stronger constraint in smooth areas of an image than in high-contrast regions. While traditional restoration methods find it difficult to implement an adaptive restoration spatially across an image, neural network based image restoration methods are particularly amendable to spatial variance of the restoration parameters.

A method based on using local image statistics to select the optimal value of regularisation parameter is considered. This method imitates the human visual system and produces superior results when compared to non-adaptive methods. In addition it was found that no disadvantage occurred when the values of regularisation parameter for each pixel were chosen by the restoration algorithm before starting the restoration, rather than during each iteration of the restoration procedure. In fact, precomputing of the regularisation parameter further increased the restoration speed.

In the final part of this chapter, the work is expanded upon to adaptively restore images degraded by a spatially variant PSF. It is shown that adaptive regularisation techniques can compensate for insufficient knowledge of the degradation in the case of spatially variant distortions. Moreover, an adaptive spatially variant restoration is shown to be able to be completed in the same order of magnitude of time as a much simpler non-adaptive spatially invariant restoration.

Chapter 5: Perception Based Cost Functions for Image Restoration

5.1 Introduction

In Chapter 2, the author introduced a new error measure based on comparing local variances which examines the image in a regional sense rather than a pixel-by-pixel sense. In this chapter, we further develop these concepts by introducing two new restoration algorithms which incorporate versions of this image error measure. The cost functions that these algorithms are based on are nonlinear and cannot be efficiently implemented by conventional methods. We therefore propose extended neural network algorithms to iteratively perform the restoration. We show that the proposed cost functions and processing algorithms perform very well when applied to both colour and grayscale images. One important property of the proposed method, compared with the neural network implementation of the constrained least square filter, is that it is very fault-tolerant in the sense that when some of the neural connections are damaged, it can still produce very satisfactory results. Comparison with some of the conventional methods will be provided to justify the new method.

This chapter is organised as follows. Section 5.2 describes the motivation for incorporating the error measure described in Chapter 2 into a cost function. Section 5.3 presents the restoration cost function incorporating a version of the proposed image error measure from Chapter 2. Section 5.4 builds on the previous section to present an algorithm based on a more robust variant of the novel image error measure. Section 5.5 describes implementation considerations.

Section 5.6 presents some experimental data from this investigation, and Section 5.7 summarises this chapter.

5.2 Motivation

In Chapter 2, we considered the problems inherent in using the MSE and SNR to compare two images. It was seen that the MSE and SNR have little relationship to the way that humans perceive the differences between two images. Although incorporating concepts involved in human perception may seem a difficult task, a new image error measure was presented there which, despite its simplicity, incorporates some concepts involved in human appraisal of images.

In Chapter 4, the basic neural network restoration algorithm described in Chapter 3 was expanded to restore images adaptively using simple human visual concepts. This adaptive algorithm obtained superior results when compared to the non-adaptive algorithm, and was shown to produce a more robust restoration when errors occurred due to insufficient knowledge of the degrading function. Despite the improved performance of the adaptive algorithm, it is still not simple to chose the correct values of the constraint parameter, λ. In the case of the adaptive algorithm, the problem is compounded by the fact that many values of λ must be selected, rather than just one value as in the non-adaptive case. In addition, the selected λ values must be related to local variance levels.

We desire to create an algorithm which can adaptively restore an image, using simple concepts involved in human perception, with only a few free parameters to be set. Such an algorithm would be more robust and easier to use than the algorithm in Chapter 4. In the previous

adaptive algorithm, minimising the MSE was still at the base of the restoration strategy. However Chapter 2 provides us with a simple alternative to the MSE. It seems logical to create a new cost function which minimises a LSMSE related term. In this way, the adaptive nature of the algorithm would be incorporated in the LSMSE term rather than imposed by the external selection of λ values.

5.3 A LVMSE-Based Cost Function

The discussion in the previous section prompts us to structure a new cost function which can properly incorporate the LSMSE into restoration. Since the formula for the neural network cost function has a term which attempts to minimise the MSE, the author investigated the problem of restoring images using a cost function with a term which endeavours to minimise a LSMSE-related error measure. The author proposes that an additional term be added to (2.3). The new term evaluates the local variance mean square error (2.12). The comparison of local variances rather than local standard deviations was chosen for the cost function since it is easier and more efficient to calculate.

Hence the new cost function we suggest is:

$$E = \frac{1}{2}\|\boldsymbol{g} - \boldsymbol{H}\hat{\boldsymbol{f}}\|^2 + \frac{\lambda}{2}\|\boldsymbol{D}\hat{\boldsymbol{f}}\|^2 + \theta \sum_{x=0}^{N-1}\sum_{y=0}^{M-1} \frac{(\sigma_A^2(\hat{\boldsymbol{f}}(x,y)) - \sigma_A^2(\boldsymbol{g}(x,y))^*)^2}{NM} \qquad (5.1)$$

where $\sigma_A^2(\hat{\boldsymbol{f}}(x,y))$ is the variance of the local region surrounding pixel (x,y) in the image estimate and $\sigma_A^2(\boldsymbol{g}(x,y))^*$ is the variance of the local region surrounding pixel (x,y) in the

degraded image scaled to predict the variance in the original image. The first two terms in (5.1) ensure a globally balanced restoration, whereas the added LVMSE term enhances local features. In (5.1), $\sigma_A^2(\boldsymbol{g}(x, y))^*$ is determined as follows. Since the degraded image has been blurred, image variances in $\boldsymbol{g}$ will be lower than the corresponding variances in the original image. In this case, the variances $\sigma_A^2(\boldsymbol{g}(x, y))^*$ would be scaled larger than $\sigma_A^2(\boldsymbol{g}(x, y))$ to reflect the decrease in variance due to the blurring function. In general, if we consider an image degraded by a process which is modelled by (2.2), then we find that a useful approximation is:

$$\sigma_A^2(\boldsymbol{g}(x, y))^* = K(x, y)(\sigma_A^2(\boldsymbol{g}(x, y)) - J(x, y)) \tag{5.2}$$

where $J(x, y)$ is a function of the noise added to the degraded image at point (x, y) and $K(x, y)$ is a function of the degrading point spread function at point (x, y). Although it may appear difficult to accurately determine the optimal values of $K(x, y)$, in fact the algorithm is extremely tolerant of variations in this factor and only a rough estimate is required. For example, if the image degradation is a moderate blurring function, with a region of support of around 5 or 7 then $K(x, y)$ would be set to 2 for all pixels in the image. This indicates that the local variances in the original image are on average approximately twice that of the degraded image. A high degree of accuracy is not required. In highly textured regions of the image where the preservation of image details are most important, the LVMSE term requires that the variance of the region be large, and the first two terms of (5.1) ensure the sharpness and accuracy of the image features.

5.3.1 The Extended Algorithm for the LVMSE Modified Cost Function

The LVMSE modified cost function does not fit easily into the neural network energy function as given by (2.4), however an efficient algorithm can be designed to minimise this cost function. One of the first considerations when attempting to implement the LVMSE cost function is prompted by a fundamental difference in the cost function which occurs due to the addition of the new term. In the case of a cost function based on minimising mean square error alone, any changes in an individual pixel value effects the entire image MSE in a simple way. The square error between any pixel in the image estimate and the corresponding pixel in the original image does not affect the square error of its neighbours. This simplifies the implementation of the cost function. In the case of the LVMSE modified cost function, it is different. When a pixel's value is altered by the algorithm, the total change in the LVMSE is not a simple function of the current pixel's change in value alone. Changing the current pixel's value changes its own local variance, and the local variances of all of its neighbours within an A by A proximity of the current pixel. Hence to truly calculate the total change in LVMSE for the entire image, the algorithm must calculate how changing the current pixel's value effects the local variances of all its neighbours and how these changes effect the overall LVMSE. This approach is computationally prohibitive. To resolve this problem we must go back to the fundamental justification for adding the LVMSE term in the first place. The justification for adding this term was the fact that we wished to create a cost function which matched the local statistics of pixels in the original image to that of the image estimate. In this case it is sufficient that the algorithm considers only minimising the difference in the local variance of the estimated image pixel to the corresponding original image pixel and not minimising the total LVMSE of the image. The great benefit arising from this approximation will become apparent as explained below.

The first step in the development of the algorithm is a change in notation. For an N by M image let $\boldsymbol{f}$ represent the lexicographically organized image vector of length NM as per the model given by (2.2) and the algorithm for the unmodified neural network cost function (2.3). The translation between the two indices x and y of $\boldsymbol{f}(x, y)$ and the single index i of $\boldsymbol{f}_i$ is given by:

$$i = x + yN \tag{5.3}$$

Define x^k and y^k as the two dimensional x and y values associated with pixel k by (5.3).

Define the two-dimensional distance between pixels i and j as:

$$\text{dis2}(i, j) = \left|x^i - x^j\right| + \left|y^i - y^j\right| \tag{5.4}$$

Let $\boldsymbol{K}^i$ represent the NM by NM matrix which has the following property:

$$\text{Let } \boldsymbol{f}^i = \boldsymbol{K}^i \boldsymbol{f} \tag{5.5}$$

$$\text{then} \qquad [\boldsymbol{f}^i]_j = \begin{cases} 0; & \text{dis2}(i, j) > \dfrac{A-1}{2} \\ \boldsymbol{f}_j; & \text{dis2}(i, j) \le \dfrac{A-1}{2} \end{cases} \tag{5.6}$$

$\boldsymbol{K}^i$ has the effect of setting to zero all pixels not within the A by A neighbourhood centered on the pixel with co-ordinates x^i, y^i. As a shortened notation we will denote $[\boldsymbol{f}^i]_j$ as $\boldsymbol{f}^i_j$. Using

this notation, the average pixel value in the *A* by *A* region surrounding pixel *i* is given by:

$$M_A(i) = \frac{1}{A^2} \sum_{j=1}^{MN} \boldsymbol{f}_j^i$$

Let $\beta_i = \sum_{j=1}^{NM} (\boldsymbol{f}_j^i)^2$ and $\gamma_i = \sum_{j}^{NM} \boldsymbol{f}_j^i$. Then the estimated variance of the *A* by *A* region surrounding pixel *i* is given by:

$$V^i = \frac{\beta_i}{A^2} - \frac{\gamma_i^2}{A^4} \tag{5.7}$$

Note that, strictly speaking, V^i is an estimate of the variance of this region given the available pixel values. The true variance of this region is the expectation of the second moment. However (5.7) is a suitable approximation given the available data. In the rest of this analysis, (5.7) will be called the “local variance” and the term “estimated local variance” will be used to refer to $\sigma_A^2(\boldsymbol{g}(x, y))^*$.

The LVMSE between the image estimate, $\hat{\boldsymbol{f}}$, and the original image, $\boldsymbol{f}$, may then be written as:

$$\text{LVMSE}(\hat{\boldsymbol{f}}, \boldsymbol{f}) = \frac{1}{NM} \sum_{i=1}^{NM} (V^i(\hat{\boldsymbol{f}}) - V^i(\boldsymbol{f}))^2 \tag{5.8}$$

Let $V^i(\hat{\boldsymbol{f}})$ be approximated by $Vf^{\,i}$. $Vf^{\,i}$ is the estimate of the local variance of pixel *i* in the original image based on the degraded image and knowledge of the degrading point spread function as per equation (5.2). $Vf^{\,i}$ is calculated before the algorithm commences and remains a constant throughout the restoration procedure.

The algorithm we propose to implement first computes the negative direction of the gradient which gives an indication of whether increasing or decreasing the current neuron value will result in a net decrease in energy. Once the negative gradient is found the neuron value is changed in unit steps and the resultant energy decrease after each step is computed. This ends when no further energy minimisation is possible. In Chapter 3, we showed that the negative gradient of the unmodified cost function, (2.3), is in fact the input to the neuron. Hence the negative gradient of the modified cost function will therefore be the input to the neuron minus the derivative of (5.8).

The gradient of (5.8) is given by:

$$\frac{\partial}{\partial \hat{\boldsymbol{f}}_i}\text{LVMSE} = \frac{2}{NM}(V^i(\hat{\boldsymbol{f}}) - Vf^{\,i})\frac{\partial}{\partial \hat{\boldsymbol{f}}_i}(V^i(\hat{\boldsymbol{f}}) - Vf^{\,i}) \tag{5.9}$$

Note that this formula is an approximation of the gradient which ignores the contributions of the local variances of the pixels adjacent to *i* to the overall LVMSE of the image.

$$\frac{\partial V^i}{\partial \hat{\boldsymbol{f}}_i} = \frac{2\hat{\boldsymbol{f}}_i}{A^2} - \frac{2\gamma_i}{A^4} \tag{5.10}$$

Taking note of the fact that $\hat{\boldsymbol{f}}_i^i = \hat{\boldsymbol{f}}$. Substituting (5.10) into (5.9) we obtain:

$$\frac{\partial}{\partial \hat{\boldsymbol{f}}_i}\text{LVMSE} = \frac{2}{NM}\left(\frac{\beta_i}{A^2} - \frac{\gamma_i^2}{A^4} - Vf^i\right)\left(\frac{2\hat{\boldsymbol{f}}_i}{A^2} - \frac{2\gamma_i}{A^4}\right)$$

$$\therefore \frac{\partial}{\partial \hat{\boldsymbol{f}}_i}\text{LVMSE} = \frac{4}{NMA^2}\left(\frac{\hat{\boldsymbol{f}}_i\beta_i}{A^2} - \frac{\hat{\boldsymbol{f}}_i\gamma_i^2}{A^4} - \hat{\boldsymbol{f}}_i Vf^i - \frac{\beta_i\gamma_i}{A^4} + \frac{\gamma_i^3}{A^6} + \frac{\gamma_i Vf^i}{A^2}\right) \quad (5.11)$$

Multiplying (5.11) by θ and subtracting it from the input to the neuron gives us the negative gradient of the cost function. Given a change in the value of pixel *i*, the resultant change in energy is the previous change in energy given by (3.4) plus θ times the change in LVMSE. The change in LVMSE is given by:

$$\Delta\text{LVMSE} = \frac{1}{NM}((V_{new}^i - Vf^i)^2 - (V_{old}^i - Vf^i)^2) \quad (5.12)$$

where

$$\gamma_i^{new} = \gamma_i^{old} + \Delta\hat{\boldsymbol{f}}_i$$

$$\beta_i^{new} = \beta_i^{old} + 2\hat{\boldsymbol{f}}_i\Delta\hat{\boldsymbol{f}}_i + (\Delta\hat{\boldsymbol{f}}_i)^2$$

$$V_{new}^i = \frac{\beta_i^{new}}{A^2} - \frac{(\gamma_i^{new})^2}{A^4}$$

$$= V_{old}^{i} + \frac{(\Delta \hat{f}_i)^2}{A^2} + \frac{2\hat{f}_i \Delta \hat{f}_i}{A^2} - \frac{2\gamma_i^{old} \Delta \hat{f}_i}{A^4} - \frac{(\Delta \hat{f}_i)^2}{A^4} \qquad (5.13)$$

where β_i^{old}, γ_i^{old}, and V_{old}^{i} are the values of these parameters before the change in the state of neuron *i* occurred. The LVMSE algorithm is therefore:

Algorithm 5.1.

repeat

{

for $i = 1, \ldots, L$ do

{

$$u_i = b_i + \sum_{j=1}^{L} w_{ij} \hat{f}_j$$

$$\beta_i^{old} = \sum_{j=1}^{NM} \left(\hat{f}_j^{i} \right)^2$$

$$\gamma_i^{old} = \sum_{j}^{NM} \hat{f}_j^{i}$$

$$V_{old}^{i} = \frac{\beta_i^{old}}{A^2} - \frac{\left(\gamma_i^{old}\right)^2}{A^4}$$

$$-\left(\frac{\partial E}{\partial \hat{f}_i}\right) = u_i - \frac{4\theta}{NMA^2}\left(\frac{\hat{f}_i \beta_i^{old}}{A^2} - \frac{\hat{f}_i (\gamma_i^{old})^2}{A^4} - \hat{f}_i V f^{i} - \frac{\beta_i^{old} \gamma_i^{old}}{A^4} + \frac{(\gamma_i^{old})^3}{A^6} + \frac{\gamma_i^{old} V f^{i}}{A^2} \right)$$

$$\Delta \hat{f}_i = G\left(-\left(\frac{\partial E}{\partial \hat{f}_i}\right)\right) \qquad \text{where } G(u) = \begin{cases} 1, u > 0 \\ 0, u = 0 \\ -1, u < 0 \end{cases}$$

$$V^i_{new} = V^i_{old} + \frac{(\Delta \hat{f}_i)^2}{A^2} + \frac{2\hat{f}_i \Delta \hat{f}_i}{A^2} - \frac{2\gamma^{old}_i \Delta \hat{f}_i}{A^4} - \frac{(\Delta \hat{f}_i)^2}{A^4}$$

$$\Delta E = -\left(\frac{1}{2}\right) w_{ii} (\Delta \hat{f}_i)^2 - u_i \Delta \hat{f}_i + \frac{\theta}{NM}((V^i_{new} - Vf^i)^2 - (V^i_{old} - Vf^i)^2)$$

repeat

$$\hat{f}_i(t+1) = K(\hat{f}_i(t) + \Delta \hat{f}_i)$$

$$\text{where } K(u) = \begin{cases} 0, u < 0 \\ u, 0 \le u \le S \\ S, u > S \end{cases}$$

$$u_i = u_i + w_{ii} \Delta \hat{f}_i$$

$$V^i_{old} = V^i_{new}$$

$$\gamma^{old}_i = \gamma^{old}_i + \Delta \hat{f}_i$$

$$V^i_{new} = V^i_{old} + \frac{(\Delta \hat{f}_i)^2}{A^2} + \frac{2\hat{f}_i \Delta \hat{f}_i}{A^2} - \frac{2\gamma^{old}_i \Delta \hat{f}_i}{A^4} - \frac{(\Delta \hat{f}_i)^2}{A^4}$$

until $\Delta E = -\left(\frac{1}{2}\right) w_{ii} (\Delta \hat{f}_i)^2 - u_i \Delta \hat{f}_i + \frac{\theta}{NM}((V^i_{new} - Vf^i)^2 - (V^i_{old} - Vf^i)^2) \ge 0$

}

t = t+1

}

until ($\hat{f}_i(t) = \hat{f}_i(t-1) \; \forall i = 1, ..., L$)

Figure 5.1 shows a flow chart representation of Algorithm 5.1.

Note that Algorithm 5.1 still utilises some features of Algorithm 3.3, specifically the use of bias inputs and interconnection strength matrices.

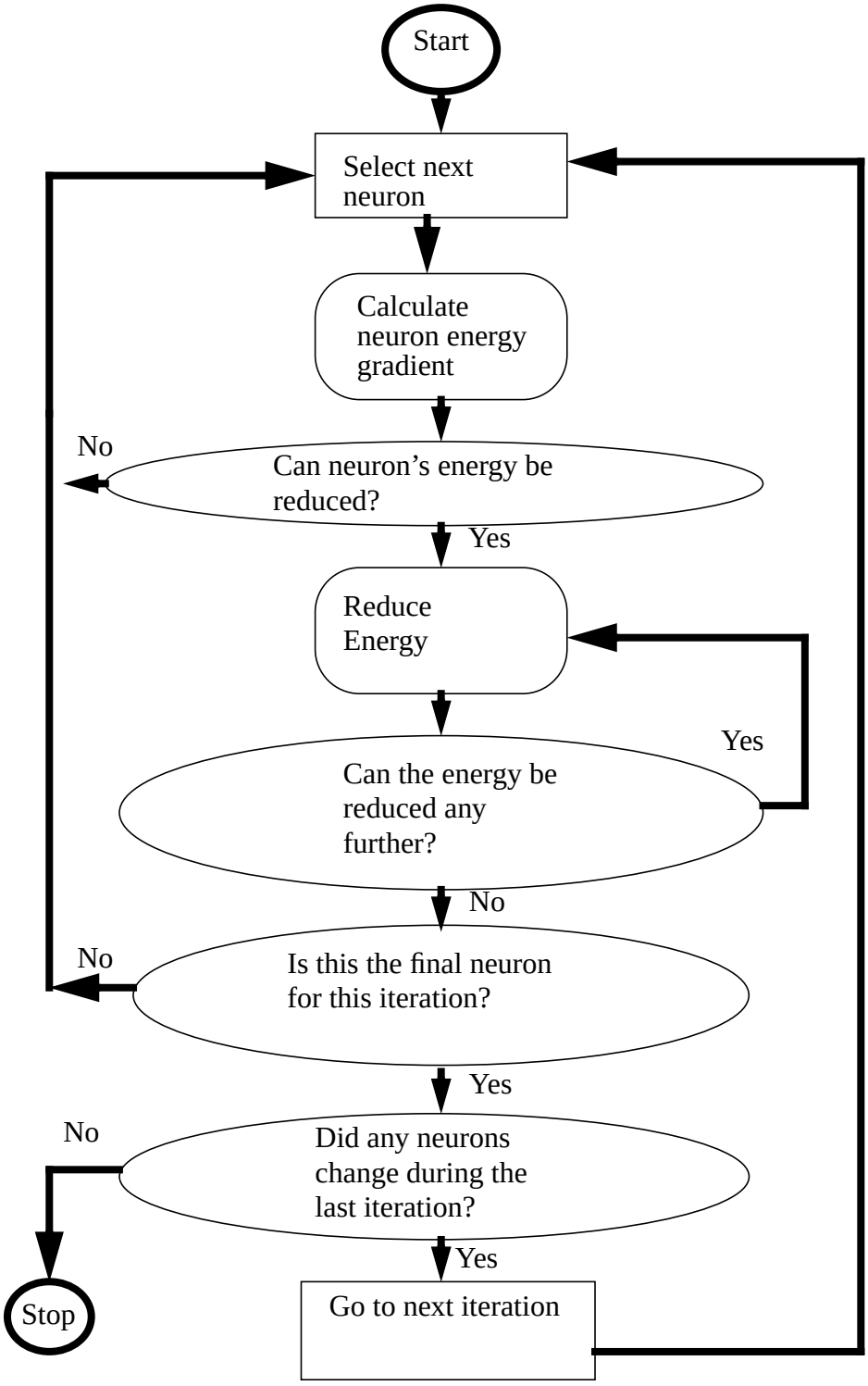


Figure 5.1: Flow-chart representing Algorithm 5.1.

5.3.2 Analysis

It is important to verify that Algorithm 5.1 acts upon a pixel in the intended manner. To verify this, we must examine the LVMSE term in Algorithm 5.1 more closely.

According to (5.11), the gradient of the LVMSE term of the cost function, when all pixels except $\hat{f}_i$ are held constant, is given by:

$$\frac{\partial}{\partial \hat{\boldsymbol{f}}_i}\text{LVMSE} = \frac{2}{NM}\left(\frac{\beta_i}{A^2} - \frac{\gamma_i^{\,2}}{A^4} - V f^i\right)\left(\frac{2\hat{\boldsymbol{f}}_i}{A^2} - \frac{2\gamma_i}{A^4}\right)$$

Note that $\beta_i = \sum_{j=1}^{NM} (\boldsymbol{f}_j^i)^2$ and $\gamma_i = \sum_{j}^{NM} \boldsymbol{f}_j^i$ can be rewritten as $\beta_i = \beta_i' + \hat{f}_i^{\,2}$ and $\gamma_i = \gamma_i' + \hat{f}_i$ where $\beta_i' = \sum_{j,\, j\neq i}^{NM} (\boldsymbol{f}_j^i)^2$ and $\gamma_i' = \sum_{j,\, j\neq i}^{NM} \boldsymbol{f}_j^i$. Note that in this way, we can extract the elements of (5.11) that depend on $\hat{f}_i$. Hence we obtain:

$$\frac{\partial}{\partial \hat{\boldsymbol{f}}_i}\text{LVMSE} = \frac{2}{NM}\left(\frac{\hat{f}_i^{\,2}}{A^2} + \frac{\beta_i'}{A^2} - \frac{(\hat{f}_i + \gamma_i')^2}{A^4} - V f^i\right)\left(\frac{2\hat{f}_i}{A^2} - \frac{2\hat{f}_i}{A^4} - \frac{2\gamma_i'}{A^4}\right)$$

$$= \frac{2}{NM}\left(\frac{\hat{f}_i}{A^2} - \frac{\hat{f}_i^{\,2}}{A^4} - \frac{2\hat{f}_i\gamma_i'}{A^4} + \frac{\beta_i'}{A^2} - \frac{\gamma_i'}{A^4} - V f^i\right)\left(\frac{2\hat{f}_i}{A^2} - \frac{2\hat{f}_i}{A^4} - \frac{2\gamma_i'}{A^4}\right) \tag{5.14}$$

Note that $\frac{\beta_i'}{A^2} - \frac{(\gamma_i')^2}{A^4}$ is an approximation to the local variance at pixel *i* neglecting the

contribution of the value of pixel *i* itself. Let $V' = \frac{\beta_i'}{A^2} - \frac{(\gamma_i')^2}{A^4}$. As *A* increases in value, V' approaches the value of the local variance at pixel *i*. Similarly, we can define an approximation to the local mean of pixel *i* as:

$$M' = \frac{1}{A^2} \sum_{j,\, j \neq i}^{NM} \hat{f}_j^i = \frac{\gamma_i'}{A^2} \tag{5.15}$$

If *A* is greater than 3, then pixel *i* contributes less than 15% of the value of the local mean and local variance of its neighbourhood. If *A* is 7 then the contribution is only 2%. Hence approximating V' as the local variance is valid.

This leaves us with:

$$\frac{\partial}{\partial \hat{\boldsymbol{f}}_i}\text{LVMSE} = \frac{4}{NMA^2}\left(\hat{f}_i^{\,2}\left(\frac{1}{A^2} - \frac{1}{A^4}\right) - \frac{2\hat{f}_i M'}{A^2} + J \right)\left(\hat{f}_i\left(1 - \frac{1}{A^2}\right) - M' \right) \tag{5.16}$$

where $J = V' - V f^i$.

The points for which Equation (5.16) is equal to zero are the stationary points of the LVMSE term in (5.1). Equation (5.16) is zero when $\hat{f}_i = \frac{M'}{\left(1 - \frac{1}{A^2}\right)}$. This corresponds to the case where $\hat{f}_i$ is approximately equal to its local mean. Equation (5.16) also has zeroes when $\hat{f}_i$ satisfies:

$$\hat{f}_i = \frac{\frac{2M'}{A^2} \pm \sqrt{\frac{4(M')^2}{A^4} - 4J\left(\frac{1}{A^2} - \frac{1}{A^4}\right)}}{2\left(\frac{1}{A^2} - \frac{1}{A^4}\right)}$$

If A is greater than or equal to 5, then the error resulting from approximating $\frac{1}{A^2} - \frac{1}{A^4}$ as $\frac{1}{A^2}$ is less than 4%. Therefore, if we assume that A is large enough that $\frac{1}{A^2} \gg \frac{1}{A^4}$ then (5.16) has zeroes at:

$$\hat{f}_i \approx M' \pm A\sqrt{\frac{(M')^2}{A^2} - J} \qquad (5.17)$$

Note that (5.17) indicates that if $J > 0$, (5.17) may not have a real-valued solution. There are three cases to examine.

CASE 1: $J < 0$

When $J < 0$, then local variance of the region surrounding pixel i is less than the estimated local variance. Equations (5.17) and (5.16) indicate that the LVMSE term in the cost function will have three stationary points. The function will then appear as Figure 5.2. The stationary point given by $\hat{f}_i = \frac{M'}{\left(1 - \frac{1}{A^2}\right)}$ becomes a maximum and the function has two minima given by (5.17). At least one of these minima will be in the domain $\hat{f}_i > 0$, and so as long as J is not

excessively negative it is possible to minimise the LVMSE term in the cost function in this case.

This is as one would expect since if $J < 0$, then the variance of the local region surrounding pixel *i* needs to be increased to meet the target variance estimate. It is always possible to increase the local variance of pixel *i* by moving that pixel's value further from the mean. This is why the stationary point $\hat{f}_i = \frac{M'}{\left(1 - \frac{1}{A^2}\right)}$, which is the point where $\hat{f}_i$ is approximately equal to the mean of the local region, becomes a maximum.

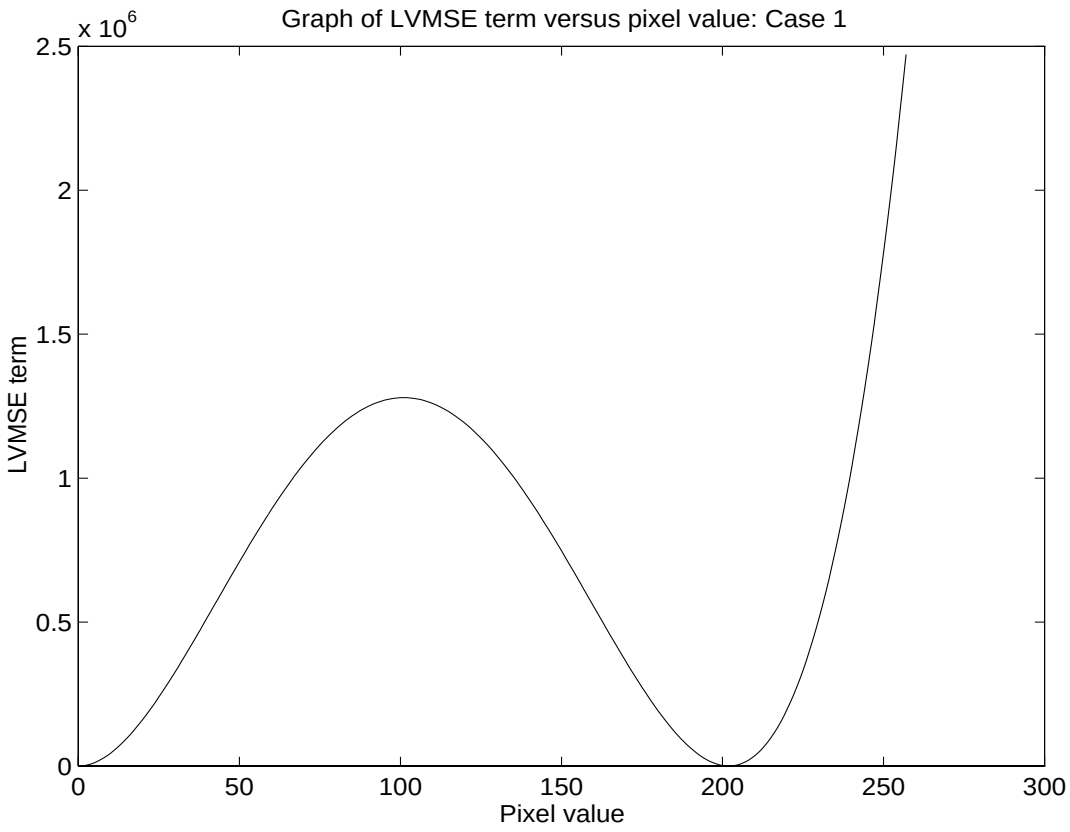


Figure 5.2: Graph of function for Case 1.

CASE 2: $0 < J \leq \frac{(M')^2}{A^2}$

When $0 \leq J \leq \frac{(M')^2}{A^2}$, then the local variance of region surrounding pixel *i* is greater than the estimated local variance, but the difference is not great. Equations (5.17) and (5.16) indicate that the LVMSE term in the cost function will again have three stationary points. The function will then appear as Figure 5.3.

The stationary point given by $\hat{f}_i = \frac{M'}{\left(1 - \frac{1}{A^2}\right)}$ becomes a maximum and the function has two minima given by (5.17). At least one of these minima will be in the domain $\hat{f}_i > 0$, and it is again possible to minimise the LVMSE term in the cost function in this case.

When $J > 0$, the local variance of pixel *i* needs to be decreased to match the target variance estimate. If $J \leq \frac{(M')^2}{A^2}$, it is possible to match the local variance of pixel *i* with the target variance estimate by moving that pixels value toward the mean pixel value in the local region. The stationary point $\hat{f}_i = \frac{M'}{\left(1 - \frac{1}{A^2}\right)}$, which is the point where $\hat{f}_i$ is approximately equal to the mean of the local region, is again a maximum, unless $J = \frac{(M')^2}{A^2}$. When $J = \frac{(M')^2}{A^2}$, all three stationary points correspond to the same value and the function has one minimum at $\hat{f}_i = \frac{M'}{\left(1 - \frac{1}{A^2}\right)} \approx M'$.

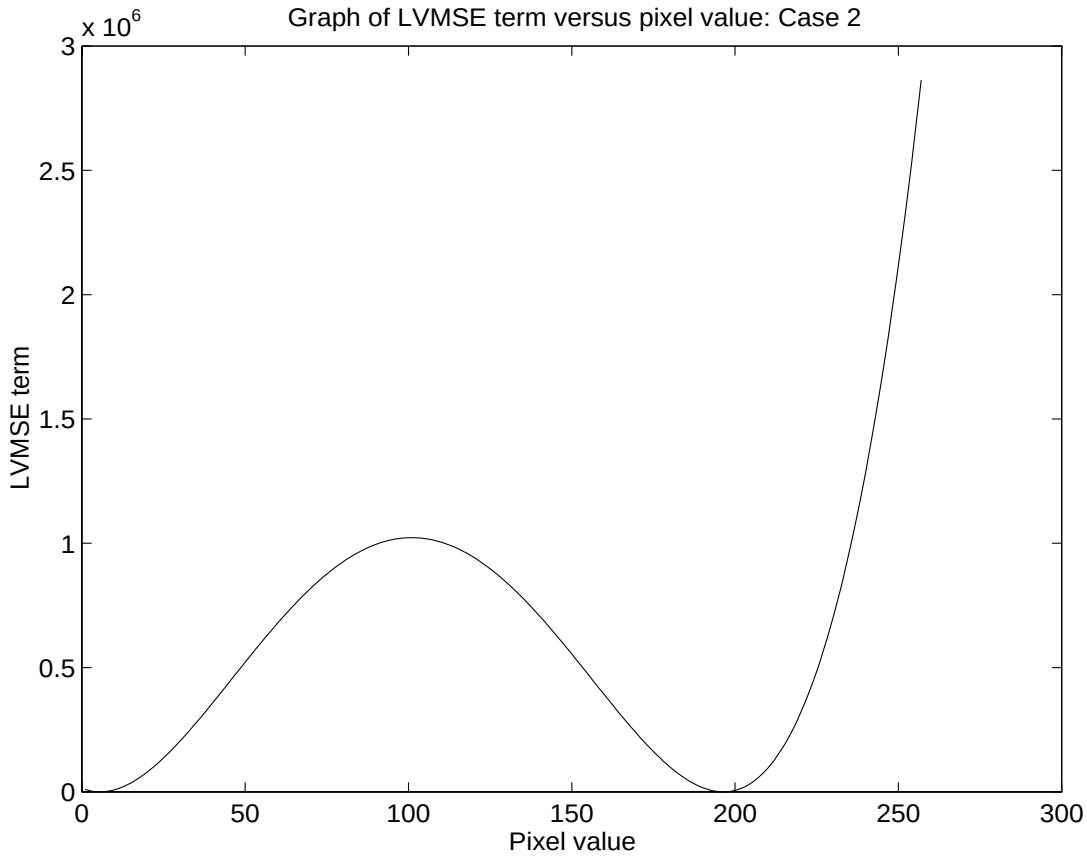


Figure 5.3: Graph of function for Case 2.

CASE 3: $J > \frac{(M')^2}{A^2}$

When $J > \frac{(M')^2}{A^2}$, then the local variance of region surrounding pixel *i* is greater than the estimated local variance, but the difference is too large for equality to be reached by changing a single pixel's value. Equation (5.17) will have no solutions and the LVMSE term in the cost function will have one stationary point. The function will then appear as Figure 5.4.

The stationary point given by $\hat{f}_i = \frac{M'}{\left(1 - \frac{1}{A^2}\right)}$ becomes a minimum, where $M' > 0$ since all pixel values are constrained to be positive. So the minima will be in the domain $\hat{f}_i > 0$, and it is possible to minimise the LVMSE term in the cost function in this case.

When $J > \frac{(M')^2}{A^2}$, the local variance of pixel *i* needs to be decreased to match the target variance estimate. The minimum local variance is obtained by decreasing the value of the current pixel to the mean value of the local region. In this case, the minimum local variance obtainable by altering one pixel's value is still greater than the estimate. The stationary point $\hat{f}_i = \frac{M'}{\left(1 - \frac{1}{A^2}\right)}$ is the point where $\hat{f}_i$ is approximately equal to the mean of the local region. Hence the pixel will move toward the mean value of the local region, hence reducing the local variance as much as possible.

By examining each of the three cases above we can see that the LVMSE term in the cost function is well behaved despite its non-linear nature. In all cases a minimum exists and by minimising the LVMSE term in the cost function, the local variance of the current pixel's region will always move closer to the target variance estimate.

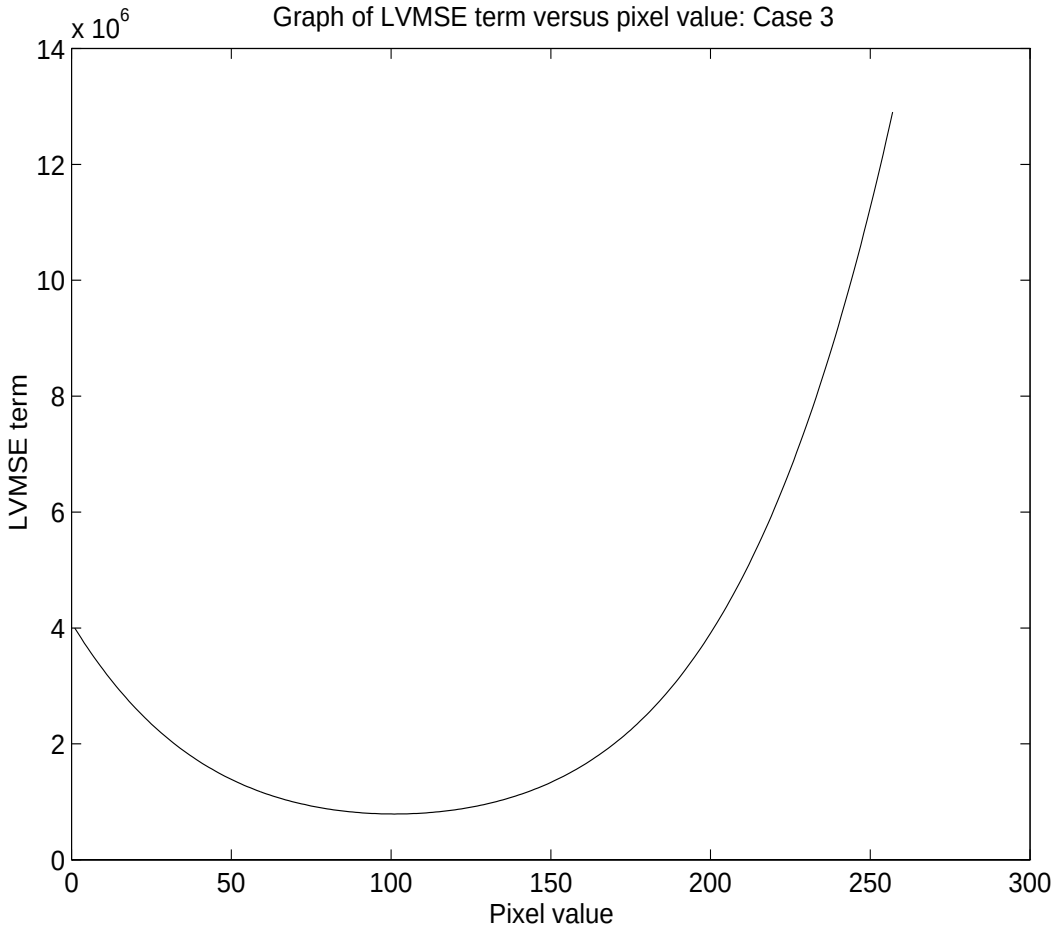


Figure 5.4: Graph of function for Case 3.

5.4 A log LVMSE-Based Cost Function

The previous section showed that the LVMSE cost term in (5.1) is well behaved, and results in an algorithm which does indeed work as intended to match the local variance of the region surrounding the current pixel to that of a target variance estimate of the original image. However improvement is still possible. The LVMSE term in (5.1) has its greatest effect when the difference between the actual local variance and the target variance estimate is large. When the difference is small, the LVMSE term in (5.1) has little effect. The strength of the LVMSE term is proportional to the square of the absolute difference between the two variances and does not

depend on the level of the variances. The dependence on the absolute difference between the variances is in fact a disadvantage. When the local variance of a region is large and the target variance for that region is also large, then noise will not be readily noticed. In this case, the first term in the cost function which ensures sharpness of edges should be allowed to dominate the restoration. However in this case, the difference between the target variance and actual variance may be large, causing the LVMSE term to dominate instead. On the other hand, when the local variance of a region is small and the target variance for that region is also small, we would want the LVMSE term to dominate, since this term would keep the local variance low and suppress noise. However, in this case since the target variance and the actual variance are small, the LVMSE term can also be too small and have an insufficient effect.

This prompts us to move away from absolute differences in local variances, and, instead, compare the ratio of the local variance and its estimate. Taking this idea one step further, we notice that taking the log of this ratio provides additional emphasising of small differences in variance at low variance levels and de-emphasising large differences in variance at high variance levels.

Hence the new cost function we propose is:

$$E = \frac{1}{2}\left\|\boldsymbol{g} - \boldsymbol{H}\hat{\boldsymbol{f}}\right\|^2 + \frac{\lambda}{2}\left\|\boldsymbol{D}\hat{\boldsymbol{f}}\right\|^2 + \theta \sum_{x=0}^{N-1}\sum_{y=0}^{M-1} \frac{\left(\ln\left(\frac{\sigma_A^2(\hat{\boldsymbol{f}}(x,y))}{\sigma_A^2(\boldsymbol{g}(x,y))^*}\right)\right)^2}{NM} \tag{5.18}$$

This new term in (5.18), we will denote the Log Local Variance Ratio (Log LVR) term.

5.4.1 The Extended Algorithm for the Log LVR Modified Cost Function

The algorithm to minimise (5.18) has the same basic strategy as the algorithm created in Section 5.3. First, the negative direction of the gradient is computed. Once the negative gradient is found, the neuron value is changed in unit steps and the resultant energy decrease after each step is computed. This ends when no further energy reduction is possible. As in the last section, multiplying the partial derivative of the Log LVR term in (5.18) by θ and subtracting it from the input to the neuron gives us the negative gradient of the cost function. As defined in Section 5.3, the local variance of the *A* by *A* region centred on pixel *i* is given by:

$$V^i = \frac{\beta_i}{A^2} - \frac{\gamma_i^2}{A^4}$$

where $\beta_i = \sum_{j=1}^{NM} (\boldsymbol{f}_j^i)^2$ and $\gamma_i = \sum_{j}^{NM} \boldsymbol{f}_j^i$.

The gradient of the Log LVR term in (5.18) is given by:

$$\frac{\partial}{\partial \hat{f}_i} LogLVR = \frac{1}{NM} 2 \ln\left(\frac{V^i}{Vf^i}\right) \frac{Vf^i}{V^i} \frac{1}{Vf^i} \left(\frac{2\hat{f}_i}{A^2} - \frac{2\gamma_i}{A^4}\right)$$

$$= \frac{4}{NM} \left(\frac{\ln\left(\frac{\beta_i}{A^2} - \frac{\gamma_i^2}{A^4}\right)}{\ln(Vf^i)} \right) \frac{\left(\frac{\hat{f}_i}{A^2} - \frac{\gamma_i}{A^4}\right)}{\left(\frac{\beta_i}{A^2} - \frac{\gamma_i^2}{A^4}\right)} \qquad (5.19)$$

$$= \frac{4}{NM}\left(\frac{\ln\left(\frac{\beta_i}{A^2}-\frac{\gamma_i^2}{A^4}\right)}{\ln(Vf^i)}\right)\frac{\left(\hat{f}_i-\frac{\gamma_i}{A^2}\right)}{\left(\beta_i-\frac{\gamma_i^2}{A^2}\right)}$$

Similarly, given a change in the value of pixel *i*, the resultant change in energy is the previous change in energy given by (3.4) plus θ times the change in Log LVR. The change in Log LVR is given by:

$$\Delta LogLVR = \frac{1}{NM}\left(\left(\ln\left(\frac{V_{new}^i}{Vf^i}\right)\right)^2-\left(\ln\left(\frac{V_{old}^i}{Vf^i}\right)\right)^2\right) \tag{5.20}$$

where

$$\gamma_i^{new} = \gamma_i^{old} + \Delta\hat{\boldsymbol{f}}_i$$

$$\beta_i^{new} = \beta_i^{old} + 2\hat{\boldsymbol{f}}_i\Delta\hat{\boldsymbol{f}}_i + (\Delta\hat{\boldsymbol{f}}_i)^2$$

$$V_{new}^i = \frac{\beta_i^{new}}{A^2}-\frac{(\gamma_i^{new})^2}{A^4}$$

$$= V_{old}^i + \frac{(\Delta\hat{\boldsymbol{f}}_i)^2}{A^2} + \frac{2\hat{\boldsymbol{f}}_i\Delta\hat{\boldsymbol{f}}_i}{A^2} - \frac{2\gamma_i^{old}\Delta\hat{\boldsymbol{f}}_i}{A^4} - \frac{(\Delta\hat{\boldsymbol{f}}_i)^2}{A^4}$$

The new algorithm is therefore:

Algorithm 5.2.

repeat

{

for $i = 1, \ldots\ldots, L$ do

{

$$u_i = b_i + \sum_{j=1}^{L} w_{ij} \hat{f}_j$$

$$\beta_i^{old} = \sum_{j=1}^{NM} \left(\hat{f}_j^i \right)^2$$

$$\gamma_i^{old} = \sum_{j}^{NM} \hat{f}_j^i$$

$$V_{old}^i = \frac{\beta_i^{old}}{A^2} - \frac{\left(\gamma_i^{old}\right)^2}{A^4}$$

$$-\left(\frac{\partial E}{\partial \hat{f}_i}\right) = u_i - \frac{4\theta}{NM}\left(\frac{\ln\left(\frac{\beta_i}{A^2} - \frac{\gamma_i^2}{A^4}\right)}{\ln(V f^i)}\right)\left(\frac{\left(\hat{f}_i - \frac{\gamma_i}{A^2}\right)}{\left(\beta_i - \frac{\gamma_i^2}{A^2}\right)}\right)$$

$$\Delta \hat{f}_i = G\left(-\left(\frac{\partial E}{\partial \hat{f}_i}\right)\right) \qquad \text{where } G(u) = \begin{cases} 1, u > 0 \\ 0, u = 0 \\ -1, u < 0 \end{cases}$$

$$V_{new}^i = V_{old}^i + \frac{(\Delta \hat{f}_i)^2}{A^2} + \frac{2 \hat{f}_i \Delta \hat{f}_i}{A^2} - \frac{2\gamma_i^{old} \Delta \hat{f}_i}{A^4} - \frac{(\Delta \hat{f}_i)^2}{A^4}$$

$$\Delta E = -\left(\frac{1}{2}\right) w_{ii} (\Delta \hat{f}_i)^2 - u_i \Delta \hat{f}_i + \frac{\theta}{NM}\left(\left(\ln\left(\frac{V_{new}^i}{V f^i}\right)\right)^2 - \left(\ln\left(\frac{V_{old}^i}{V f^i}\right)\right)^2\right)$$

repeat

$$\hat{f}_i(t+1) = K(\hat{f}_i(t) + \Delta \hat{f}_i)$$

$$\text{where } K(u) = \begin{cases} 0, u < 0 \\ u, 0 \le u \le S \\ S, u > S \end{cases}$$

$$u_i = u_i + w_{ii}\Delta\hat{\boldsymbol{f}}_i$$

$$V_{old}^i = V_{new}^i$$

$$\gamma_i^{old} = \gamma_i^{old} + \Delta\hat{\boldsymbol{f}}_i$$

$$V_{new}^i = V_{old}^i + \frac{(\Delta\hat{\boldsymbol{f}}_i)^2}{A^2} + \frac{2\hat{\boldsymbol{f}}_i\Delta\hat{\boldsymbol{f}}_i}{A^2} - \frac{2\gamma_i^{old}\Delta\hat{\boldsymbol{f}}_i}{A^4} - \frac{(\Delta\hat{\boldsymbol{f}}_i)^2}{A^4}$$

$$\text{until} \quad \Delta E = -\left(\frac{1}{2}\right)w_{ii}(\Delta\hat{\boldsymbol{f}}_i)^2 - u_i\Delta\hat{\boldsymbol{f}}_i + \frac{\theta}{NM}\left(\left(\ln\left(\frac{V_{new}^i}{Vf^i}\right)\right)^2 - \left(\ln\left(\frac{V_{old}^i}{Vf^i}\right)\right)^2\right) \geq 0$$

}

t = t+1

}

until ($\hat{\boldsymbol{f}}_i(t) = \hat{\boldsymbol{f}}_i(t-1)\ \forall i = 1, ..., L$)

Note that Algorithm 5.2 is almost identical to Algorithm 5.1 and still utilises some features of Algorithm 3.3, specifically the use of bias inputs and interconnection strength matrices.

5.4.2 Analysis

As in the previous section, we will verify the correct operation of Algorithm 5.2. To verify this, we must examine the Log LVR term in (5.18) more closely.

The gradient of the Log LVR term of the cost function when all pixels except $\hat{f}_i$ are held constant is given by:

$$\frac{\partial}{\partial \hat{f}_i} LogLVR = \frac{4}{NM}\left(\frac{\ln\left(\frac{\beta_i}{A^2}-\frac{\gamma_i^2}{A^4}\right)}{\ln(Vf^i)}\right)\frac{\left(\frac{\hat{f}_i}{A^2}-\frac{\gamma_i}{A^4}\right)}{\left(\frac{\beta_i}{A^2}-\frac{\gamma_i^2}{A^4}\right)} \tag{5.21}$$

Note that $\beta_i = \sum_{j=1}^{NM} (\boldsymbol{f}_j^i)^2$ and $\gamma_i = \sum_{j}^{NM} \boldsymbol{f}_j^i$ can be rewritten as $\beta_i = \beta_i' + \hat{f}_i^{\,2}$ and $\gamma_i = \gamma_i' + \hat{f}_i$ where $\beta_i' = \sum_{j,\, j\neq i}^{NM} (\boldsymbol{f}_j^i)^2$ and $\gamma_i' = \sum_{j,\, j\neq i}^{NM} \boldsymbol{f}_j^i$. Note that in this way we can extract the elements of (5.21) that depend on $\hat{f}_i$. Hence we obtain:

$$\frac{\partial}{\partial \hat{f}_i} LogLVR = \frac{4}{NM}\frac{\left(\ln\left(\hat{f}_i^{\,2}\left(\frac{1}{A^2}-\frac{1}{A^4}\right)-\frac{2\hat{f}_i\gamma_i'}{A^4}+\frac{\beta_i'}{A^2}-\frac{(\gamma_i')^2}{A^4}\right)-\ln(Vf^i)\right)\left(\frac{\hat{f}_i}{A^2}-\frac{\gamma_i'}{A^4}-\frac{\hat{f}_i}{A^2}\right)}{\left(\hat{f}_i^{\,2}\left(\frac{1}{A^2}-\frac{1}{A^4}\right)-\frac{2\hat{f}_i\gamma_i'}{A^4}+\frac{\beta_i'}{A^2}-\frac{(\gamma_i')^2}{A^4}\right)}$$

If we assume that A is large enough so that $\frac{1}{A^2} \gg \frac{1}{A^4}$, then:

$$\frac{\partial}{\partial \hat{f}_i} LogLVR = \frac{4}{NMA^2}\frac{\left(\ln\left(\frac{\hat{f}_i^{\,2}}{A^2}-\frac{2\hat{f}_i\gamma_i'}{A^4}+\frac{\beta_i'}{A^2}-\frac{(\gamma_i')^2}{A^4}\right)-\ln(Vf^i)\right)\left(\hat{f}_i\left(1-\frac{1}{A^2}\right)-\frac{\gamma_i'}{A^2}\right)}{\left(\frac{\hat{f}_i^{\,2}}{A^2}-\frac{2\hat{f}_i\gamma_i'}{A^4}+\frac{\beta_i'}{A^2}-\frac{(\gamma_i')^2}{A^4}\right)}$$

Note that $\frac{\beta_i'}{A^2}-\frac{(\gamma_i')^2}{A^4}$ is an approximation to the local variance at pixel *i* neglecting the

contribution of the value of pixel *i* itself. Let $V' = \frac{\beta_i'}{A^2} - \frac{(\gamma_i')^2}{A^4}$. As *A* increases in value, V' approaches the value of the local variance at pixel *i*. Similarly, we can define an approximation to the local mean of pixel *i* as:

$$M' = \frac{1}{A^2} \sum_{j,\, j \neq i}^{NM} \hat{f}_j^i = \frac{\gamma_i'}{A^2}$$

This leaves us with:

$$\frac{\partial}{\partial \hat{f}_i} LogLVR = \frac{4}{NMA^2} \frac{\left(\ln\left(\frac{\hat{f}_i^{\,2}}{A^2} - \frac{2\hat{f}_i M'}{A^2} + V' \right) - \ln(V f^i) \right) \left(\hat{f}_i \left(1 - \frac{1}{A^2} \right) - M' \right)}{\left(\frac{\hat{f}_i^{\,2}}{A^2} - \frac{2\hat{f}_i \gamma_i'}{A^4} + V' \right)} \qquad (5.22)$$

There are some points for which (5.22) is undefined. These points are given by:

$$\hat{f}_i \approx M' \pm A\sqrt{\frac{(M')^2}{A^2} - V'}$$

At these points the function will have an infinitely negative gradient. In the event that $V' > \frac{(M')^2}{A^2}$ then the function will be defined for all values of $\hat{f}_i$. Fortunately this will almost always happen. This can be seen by examining the condition for undefined points to exist:

$$V' \leq \frac{(M')^2}{A^2}$$

Using the formulas for V' and M' we get:

$$\frac{\beta_i'}{A^2} - \frac{(\gamma_i')^2}{A^4} \leq \frac{1}{A^2}\left(\frac{\gamma_i'}{A^2}\right)^2$$

$$\frac{\beta_i'}{A^2} - \frac{(\gamma_i')^2}{A^4} \leq \frac{1}{A^6}(\gamma_i')^2$$

Hence we require that:

$$\frac{\beta_i'}{A^2} \leq (\gamma_i')^2\left(\frac{1}{A^4} + \frac{1}{A^6}\right) \qquad (5.23)$$

However the variance of a set of numbers is always greater than or equal to zero. V' is the variance of the local region of pixel *i* obtained when the value of pixel *i* is set to zero. Hence we have:

$$V' = \frac{\beta_i'}{A^2} - \frac{(\gamma_i')^2}{A^4} > 0$$

Which means that:

$$\frac{\beta_i'}{A^2} \geq \frac{(\gamma_i')^2}{A^4} \tag{5.24}$$

Equation (5.24) of course means that condition (5.23) will only be satisfied when the local variance is very close to zero. As long as steps are taken to ensure that this does not occur, the function will be well defined for all values of $\hat{f}_i$.

The points for which Equation (5.22) is equal to zero are the stationary points of the Log LVR term in (5.18). Equation (5.22) is zero when $\hat{f}_i = \frac{M'}{\left(1 - \frac{1}{A^2}\right)}$. This corresponds to the case where $\hat{f}_i$ is approximately equal to its local mean. Equation (5.22) also has zeroes when $\hat{f}_i$ satisfies:

$$\ln\left(\frac{\hat{f}_i^{\,2}}{A^2} - \frac{2\hat{f}_i M'}{A^2} + V'\right) - \ln(V f^i) = 0$$

This is equivalent to:

$$\frac{\hat{f}_i^{\,2}}{A^2} - \frac{2\hat{f}_i M'}{A^2} + J = 0$$

where $J = V' - V f^i$.

The stationary points are thus given by:

$$\hat{f}_i \approx M' \pm A\sqrt{\frac{(M')^2}{A^2} - J} \tag{5.25}$$

Note that (5.25) indicates that if $J > 0$, (5.25) may not have a real-valued solution. (5.25) is identical to (5.17) in the previous section and so the case by case analysis in Section 5.3.2 is identical for a Log LVR modified cost function as it was for the LVMSE modified cost function.

As with the LVMSE term, the Log LVR term in the cost function is well behaved despite its non-linear nature. In all cases minima exist and by minimising the Log LVR term in the cost function, the local variance of the current pixel's region will always move closer to the target variance estimate.

5.5. Implementation Considerations

A problem to be overcome is that the third terms in the LVMSE based cost functions are not quadratic in nature. When the local variance in the image estimate is much lower than the projected local variances of the original image, the LVMSE term in Algorithm 5.1 becomes large and may force the pixel values to an extreme of the range of acceptable values in order to create a high variance region. The LVMSE term should never completely dominate over the first term in (5.1) since the LVMSE term only attempts to match regions, not pixels, and fine structure within the region will be lost. To remedy this situation, the pixel values are not allowed to change by more than a set amount per iteration. This method appears to work well in practice

and the pixel values converge to a solution after a finite number of iterations. This method however is not required to the same degree in Algorithm 5.2. Algorithm 5.2 was designed to avoid this effect, however this method may still be employed to improve results.

The addition of the LVMSE term into the cost function allows a powerful optimisation to be made to Algorithm 5.1. In regions were the degraded image is very smooth and the variance estimate of the original image is very small, improvement in image processing speed can be achieved by not restoring these pixels. This will not affect the quality of processing since attempting to deconvolve regions where the blurring effect is not noticable by humans can only serve to amplify noise. It is logical not to attempt to restore such regions when using Algorithm 5.1 since the LVMSE based term in the cost function for this algorithm has little effect at low variance regions. Algorithm 5.2 on the other hand was designed to smooth these regions and so it is not necessary to avoid attempting to restore these regions.

5.6 Experimental Results

A number of experiments were conducted to evaluate the performance of the proposed method. Comparisons were made with some well known methods in the literature.

5.6.1 Colour Image Restoration

For the first experiment, colour images were used consisting of three colour planes, red, green, and blue. The image was degraded by a 5 by 5 Gaussian PSF of standard deviation 2.0 applied to each of the colour planes. In addition, additive noise of variance 369.31 was also added to each colour plane. Figure 5.5a shows the original image and Figure 5.5b shows the

degraded image. The degraded image has a SNR of 19.81dB and a LSMSE of 313.05. The SNR was calculated by adding together the signal to noise ratio of each colour plane:

$$SNR = 20\log\left(\frac{\sigma_o^r}{\sigma_n^r} + \frac{\sigma_o^g}{\sigma_n^g} + \frac{\sigma_o^b}{\sigma_n^b}\right) \tag{5.26}$$

Similarly the LSMSE for the entire image was calculated by summing the LSMSEs of each colour plane. A 9 by 9 neighbourhood was used for calculating the local variance. We compared our algorithm with the Wiener filter and the constrained least square (CLS) filter. In this investigation, we assumed that each colour plane in our test image does not have a high level of correlation and so the filters are applied to each colour plane separately. The Wiener restored image is shown in Figure 5.5c and has a SNR of 16.65 dB and a LSMSE of 859.80. The image was also restored using Algorithm 3.3, without the LSMSE term. A constraint factor of λ = 0.001 was chosen. The CLS restored image is shown in Figure 5.5d and has a SNR of 17.26 dB and a LSMSE of 634.04. The image was also restored using the adaptive constraint algorithm from Chapter 4. This image is shown in Figure 5.5e and has a SNR of 19.19 dB and a LSMSE of 195.68. The same degraded image was also restored using the LSMSE modified cost function, Algorithm 5.1. In the LSMSE modified cost function, the value of λ was set to 0.0005. The factor θ was set to be 0.00001 and the image local variance estimate was computed as:

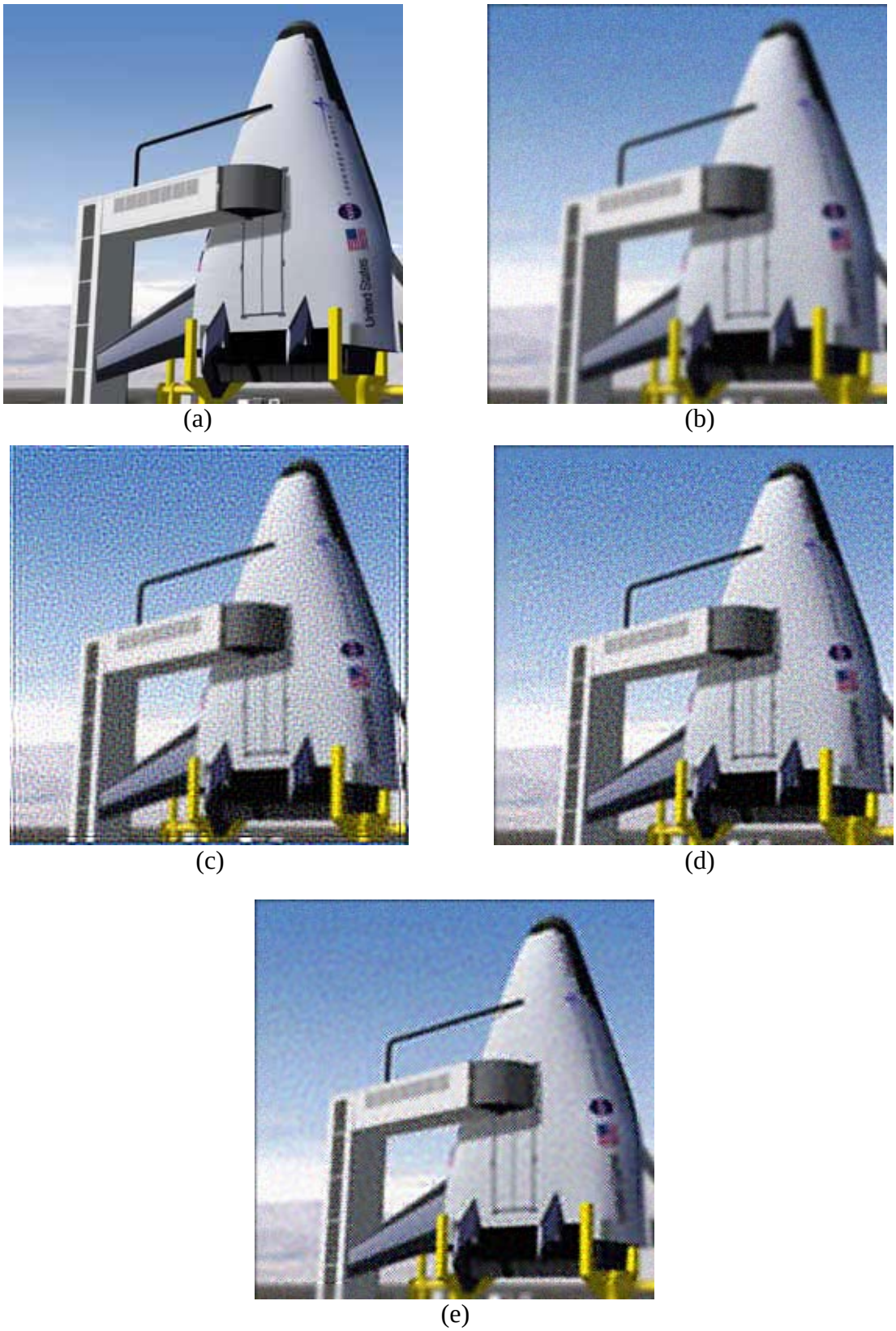
$$\sigma_A^2(\boldsymbol{g}(x,y))^* = 2(\sigma_A^2(\boldsymbol{g}(x,y)) - 200)$$

This image is shown in Figure 5.5f and has a SNR of 19.89 dB and a LSMSE of 180.81. Finally the degraded image was restored using the Log LVR modified cost function, Algorithm

5.2. In the Log LVR modified cost function, the value of λ was set to 0.0005. The factor θ was set to be 50 and the image local variance estimate was computed as for Algorithm 5.1. This image is shown in Figure 5.5g and has a SNR of 21.65 dB and a LSMSE of 88.43.

By visual observation it can be seen that Figures 5.5f and 5.5g, produced by the LSMSE and Log LVR based cost functions, display better noise suppression in background regions and are at the same time sharper than Figure 5.5c and Figure 5.5d, produced by the Wiener and the CLS approaches. Figures 5.5f and 5.5g also display a better SNR and LSMSE than Figures 5.5c, 5.5d and 5.5e. Although the LSMSE restored images are visually closer to the original image than the degraded image, their SNRs are only slightly higher than the degraded image. This is not surprising in view of the arguments above that SNR does not correspond well with human visual perception. However LSMSE does match with human observation and assigns a much lower value to Figures 5.5f and 5.5g. Comparing the two different forms of the LSMSE-based cost functions, we find that Algorithm 5.2, (Figure 5.5g), is superior, with a similar level of sharpness when compared to Figure 5.5f, yet better noise suppression in background regions.

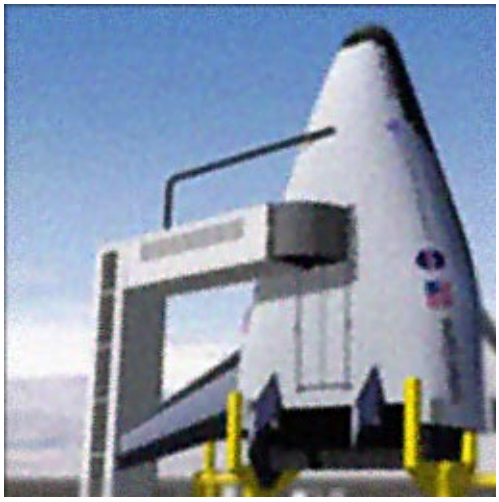
We see that the adaptive constraint method produces a similar result to Algorithm 5.1. This is primarily because both algorithms use the concept of a variance threshold. As mentioned in Section 5.5, if the local variance is below the threshold, the pixel is not adjusted. Both algorithms 3.3 and 5.1 use identical thresholds and so have similar LSMSEs. Algorithm 5.2, however, was designed not to require the variance threshold and instead provides additional smoothing to background regions and hence a much lower LSMSE.



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

Figure 5.5: Colour images restored using various algorithms.

(f)



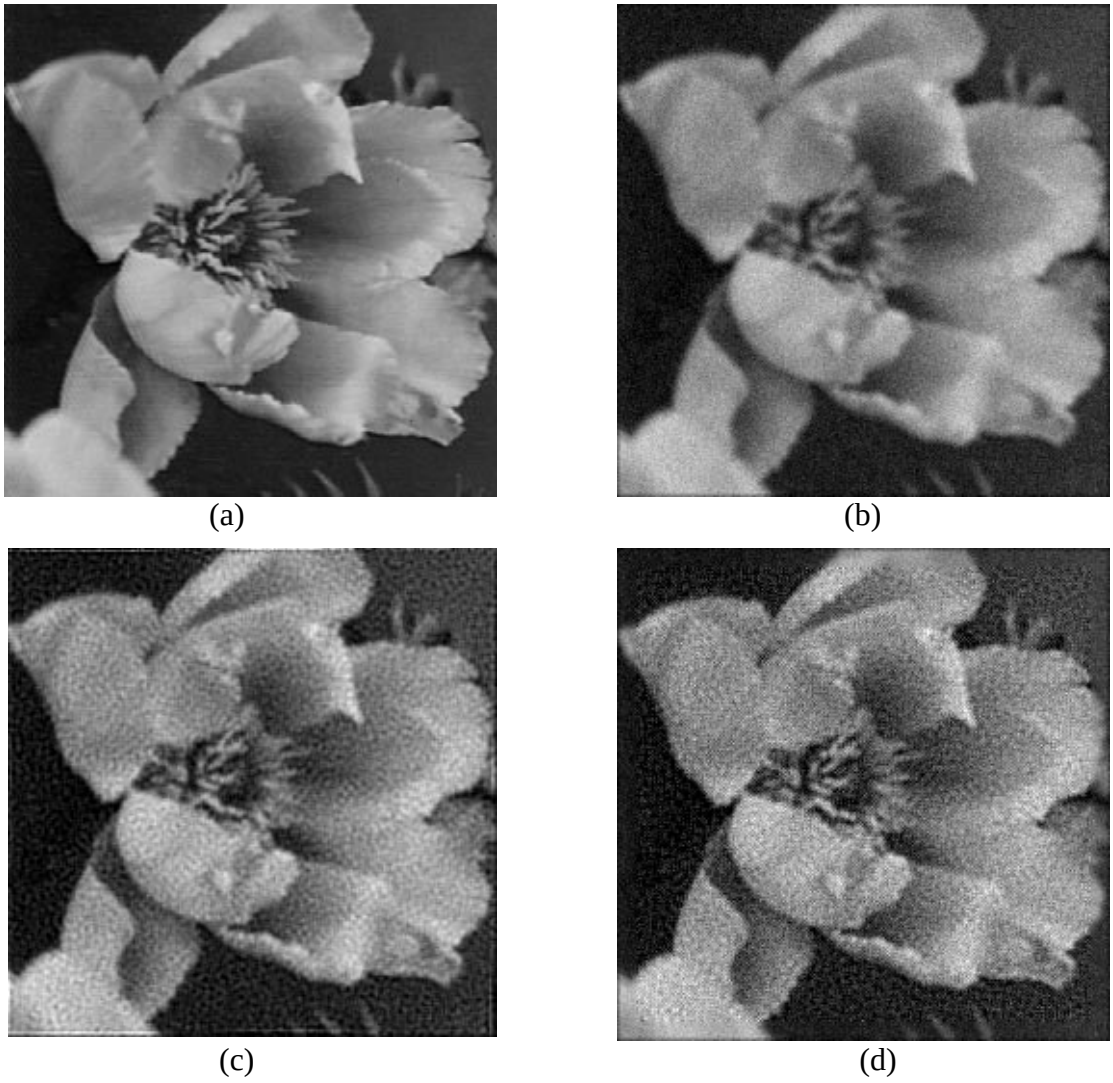
(g)

Figure 5.5 (cont.): Colour images restored using various algorithms.

5.6.2 Grayscale Image Restoration

For the second experiment, a grayscale image was degraded by a 5 by 5 Gaussian PSF of standard deviation 2.0. Additive noise of variance 87.62 was also added. Figure 5.6a shows the original image and Figure 5.6b shows the degraded image. The degraded image has a SNR of 12.58 dB and a LSMSE of 28.13. The degraded image was first restored using a Wiener filter approach. The Wiener restored image is shown in Figure 5.6c and has a SNR of 11.66 dB and a LSMSE of 38.69. The image was also restored using the CLS algorithm (Algorithm 3.3). Figure 5.6d shows the image restored using the CLS algorithm with a constant factor of λ = 0.001. Figure 5.6d has a SNR of 8.76 dB and a LSMSE of 128.09. Figure 5.6e shows the image restored using the CLS algorithm with a constant factor of λ = 0.002. Figure 5.6e has a SNR of 11.93 dB and a LSMSE of 36.91. Figure 5.6f shows the image restored using the adaptive constraint algorithm presented in Chapter 4 using a range of constraint values from 0.02 to 0.0015 associated with levels of local variance. Figure 5.6f has a SNR of 11.97 dB and a LSMSE of 22.28. The degraded image was also restored using the LVMSE modified cost function imple-

mented using Algorithm 5.1. Figure 5.6g shows this image which has a SNR of 12.15dB and a LSMSE of 22.71. Finally the degraded image was restored using the Log LVR modified cost function implemented using Algorithm 5.2. Figure 5.6h shows this image which has a SNR of 12.07 dB and a LSMSE of 20.59. By observation, it can be seen that Figure 5.6h is visually closest to the original image. LSMSE confirms visual inspection and indicates that Figure 5.6h is the most well restored. Note that once again the adaptive algorithm from Chapter 4 performs similarly to the LSMSE-based algorithms. The advantage of the LSMSE algorithms is that they have less free variables to set up.



(a) (b)

(c) (d)

Figure 5.6: Grayscale images restored using various algorithms.

(e)

(f)

(g)

(h)

Figure 5.6 (cont.): Grayscale images restored using various algorithms.

5.6.3 LSMSE of Different Algorithms

For the third experiment, the original flower image was blurred using a 5 by 5 Gaussian blur of standard deviation 2.0. A number of images were created, each suffering a different value of noise. The images were restored using Algorithm 3.3, Algorithm 5.1, Algorithm 5.2, a Wiener filter, and the adaptive constraint algorithm from Chapter 4. For each image, the same value of λ was used in Algorithm 3.3, Algorithm 5.1 and Algorithm 5.2. This meant that the restored

images from Algorithm 3.3, Algorithm 5.1, and Algorithm 5.2 had the same degree of sharpness, but differed in the level of noise suppression. In this way the effects of the LSMSE-based terms in (5.1) and (5.18) could be examined in isolation. Figure 5.7 shows the results of this experiment. It can be clearly seen that in terms of the LSMSE, Algorithms 5.1 and 5.2 outperform the other algorithms, especially the standard CLS approach for the same level of sharpness.

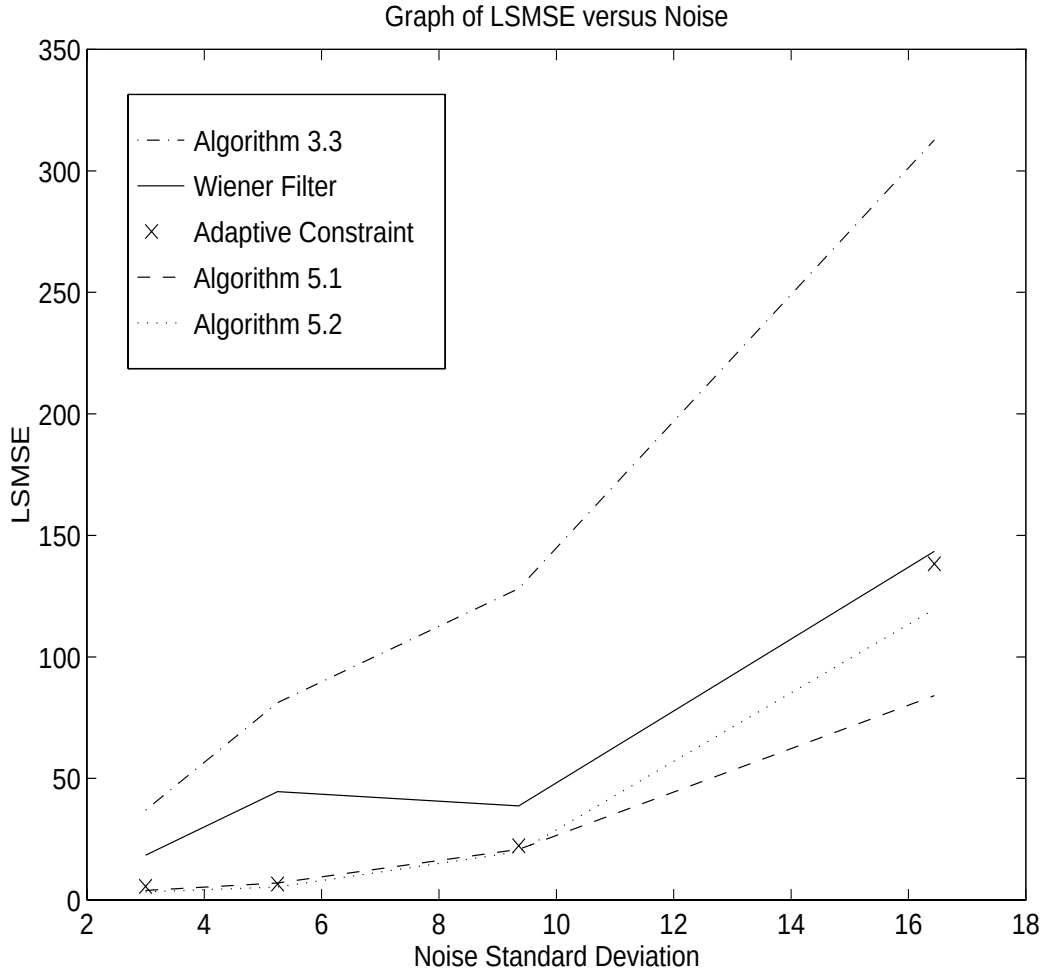
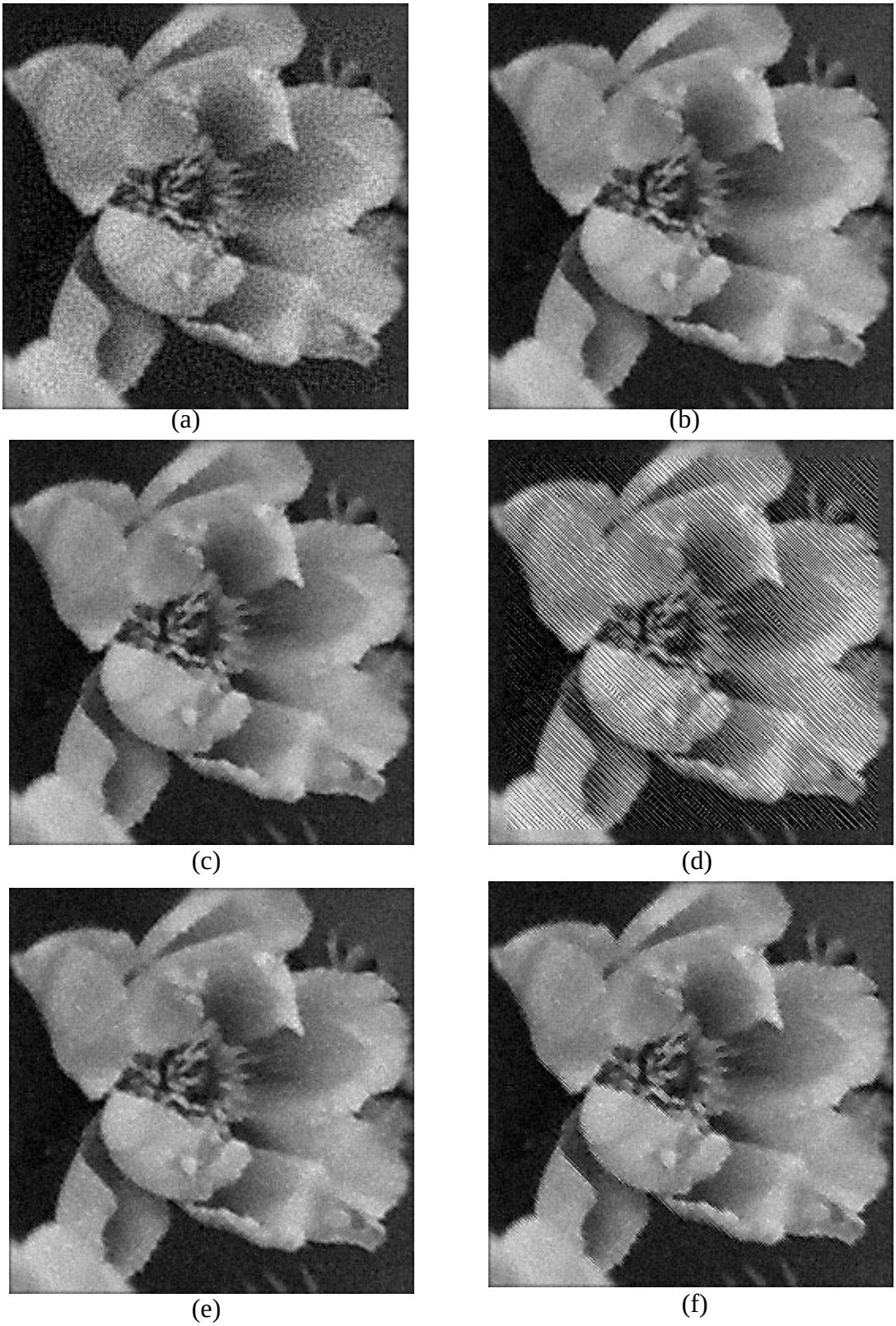


Figure 5.7: Graph of LSMSE for various algorithms and levels of noise.

5.6.4 Robustness Evaluation

For the fourth experiment, the original flower image was blurred using a 5 by 5 Gaussian blur of standard deviation 2.0. Additive noise of variance 87.62 was also added. The degraded image was restored using Algorithm 3.3, Algorithm 5.1 and Algorithm 5.2. In each algorithm, λ was set to 0.001 to maintain the same level of sharpness. Figure 5.8a shows the results of Algorithm 3.3. This image is identical to Figure 5.6d and has a SNR of 8.76 dB and a LSMSE of 128.09. Figure 5.8b shows the results of Algorithm 5.1. Figure 5.8b has a SNR of 12.45 dB and a LSMSE of 20.76. Figure 5.8c shows the results of Algorithm 5.2. Figure 5.8c has a SNR of 12.25 dB and a LSMSE of 19.76. Next we severed one of the neural interconnections to a neighbouring neuron for every neuron in the network. The same connection was severed for each neuron in the network. This would be expected to degrade the performance of the network. Using the same parameters, the restorations were performed again. Figure 5.8d shows the results of restoring the image using Algorithm 3.3 with a faulty network. The SNR is -4.21 dB and the LSMSE is 5889.42. Figure 5.8e shows the results of restoring the image using Algorithm 5.1 with a faulty network. The SNR is 12.15 dB and the LSMSE is 23.23. Figure 5.8f shows the results of restoring the image using Algorithm 5.2 with a faulty network. The SNR is 10.06 dB and the LSMSE is 40.13. From these results we can see that Algorithm 3.3 is not very tolerate of errors in weights. The image produced by the faulty network is very degraded and has poor values of SNR and LSMSE. On the other hand, Algorithm 5.1 and Algorithm 5.2 have almost no visual differences between images restored using the correct network and images restored using the faulty network. The images restored using the faulty network have only slightly worse values of SNR and LSMSE compared to the image restored using the correct network. The reason that Algorithm 5.1 and 5.2 are more fault-tolerant than Algorithm 3.3 is due to the LSMSE-related terms in these algorithms. The damaged weights in

Algorithm 3.3 produced streaks in the image. These streaks would cause the pixels in their vicinity to have very high local variances. Since Algorithm 3.3 does not consider the local regional statistics of the image, the streaks are not suppressed. However Algorithm 5.1 and Algorithm 5.2 attempt to match local variances in the restored image with an estimate of the original image. The streaks are therefore suppressed by Algorithms 5.1 and 5.2. It is clear that Algorithms 5.1 and 5.2 are very robust and are not greatly affected by errors in the network. Algorithm 5.1 is more robust than Algorithm 5.2 on the edges because of the fact that the log-ratio relationship between the local variance and the target variance used by Algorithm 5.2 was developed to de-emphasise the LSMSE effect on edges. Algorithm 5.1 has its greatest effect on edges, whereas Algorithm 5.2 was specifically designed to have the least effect on edges and the greatest effect on smooth regions. However both algorithms are still quite tolerate of net-work errors. This is due to the fact that the LVMSE based terms in these algorithms can never be affected by severed neural connections. The author believes that this aspect is a powerful feature of the proposed algorithms.



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

Figure 5.8: Images restored using correct and faulty networks.

5.7 Summary

In Chapter 2, a novel error measure was introduced which compares two images by consideration of their regional statistical differences rather than their pixel-level differences. It was found that this error measure more closely corresponds to human visual perception of image quality. Based on the new error measure, two cost functions were developed. The first cost function was based closely on the LVMSE error measure introduced in Chapter 2. This cost function was analysed and shown to be well behaved. The analysis of the first modified cost function suggested that improvements could be made by incorporating a logarithmic version of the LVMSE into the standard cost function. The second cost function was hence developed and shown to be well behaved. Algorithms to optimise these cost functions were designed based on adaptation of the neural network approach to optimising constrained least square error.

The proposed algorithms were shown to suppress noise strongly in low variance regions while still preserving edges and highly textured regions of an image. The algorithms were shown to perform well when applied to both grayscale and colour images. It was also shown that the proposed iterative algorithms are very robust.

Chapter 6: Conclusion

6.1 Introduction

This chapter summarises the material presented in this thesis. Section 6.2 details the overall objectives of this thesis. Section 6.3 summarises the background material presented in Chapter 1. Section 6.4 considers the material presented in this thesis regarding the problem of restoring images for human observers. Section 6.5 summarises advances made to the basic neural network algorithm. Section 6.6 details the extension of the basic neural network algorithms to incorporate some concepts of human perception. Section 6.7 examines some future research directions for this field. Section 6.8 concludes this chapter.

6.2 Overview of Thesis Objectives

In the field of image restoration, the recent advances in the use of Hopfield-based neural networks have been exciting. The neural network approach provides a level of flexibility and adaptability which the author believes is far from being fully exploited. For this reason, this thesis shows how this approach can be expanded to handle the cases of semi-blind image restoration, adaptive constraint restoration and the restoration of images suffering from spatially variant degradations.

Another underdeveloped research field is the problem of incorporating concepts of the human visual system into the field of image processing, especially image restoration. The author feels that the field of image processing is extensively intertwined with these concepts,

and as such, they deserve further investigation. Therefore a very important objective of this thesis is to develop usable image fidelity measures, which can describe important human visual criteria, and yet be of a form which allows easy incorporation into image restoration (and general image processing) algorithms. Some such algorithms have been presented in this thesis.

6.3 Background

In Chapter 1, background material was presented. Various types of image degradation and their causes were described. Image degradations could be described as linear or non-linear and spatially variant or spatially invariant. This was followed by a description of the basic concepts involved in the problem of restoring degraded images to approximations of their original forms.

In that chapter a review of the basic classical methods of image restoration was given. These methods were classified into transform-based techniques or algebraic techniques.

Among the transform related techniques we considered restoration methods such as the Inverse filter, Wiener filter, parametric estimation filters, Kalman filters and homomorphic filters.

For the algebraic restoration techniques, we considered methods such as pseudoinverse restoration, singular value decomposition pseudoinverse restoration, Wiener estimation, and constrained restoration.

We pointed out each of these classical methods have one serious drawback or another. Some

of them can be computationally expensive, while most do not adapt easily to varying algorithmic parameters spatially during the restoration procedure. For this reason, we looked at emergent image restoration methods and research directions.

We considered the recent neural network approaches to image restoration. These approaches are increasing in popularity due to the ease of implementing a wide range of filters and the ability of neural networks to alter neuron weights to implement truly adaptive image restoration.

We also studied some new methods for the restoration of images degraded by spatially variant distortions, and we considered general algorithms for adaptive image restoration. Recent research in the difficult field of blind image restoration was presented and a number of recent algorithms described. The problems involved in the restoration of colour images were detailed and recent research papers in this field were summarised.

6.4 Perception Motivated Error Measures

This thesis studied the problem of incorporating concepts from the human visual system into image restoration algorithms. It was seen that images restored without using any concepts from the human visual system can hardly take into account the human concepts of a “good” or “bad” image. Without these concepts, a restoration algorithm will invariably produce suboptimal results from a human point of view.

Most image restoration algorithms are based on some measure of image quality, such as the mean square error (MSE). However error measures such as the MSE are derived from statisti-

cal considerations, such as the idea that a bad image is one where the noise power is too great. These considerations have little relationship to the way that humans perceive images. Humans do not concentrate on the values of individual pixels or the global statistics of the image, but instead are concerned with edges and textures within the image. Edges and textures can be considered as local statistical properties, rather than global ones. Research has been done into various forms of image error measures. Many of these measures rely on complex models, or empirically derived constants, and are hence difficult to incorporate into image restoration algorithms. In order to feasibly incorporate perception into restoration, the author developed a simple, yet powerful, image error measure based on the comparison of local regional statistics. The proposed error measures have two formats; LSMSE (local standard deviation mean square error) and LVMSE (local variance mean square error). An experiment was performed which showed that the LSMSE and the LVMSE provide a better match to human visual appraisal when compared to the SNR.

6.5 Advances on the Basic Neural Network Algorithm

In Chapter 3, previous work performed in the field of neural network based image restoration was reviewed and the basic algorithms described. The author then presented a modified algorithm which uses a more advanced neuron. The new algorithm is able to minimise each neuron's contribution to the overall energy function in one step, rather than the many steps required by previous algorithms. The author then shows that this results in a great increase in speed without any subsequent reduction in restoration quality. The author shows how the algorithm may be extended to the case of restoring images suffering from spatially variant degradations through the use of multiple "weighting masks".

In Chapter 4, the new algorithm is used as a basis for extending the functionality of the neural network restoration technique. In particular, adaptive regularisation is introduced. The adaptive constraint algorithm requires a method of selecting the correct constraint value for each neuron (pixel) in the network. In Chapter 4, the author considers how this can be achieved. It is first shown that using the technique of gradient descent to select the regularisation constant will produce poor results. Given the relevance of local statistics and the human visual system, the author presents a method for assigning constraint values to each pixel based on a log-linear relationship between local variance and the value of the regularisation constant. It is described how using this constraint selection method may not only be able to implement an adaptive regularisation term with a view to improving the visual appearance of images for a human observer, but may also be applicable to the problem of semi-blind image restoration. This is due to the fact that a varying regularisation term can alternately be modelled as a varying estimation of the degrading PSF. The multiple weighting mask concept for space variant processing is then taken a step further to allow adaptive regularisation.

A number of experiments are performed which show the results of applying the adaptive constraint network approach to the problems of standard image restoration, restoration efficiency, and semi-blind image restoration. The adaptive algorithm is shown to perform better than other restoration techniques for these problems.

6.6 Perception-Based Algorithm Modifications

Despite the success of the adaptive constraint algorithm presented in Chapter 4, the author considered the problems involved with using the MSE as the sole measure of image fidelity. In Chapter 5, the author presented two new restoration algorithms, which incorporate error meas-

ures based on the LVMSE error measure presented in Chapter 2.

The first restoration algorithm is based on minimising the constrained least square error function with an additional term which describes the LVMSE of the image and the second cost function is a variation of LVMSE, but with greater emphasis on low levels of local variance, by taking the logarithm of the ratio of the local statistics.

The author presented algorithms to minimise both of these cost functions and presented a number of experiments which show that the LSMSE-modified cost functions perform better than the standard constrained least square error cost functions for the problems of standard grayscale image restoration, colour image restoration, and network error tolerance.

One of the most important findings regarding the modified cost functions is that they are very robust. When interconnections between neurons were severed in the networks, the standard neural network algorithms produced very poor results. However by including details regarding the local statistical properties of the image into the restoration cost function by the use of a LVMSE-related term, images could be restored with very little reduction in quality. This is a powerful feature of the algorithms presented in Chapter 5.

6.7 Future Work

There are a number of future research topics which still remain to be pursued in this field. Better models of the adaptive constraint algorithm could be developed, especially in regard to the method of constraint assignment, which needs to be formalised.

In Chapter 5 we examined the incorporation of LSMSE based terms into the standard constrained least square error cost function. Two variants of LVMSE were considered as candidates for inclusion in modified cost functions. Both measures restored the image adaptively and produced better results when compared to non-adaptive methods, especially when noise levels were high. However there exists other variations of the LSMSE measure. It would be interesting to examine the many possible variations of the LSMSE and the resultant restoration algorithms. Some variations may prove superior to others, or be tailored to a particular image or problem. Here we have tailored the standard restoration algorithms to the problem of restoring images for human appraisal, but more research needs to be done into the more general concept of “problem tailored algorithms”. A future research direction would be incorporating into the algorithm information about the correlation between the colour co-ordinates and comparing this with existing techniques such as the well known Wiener filter proposed by Hunt and Kubler [81].

6.8 Summary

This chapter summarised the material covered in this thesis. The major developments and contributions have been described, as have been the experiments performed and their results.

Appendix A: Calculation of Weights and Bias Inputs

The cost function we are attempting to minimise is given by:

$$E = \frac{1}{2}\|\boldsymbol{g} - \boldsymbol{H}\hat{\boldsymbol{f}}\|^2 + \frac{1}{2}\lambda\|\boldsymbol{D}\hat{\boldsymbol{f}}\|^2 \tag{A1.1}$$

where $\hat{\boldsymbol{f}}$ is the restored image estimate, λ is a constant, and $\boldsymbol{D}$ is a smoothness constraint operator. Let $L = MN$ be the number of neurons (pixels) in the network.

The formula for the energy of the neural network is given by:

$$E = -\frac{1}{2}\hat{\boldsymbol{f}}^T\boldsymbol{W}\hat{\boldsymbol{f}} - \boldsymbol{b}^T\hat{\boldsymbol{f}} + c \tag{A1.2}$$

where the (i,j)th element of $\boldsymbol{W}$ corresponds to the interconnection strength between neurons (pixels) i and j in the network and vector $\boldsymbol{b}$ corresponds to the bias input to each neuron.

Expanding (A1.1) we get:

$$E = \frac{1}{2}\sum_{p=1}^{L}\left(g_p - \sum_{i=1}^{L} h_{pi}\hat{f}_i\right)^2 + \frac{1}{2}\lambda\sum_{p=1}^{L}\left(\sum_{i=1}^{L} d_{pi}\hat{f}_i\right)^2$$

$$= \frac{1}{2}\sum_{p=1}^{L}\left(g_p - \sum_{i=1}^{L} h_{pi}\hat{f}_i\right)\left(g_p - \sum_{j=1}^{L} h_{pj}\hat{f}_j\right) + \frac{1}{2}\lambda\sum_{p=1}^{L}\left(\sum_{i=1}^{L} d_{pi}\hat{f}_i\right)\left(\sum_{j=1}^{L} d_{pj}\hat{f}_j\right)$$

$$= \frac{1}{2}\sum_{p=1}^{L}\left((g_p)^2 - 2g_p\sum_{i=1}^{L}h_{pi}\hat{f}_i + \sum_{i=1}^{L}h_{pi}\hat{f}_i\sum_{j=1}^{L}h_{pj}\hat{f}_j\right) + \frac{1}{2}\lambda\sum_{p=1}^{L}\left(\sum_{i=1}^{L}d_{pi}\hat{f}_i\right)\left(\sum_{j=1}^{L}d_{pj}\hat{f}_j\right)$$

$$= \frac{1}{2}\sum_{p=1}^{L}(g_p)^2 - \sum_{p=1}^{L}\sum_{i=1}^{L}g_p h_{pi}\hat{f}_i + \frac{1}{2}\sum_{p=1}^{L}\sum_{i=1}^{L}h_{pi}\hat{f}_i\sum_{j=1}^{L}h_{pj}\hat{f}_j + \frac{1}{2}\lambda\sum_{p=1}^{L}\sum_{i=1}^{L}d_{pi}\hat{f}_i\sum_{j=1}^{L}d_{pj}\hat{f}_j$$

$$= \frac{1}{2}\sum_{p=1}^{L}\sum_{i=1}^{L}\sum_{j=1}^{L}h_{pj}\hat{f}_j h_{pi}\hat{f}_i + \frac{1}{2}\lambda\sum_{p=1}^{L}\sum_{i=1}^{L}\sum_{j=1}^{L}d_{pj}\hat{f}_j d_{pi}\hat{f}_i - \sum_{p=1}^{L}\sum_{i=1}^{L}g_p h_{pi}\hat{f}_i + \frac{1}{2}\sum_{p=1}^{L}(g_p)^2$$

Hence

$$E = \frac{1}{2}\sum_{i=1}^{L}\sum_{j=1}^{L}\left(\sum_{p=1}^{L}h_{pj}h_{pi} + \lambda\sum_{p=1}^{L}d_{pj}d_{pi}\right)\hat{f}_i\hat{f}_j - \sum_{i=1}^{L}\sum_{p=1}^{L}g_p h_{pi}\hat{f}_i + \frac{1}{2}\sum_{p=1}^{L}(g_p)^2 \quad \text{(A1.3)}$$

Expanding (A1.2) we get:

$$E = -\frac{1}{2}\sum_{i=1}^{L}\sum_{j=1}^{L}w_{ij}\hat{f}_i\hat{f}_j - \sum_{i=1}^{L}b_i\hat{f}_i + c \quad \text{(A1.4)}$$

By equating the terms in equations (A1.3) and (A1.4) we find that the neural network model can be matched to the constrained least square error cost function by ignoring the constant, *c*, and setting:

$$w_{ij} = -\sum_{p=1}^{L}h_{pi}h_{pj} - \lambda\sum_{p=1}^{L}d_{pi}d_{pj} \quad \text{(A1.5)}$$

and

$$b_i = \sum_{p=1}^{L} g_p h_{pi} \tag{A1.6}$$

where w_{ij} is the interconnection strength between pixels i and j, and b_i is the bias input to neuron (pixel) i. In addition, h_{ij} is the (i,j)th element of matrix $\boldsymbol{H}$ from equation (A1.1) and d_{ij} is the (i,j)th element of matrix $\boldsymbol{D}$.

Appendix B: Analysis of the Adaptive Constraint Approximation

The weights for the adaptive constraint method can be computed by generalising the constrained least square error cost function.

$$E = \frac{1}{2}\|\boldsymbol{g} - \boldsymbol{H}\hat{\boldsymbol{f}}\|^2 + \frac{1}{2}\|\sqrt{\Lambda}\boldsymbol{D}\hat{\boldsymbol{f}}\|^2 \qquad \text{(A2.1)}$$

where

$$\sqrt{\Lambda} = \begin{bmatrix} \sqrt{\lambda_1} & 0 & 0 \\ 0 & \dots & 0 \\ 0 & 0 & \sqrt{\lambda_{NM}} \end{bmatrix} \qquad \text{(A2.2)}$$

is a L by L diagonal matrix to reflect the adaptive processing nature. In this model each pixel i is assigned a constraint value λ_i. Define $\boldsymbol{D}' = \sqrt{\Lambda}\boldsymbol{D}$. Note that the elements of $\boldsymbol{D}'$ are given by:

$$d_{ij}' = \sqrt{\lambda_i} d_{ij}$$

Using equation (A1.5), the weight between neurons i and j in the network is given by:

$$w_{ij} = -\sum_{p=1}^{L} h_{pi}h_{pj} - \sum_{p=1}^{L} d_{pi}'d_{pj}'$$

$$= -\sum_{p=1}^{L} h_{pi}h_{pj} - \sum_{p=1}^{L} \lambda_p d_{pi}d_{pj} \qquad \text{(A2.3)}$$

Equation (A2.3) is difficult to implement compared to the single constraint case, (A1.5). However (A2.3) can be greatly simplified. Consider the approximation:

$$w_{ij} = -\sum_{p=1}^{L} h_{pi}h_{pj} - k\lambda_i \sum_{p=1}^{L} d_{pi}d_{pj} \qquad \text{(A2.4)}$$

where λ_i is the constraint value assigned to neuron *i*, and *k* is a scaling constant. Equation (A2.4) is much easier to calculate for each neuron since it only relies on the value of λ assigned to neuron *i*, and does not depend on the values of λ assigned to other neurons in the neighbourhood of neuron *i*. If we have a finite set of available values of λ, then by using (A2.4) we will require only one weighing mask for each λ value in the set. If we instead use (A2.3) we will have to recalculate the weights for each neuron in the network. This will, for obvious reasons, greatly degrade the performance of the network. The next question we must ask is how well does (A2.4) approximate (A2.3) and when can we expect the approximation to break down.

Note that the concept of a “weighting mask” described in Chapter 3 used the fact that a neuron only has non-zero connections to other neurons within a certain neighbourhood centred on the neuron. If a neuron lies in a region where all its neighbours were assigned the same value of λ, then (A2.3) would be the same as the weighting mask produced by using a constant value of λ for all pixels in the image. This means that a constant can be found such that (A2.3) can be accurately approximated by (A2.4) and $w_{ij} = w_{ji}$ in the vicinity of this neuron.

If some of the neurons in the neighbourhood of neuron *i* have different values of λ, but the difference between λ_i and λ_j is not great, then (A2.4) is still a good approximation although $w_{ij} \neq w_{ji}$. However we can expect that $w_{ij} \approx w_{ji}$

If some of the neurons in the neighbourhood of neuron *i* have very different values of λ then the approximation will not hold. We cannot expect that $w_{ij} \approx w_{ji}$. This is not a big problem however, since regions where neighbouring neurons have greatly different values of λ must be edges and highly textured regions. The reasons for this will be further explained in Chapter 4. In such regions, noise is not readily noticable and so any errors made by approximating (A2.3) as (A2.4) will be masked. Hence the approximation provides increased ease of use and only breaks down in those regions where the effects are least noticable.

Appendix C: Analysis of the Factors in the Gradient Descent Method.

FACTOR A

Factor *A* is given by:

$$A = \sum_{j=1}^{L} \sum_{p=1}^{L} h_{pi} h_{pj} \hat{f}_j - b_i = K - b_i$$

where $K = \sum_{j=1}^{L} \sum_{p=1}^{L} h_{pi} h_{pj} \hat{f}_j = [\boldsymbol{H}^T \boldsymbol{H} \hat{\boldsymbol{f}}]_i$.

The factor *K* is the value of pixel $\hat{f}_i$ after a double application of the degrading PSF to the image estimate. Using the approximate formula for $\boldsymbol{H}$, then $\boldsymbol{H}$ is symmetric, $\boldsymbol{H}^T=\boldsymbol{H}$. Note that since $b_i = \sum_{p=1}^{L} h_{pi} y_p$, b_i is the value of pixel $\hat{f}_i$ after a single application of the degrading PSF to the degraded image, which is itself the result of applying the degrading PSF to the original image. Therefore in effect;

$$b_i = \sum_{p=1}^{L} \sum_{j=1}^{L} h_{pi} h_{pj} f_j = [\boldsymbol{H}^T \boldsymbol{H} \boldsymbol{f}]_i \qquad \text{(A3.1)}$$

where $\boldsymbol{f}$ is the estimate of the original when noise is not considered. The algorithm would

eventually return $\boldsymbol{f}$ as the restored image when $\lambda = 0$. Hence we obtain:

$$\boldsymbol{A}_k = \boldsymbol{H}^T\boldsymbol{H}\hat{\boldsymbol{f}} - \boldsymbol{H}^T\boldsymbol{H}\boldsymbol{f} = \boldsymbol{H}^T\boldsymbol{H}(\hat{\boldsymbol{f}} - \boldsymbol{f}) \qquad \text{(A3.2)}$$

where $\boldsymbol{A}_k$ is the column vector whose element i is the value of factor A computed at neuron i during the kth iteration of the algorithm.

Considering (A3.2), it is obvious that the entries in $\boldsymbol{A}_k$ may be positive or negative and will approach zero as the image estimate approaches the original image in the event of no noise. We can expect $\boldsymbol{A}_k$ to have its greatest values in the initial iteration of the algorithm.

If the initial image estimate is $\hat{\boldsymbol{f}} = \boldsymbol{g} = \boldsymbol{H}\boldsymbol{f}$, then:

$$\boldsymbol{A}_1 = \boldsymbol{H}^T\boldsymbol{H}(\boldsymbol{H}\boldsymbol{f} - \boldsymbol{f}) = \boldsymbol{H}^T\boldsymbol{H}(\boldsymbol{H} - \boldsymbol{I})\boldsymbol{f} \qquad \text{(A3.3)}$$

As long as the additive noise is not too severe, the double application of the low-pass degrading PSF given by $\boldsymbol{H}$ will remove most of the noise in the smooth regions of the image. Since $\boldsymbol{H} - \boldsymbol{I}$ is a high pass filter, factor A would tend to be large in high variance regions and small in low variance regions of the image.

FACTOR B

For factor B we have $B = \sum_{j=1}^{L}\sum_{p=1}^{L} d_{pi}d_{pj}\hat{f}_j$, which can be rewritten as:

$$B = [\boldsymbol{D}^T \boldsymbol{D} \hat{\boldsymbol{f}}]_i \qquad \text{(A3.4)}$$

Since $\boldsymbol{D}^T = \boldsymbol{D}$, factor B is hence the value of pixel $\hat{f}_i$ after a high-pass filter has been applied twice to the image estimate. On edges and high texture regions, we would expect the magnitude of factor B to be large. However since the high-pass filter is applied twice, noise and very sharp edges would produce a larger magnitude of B than more gradual edges in the image. It is important to note that B may be positive or negative depending on whether the value of $\hat{f}_i$ is higher or lower than the mean of its neighbours, however B will tend to zero in low variance regions of the image.

FACTORS C & D

Factors C and D are given by $C = \frac{1}{2} \sum_{p=1}^{L} h_{pi}^2$ and $D = \frac{1}{2} \sum_{p=1}^{L} d_{pi}^2$. Both these factors are constant. C will always be quite small and in the case of a degrading PSF modelled by a 5 by 5 Gaussian blur of standard deviation 2.0 in both x and y directions, $C = 0.28$. Factor D depends upon the type of high-pass filter chosen and typical values are of the order of 50 to 500.

REFERENCES

[1] H. C. Andrews and B. R. Hunt, *Digital Image restoration*. Englewood Cliffs, NJ: Prentice-Hall, 1977.

[2] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*. Addison-Wesley, 1992.

[3] F. Roddier, "The effects of atmospheric turbulence in optical astronomy," in *Progress in Optics*, E. Wolf, ed., vol. 19, pp. 281-376, North-Holland, Amsterdam, 1981.

[4] J.P. Oakley and B.L. Satherley, "Improving Image Quality in Poor Visibility Conditions Using a Physical Model for Constrast Degradation," *IEEE Trans. Image Proc.*, vol. 17, no. 2, February, 1998.

[5] E.J. McCartney, *Optics of the Atmosphere: Scattering by Molecules and Particles*, New York, Wiley, 1976.

[6] V.E. Zuev, *Propagation of Visible and Infrared Radiation in the Atmosphere*, New York, Wiley, 1974.

[7] I. Dinstein, H. Zoabi, and N.S. Kopeika, "Prediction of effects of weather on image quality: Preliminary results of model validation," *Appl. Opt.*, vol. 27, pp. 2539-2545, 1988.

[8] M. Bertero, T. Poggio, and V. Torre, "Ill-posed problems in early vision," *Proc. IEEE*, vol.

76, no. 8, pp. 1516-1523, 1991.

[9] A. Katsaggelos, J. Biemond, R.Schafer, and R. Mersereau, “A regularized iterative image restoration algorithm,” *IEEE Trans. Sig. Proc*. vol. SP-39, no. 4, pp. 914-929, 1991.

[10] A. Tikhonov, A. Goncharsky, and V. Stepanov, “Inverse problem in image processing,” in *Ill-Posed Problems in the Natural Sciences,* A. Tikhonov and A. Goncharsky, Eds., pp. 220-232, Mir Publishers, Moscow, 1987.

[11] M. I. Sezan and A. M. Tekalp, “Survey of recent developments in digital image restoration,” *Opt. Eng*., vol. 29, pp. 393-404, May 1990.

[12] S. German and D. German, “Stochastic relaxation, Gibbs distribution, and the Bayesian restoration of images,” *IEEE Trans. Pattern Anal. Machine Intell*., vol. PAMI-6, pp. 721-741, Nov. 1984.

[13] G. Demoment, “Image reconstruction and restoration: Overview of common estimation structures and problems,” *IEEE Trans. Acoust., Speech Sig. Proc.*, vol. 37, pp. 2024-2073, Dec 1989.

[14] A. K. Katsaggelos, ed., *Digital Image Restoration*, New York, Springer-Verlag, 1991.

[15] M.M. Sondhi, "Image Restoration: The Removal of Spatially Invariant Degradations.", *Proc. IEEE*, 60, 7, July 1972, 842-853.

[16] H.C. Andrews, "Digital Image Restoration: A Survey.", *IEEE Computer*, 7, 5, May 1974, 36-45.

[17] B.R. Hunt, "Digital Image Processing", *Proc. IEEE*, 63, 4, April 1975, 36-45.

[18] B.R. Frieden, "Image Enhancement and Restoration.", in *Picture Processing and Digital Filtering*, T.S. Huang, ed., Springer-Verlag, New York, 1975.

[19] M.K. Ozkam, et al, "Efficient Multiframe Wiener Restoration of Blurred and Noisy Image Sequences.", *IEEE Trans. Image Processing*, vol. 1, no. 4, October 1992, 453-478.

[20] E.R. Cole, "The Removal of Unknown Image Blurs by Homomorphic Filtering," Ph.D. Dissertation, Department of Electrical Engineering, University of Utah, Salt Lake City, June 1973.

[21] B.R. Hunt, "The Application of Constrained Least Squares Estimation to Image Restoration by Digital Computer.", *IEEE Trans. Computers*, c-23, 9, September 1973, 805-812.

[22] K. Erler and E. Jernigan, "Adaptive Image Restoration Using Recursive Image Filters.", *IEEE Trans. Signal Proc.*, vol. 42, no. 7, May 1994, 1877-1881.

[23] H.S. Mallikarjuna and L.F. Chaparro, "Iterative Composite Filtering for Image Restoration.",*IEEE Trans. Pattern Analysis and Machine Intelligence*., vol. 14, no. 6, June 1992, 674-677.

[24] W.Wu and A.Kundu, "Image Estimation Using Fast Modified Reduced Update Kalman Filter.", *IEEE Trans. Signal Proc*., vol. 40, no. 4, April 1992, 915-926.

[25] S. Citrin and M.R. Azimi-Sadjadi, "A Full-Plane Block Kalman Filter for Image Restoration.", *IEEE Trans. Image Processing*, vol. 1, no. 4,October 1992, 488-495.

[26] S.J. Reeves and R.M. Mesereau, "Automatic Assessment of Constraint Sets in Image Restoration.", *IEEE Trans. Image Processing*, vol. 1, no. 1, January 1992, 119-122.

[27] D.L. Snyder, et al, "Deblurring Subject to Non-negativity Constraints.", *IEEE Trans. Signal Proc*., vol. 40, no. 5, May 1992, 1143-1150.

[28] J.P. Burg, "Maximum Entropy Spectral Analysis," *37th Annual Society of Exploration Geophysicists Meeting*, Oklahoma City, 1967.

[29] J.A. Edward and M.M. Fitelson, "Notes on Maximum Entropy Processing.", *IEEE Trans. Information Theory*, IT-19, 2, March 1973, 232-234.

[30] P.A. Jansson, R.H. Hunt and E.K. Peyler, "Resolution Enhancement of Spectra.", *J. Opt. Soc. Am.*, 60, 5, May 1970, 596-599.

[31] T.S. Huang, D.S. Baker, and S.P. Berger, "Iterative Image Restoration.", *Appl. Opt*., 14, 5, May 1975, 1165-1168.

[32] M. Gokmen and C. Li, "Edge Detection and Surface Reconstruction Using Refined Regularization.", *IEEE Trans. Pattern Analysis and Machine Intelligence.*, vol. 15, no. 5, May 1993, 492-498.

[33] M.E. Zervakis and A.N. Venetsanopoulos, "Iterative Least Squares Estimators in Nonlinear Image Restoration.", *IEEE Trans. Signal Proc*., vol. 40, no. 4, April 1992, 927-945.

[34] D. P. Bertsekas and J. N. Tsitsiklis, *Parallel and Distributed Computation: Numerical Methods*. Englewood Cliffs, NJ: Prentice Hall, 1989.

[35] Y.T. Zhou, R. Chellappa, Aseem Vaid and B.K. Jenkins, "Image Restoration using a neural network.", *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-36, 1141-1151, July 1988.

[36] J.K. Paik and A.K. Katsaggelos, "Image Restoration Using a Modified Hopfield Network.", *IEEE Trans. Image Processing*, vol. 1, no. 1, January 1992, 49-63.

[37] Y. Iiguni, et al, "A Real-Time Learning Algorithm for a Multilayered Neural Network Based on the Extended Kalman Filter.", *IEEE Trans. Signal Proc*., vol. 40, no. 4, April 1992, 959-966.

[38] R.J. Steriti and Micheal A. Fiddy, "Regularised Image Reconstruction using SVD and a Neural Network Method for Matrix Inversion.", *IEEE Trans. Signal Proc*., vol. 41, no. 10, October 1993, 3074-3077.

[39] S. W. Perry and L. Guan, “Neural network restoration of images suffering space-variant distortion,” *Electronic Letters*, vol. 31, no. 16, pp. 1358-1359, Aug 1995.

[40] S. Koch, H. Kaufman, and J. Biemond, “Restoration of spatially varying blurred images using multiple model-based Kalman filters,” *IEEE Trans. Image Proc*., vol. 4, no. 4, pp. 520-523, April 1995.

[41] M. K. Ozkan, A. M. Tekalp, and M. I. Sezan, “POCS-based restoration of space-varying images,” *IEEE Trans. Image Proc*., vol. 3, no. 4, pp. 450-454, July 1994.

[42] H.K. Sung and H.M. Choi, “Nonlinear restoration of Spatially Varying Blurred Images Using Self-Organising Neural Network,” *Proc. International Conference on Acoustics, Speech and Signal Processing*., vol. 2, pp. 1097, Seattle, Wash., USA, May 12-15, 1998

[43] A. J. Patti, M. Tekalp, and M. I. Sezan, “A New Motion-Compensated Reduced Order Model Kalman Filter for Space-Varying Restoration of Progressive and Interlaced Video,” *IEEE Trans. Image Processing*, vol. 7, no. 4, pp. 543-554, April 1998.

[44] N. P. Galatsanos and A. K. Katsaggelos, “Methods for choosing the regularization paraeter and estimating the noise variance in image restoration and their relation,” *IEEE Trans. Image Proc*., vol. 1, pp. 322-336, July 1992.

[45] A. M. Thompson, J. C. Brown, J. W. Kay, and D. M. Titterington, “A study of methods of choosing the smoothing parameter in image restoration by regularization,” *IEEE Trans. Pat-*

tern Anal. Machine Intell., vol. 13, pp. 703-714, July 1991.

[46] N. B. Karayiannis and A. N. Venetsanopoulos, “Regularization theory in image restoration-The stabilizing functional approach,” *IEEE Trans. Acoust., Speech, Sig. Proc.*, vol.38, no. 7, pp. 1155-1179, July 1990.

[47] W. Qian, and L. P. Clarke, “Wavelet-Based Neural Network with Fuzzy-Logic Adaptivity for Nuclear Image Restoration,” *Proc. of the IEEE*, vol. 84., no. 10., pp. 1458-1473, October 1996.

[48] H.S. Wong and L. Guan, “Adaptive regularization in image restoration using a model-based neural network,” *Opt. Eng*., vol. 36, no. 12, pp. 3297-3308, Dec. 1997.

[49] M. G. Kang and A. K. Katsaggelos, “General Choice of the Regularization Functional in Regularized Image Restoration,” *IEEE Trans. Image Proc*., vol. 4, no. 5, pp. 594-603, May. 1995.

[50] S. J. Reeves, “Optimal Space-Varying Regularization in Iterative Image Restoration,” *IEEE Trans. Image Proc*., vol. 3, no. 3, pp. 319-324, May 1994.

[51] G. Wang, J. Zhang, and G. W. Pan, “Solution of inverse problems in image processing by wavelet expansion,” *IEEE Trans. Image Proc*., vol. 4, no. 5, pp. 579-593, May 1995.

[52] M. R. Banham and A. K. Katsaggelos, “Spatially Adaptive Wavelet-Based Multiscale Image Restoration,” *IEEE Trans. Image Proc.*, vol. 5, no. 4, pp. 619-634, April 1996.

[53] D. Kundur and D. Hatzinakos, “Blind Image Deconvolution,” *IEEE Signal Processing Magazine*, pp. 43-64, May 1996.

[54] D. Kundur and D. Hatzinakos, “Blind Image Deconvolution Revisited,” *IEEE Signal Processing Magazine*, pp. 61-63, November 1996.

[55] A.E. Savakis and H.J. Trussell, "Blur Identification by Residual Spectral Matching.", *IEEE Trans. Image Processing*, vol. 2, no. 2, April 1993, 141-151.

[56] S.J. Reeves and R.M. Mersereau, "Blur Identification by the Method of Generalized Cross-Validation.", *IEEE Trans. Image Processing*, vol. 1, no. 3, July 1992, 301-311.

[57] N.P. Galatsanos and A.K. Katsaggelos, "Methods for Choosing the Regularisation Parameter and Estimating the Noise Variance in Image Reconstruction and Their Relation.", *IEEE Trans. Image Processing*, vol. 1, no. 3, July 1992, 322-336.

[58] G.Pavlovic and A.M Tekalp, "Maximum Likelihood Parametric Blur Identification Based on a Continuous Spatial Domain Model.", *IEEE Trans. Image Processing*, vol. 1, no. 4, October 1992, 496-504.

[59] H.J. Trussell and S. Fogel, "Identification and Restoration of Spatially Variant Motion Blurs in Sequential Images.", *IEEE Trans. Image Processing*, vol. 1, no. 1, January 1992, 123-126.

[60] E. Anarim, H. Ucar and Y. Istefanopulos, “Identification of Image and Blur Parameters in Frequency Domain Using the EM Algorithm,” *IEEE Trans. Image Proc.*, vol. 5, no. 1, January 1996.

[61] T.G. Stockham Jr., T.M. Cannon, and P.M. Ingebretsen, "Blind Deconvolution Through Digital Signal Processing.", *Proc. IEEE*, 63, 4, April 1975, 678-692.

[62] D. Slepian, "Restoration of Photographs Blurred by Image Motion", *Bell Syst. Tech. J.*, XLVI, 10, December 1967, 2353-2362.

[63] L. Guan and R.K. Ward, "Restoration of Randomly Blurred Images Via the Maximum A Posteriori Criterion.", *IEEE Trans. Image Processing*, vol. 1, no. 2, April 1992, 256-262.

[64] R.K. Ward, "Restoration of Differently Blurred Versions of an Image with Measurement Errors in the PSF.", *IEEE Trans. Image Processing*, vol. 2, no. 3, July 1993, 369-381.

[65] K. Sivakumar and U.B. Desai, "Image Restoration Using a Multilayer Perceptron with a Multilevel Sigmoidal Function.", *IEEE Trans. Signal Proc.*, vol. 40, no. 5, May 1992, 2018-2021.

[66] V. Mesarovic, N. Galatsanos, R. Molina, A. K. Katsaggelos, “Hierarchical Bayesian Image Restoration From Partially-Known Blurs,” *Proc. International Conference on Acoustics, Speech and Signal Processing*., vol. 5, pp. 2905, Seattle, Wash., USA, May 12-15, 1998.

[67] H. T. Pai, J. W. Havlicek, A. C. Bovik, “Generically Sufficient Conditions For Exact Multichannel Blind Image Restoration,” *Proc. International Conference on Acoustics, Speech and Signal Processing*., vol. 5, pp. 2861, Seattle, Wash., USA, May 12-15, 1998.

[68] K. May, T. Stathaki, A. K. Katsaggelos, “Blind Image Restoration Using Local Bound Constraints,” *Proc. International Conference on Acoustics, Speech and Signal Processing*., vol. 5, pp. 2929, Seattle, Wash., USA, May 12-15, 1998.

[69] Y. L. You and M. Kaveh, “A Regularization Approach to Joint Blur Identification and Image Restoration,” *IEEE Trans. Image Proc*., vol. 5, no. 3, pp. 416-428, March 1996.

[70] G. R. Ayers and J. C. Dainty, “Iterative blind deconvolution method and its applications,” *Opt. Lett.*, **13**, pp. 547-549, 1988.

[71] B. L. K. Davey, R. G. Lane, and R. H. T. Bates, “Blind deconvolution of noisy complex-valued image,” *Opt. Commun.*, **69**, pp. 353-356, 1989.

[72] J. R. Feinup, “Phase retrieval algorithms: a comparison,” *Appl. Opt.*, **21**, pp. 2758-2769, 1982.

[73] D. A. Fish, A. M. Brinicombe, and E. R. Pike, “Blind deconvolution by means of the Richardson-Lucy algorithm,” *J. Opt. Soc. Am.*, vol. 12, no. 1, pp. 58-65, January 1995.

[74] W. H. Richardson, “Bayesian-based iterative method of image restoration,” *J. Opt. Soc. Am.*, **62**, pp. 55-59, 1972.

[75] L. B. Lucy, “An iterative technique for the rectification of observed distributions,” *Astron. J.*, **79**, pp. 745-754, 1974.

[76] T. J. Holmes, “Blind deconvolution of quantum-limited incoherent imagery: maximum likelihood approach,” *J. Opt. Soc. Am.*, A **9**, pp. 1052-1061, 1992.

[77] A. P. Dempster, N. M. Laird, and D. B. Rubin, “Maximum likelihood from incomplete data via the EM algorithm,” *J. R. Stat. Soc.*, **39**, pp. 1-38, 1977.

[78] E. Thiebaut and J. M. Conan, “Strict *a priori* constraints for maximum-likelihood blind deconvolution,” *J. Opt. Soc. Am.*, vol. 12, no. 3, March 1995.

[79] R. G. Lane, “Blind deconvolution of speckle images,” *J. Opt. Soc. Am.*, A **9**, pp. 1508-1514, 1992.

[80] G. Sharma and H. J. Trussell, “Digital Color Imaging,” *IEEE Trans. Image Proc.*, vol. 6, no. 7, pp. 901-932, July 1997.

[81] B. R. Hunt, O. Kubler, “Karhunen-Loeve multispectral image restoration, Part I:Theory,” *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-32, no. 3, pp. 595-600, June 1984.

[82] N. P. Galatsanos, A. K. Katsaggelos, R. T. Chin and A. D. Hillery, “Least Squares Restoration of Multichannel Images,” *IEEE Transactions on Signal Proc.*, vol. 39, no. 10, Octo-

ber 1991.

[83] J. Beck, B. Hope, and A. Rosenfeld, ed., *Human and Machine Vision*, Orlando, Florida, Academic Press, INC., 1983.

[84] J. Beck, "Effect of orientation and of shape similarity on perceptual grouping," *Perception and Psychophysics*, **1**, pp. 300-302, 1966.

[85] A. Gagalowicz, "A new method for texture field synthesis: Some applications to the study of human vision," *IEEE Trans. Pattern Analys. Mach. Intell.*, **3**, pp. 520-533, 1981.

[86] B. Julesz, "Experiments in the visual perception of texture," *Scientific American*, **232** (4), pp. 34-43, 1975.

[87] M. Fahle and T. Poggio, "Visual hyperacuity: Spatiotemporal interpolation in human stereo vision," *Proc. of the Royal Society,* London, **B213**, pp. 451-477, 1981.

[88] D. Marr, S. Ullman, and T. Poggio, "Bandpass channels, zero-crossings and early visual information processing," *J. Opt. Soc. Am.*, **69**, pp. 914-916, 1979.

[89] F. Attneave, "Some informational aspects of visual perception, " *Psychological Review*, **61**, pp. 183-193, 1954.

[90] G. Johansson, "Visual motion perception," *Scientific American*, **232** (6), pp. 76-88, 1975.

[91] R. F. Hess and D. R. Badcock, “Metric for separation discrimination by the human visual system,” *J. Opt. Soc. Am.*, vol. 12, no. 1, pp. 3-16, January, 1995.

[92] D. M. Levi and S. A. Klein, “Weber law for position: the role of spatial frequency and contrast,” *Vision Res.*, **32**, pp. 2235-2250, 1992.

[93] R. F. Hess and I. Holliday, “The coding of spatial position by the human visual system: Spatial scale and contrast,” *Vision Res.*, **32**, pp. 1085-1097, 1992.

[94] G. L. Anderson and A. N. Netravali, “Image Restoration Based on a Subjective Criterion,” *IEEE Trans. Systems, Man, and Cybernetics.*, vol. SMC-6, no. 12, pp. 845-853, December 1976.

[95] T. A. Hentea and V. R. Algazi, “Perceptual Models and the Filtering of High Contrast Achromatic Images,” *IEEE Trans. Systems, Man, and Cybernetics.*, vol. SMC-14, no. 2, pp. 230-246, April 1984.

[96] V. R. Algazi, G. E. Ford, and H. Chen, “Linear Filtering of Images Based on Properties of Vision,” *IEEE Trans. Image Proc*., vol. 4, no. 10, pp. 1460-1464, October. 1995.

[97] X. Ran and N. Farvardin, “A Perceptually Motivated Three-Component Image Model - Part I: Description of the Model,” *IEEE Trans. Image Proc*., vol. 4, no. 4, pp. 401-415, April. 1995.

[98] A. Bellini, A. Leone, R. Rovatti, “Fuzzy Perceptual Filtering for Still Image Compres-

sion," *Proc. 1998 IEEE World Congress on Computational Intelligence*, pp. 1637-1642, Anchorage, Alaska, USA, May 4-9, 1998.

[99] I. Hontsch and L. J. Karam, "Locally-Adaptive Image Coding Based on a Perceptual Target Distortion," *Proc. International Conference on Acoustics, Speech and Signal Processing.*, vol. 5, pp. 2569, Seattle, Wash., USA, May 12-15, 1998.

[100] J. J. Huang and E. J. Coyle, "Perceptually Optimal Restoration of Images with Stack Filters," *Proc. International Conference on Acoustics, Speech and Signal Processing.*, vol. 5, pp. 2873, Seattle, Wash., USA, May 12-15, 1998.

[101] S. Daly, "The visible difference predictor: an algorithm for the assessment of image fidelity," Chapter 14 in *Digital Images and Human Vision*, ed. by A. B. Watson, MIT Press, 1993.

[102] J. A. Anderson and J. P. Sutton, "A network of networks: Computation and neurobiology," *World Congress of Neural Networks*, vol. 1, pp. 561-568, 1995.

Publication List

Papers Published

Stuart W. Perry and Ling Guan, “Neural Network Restoration of Images Suffering Space-Variant Distortion”, *Electronic Letters*, vol. 31, no. 16, pp 1358-1359, 1995.

Stuart W. Perry and Ling Guan, “Image restoration using a neural network with an adaptive constraint factor”, *Proc. International Conference on Neural Networks in Signal Processing*., pp. 1031-1034, Nanjing, P. R. China., Dec. 10-20, 1995.

Stuart W. Perry and Ling Guan, “Restoration of Images Degraded by Space-Variant Distortion Using a Neural Network”, *Proc. IEEE Conference on Neural Networks*., Vol. 4., pgs 2067-2070, Perth, Australia., Nov. 26-Dec. 2., 1995.

Stuart W. Perry and Ling Guan., “A Partitioned Modified Hopfield Neural Network Algorithm for Real-Time Image Restoration”, *Journal of Real-Time Imaging*., vol. 2, pp. 215-224, 1996.

Stuart W. Perry and Ling Guan, “Image Restoration using a Neural Network with an Adaptive Constraint Factor”, (invited paper) *Proc. Computational Engineering in Systems Applications Conference*, pp.854-859, Lille, France., July 9-12, 1996.

Stuart W. Perry and Ling Guan, “Segmentation and Visualisation of 3D Underwater Ultrasonic Images,” *Proc. Australian Pattern Recognition Society Workshop*, pp. 69-74, University of

Technology, Sydney, December. 1996.

Stuart W. Perry and Ling Guan, “Adaptive Constraint Restoration and Error Analysis Using a Neural Network”, *Proc. Australian Joint Conference on Artificial Intelligence*, pp. 87-95, Perth, Australia, 1997.

Stuart W. Perry and Ling Guan, “Perception Based Adaptive Image Restoration”, *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, vol. V, pp. 2893-2896, Seattle, Washington, USA, May 12-15, 1998.

Stuart W. Perry and Ling Guan, “A Statistics-Based Weight Assignment in a Hopfield Neural Network for Adaptive Image Restoration”, *Proc. IEEE International Joint Conference on Neural Networks*, pp. 992-997, Anchorage, Alaska, USA, May 5-9, 1998.

Edwin P.K. Wong, Ling Guan and Stuart W. Perry., “A Neural Network Implementation of the SMSE Filter for Image Processing”, *Proc. IS&T/SPIE Symposium on Electronic Imaging: Science & Technology*., Vol. 2661, pp. 77-85, San Jose, USA, Jan. 29-30, 1996.

L Guan, S. Perry, R. Romagnoli, H.S Wong and H.S. Kong, “Neural Vision System and Applications in Image Processing and Analysis”, (invited paper) *Proc. IEEE Int. Conf. on Acoustics, Speech and Sig. Process*., vol. II, pp. 1245-1248, Seattle, USA, May 12-15, 1998.

Ling Guan, Stuart W. Perry, and Edwin P.K. Wong., “Adaptive Statistical Filtering by Neural Networks”, *Journal of Electronic Imaging*., (Special Issue on Real-Time Imaging), vol. 5, pp. 460-465, October 1996.

J. P. Sutton, L. Guan and S. Perry, “Sonar image enhancement in turbid environments”, *Proc. Joint Conference on Information Sciences.*, vol. II, pp. 1-3, ed. Paul P. Wang, Assoc. for Intelligent Machinery, Inc., Research Triangle Park, North Carolina, USA, 1998.

Papers in review

Stuart W. Perry and Ling Guan, “Perception Based Adaptive Image Restoration”, conditional acceptance for publication in *IEEE Transactions on Image Processing*.

Stuart W. Perry and Ling Guan, “Weight Assignment for Adaptive Image Restoration by Neural Networks”, under review for possible publication in *IEEE Transactions on Neural Networks*.

GLOSSARY

Cost Function:	A function of any number of variables whose output is a single number. The function can be minimised to find an optimal set of parameters.
Blind Image Restoration:	Image restoration performed with no knowledge of the degrading PSF details.
Human Visual System:	The human biological system for the processing of visual information. This description includes visual properties and processing inherent in the eyes, optical nerve and visual centres of the brain.
Ill-Posed Problem:	A problem with too few constraints and hence has a number of possible solutions. Image processing problems are often of this form.
Neural Networks:	A computational model consisting of numerous simple processing units connected in a grid to each other. Motivated by biological systems.
Point Spread Function (PSF):	A two dimensional impulse response. The response of a single point source to a linear imaging system.
Restoration:	The correction of an image suffering some form of distortion, noise or degradation.
Semi-Blind Image Restoration:	Image restoration performed with incomplete knowledge of the degrading PSF.
Spatially Invariant PSF:	A PSF whose form does not depend on its position in the image.

Spatially Variant PSF:	A PSF whose form varies according to its position in the image.