Parallel algorithms for restoration of degraded images using Gibbs field models.

Magdi Abd-Elsalam Elgabali

University of Windsor

Follow this and additional works at: https://scholar.uwindsor.ca/etd

Recommended Citation
https://scholar.uwindsor.ca/etd/2989

This online database contains the full-text of PhD dissertations and Masters’ theses of University of Windsor students from 1954 forward. These documents are made available for personal study and research purposes only, in accordance with the Canadian Copyright Act and the Creative Commons license—CC BY-NC-ND (Attribution, Non-Commercial, No Derivative Works). Under this license, works must always be attributed to the copyright holder (original author), cannot be used for any commercial purposes, and may not be altered. Any other use would require the permission of the copyright holder. Students may inquire about withdrawing their dissertation and/or thesis from this database. For additional inquiries, please contact the repository administrator via email (scholarship@uwindsor.ca) or by telephone at 519-253-3000 ext. 3208.
NOTICE

The quality of this microform is heavily dependent upon the quality of the original thesis submitted for microfilming. Every effort has been made to ensure the highest quality of reproduction possible.

If pages are missing, contact the university which granted the degree.

Some pages may have indistinct print especially if the original pages were typed with a poor typewriter ribbon or if the university sent us an inferior photocopy.

Reproduction in full or in part of this microform is governed by the Canadian Copyright Act, R.S.C. 1970, c. C-30, and subsequent amendments.

AVIS

La qualité de cette microforme dépend grandement de la qualité de la thèse soumise au microfilmage. Nous avons tout fait pour assurer une qualité supérieure de reproduction.

S'il manque des pages, veuillez communiquer avec l'université qui a conféré le grade.

La qualité d'impression de certaines pages peut laisser à désirer, surtout si les pages originales ont été dactylographiées à l'aide d'un ruban usé ou si l'université nous a fait parvenir une photocopie de qualité inférieure.

La reproduction, même partielle, de cette microforme est soumise à la Loi canadienne sur le droit d'auteur, SRC 1970, c. C-30, et ses amendements subséquents.
PARALLEL ALGORITHMS FOR RESTORATION OF DEGRADED IMAGES USING
GIBBS FIELD MODELS

by

© MAGDI ABD-EL-SALAM ELGABALI

Submitted to the
Faculty of Graduate Studies and Research
through the Department of
Electrical Engineering in Partial Fulfillment
of the requirements for the degree
of Doctor of Philosophy at
the University of Windsor

Windsor, Ontario, Canada
1988
The author has granted an irrevocable non-exclusive licence allowing the National Library of Canada to reproduce, loan, distribute or sell copies of his/her thesis by any means and in any form or format, making this thesis available to interested persons.

The author retains ownership of the copyright in his/her thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without his/her permission.

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

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

DEDICATION

TO MY MOTHER AND THE MEMORY OF MY FATHER
TO MY WIFE SAFAA, MY SON SHERIF and MY DAUGHTER HODA
TO MY BROTHER MOUSTAFA and MY SISTER HODA
ABSTRACT

In this thesis, restoration of noisy images using Markov Random Field (MRF) models for the clean images and the Maximum A Posteriori Probability (MAP) approach is considered.

The degradation model is very general, in that it incorporates (a) additive and multiplicative random noise (b) nonlinear distortion and (c) focus blur.

Several algorithms are derived for restoration purposes, depending on the a priori information about the nature of degradations. These algorithms will allow for a very efficient implementation to obtain the restored images. Also a general restoration algorithm has been developed for the case when the image has been corrupted by all the different components of the degradation. The algorithm was tested on a class of images representing:

1. Images ranging from binary (two levels) to continuous grey scale.

2. Degradation ranging from simple additive noise to the general degradation including additive/multiplicative noise, focus blur and nonlinear distortion.

Test results indicate both the feasibility as well as the robustness of the algorithms.

Finally, it is shown that the algorithms lend themselves to efficient hardware implementation, using SIMD/MIMD architecture as well as the more modern parallel architecture.
ACKNOWLEDGEMENTS

My sincere thanks and gratitude are due to his Almighty ALLAH, who helped and blessed me during the painful days of my study.

I wish to express my appreciation and thanks to my advisors: Professor M. Ahmadi and Professor M. Shridhar. Professor M. Shridhar for initiating the research project, for his continuous support, and more important for guiding me toward a career in computer vision. Professor M. Ahmadi for his help, encouragement and inspiration throughout the duration of my graduate studies at the University of Windsor.

I also wish to sincerely thank Dr. J. J. Soltis and Dr. R. G. Gaspar for serving on my research committee. Sincere thanks also due to the Dean of Graduate Studies and Research, Dr. Lois. K. Smedick for carrying out her big responsibilities in supporting and encouraging the Graduate School at the University of Windsor. I wish also to express my gratitude to the NSERC of Canada for financially supporting this research.

My wife Safaa, my son Sherif and my daughter Hoda deserve special recognition for their support and moral assistance during the hectic days of dissertation preparation.

Finally, I am grateful to my mother, my late father, my brothers and my sisters for their support and encouragement.
**ABBREVIATIONS AND NOTATION**

### Abbreviations

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AAP</td>
<td>Adaptive Array Processor</td>
</tr>
<tr>
<td>BRES</td>
<td>Binary Restoration</td>
</tr>
<tr>
<td>CLIP</td>
<td>Cellular Logic Image Processor</td>
</tr>
<tr>
<td>CRT</td>
<td>Cathod Ray Tube</td>
</tr>
<tr>
<td>DAP</td>
<td>Distributed Array Processor</td>
</tr>
<tr>
<td>GD</td>
<td>Gibbs Distribution</td>
</tr>
<tr>
<td>i.i.d.</td>
<td>Identically and Independent Distributed</td>
</tr>
<tr>
<td>LSB</td>
<td>Least Significant Bit</td>
</tr>
<tr>
<td>MAP</td>
<td>Maximum A Posteriori</td>
</tr>
<tr>
<td>MIMO</td>
<td>Multi Input Multi Output</td>
</tr>
<tr>
<td>ML</td>
<td>Maximum Likelihood</td>
</tr>
<tr>
<td>MMSE</td>
<td>Minimum Mean-Squared Error</td>
</tr>
<tr>
<td>MRF</td>
<td>Markov Random Field</td>
</tr>
<tr>
<td>MPP</td>
<td>Massively Parallel Processor</td>
</tr>
<tr>
<td>MSB</td>
<td>Most Significant Bit</td>
</tr>
<tr>
<td>NSHP</td>
<td>Non-Symmetrical Half Plane</td>
</tr>
<tr>
<td>PBR</td>
<td>Partial Binary Restoration</td>
</tr>
<tr>
<td>PE</td>
<td>Processing Element</td>
</tr>
<tr>
<td>PET</td>
<td>Photon Emission Tomography</td>
</tr>
<tr>
<td>PDF</td>
<td>Probability Density Function</td>
</tr>
<tr>
<td>Acronym</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>----------------------------------</td>
</tr>
<tr>
<td>Pixel</td>
<td>Picture Element</td>
</tr>
<tr>
<td>PSF</td>
<td>Point Spread Function</td>
</tr>
<tr>
<td>RV</td>
<td>Random Variable</td>
</tr>
<tr>
<td>SAR</td>
<td>Synthetic Aperture Radar</td>
</tr>
<tr>
<td>SD</td>
<td>Standard Deviation</td>
</tr>
<tr>
<td>SIPSF</td>
<td>Space Invariant Point Spread Function</td>
</tr>
<tr>
<td>SVPSF</td>
<td>Space Variant Point Spread Function</td>
</tr>
<tr>
<td>TV</td>
<td>Television</td>
</tr>
<tr>
<td>WGN</td>
<td>White Gaussian Noise</td>
</tr>
</tbody>
</table>
NOTATION

Lt is a representation of a lattice of picture elements defined on a rectangular array as follows: \( \text{Lt} = \{ (i,j) \in \mathbb{Z}^2 : 1 \leq i \leq N1, 1 \leq j \leq N2 \} \)

X is a representation of the true scene Random Field

Y is a representation of the corrupted image Random Field

N is a representation of the additive/multiplicative WGN Random Field

U is a representation of the Multiplicative WGN Random Field

V is a representation of the Compound WGN Random Field

W is a representation of the Additive WGN Random Field

\( X_{ij}, Y_{ij}, N_{ij} \ldots \) is a representation of the associated Random Variables, \((i,j) \in \text{Lt}\)

x, y, n \ldots is a representation of specific realizations

\( \eta_{ij} \) is a representation of a neighbourhood system of order k

\( \eta_{ij}^k \) is a representation of a neighbourhood of order k at \((i,j)\)

F is a Linear Mapping Function

H is a representation of the blurring window PSF

\( \phi \) is a representation of the blurring angle.

C, t are the non-linear distortion parameters

W, h are the width and height of the blurring window.

\( \alpha \) and \( \beta \) are the MRF model parameters.

\( V_s(x_{ij}) \) is a measure of the Posterior Marginal probability at pixel \( x_{ij} \) in case of simple additive noise.

\( V_m(x_{ij}) \) is a measure of the Posterior Marginal probability at pixel \( x_{ij} \) in case of simple multiplicative noise.
$V_e(x_{ij})$ is a measure of the Posterior Marginal probability at pixel $x_{ij}$ in case of compound noise.

$V_b(x_{ij})$ is a measure of the Posterior Marginal probability at pixel $x_{ij}$ in case of excessive blur with additive noise.

$n_w$ is the number of pixels neighbouring (i,j) inside a window centred at (i,j).

$f_m$ is the mean of the highest and lowest grey levels in $X$.

$f_d$ is the symmetric difference of the highest and the lowest grey levels in $X$.

$\chi$ is the sample space (space of all possible realizations) of $X$.

$\theta$ weights for pairwise interactions MRF scene model.

$\zeta_{ij}$ a set of pixels neighbouring (i,j) inside a window centered at (i,j).

$\zeta_{oij}$ all pixels inside a window centered at (i,j).

$\zeta_{eij}$ a set of pixels neighbouring (i,j) inside an expanded window centered at (i,j).

$\zeta_{e_oij}$ all pixels inside an expanded window centered at (i,j).
TABLE OF CONTENTS

DEDICATION .................................................. iii
ABSTRACT .................................................. iv
ACKNOWLEDGEMENTS ....................................... v
ABBREVIATIONS AND NOTATION .......................... vi
TABLE OF CONTENTS ....................................... x
LIST OF FIGURES ......................................... xiii

Chapter I:  INTRODUCTION .................................. 1
  Preamble ................................................. 1
  A Brief Survey: ......................................... 1
  Research Proposal ...................................... 5
  Organization of the Dissertation: ..................... 5

Chapter II:  MARKOV RANDOM FIELD SCENE MODELS .... 6
  Background on MRF and GD ............................. 6
  Problem Statement ..................................... 12

Chapter III:  BINARY RESTORATION ALGORITHM AND ITS EXTENSION TO RESTORATION OF MULTILEVEL IMAGES . 16
  Overview .................................................. 16
  MAXIMUM A POSTERIORI (MAP) ESTIMATION: .......... 18
  Binary restoration algorithm: ......................... 19
  Test Results: .......................................... 23
  CONVERGENCE OF THE ALGORITHM ..................... 27
  Extension to Restoration of Multi-level Images ....... 28
  Repeated Application of Binary Restoration (BRES) Without Subtraction .................................... 28
  Repeated Application of Binary Restoration With Subtraction ..................................................... 31
  New Algorithm for Repeated Application of Binary Restoration Using Partial Binary Restoration (PBR) ............. 35
  Results and Conclusion ................................ 38
Chapter IV: **ONE STEP MULTILEVEL RESTORATION ALGORITHMS**

Overview ........................................... 42
Restoration Using Brute Force Optimization[58],[35]. ................. 44
Quadratic Pairwise Interactions Potential Function and Gradient Based Algorithm ........................................... 46
Choice of Model Parameters ........................................... 48
Results and Conclusion ........................................... 50

Chapter V: **RESTORATION OF IMAGES CORRUPTED BY DIFFERENT SOURCES OF DEGRADATION USING A SPECIAL CLASS OF MARKOV RAN DON FIELD SCENE MODELS**

Overview ........................................... 62
Restoration of Images Corrupted by Additive/Multiplicative Noise With Blur and Nonlinearity Using MRF Scene Models ........................................... 63
Mathematical Foundation ........................................... 64
Applications ........................................... 68
Restoration of Images Corrupted By Compound Noise With Blur and Nonlinearity Using MRF Scene Models ........................................... 76
Optimal Linear Model and Restoration of Images Degraded by Multiplicative Noise With Blur and Nonlinearity Using MRF Scene model ........................................... 78
Restoration of Images Degraded by Combined Additive and Multiplicative Noise With Blur and Nonlinearity Using MRF scene Models ........................................... 81
Sensitivity Analysis ........................................... 85
Results and Conclusion ........................................... 90

Chapter VI: **SPECIAL PURPOSE ALGORITHM FOR RESTORATION OF IMAGES DEGRADED BY EXCESSIVE BLUR AND NOISE USING MRF SCENE MODELS**

Overview ........................................... 109
Mathematical Foundation ........................................... 111
Results and Conclusion ........................................... 115

Chapter VII: **GIBBS FIELD MODELS RESTORATION ALGORITHMS HARDWARE FEASIBILITY ON SMID MACHINE ARCHITECTURE**

Overview ........................................... 123
Serial Implementation ........................................... 124
Parallel Implementation ........................................... 125
Comments and Conclusions ........................................... 136
Chapter VIII: CONCLUSION .......................................................... 137

Summary ................................................................. 137
Discussion ............................................................. 138
Advantages ........................................................... 138
Limitations ............................................................ 139
SUGGESTIONS FOR FUTURE WORK ..................................... 139

Appendix A: Mathematical derivation of $A_1$, $A_2$, and $A_3$ in 5.3.1 .................. 140

Appendix B: Derivation of Eqns. (5.54) and (5.55) ........................................... 142

Appendix C: Derivation of Eqn. (6.10) ...................................................... 145

REFERENCES .......................................................... 149

VITA AUCTORIS ......................................................... 155
LIST OF FIGURES

2.1 Neighbourhood orders relative to (i,j) ......................................................... 7
2.2 Cliques associated with first, second, and third order neighbourhood systems .... 9
3.1 ...................................................................................................................... 20
3.2 Piston head corrupted by simple additive noise .............................................. 24
3.3 Piston head corrupted by additive noise ......................................................... 25
3.4 Simulated image corrupted by simple additive-noise ..................................... 26
3.5 Repeated Application of BRES Without Subtraction. procedures for 
    restoration of five level image ......................................................................... 30
3.6 Repeated Application of BRES With Subtraction. procedures for restoration 
    of five level image ......................................................................................... 34
3.7 Partial Binary Restoration. example of an eight level image ............................ 36
3.8 Partial Binary Restoration .............................................................................. 37
3.9 Repeated Application of Binary Restoration. Without Subtraction ................. 39
3.10 Repeated Application of Binary Restoration. With Subtraction ..................... 40
3.11 Partial Binary Restoration. example of five level image .............................. 41
4.1 The effect of changing r on the quality of the restored image ......................... 45
4.2 Comparison between direct and gradient based restoration techniques .......... 51
4.3 Performance of the quadratic potential function algorithm at two levels of 
    noise .............................................................................................................. 52
4.4 Convergence properties of the gradient based restoration algorithm .............. 53
4.5 Continuous intensity scene corrupted by additive noise ................................ 54
4.6 Continuous intensity scene corrupted by additive noise ............................... 55
4.7 Controlling the resolution of a continuous intensity scene ............................... 56
4.8 Controlling the resolution of a continuous intensity scene ............................... 57
4.9 Performance at high level of noise for simulated image. ............................... 58
Chapter I
INTRODUCTION

1.1 Preamble

Image Restoration and Classification are two extremely important subjects in scene understanding and image analysis. Image restoration is of great interest to many researchers because of the urgent need to reconstruct a scene from its degraded version.

Restoration of degraded images is a class of digital picture processing closely related to image segmentation and boundary finding, and is extensively studied for its evident practical importance as well as theoretical interest [1].


1.2 A Brief Survey:

Although many restoration techniques have been reported in the literature, most of them are linear and deal with the assumed spectral content of image or noise. These usually involve iterative spatial-domain techniques, or frequency-domain techniques using the Fast Fourier Transform (FFT) [2]. In this class of techniques, it is implicitly assumed that the spectra of the image and the noise do not overlap. Based on a priori information about the image spectrum, a suitable linear filter (recursive or non-recursive) is chosen and convolved with the noisy image to.
obtain the restored image. There are many techniques for the design of linear filters (recursive or non-recursive) which can be used in this process [3-6]. In most cases these filters are of the low pass type and they suppress the high frequencies in the noisy image. Two major criticisms of this approach are (a) the invalidity of the assumption about the non-overlapping spectra of the image and the noise causing a severe loss of sharpness and edge information in the restored image and (b) the distortion caused by non-constant group delay characteristics of the smoothing filter. However, a linear high pass filter might preserve or even enhance edge information but these filters are expected to be very sensitive to high frequency noise components and very insensitive to slow intensity variations in scenes.

Therefore, in order to eliminate the noise and at the same time to preserve edge information, one is obliged to use nonlinear filtering. This class of techniques is generally effective for suppressing some specific types of noise in the image. These include impulse noise and wide-band uncorrelated noise. In the case of an image corrupted by impulse noise, one generally uses a median filter [7] or extensions of it [8], [9] in which the pixel value at any location may be replaced by the median value within a suitably chosen window centered at the pixel location. These filters, however, are not very effective for other types of noises that may be present in the image. A second class of filters called the generalized mean filter [10], has been used for the removal of impulse noise from images. These filters have been found to be quite effective; further, they lend themselves to easy hardware implementation. A third class of filters uses local statistics obtained from the noisy image to derive the smoothing or enhancing operation through the adjustment of single parameter [11].

The main drawback of these nonlinear approaches is the fact that they are geared to processing one specific type of noise in the image. Further, the methods do not utilize any valid statistical models of the image or the degradation phenomena.
Therefore, statistical approaches to noise suppression are needed. Some of the earliest work on image restoration that incorporates statistical models is due to Nahi [12], Nahi and Assefi [13], Nahi and Franco [14], Habibi [15], Manry and Aggarwal [16], Aboutalib and Silverman [17], Aboutalib et al. [18], Woods and Radewan [19]. Most of this work involves attempts to extend Kalman filter in two-dimensions. Nahi [12] suggested using a piecewise stationary approximation over various sections of each scanned line. In order to limit the effect of stationarity approximation, Nahi and Franco [14] suggested vector scanning. Manry and Aggarwal [16] suggested overlapping sections to eliminate edge problems. These methods required separable image covariance. Motion blur along one direction was considered by Aboutalib and Silverman [17]. An expression to general motion blur was proposed by Aboutalib et al. [18]. In this work recursive linear dynamical model was used for deblurring space variant/invariant point spread function (SVPSF/SIPSF) in the noise free case, which is augmented with Kalman filter in the noisy case. The main disadvantage of this approach is the high sensitivity to errors in knowledge of the point spread function (PSF). A two-dimensional recursive model for images was first considered by Habibi [15]. Woods and Radewan [19] proposed two 2-D Kalman processors known as Kalman strip filter (with vector scanning) and reduced update Kalman filter (RUKF) (with scalar scanning). The RUKF scheme overcomes the forbidden computational problems of the 2-D Kalman filter, where the updating process was limited to a small region in the vicinity of the current pixel (NSHP). More recently, Woods and Ingle [20] extended RUKF to include blur in their model. Suresh and Shenoi [21] used Kalman strip filtering and modeled the blur by 2-D state space structure with a propagation of the state space in one dimension. Azimi and Wong [22] proposed 2-D block realization technique in which the blur was modeled by two-dimensional Multi_Input Multi_Out (MIMO) block state space structure with considerable computational advantages when vector scanning scheme is available.
Although Bayesian estimate was mentioned, the methods were essentially equivalent to linear mean square estimation. In addition, the basic Kalman filtering algorithm suffers from two main problems: (i) The cost in terms of the number of multiplications and additions is related to $k^3$ where $k$ is the dimension of the state vector of the signal model; and (ii) There may be numerical instability in some cases [23].


The MAP estimate (which is the mode of the posterior distribution) is a stochastic estimation approach that offers potential promise for image restoration problems where the degradation (observation) model is nonlinear. Explicit knowledge of the posterior density need not be known, which is fortunate because of its computational complexities. The MAP estimate is basically Bayesian. It has been used successfully in special settings, see Hansen and Elliott [27] and Hunt [28]. see also Habibi [15], Richardson [29] and Nahi and Assefi [13]. Boundary estimation in noisy images using MAP estimate was studied by Elliott and Srinivasan [30], Elliott et al. [31] and Cooper et al. [32]. Hansen and Elliott [27] used MAP estimate for the segmentation of remotely sensed data at high level of additive noise. The image model used was a first order neighbourhood with binary variables. However, Gibbs Distribution (GD) was not used to give the joint distribution of the autologistic model, and the conditional probabilities were approximated by one-dimensional transition probabilities. Dynamic programming was used in segmenting each row and then for the entire image. In more recent work, Elliott et al. [33] used GD along the same line in [27] and improved results were obtained. However, the local characteristics of the true scene represented by MRF scene models may reflect undesirable large scale problems and the MAP estimate would present formidable computational task.
1.3 Research Proposal

The main concern of this dissertation is to devise restoration techniques based on stochastic estimation (MAP estimate) for images corrupted by a variety of degradation sources (including noises in combination) and whose computation and memory requirements are kept to a minimum. These restoration algorithms should not be affected by the large scale properties of the scene MRF model. Parallel implementation using modern VLSI technology will be taken into consideration.

1.4 Organization of the Dissertation:

The problem statement is described in the next chapter after preliminary material on MRF and GD. Chapter III contains the binary restoration algorithm and its extension to multilevel restorations. One step approach to multilevel restoration is then given in chapter IV. Chapter V is composed of four main sections, the first section is concerned with the restoration of images corrupted by additive or multiplicative noise with blur and nonlinearity. The second section is concerned with a linear model approximation of the multiplicative noise and the restoration of images corrupted by compound noise (additive and/or multiplicative) with blur and nonlinearity. The third section concerns the sensitivity analysis of the restoration algorithms followed by results and conclusion in the fourth section. The special purpose algorithm for restoration of images corrupted by blur and additive noise is the subject of chapter VI. The parallel structure of the restoration algorithms presented in this dissertation is the subject of chapter VII. Finally, in chapter VIII conclusions and suggestions for further research are discussed.
Chapter II
MARKOV RANDOM FIELD SCENE MODELS

Image modelling is an essential element of image understanding system. It is the key behind successful restoration in image processing applications. MRF is found to be a useful model incorporating the spatial dependance among pixels in close proximity of each other [34]. This scene model is sometimes called locally dependant MRF [35].

The joint distribution given by the local characteristics of the MRF can only be approximated by a set of transition probabilities resulting in severe consistency problems. The positivity condition given by the Hammersly and Clifford theorem [36] and the discovery of Gibbs Distribution Markov Random Field (GD-MRF) equivalence results in a unique and consistent expression for the joint distribution. This significant result constitutes a breakthrough of sorts in making MRF models readily accessible as an appropriate scene model.

2.1 Background on MRF and GD

A neighbourhood system $\eta^k$ of order $k$ ($k = 1, 2, ...$), defined over a finite lattice $Lt$ of picture elements (pixels), is a collection of subsets $\eta^k_j$ the neighbourhood at location $(i,j)$ of the lattice $Lt$ where:

1. $\eta^i_j \in Lt$
2. $(i,j) \notin \eta^i_j$
3. $(i,m) \in \eta^i_j \leftrightarrow (i,j) \in \eta^m_l$
Hierarchically ordered neighbourhoods are illustrated in Fig. 2.1. As indicated, a first order MRF is one for which the neighbour set consists of the nearest four pixels labeled '1' for \( \eta^1 \) while labels '1' and '2' for \( \eta^2 \) and so on.

\[
\begin{array}{ccc}
3 & & \\
2 & 1 & 2 \\
3 & 1 & (i, j) & 1 & 3 \\
2 & 1 & 2 \\
3 & & \\
\end{array}
\]

\textit{Figure 2.1:} Neighbourhood orders relative to \((i,j)\)

\[
\eta_k^{x} = \{ \eta_{ij}^{x}, (i,j) \in Lt \} \\
\eta_{ij}^{x} = \{ (l,m) \in Lt : 1 \leq (l-i)^2 + (m-j)^2 \leq r_k^2 \}
\]

where:

\[
k = 1, 2, 3, 4, 5, 6, \ldots
\]

\[
r_k = 1, \sqrt{2}, 2, \sqrt{5}, \sqrt{8}, 3, \ldots
\]

For convenience we will drop the suffix \(k\) in the notation of the neighbourhood system.

\textbf{Definition 1:}

Let \( \eta \) be a neighbourhood system defined over a finite lattice \( Lt \), a random field \( X \) defined over \( Lt \) is a MRF with respect to \( \eta \) if and only if:

1. \( P(X = x) > 0 \)

2. \( P(X_{ij} = x_{ij} \mid X_{im} = x_{im}, (l,m) \in Lt \cap (i,j)) = P(X_{ij} = x_{ij} \mid X_{im} = x_{im}, (l,m) \in \eta_{ij}) \)
Pixels on the boundary have fewer neighbours than interior pixels, and for higher order neighbourhood systems, pixels close enough to the boundary have also fewer neighbours than those relatively farther from the boundary. This phenomenon is called "free boundary" and is more natural than periodic boundaries in image processing applications.

The joint probability of the MRF is given by GD and the latter is based on the definition of cliques which is given by:

**Definition 2:**

A clique of the lattice neighborhood system pair \((L_t, \eta)\) denoted by \(c\), has the following properties:

(i) \(c \in L_t\)

(ii) \(c\) consists of a single pixel, or any pair of distinct pixels belonging to \(c\) must be neighbours,

\[\text{i.e.} \quad \text{if } (l, m) \in c \text{ and } (r, t) \in c\]

\[\text{then:} \quad (r, t) \in \eta_{lm}\]

The collection of all cliques of \((L_t, \eta)\) is denoted by \(C(L_t, \eta)\).

The cliques associated with \(\eta^1\), \(\eta^2\) and \(\eta^3\) are shown in Fig. 2.2.

**Definition 3:**

Consider a lattice neighbourhood system pair \((L_t, \eta)\), a random field \(X\) indexed by \(L_t\) has GD with respect to \(\eta\) if and only if [40]:

\[P(X = x) = \frac{e^{\psi(x)}}{Z} \quad (2.1)\]

where:
<table>
<thead>
<tr>
<th>Order of neighbourhood</th>
<th>Neighbourhood structure</th>
<th>Associated cliques</th>
</tr>
</thead>
<tbody>
<tr>
<td>FIRST</td>
<td>*  o  *</td>
<td>o — o</td>
</tr>
<tr>
<td></td>
<td>*</td>
<td></td>
</tr>
<tr>
<td>SECOND</td>
<td>*  *  *</td>
<td>o — o</td>
</tr>
<tr>
<td></td>
<td>*  o  *</td>
<td>o — o — o</td>
</tr>
<tr>
<td></td>
<td>*  *  *</td>
<td>o — o</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>THIRD</td>
<td>*  *  o  *  *</td>
<td>o — o — o — o — o</td>
</tr>
<tr>
<td></td>
<td>*  *  *</td>
<td>o — o — o — o — o</td>
</tr>
<tr>
<td></td>
<td>*</td>
<td></td>
</tr>
</tbody>
</table>

*Figure 2.2:*Cliques associated with first, second, and third order neighbourhood systems
\[ U(x) = \sum_{c \in \mathcal{C}(L, \eta)} V_c(x) \]  

\( U(x) \) is the energy function.

\( V_c(x) \) is the potential function associated with clique \( c \).

\( Z \) is a partition function or simply a normalizing constant such that the total probability of all possible realizations is 1.

**GD-MRF equivalence:**

Theorem:

A random field \( X \) indexed by \( L \) with respect to \( \eta \) is a MRF if and only if its joint distribution is Gibbsian with cliques associated with \( \eta \).

Ernst Ising [37], the German scientist in 1925 used GD in what is now known as the Ising model to describe properties of ferromagnetic materials. Ising used a first order neighbourhood system with binary variables. The energy function used by Ising (see Kindermann and Snell [38]) can be generalized as:

\[ U(x) = \sum_{ij \in L} \left\{ A_{ij} (x_{ij}) + \sum_{kl} B_{ij} (x_{ij}, x_{kl}) + \sum_{klm} C_{ij} (x_{ij}, x_{kl}, x_{mn}) + \ldots \right\} \]

The summation in the second term in Eqn. (2.3) is counted only once for each pair (cliques of size two), while the summation in the third term is counted only once for each triple (cliques of size three) and so on. As an explanation for the significance of the different terms in Eq. (2.3), it is stated that [38]:

1. \( A_{ij} (x_{ij}) \) represents an external agent, which is a function that is proportional to the percentage of each level present in \( X \).

2. \( B_{ij} (x_{ij}, x_{kl}) \) model special features in the scene such as size, shape, orientation of clusters,...
3. $C_{ij}(x_{ij}, x_{kl}, x_{mn})$ has a similar property as in 2. with additional complications in the model (complex scenes require complex energies).

It is generally agreed [1], [33] that the first two terms (especially the second) are the dominant terms in this series representation of $U(x)$. In addition many investigators [38]-[40] have used the following formulation for $U(x)$:

$$U(x) = \sum_{ij} \left\{ \alpha x_{ij} + \sum_{kl} \beta R(x_{ij}, x_{kl}) \right\}$$

where:

$\alpha$ and $\beta$ are parameters that characterize the scene. Elliott has described a technique [40] for estimating $\alpha$ and $\beta$ given a sample of noise free image. However, if the image is noisy, there is no reliable method for a priori information of $\alpha$ and $\beta$. The specific choices of these parameters are described in a later section of this thesis.

In this thesis a formulation of $U(x)$ similar to that proposed by Derin [57] is used for the derivation of the globally optimal restoration.
2.2 Problem Statement

In this section the assumptions about the statistical characterization of the scene, the corrupting noise and the basic problem statement will be presented.

We will restrict the definitions to two-dimensional discrete random fields defined over a finite lattice $\mathbf{L}_t$ of picture elements (pixels). Capital letters will be used to represent random variables and random fields while lower case letters will be used to represent specific realizations.

Assumptions:

1. The model of the degraded image is given by:

$$ Y = C \{H \ast F(X)\} \cdot U + W $$

(2.4)

It can be expressed as:

$$ y_{ij} = C \left\{ \sum_{kl} H(k, l) F[x(i+k, j+l)] \right\} \cdot u_{ij} + w_{ij} $$

(2.5)

for all $(i,j) \in \mathbf{L}_t$

Where:

- $W$ is a representation for additive noise.
- $U$ is a representation for multiplicative noise.
- $C$, $t$ are the nonlinearity constants.
- $H$ is the blurring window of angle $\phi$

$$ H(k, l) = \begin{cases} \cos^2 \phi & (k, l) = (0, 0) \\ \frac{\sin^2 \phi}{n_w} & (k, l) \in \zeta_{ij} \end{cases} $$

(2.6)

$\zeta_{ij}$ represents all pixels neighbouring $(i,j)$ inside a window centered at $(i,j)$. The figure below shows an example of 3 by 5 window.
\[ l_w = 2 \quad l_h = 1 \]
\[ W = 2l_w + 1 \quad h = 2l_h + 1 \]
\[ n_w = W \cdot h - 1 \]

\( n_w \) represents the number of pixels inside the blurring window excluding the center pixel.

2. \( X \) is homogeneous and isotropic MRF with \( \{X = x\} \) as a specific realization.

Note that:
\( \{X = x\} \) stands for \( \{X_{ij} = x_{ij} \ . (i,j) \in \mathbb{L}t \} \)
\( x_{ij} \in [-1, 1] \) is a real number representing a 'label' at location \((i,j)\).
\( F(x_{ij}) \geq 0 \) is a nonnegative integer representing the corresponding grey level.

The state space of \( X \) contains \( N \) elements, i.e., the number of grey levels in \( X \) is finite \( = N \).

3. \( F \) is a linear mapping and is given by:
\[ F(x_{ij}) = f_m + f_d \cdot x_{ij} \quad (2.7) \]

arranging the state space of \( X \) in increasing order we can write:
\[ x_{ij} \in \{x_1, x_2, \ldots, x_N\} \quad (2.8) \]

Where:
\[ x_1 = -1 \quad x_N = 1 \]
\[ x_r > x_t \text{ for } r > t \quad 1 \leq t < r \leq N \]
It can be verified that:

\[
    f_m = \frac{1}{2} \left( F(1) + F(-1) \right)
\]

\[
    f_d = \frac{1}{2} \left( F(1) - F(-1) \right)
\]

(2.9)

(2.10)

Note that:

\[F(1) = f_h \text{ is the highest grey level in } X.\]

\[F(-1) = f_l \text{ is the lowest grey level in } X.\]

4. Given \( X \) the random variables \( Y_{ij}, (i, j) \in L \) are conditionally independent and depend only on the \( X \) values inside the blurring window centered at \((i, j)\).

5. The noisy image \( y \) consists of a number of grey levels which might exceed the number of grey levels in the uncorrupted image.

6. The corrupting noise sources \( W \) and \( U \) are Gaussian independent and identically distributed (i.i.d.) with zero and unity means (respectively) and known (estimable) variances \( W = N(0, \sigma^2_W) \), \( U = N(1, \sigma^2_U) \).

7. \( X, U \) and \( W \) are independent stochastic processes (in pairs).
The Statement of The Problem:

The basic problem statement is given by:

Given the degraded image $y$, determine an estimate $\hat{x}$ of $X$

such that:

$P(X = \hat{x} | y) = \alpha$ is a maximum.

REMARK:

The degradation model is made quite general since it allows for additive and/or
multiplicative noise sources, local blur and nonlinear distortion.
Chapter III

BINARY RESTORATION ALGORITHM AND ITS EXTENSION TO
RESTORATION OF MULTILEVEL IMAGES

3.1 Overview

In the restoration problem, we basically search for a realization of the scene random field $X$ which maximizes the conditional probability of the scene given the records (degraded image), i.e. we search for the mode of the posterior distribution of $X$ given $Y$.

Scenes may contain 512 by 512 pixels and the reconstruction problem may reflect undesirable large scale properties of the assumed MRF, leading to enormous computational problems. Exceptions occur by design, as an example, Derin et al. [34] used Bayes smoothing algorithm with Markov Mesh Random Field (MMRF) scene model and overlapping strips. The algorithm was quite satisfactory for restoration of very simple binary images, corrupted by high level of noise. However, the MMRF model specifications are unnatural in a spatial context and would be very restrictive [since the unilateral nature of the model implies the unobvious consistency conditions given by the Hammersly and Clifford theorem and this class must fail when substantial local dependence is present [35]]. In order to avoid such problems we will use a noncausal MRF and the symmetry in the neighbourhood definition will be preserved.
For clarity of presentation, and in order to compare the performance of the restoration algorithms with others, we will consider the case of simple additive noise in this chapter. Thus, neglecting the effect of blur, multiplicative noise and nonlinearity in Eqn. (2.3) we get:

\[ Y = F(X) + W \]

or

\[ y_{ij} = F(x_{ij}) + w_{ij} \]  

for all \((i,j) \in L\)
3.2 MAXIMUM A POSTERIORI (MAP) ESTIMATION:

The estimation of the original scene $X$ can be formulated as the maximization of the posterior probability density $P(X = x | Y = y)$ with respect to $x$. Using Bayes' theorem we get:

$$P(X = x | Y = y) = rac{P(Y = y | X = x) \cdot P(X = x)}{P(Y = y)}$$

Since $P(Y = y)$ is independent of $X$, the estimation can be achieved by maximizing the joint probability $P(X = x, Y = y)$ with respect to $x$.

$$P(X = x, Y = y) = P(Y = y | X = x) \cdot P(X = x)$$

Using assumption 4 in chapter II we get:

$$P(Y = y | X = x) = \prod_{i,j} P(Y_{ij} = y_{ij} | X_{ij} = x_{ij})$$

Assumptions (6) & (7) in chapter II result:

$$P(Y = y | X = x) = \frac{1}{(2\pi\sigma^2)^N} \prod_{i,j} \exp \left\{ \frac{-[y_{ij} - F(x_{ij})]^2}{2\sigma^2} \right\}$$

Where: $\mu = \frac{N1 \cdot N2}{2}$, and the image array is assumed to be rectangular of size $N1 \cdot N2$

Taking the natural logarithm of Eqn. (3.2) and using Eqns. (3.3),(2.1) after neglecting constant terms, it can be shown that the best estimate of $x$ is obtained by maximizing:

$$V(X) = U(X) - \frac{1}{2\sigma^2} \sum_{i,j \in L} \left[ y_{ij} - F(x_{ij}) \right]^2$$

with respect to $x$.

where:

$$V(x) = \ln P(X = x, Y = y)$$
The algorithms that have been devised to solve the above problem are discussed in the following sections.

3.3 Binary restoration algorithm:

For a binary image, the number of grey levels $N = 2$, using Eqns. (2.5) and (2.6) result:

\[ x_{ij} \in \{-1, 1\} \quad \text{for all } (i,j) \in Lt \]

\[ F(x_{ij}) = f_m + f_d \cdot x_{ij} \quad (3.5) \]

where:

\[ f_m = \frac{F(1) + F(-1)}{2} \quad (3.6) \]

\[ f_d = \frac{F(1) - F(-1)}{2} \quad (3.7) \]

For reasons of computational simplicity and modal adequacy we will assume a second order neighbourhood system, autologistic model. The energy function is assumed to be in the form:

\[ U(x) = \alpha \sum_{ij} x_{ij} + \beta \sum_{ij} \sum_{kl} \theta_{kl} x_{ij} x_{kl} \quad (3.8) \]

where:

(i) $\Theta$ represents positive weights used to reward the interaction between the center pixel and its neighbours based on their relative distances with respect to the center pixel. A typical choice of $\Theta$ for second order neighbourhood system is shown in Fig. 3.1.

(ii) $\alpha$ and $\beta$ are the model parameters.
Figure 3.1 $\theta_{ij}$ dependence on the distance from the center pixel and its neighbours

Thus the function to be maximized is given by:

$$V(x) = \sum_{ij} \left\{ \alpha x_{ij} + \beta \sum_{kl} \theta_{kl} x_{ij} x_{kl} - \frac{(y_{ij} - f_m - f_d x_{ij})^2}{2 \sigma^2} \right\}$$  \hspace{1cm} (3.9)

let:

$$\bar{y}_{ij} = y_{ij} - f_m$$  \hspace{1cm} (3.10)

and

$$\rho = \frac{2f_d}{\sigma} = \frac{F(1) - F(-1)}{\sigma}$$  \hspace{1cm} (3.11)

$\rho$ is defined as the signal to noise ratio (SNR).

After some manipulation, the function needed to be maximized, with respect to $x$, can be written in the form:
\[ V(x) = \sum_{ij} \left\{ (\alpha + \frac{\rho}{2\sigma} \tilde{y}_{ij}) + \beta \sum_{kl} \theta_{kl} \tilde{x}_{kl} \right\} x_{ij} \] (3.12)

A direct maximization of \( V(x) \) will involve the computation of \( V(x) \) for each of \( 2^{N1N2} \) possible states of \( x \), thus rendering a practical realization is a hopeless task.

In a recent paper by Elliott et al. [33], a sub-optimal procedure was derived by restricting the maximization to strips of size \( 3N2 \), thereby significantly reducing the number of possible states of \( x \) to \( 2^{3N2} \). Dynamic programming was utilized to obtain the estimate of the clean image. However, even with this modification, the algorithm was quite slow and was not suitable for high speed processing.

In the scheme to be presented below an iterative, highly parallel relaxation algorithm is to be derived, where a local maximization (maximizing the posterior marginal probability at each individual pixel) is utilized in the derivation of the clean image. In this procedure an initial estimate is derived from the original record by using a thresholding technique. Using this estimate the expression between brackets in Eqn. (3.12) is computed. The new value of \( x_{ij} \) is then determined as the value (1 or -1) which maximizes each term of Eqn. (3.12). When all the pixels have been processed in this manner, a new estimate is realized. The process is repeated until convergence is obtained. Thus the procedure involves:

1. An initial estimate \( (x)_0 \) of the clean image with \( t = 0 \)

2. A new estimate is found after computing:

\[ P_{ij} = \alpha + \frac{\rho}{2\sigma} \tilde{y}_{ij} + \beta \sum_{kl} \theta_{kl} \tilde{x}_{kl} \] (3.13)

3. The new estimate of \( x \) is found as follows:
\[
(x_{ij})_{t+1} = \begin{cases} 
1 & \text{if } P_{ij} > 0. \\
(x_{ij})_t & \text{if } P_{ij} = 0. \\
-1 & \text{if } P_{ij} < 0.
\end{cases}
\]

It is clear that \((x_{ij})_{t+1}\) can be computed using only one function evaluation.

\[
(x_{ij})_{t+1} = \text{sign}(P_{ij})
\tag{3.14}
\]

The procedure continues until all pixels are processed and a new image \((x)_{t+1}\) is obtained.

4. Update \(\beta\), then GOTO step 2. for next iteration with \(t \rightarrow t+1\) until convergence.

The main features of the proposed algorithm can be summarized in the following two points:

1. One function evaluation is only needed to classify each pixel. This should be compared with others where two functions are calculated first (corresponding to two values of \(x_{ij}\)) then comparison is made before classification is finally obtained.

2. Eqn. (3.12) is not only very simple to implement, but also parallel implementation has been made possible since the evaluation of the new estimate is totally restricted to the values of its immediate neighbours.
3.4 Test Results:

The algorithm was applied on a variety of images at different levels of noise. Fig. 3.2 shows the piston head corrupted by simple additive noise at (a) SNR = 0.5, (b) SNR = 1.0, and (c) SNR = 2.0. Fig. 3.3 shows the effect of increasing $\beta$ in case of SNR = 2. Fig. 3.4 shows the results obtained for simulated image at (a) SNR = 0.5, (b) SNR = 1.0, and (c) SNR = 2.0.
Figure 3.2: Piston head corrupted by simple additive noise.
1 = original,
2, 3, 4 = noisy (SNR = 0.5, 1.0, 2.0).
Standard Deviation of Noise (220, 110, 55)
5, 6, 7 = restored (SNR = 0.5, 1.0, 2.0)
$Q/Q^0 = (7.2, 25.6, 86.2)$
(final quality w.r.t that of the noisy image)
Figure 3.3: Piston head corrupted by additive noise. The effect of increasing $\beta$
SNR=2.0,
1=original, 2=noisy, 3=restored
$Q/Q^0 = 10.36$
Figure 3.4: Simulated image corrupted by simple additive noise.
1 = original,
2,3,4 = noisy (SNR = 0.5, 1.0, 2.0),
Standard Deviation of Noise (200, 100, 50)
5,6,7 = restored (SNR = 0.5, 1.0, 2.0)
Q/Q^0 = (6.851, 37.910, 99.330)
3.5 CONVERGENCE OF THE ALGORITHM

In this section it will be proved that the proposed algorithm for maximizing (3.12) will converge.

\[ V(x) = \sum_{ij} P_{ij} x_{ij} \]

and \( P_{ij} \) is given in Eqn. (3.13).

For any \((i,j) \in L_i\), the term in the summation is maximized by choosing \( x_{ij} \) at iteration \((t+1)\) to be:

\[ (x_{ij})_{t+1} = \text{sign}(P_{ij}) \]

In the proposed parallel implementation (details will be given in chapter VII) at iteration \((t+1)\), all pixels labelled '1', '2', '3' and '4' will be processed in four synchronous updates (clock cycles). At the end of clock cycle '1', it is clear that:

\[ V^1(x_{t+1}) \geq V^0(x_{t+1}) \]

where:

\[ V^i(\cdot) \text{ indicates the value of } V(\cdot) \text{ at the end of clock cycle } 'i' \]

and \( V^0(x_{t+1}) = V(x) \)

It is clear that the function \( V(x) \) can only be nondecreasing after each clock cycle. Therefore:

\[ V^{j+1}(x_{t+1}) \geq V^j(x_{t+1}) \quad j = 0, 1, 2, 3 \]

Hence:

\[ V(x_{t+1}) \geq V(x) \]

Convergence is assumed to be attained when the number of pixel changes in a given clock cycle is less than a threshold value.
3.6 Extension to Restoration of Multi-level Images

In the previous section a parallel algorithm was developed for restoration of binary images. The extension of this algorithm to the restoration of multilevel images is considered in this section.

The simplicity of the restoration algorithm is lost when the original scene is characterized by several grey levels (region types). To justify the performance of the proposed algorithm in terms of computational complexity and storage requirements in this section a comparative study is also carried out by considering the work of Shridhar and Cristi [39] and Elliott et al. [33]. An account of the advantages and disadvantages of these techniques with examples will be considered. The details of the two techniques mentioned above are presented below.

3.7 Repeated Application of Binary Restoration (BRES) Without Subtraction

Multilevel restoration (segmentation) through repeated application of binary restoration can be done in many different ways [33]. In the algorithm that will be presented here, the algorithm for two region types (binary algorithm) is to be applied N-1 times, where N represents the number of region types in the original image. Thus the computational cost is expected to grow with the number of regions linearly.

In this technique N-1 binary arrays should first be created, then the required estimate is basically a linear combination of these binary arrays. This technique will be illustrated by an algorithm and an example.

Suppose we have a scene X composed of N distinct region types and each region type has a constant grey level with the possibility of each grey level to appear at different locations in the scene. Let $f_0, f_1, \ldots, f_{N-1}$ represent the values of these grey levels.

Let:

$$Z = BRES(f_{e1}, f_{e2}; y); \quad Z \in \{0, 1\}^{N^2}$$
This expression means that binary restoration algorithm is applied on $y$ using the grey levels $f_{s_1}, f_{s_2}$ and the resulting cleaned image is stored in the array $Z$ with the label '0' for the lower level and the label '1' for the upper level. Now consider the following algorithm:

\begin{verbatim}
J = N
FOR I = 1 TO N-1 BY 1
    Z^{(I)} = BRES(f_I, f_{I-1}; y)
    J = J - 1
END FOR

S = \sum_{I=1}^{N-1} Z^{(I)} \quad S = \{ S_{ij} \}, (i, j) \in Lt

\hat{x}_{ij} = f_{s_{ij}} \quad 1 \leq i \leq N1, 1 \leq j \leq N2
\end{verbatim}

Example:

Fig. 3.5 shows an example for restoration of five region types ($A_0, A_1, \ldots, A_4$)

\[\hat{x}\]

Analysis

The final cost in terms of computation time and memory burden

1. Computation time grows linearly with the number of regions.
2. Memory burden grows linearly with the number of regions ($N-1$ binary arrays are required to create the cleaned image).

Since the cost of the proposed binary restoration algorithm requires one simple function evaluation at each pixel, then a total of $N$ simple function evaluations are required in the restoration of $N$-level image.
Figure 3.5: Repeated Application of BRES without subtraction, procedures for restoration of five level image.
3.8  \textit{Repeated Application of Binary Restoration With Subtraction}

Since the computational cost of the algorithm described in the previous section grows linearly with the number of regions, it is not suitable for high speed processing. There are several approaches one may take in resolving this problem. In an earlier method proposed by Cristi and Shridhar [39], a sequential binary restoration procedure was adopted for images whose grey levels spacing was uniform. The procedure was essentially equivalent to sequentially estimating the digits of a binary expansion of \( x \). Also, after each digit is estimated a new image had to be derived through a process of subtraction involving the noisy image \( y \).

Now we will illustrate the method in detail using an example. Consider \( N \) grey level scene given by:

\[
f_i = f_0 + L i \quad ; i = 0, 1, \ldots, N-1
\]

Where:

\( L \) is the grey level spacing

Let:

\[
K = \log_2(N)
\]

The algorithm can be summarized as follows:
\( t(K) = y \): temporary array of the image size

\[
\text{FOR } J = K \text{ TO } 1 \text{ BY } -1 \\
Z^{(J)} = BRES \left( f_{J-1}, f_0 : t(J) \right) \quad Z^{(J)} \in \{0, 1\}^{N_1 \times N_2} \\
N(J) = 2^{J-2} L \\
t(J-1) = t(J) - 2N(J) \quad Z^{(J)} \\
\text{END FOR}
\]

\[ \hat{x} = \{ Z^{(K)}, Z^{(K-1)}, \ldots, Z^{(1)} \} \]

\( \hat{x} \) is the final estimate of \( x \) using \( Z^K, Z^{K-1}, \ldots, Z^1 \) binary arrays.

**Example:**

Fig. 3.6 shows an example of five region types with

\[ f_0 = 20, L = 50 \]
\[ K = 3; \quad t(3) = y \]
\[ Z^{(3)} = BRES(f_7, f_0; t(3)); Z_{ij}(3) \in \{0, 1\} \]
\[ t(2) = t(3) - 200 \cdot Z^{(3)} \]
\[ Z^{(2)} = BRES(f_3, f_0; t(2)) \]
\[ N(2) = 50 \]
\[ t(1) = t(2) - 100 \cdot Z^{(2)} \]
\[ Z^{(1)} = BRES(f_1, f_0; t(1)) \]
\[ \tilde{x} = \{ Z^{(3)}, Z^{(2)}, Z^{(1)} \} \]

**Analysis:**

The final cost in terms of computation time and memory burden

1. Computation time is proportional to \( K \) simple function evaluations.
2. Memory burden is of the order of \( K \) binary arrays.

In comparison with [33] this algorithm has considerable computational advantages. However, the method requires the estimation of \( K \) binary arrays which are then required to be concatenated to obtain the final image. It is implicitly assumed that the grey levels in the scene are uniformly spaced, a requirement that is rather restrictive. Further, the subtractive operation performed after each binary restoration results in a new image that has significantly lower signal to noise ratio (SNR) than the original noisy image, thus affecting the estimation of the least component of \( x \).
Figure 3.6: Repeated Application of BRES With Subtraction, procedures for restoration of five level image
3.9 New Algorithm for Repeated Application of Binary Restoration Using Partial Binary Restoration (PBR)

In order to maintain the advantages and at the same time to eliminate the disadvantages in the methods described in the previous sections, a new algorithm is proposed in which the restored image is obtained in an efficient manner.

In this method a binary restoration is used to obtain the significant component of the L-tuple expansion. This binary image consists of pixels having labels '1' or '-1'. In the second step binary restoration is again applied separately to the regions that are characterized by pixel labels '1' or '-1' to obtain the estimate of the second component of the L-tuple. This procedure is repeated until all the components are extracted. The computational cost of this technique is of the order of \( \log_2(N) \) times the cost of one binary restoration i.e. \( \log_2(N) \) function evaluations are needed in order to classify each pixel. Fig. 3.7 shows the procedure for an eight level image (equally spaced 30, 60,..., 210). The general case where \( N \) is not a power of 2 is shown in Fig. 3.8. The grey levels in this example (unequally spaced) are, \( f_0 = 30, f_1 = 55, f_2 = 90, f_3 = 120, \) and \( f_4 = 250 \).

The computation time is approximately given by \( (3 - 2n_5/N1N2) \) function evaluations. Where \( n_5 \) is equal to the number of pixels (cardinality) of region \( A_5 \).

Analysis:

The final cost in terms of computation time and memory burden

1. Computation time is of the order of \( \log_2(N) \) simple function evaluations.
2. No extra memory is needed.
3. Subtraction operation is not required, so the estimation of the least significant component of \( x \) is not affected.
4. The hierarchically ordered binary restoration is essentially a one step method and is reliable and efficient.
5. The proposed technique is general without restrictions.
Figure 3.7: Partial Binary Restoration, example of an eight level image.
SNR=2.0, Equally-spaced, $\sigma = 15$
1=original, 2=noisy, 3=initial-condition,
4=first-realization, 5=second-realization,
6=estimated-image, 7=post-processed.
$Q(6)/Q^0 = 17.15$, $Q(7)/Q^0 = 18.89$

In comparison with [33], where the dynamic programming algorithm is used, the proposed algorithm has considerable computational advantages without restrictions.
**Figure 3.8:** Partial Binary Restoration. five level image (Unequally spaced)
3.10 Results and Conclusion

In this section, examples were presented for the restoration algorithms described in sections 3.5.1, 3.5.2 and 3.5.3. Fig. 3.9 is an example for repeated application of binary restoration without subtraction for four level image (unequally spaced, 40, 90, 160, 240). Three binary arrays were created to construct the restored image. Regardless of the computational cost, the quality of the restored image was quite good. Fig. 3.10 is for repeated application of binary restoration with subtraction at high level of noise. The least significant bit was almost lost as a direct result of the subtraction operation involved in this technique. Fig. 3.11 is an example of PBR. Where the procedure given in Fig. 3.8 above was validated by computer simulation.
**Figure 3.9:** Repeated Application of Binary Restoration, Without Subtraction. SNR=1.0, four level image (Un-equally spaced).

1=original, 2=noisy, 3=first element $Z^1$,
4=second element $Z^2$, 5=third element $Z^3$,
6=restored-image ($Q/Q^0 = 6.341$).
Figure 3.10: Repeated Application of Binary Restoration, With Subtraction. 
$SNR=1.0, \sigma = 30$, eight-level-image (equally spaced).
1=original, 2=noisy, 3=first-element (MSB),
4=second-element, 5=third-element (LSB),
6=restored-image ($Q/Q_0 = 3.20$)
Figure 3.11: Partial Binary Restoration, example of five level image.
1 SNR=1.0, Unequally spaced, $\sigma = 25$,
1=original, 2=noisy,
3=first-realization, 4=second-realization,
5=estimated-image ($Q/Q_0 = 4.21$)
Chapter IV
ONE STEP MULTILEVEL RESTORATION ALGORITHMS

4.1 Overview

In the previous chapters binary restoration algorithm was developed and it was extended to the restoration of multilevel images through sequential schemes. In this chapter we will tackle the multilevel restoration problem directly in one step.

For homogeneous (strictly stationary) and isotropic (rotationally invariant), auto-model, pairwise interactions MRF, Eqn. (3.9) can be written in the form:

\[ V(x) = \sum_{i,j \in L_i} \left\{ \alpha x_{ij} + \beta \sum_{k,l} \theta_{kl} R(x_{ij}, x_{kl}) \right\} - \frac{(\bar{y}_{ij} - f_D x_{ij})^2}{2 \sigma^2}, \quad (4.1) \]

Where \( R(x_{ij}, x_{kl}) \) is a suitably chosen function. A typical choice of \( R(x_{ij}, x_{kl}) \) is given by:

\[ R(x_{ij}, x_{kl}) = \begin{cases} 
1 & x_{ij} = x_{kl} \\
-1 & \text{otherwise}
\end{cases} \]

This simple form has been used by Geman and Geman [11] and Elliott et al. [40] among others.

We will assume \( R(x_{ij}, x_{kl}) \) to be in the form:

\[ R(x_{ij}, x_{kl}) = 1 - \frac{|x_{ij} - x_{kl}|}{2^r} \quad (4.2) \]

\( r \) is a tuning constant which might be chosen to match the given scene.

Clearly:

\[-1 \leq R(x_{ij}, x_{kl}) \leq 1\]

Substituting (4.2) into (4.1) we get:
\[ V(x) = \sum_{ij \in L} \left\{ \alpha \cdot x_{ij} + \beta \sum_{kl} \theta_{kl} \left( 1 - \frac{|x_{ij} - x_{k+l}|}{2^{r-1}} \right) - \frac{(|\nabla x_{ij} - f_x_{x_{ij}}|^2)}{2\sigma^2} \right\} \]

The maximum a posteriori estimation of \( x \) given by maximizing (4.3) with respect to \( x \) can be done in different ways. Elliott et al. [33] process the image in strips of up to 4 rows at a time. They used dynamic programming to achieve segmentation. The algorithm is quite slow and it is not suitable for high speed processing. Furthermore it can only be used in restoration of up to four grey levels (due to computational overheads). Geman and Geman [1] tackled the computational problem using stochastic relaxation through the medium of simulated annealing [55]. Simulated annealing proceeds by running a time inhomogeneous Markov chain with a transition matrix controlled by \( T \), which is the temperature. \( T \) is progressively decreased according to a prescribed "schedule" to a value close enough to zero. The stochastic nature of the procedure enables escapes from any local maxima to occur, but to ensure such escapes, the chain must run for a long time and \( T \) must be decreased extremely slowly. Therefore, an immense amount of computation is required for arrays of modest dimensions, so this is also not suitable for high speed processing.

In order to overcome the computational problems described above, a deterministic type of relaxation algorithms is proposed, with no claim at this point about its global optimality. However, a satisfactory local optimal is obtained in almost all of the cases.

We now will consider two different techniques in achieving a final estimate of \( x \) which maximizes the posterior marginal probability at each pixel. The first is based on a brute force (direct search) optimization procedure. The second is gradient based optimization and it is based on a specific choice of the tuning constant \( r \) in Eqn.(4.2) which leads to a convex optimization problem with unique global optimal solution. In the next chapters we will use this type of approach in solving complicated degradation models easily and efficiently.
4.2 Restoration Using Brute Force Optimization[57],[35].

The required estimate of $x$ is obtained by maximizing the local $V(x)$ in Eqn. (4.3) with respect to $x$. In N-level deterministic relaxation, one calculates the local $V(x)$ for $x_j \in \{x_1, x_2, \ldots, x_N\}$, and chooses the one, say $x_j = x_k$, that maximizes $V(x)$. This approach requires $N$ function evaluations at each pixel for each iteration. For this type of optimization there is no restriction on the choice of the tuning constant $r$; however, $r = 1$ is found to be suitable in the majority of cases with different classes of scenes.

A special case occurs when $r$ is close enough to zero. Thus,

$$
\lim_{r \to 0} R(x_j, x_k) = \begin{cases} 
1 & x_j = x_k \\
-1 & \text{otherwise}
\end{cases}
$$

Accordingly it is possible to simulate the class of scenes used by other researchers, (see [1], [40] and [41]) by selecting $r$ sufficiently small.

Let $Q$ represent the quality of the current realization $x$ in the iterative algorithm and $Q^0$ represent the quality of $x$ at iteration zero (the quality of the corrupted image). The quality could be measured in terms of the 'distance' between the ideal image and the current realization $x$ or it may be measured directly as the value of the posterior distribution. The effect of changing $r$ on the quality (measured in terms of the distance between the current realization and the ideal scene) at different levels of degradation for a typical image is given in Fig. 4.1.

It should be noticed that a single measure of the quality could be misleading. However, there is a fair closeness between the calculated qualities and the observed qualities. This is made possible by changing the raster scan (in series implementation of the algorithm) from one iteration to the following iteration in order to eliminate the directional effects which might otherwise occur [35]. The quality measured in terms of the distance between the original and restored image is given by:
Figure 4.1: The effect of changing $r$ on the quality of the restored image, $0 < r < 5.0$.
\[ q = \frac{a}{1 + b \parallel x - \hat{x} \parallel} \]

where \( a, b \) are constants.

**Analysis:** Clearly, the one step approach to multilevel restoration does not require extra memory (because the calculations are done in place), however, the computational cost in terms of the number of function evaluations at each pixel grows linearly with the number of region types (grey levels). So, when the number of grey levels in the original scene is moderate to high, the algorithm becomes computationally demanding and consequently it is not suitable for high speed processing. In the serial implementation of the algorithm the cost grows linearly with the size of the image as well.

However, regardless of the computational cost, the brute force optimization techniques are flexible and more accurate since the scene model could be appropriately chosen and the degree of interactions could be arbitrarily increased.

### 4.3 Quadratic Pairwise Interactions Potential Function and Gradient Based Algorithm

We presumably want the restoration algorithms to be useful for large scale images and ultimately for real-time film processing, this will require virtually instantaneous restorations and favour reliable, simple techniques [56].

In the formulation to be presented below auto-model pairwise interactions MRF represented by quadratic potential function enables the derivation of strictly convex posterior distribution with one solution. The model is a typical one for continuous intensities scenes and is given by using \( r = 2 \) in Eqn. (4.3)
\[
V(x) = \sum_{ij \in L_T} \left( \alpha x_{ij} + 1 - \frac{[x_{ij} - x_{kl}]^2}{2} \right) - \frac{(\bar{y}_{ij} - f_d x_{ij})^2}{2 \sigma^2}
\]  \hspace{1cm} (4.4)

Where \( R(x_{ij}, x_{kl}) \) in Eq. (4.2) is given by:

\[
R(x_{ij}, x_{kl}) = 1 - \frac{[x_{ij} - x_{kl}]^2}{2}
\]  \hspace{1cm} (4.5)

\( V(x) \) may be maximized by differentiating Eq. (4.4) with respect to \( x_{ij}; (i,j) \in L_T \) and equating the partial derivatives to zero. This yields the following estimate:

\[
\hat{x}_{ij} = \frac{\alpha + \beta \sum_{kl} \theta_{kl} x_{kl} + \bar{y}_{ij} f_d}{\beta \sum_{kl} \theta_{kl} + \left( \frac{f_d}{\sigma} \right)^2}
\]  \hspace{1cm} (4.6)

The convexity of the problem is clear, resulting in a unique global optimal solution.

It is quite interesting to see that the solution given by Eq. (4.6) in case of binary images coincides with the estimate given by Eq. (3.14)

\textit{Proof:}

If the estimate given by Eq. (4.6) is not among the feasible values \( x_1, x_2, \ldots, x_N \), the procedure is to round \( x_{ij} \) up or down to the closest allowable level. This is made possible since the optimization problem is strictly convex. However, by doing such a procedure we avoid the unnecessary function evaluations at those levels, hence avoiding two function evaluations per pixel at each iteration. For binary images, the possible scene values are 1 or -1 and the procedure is to round the estimate given by Eqn. (4.6) to +1 or -1, depending on which is closer to that estimate. This is done by finding the sign of the right hand-side of Eq. (4.6) i.e.
\[ x_{ij} = \text{sign} \left( \frac{\alpha + \beta \sum_{kl} \theta_{kl} x_{kl} + \frac{y_{ij}}{\sigma^2}}{\beta \sum_{kl} \theta_{kl} + \left( \frac{f_{ij}}{\sigma} \right)^2} \right) \]  

(4.7)

Since the denominator of (4.7) is positive definite, by using Eqn. (3.11) we get:

\[ x_{ij} = \text{sign} \left\{ \alpha + \frac{\beta}{2\sigma^2} y_{ij} + \beta \sum_{kl} \theta_{kl} (x_{kl}) \right\} \]  

(4.8)

which completes the proof.

This interesting observation means that the estimate given by Eq. (4.6) which was derived on the basis of continuous intensity scenes is valid also for unordered colour scenes with just a few levels including binary images. Computer simulation on different scenes with different grey levels agrees fully with this extremely important result, that the computational cost becomes quite independent of the number of grey levels (some time might be lost in rounding the estimates up or down to the feasible figures).

When the number and/or the values of the grey levels are unknown, the method can still be applied to provide an estimate that can be used to generate the final restored image through histogram mode analysis.

4.4 Choice of Model Parameters

The restoration algorithms require that the model parameters are known or estimated prior to (or during) restoration. The performance of the restored image is expected to depend on the choice of the model parameters.

A general theory of interactive, self-adjusting models that is practical and mathematically coherent may lie far ahead [1].
The following guidelines are the basis for the choice of the GD parameters, \( \alpha \) and \( \beta \), in this research.

1. For most scenes in which no one region is dominant a suitable value of \( \alpha \) is zero. In the author's study with different images, satisfactory results were obtained with \( \alpha = 0 \).

2. The choice of \( \beta \) is governed by the noise level. For univariate record the iterative algorithm is started with \( \beta = 0 \). This value corresponds to the ML classifier. The parameter \( \beta \) is incremented from iteration to iteration by a value depending on the noise level. If the noise level is not high, then large increments of \( \beta \) accelerate convergence. While if the noise level is high, a small increment of \( \beta \) must be used, fortunately, the algorithm is not sensitive to the specific value of \( \Delta \beta \). Clearly the number of iterations required for convergence might depend on the selected value of \( \Delta \beta \).

**REMARKS:**

1. For low SNR, using a value of \( \Delta \beta \) not consistent with the noise level will cause the program be trapped in a poor local maximum.

2. The economical stopping criterion is to measure the distance between two successive realizations and the program stops when this difference is less than a threshold value (as a direct result of the algorithm convergence).

3. As long as \( \Delta \beta \) is consistent with the noise level (typical value of \( \Delta \beta \) could be taken as \( 0.1 \times \rho \) where \( \rho \) is the SNR), the parameter \( \beta \) need not be estimated.
4.5 Results and Conclusion

In this section examples have been presented for the restoration algorithms described in sections 4.2 and 4.3. Fig. 4.3 shows an example of a four level image corrupted by two levels of additive noise. Restoration was done using the brute force technique and the quadratic pairwise interactions potential function. The results obtained show that the brute force technique is slightly more accurate. Fig. 4.4 shows the performance of the restoration algorithm using the quadratic potential function at two levels of additive noise for a five level scene. The convergence properties of the restoration algorithm are shown in Fig. 4.5 for an eight level scene corrupted by additive noise (using the quadratic potential function MRF scene model). Restoration of continuous intensity scenes are given in Fig. 4.6 and Fig. 4.7. Fig. 4.8 and Fig. 4.9 show the effect of controlling the resolution of the scene. The performance of the gradient based algorithm (using the quadratic potential function scene model) for a simulated image at different levels of additive noise is given in Fig. 4.10, 4.11 and 4.12. Finally, the algorithm performance for real image is given in Fig. 4.13.
Figure 4.2: Comparison between direct and gradient based restoration techniques. Four level image, \( \sigma = 30 \),
1 = original, 3 = noisy (SNR = 1.0), 6 = noisy (SNR = 2.0)
2, 5 = restorations using brute force optimization.
4, 7 = restorations using quadratic potential function.
\( C(2)/Q^0 = 3.675 \), \( Q(4)/Q^0 = 3.171 \),
\( Q(5)/Q^0 = 3.861 \), \( Q(7)/Q^0 = 3.122 \).
Figure 4.3: Performance of the quadratic potential function algorithm at two levels of noise. 
Five level image, $\sigma = 40$, $l$=original.
2, 3 = noisy (SNR = 0.5, 1.0).
4, 5 = restored (SNR = 0.5, SNR = 1.0).
$Q(4)/Q^0 = 13.63$, $Q(5)/Q^0 = 19.27$. 
Figure 4.4: Convergence properties of the gradient based restoration algorithm.
eight level image, SNR = 2.0, $\sigma = 40$, 1=original, 2=noisy, 3=after one iteration,
4 = after two iterations, ..., 9 = restored image.
$Q/Q_0 = (1.06, 1.31, 1.57, 1.58, 2.21, 2.47, 2.53)$
Figure 4.5: Continuous intensity scene corrupted by additive noise.
1 = original, \( \sigma = (67.96, 48.17, 21.76) \)
2, 3, 4 = noisy (SNR = 0.5, 1.0, 5.0),
5, 6, 7 = restored (SNR = 0.5, 1.0, 5.0)
\( Q/Q^0 = (6.12, 7.17, 16.67) \)
Figure 4.6: Continuous intensity scene corrupted by additive noise.
1 = original, $\sigma = (56.31, 39.93, 18.08)$
2, 3, 4 = noisy (SNR = 0.5, 1.0, 5.0).
5, 6, 7 = restored (SNR = 0.5, 1.0, 5.0)
$Q/Q^0 = (4.27, 5.05, 7.25)$
Figure 4.7: Controlling the resolution of a continuous intensity scene.
1 = original, 2 = noisy (SNR = 5), 3 = restored (256)
4, 5, 6 = restored (8 levels, 16 levels, 32 levels)
\( Q/Q^0 = (1.28, 1.41, 1.49) \)
Figure 4.8: Controlling the resolution of a continuous intensity scene:
1 = original, 2 = noisy (SNR = 5), 3 = restored (256)
4, 5, 6 = restored (8 levels, 16 levels, 32 levels)
Q/Q^0 = (1.43, 1.54, 1.64)
Figure 4.9: Performance at high level of noise for simulated image.
three level image, SNR = 0.55, $\sigma = 127.3$
1 = original, 2 = Noisy, 3 = Restored
$Q/Q^0 = 8.81$
Figure 4.10: Performance at medium level of noise for simulated image.
three level image, SNR = 0.55, $\sigma = 57.0$
1 = original, 2 = Noisy, 3 = Restored
$Q/Q^0 = 10.56$
Figure 4.11: Performance at low level of noise for simulated image.
three level image, SNR = 0.55, $\sigma = 35.0$
1 = original, 2 = Noisy, 3 = Restored
$Q/Q_0 = 10.89$
Figure 4.12: Performance at different levels of noise for real image.
four level image. 1 = original, \( \sigma = (40.0, 20.0, 10.0) \)
2, 3, 4 = noisy (SNR = 1.0, 2.0, 3.0),
5, 6, 7 = restored (SNR = 1.0, 2.0, 3.0)
\( Q/Q^0 = (3.51, 4.70, 2.74) \)
Chapter V

RESTORATION OF IMAGES CORRUPTED BY DIFFERENT SOURCES OF DEGRADATION USING A SPECIAL CLASS OF MARKOV RANDOM FIELD SCENE MODELS

5.1 Overview

Physical imaging systems are always subject to point and spatial degradation effects and corrupted by deterministic and stochastic disturbances. Sources of degradation include sensor/film nonlinearities, diffraction in the optical system, aberrations, atmospheric turbulence effects and motion blur. Noise disturbances may be caused by electronic imaging sensors, film granularity and channel transmission.

Accurate and precise image modelling is the key to successful and efficient restoration. Suppose a single image is available and this image is then used to develop parameters to describe the imaging system, a posteriori knowledge is then required to determine the point spread function (PSF) and the noise source(s). A posteriori determination of the degradation parameters can be found in [44] among others.

The degradation models will follow the standard modelling of image formation and recording processes. For more details see [28] and [45]. A space invariant point spread function (SIPSF) matrix represented by $H_\phi$ where the parameter $\phi$ is called blurring angle which is a measure of the amount of focus blur is assumed. This focus blur is due to the formation of the image, while recording the image result in sensor nonlinearities which is given by the parameters $C$ and $t$, in addition to random sensor noise which may be additive/multiplicative or both. For
clarity of presentation attention will be focused on the traditional Gaussian white noise source(s).

Extension to a general noise process is a matter of notation.

In previous chapters, the case of simple additive noise was considered. In this chapter the
most general degradation model given by Eq.(2.4) will be considered. In section 5.2, below, the
degradation model contains additive/multiplicative noise with blur and nonlinearity. The
implementation of the derived algorithms is the subject of section 5.3. Section 5.4 considers the
most general model (additive and/or multiplicative noise with blur and nonlinearity). Finally, the
results obtained by computer simulation are the subject of section 5.5.

5.2 Restoration of Images Corrupted by Additive/Multiplicative Noise With Blur
and Nonlinearity Using MRF Scene Models

The model given by Eqn. (2.4) is:

\[ y_{ij} = C \left\{ \sum_{kl} H(k, l) F(x[i + k, j + l]) \right\} u_{ij} + w_{ij} \]

where one of the two noise sources is present at a time. This Eqn. can be written in the compact
form:

\[ y_{ij} = C \left\{ \sum_{kl} H(k, l) F(x[i + k, j + l]) \right\} (\ast) N_{ij} \]

(5.1)

Where:

\( (\ast) \) is an invertible operation (additive or multiplicative).

\[ N = W \quad \text{if} \ (\ast) = + \]

\[ N = U \quad \text{if} \ (\ast) = . \]

Using Eqn. (2.6) and (2.7) Eqn. (5.1) can be written in the form:
\[ y_{ij} = C \left( K_0(\phi) + K_1(\phi) x_{ij} \right) N_{ij} \quad (5.2) \]

Where:

\[ K_0(\phi) = \left\{ f_m + \frac{f_d \sin^2 \phi}{n_w} \sum_{k \in \zeta_{ij}} x_{ki} \right\} \geq 0 \quad (5.3) \]

\[ K_1(\phi) = f_d \cos^2 \phi \geq 0 \quad (5.4) \]

### 5.2.1 Mathematical Foundation

The restoration algorithm is iterative and at each iteration the state of the current pixel is to be updated based on the available data. The available data include the original record and the current realization elsewhere. In the formulation to be given below we will maximize the posterior marginal probability at each pixel. Let \([K]\) represent the clock cycle (timing) of the current realization, then \([K + 1]\) represents the timing of the updated state.

The maximization problem can be written in the form:

\[
\max_{X_{ij}} P(Y_{ij}, X_{ij}, \{X \setminus X_{ij}\})
\]

Using Baye's theorem, after neglecting the effect of the constant terms, we get:

\[
P\left(X_{ij}^{[K+1]} \mid Y_{ij}, X_{ij}^{[K]} \right) \propto P\left(Y_{ij} \mid X_{ij}^{[K+1]}, X_{\zeta_{ij}}^{[K]} \right) P_{ij}\left(X_{ij}^{[K+1]} \mid X_{\eta_{ij}}^{[K]} \right) \quad (5.5)
\]

Where:

- \(X \setminus X_{ij}\) means excluding the random variable \(X_{ij}\) from the random field \(X_{ij}\).
- \(X_{\zeta_{ij}}, X_{\eta_{ij}}\) represent a set of random variables belonging to \(X\) and indexed by \(\zeta_{ij}\) and \(\eta_{ij}\) respectively.
$P_{ij}(X_{ij} | X_{n_i})$ represents the contribution of the neighboring pixels to the conditional probability of the center pixel.

For ease of notation Eq. (5.5) is simplified as:

$$P(X_{ij} | Y, (X \setminus X_{ij})) \propto P(Y_{ij} | X_{n_{ij}}) \cdot P_{ij}(X_{ij} | X_{n_{ij}})$$  \hspace{1cm} (5.6)

Where $P_{ij}(X_{ij} | X_{n_{ij}})$ is given by:

$$P_{ij}(X_{ij} | X_{n_{ij}}) \propto e^{\alpha x_{ij} + \beta \sum_{u \in n_{ij}} R(x_{ij}, x_{ui})}$$  \hspace{1cm} (5.7)

and

$$P(Y_{ij} | X_{n_{ij}} = P[ Y_{ij} = C \{K_0(\phi) + K_1(\phi) x_{ij}\}^f | X_{n_{ij}} ]$$  \hspace{1cm} (5.8)

CASE I

Additive noise with blur and nonlinearity:

In this case $(\ast) = + \& N = W$, thus:

$$P(Y_{ij} | X_{n_{ij}} = P[ W_{ij} = y_{ij} - C \{K_0(\phi) + K_1(\phi) x_{ij}\}^f | X_{n_{ij}} ]$$

Since $X$ & $W$ are independent stochastic processes we get:

$$P(Y_{ij} | X_{n_{ij}} = P[ W_{ij} = y_{ij} - C \{K_0(\phi) + K_1(\phi) x_{ij}\}^f ]$$

But:

$$W = N(0, \sigma_w^2)$$

Then:

$$P(Y_{ij} | X_{n_{ij}} = \frac{1}{\sqrt{2\pi\sigma_w^2}} e^{-\frac{-(y_{ij} - C \{K_0(\phi) + K_1(\phi) x_{ij}\}^f)^2}{2\sigma_w^2}}$$  \hspace{1cm} (5.9)
CASE 2.

Multiplicative noise with blur and nonlinearity:

In this case (*) = s & N = U

It can be shown that:

\[
P(Y_{ij} | X_{ij}) = \frac{1}{\sqrt{2\pi\sigma_u^2}} \exp\left(-\frac{1}{2\sigma_u^2} \left(\frac{y_{ij} - (C(K_0(\phi) + K_1(\phi)) x_{ij}'}{2\sigma_u^2}\right)^2\right)
\]  

(5.10)

Now it is possible to write expressions for the MAP estimate and the maximum marginal probability at each individual pixel.

1- Additive Noise With Blur and Nonlinearity.

The MAP estimate is given by:

\[
\text{Max} \sum_{x \in \chi} \sum_{ij \in L_i} \left\{ \alpha x_{ij} + \beta \sum_{kl} \theta_{kl} R(x_{ij}, x_{kl'}) - \frac{(y_{ij} - C(K_0(\phi) + K_1(\phi)) x_{ij})^2}{2\sigma_u^2}\right\}
\]

Maximization of the posterior marginal probability at each pixel is given by:

\[
\text{Max} \left\{ \alpha x_{ij} + \beta \sum_{kl} \theta_{kl} R(x_{ij}, x_{kl'}) - \frac{(y_{ij} - C(K_0(\phi) + K_1(\phi)) x_{ij})^2}{2\sigma_u^2}\right\}
\]

(5.11)

2- Multiplicative Noise With Blur and Nonlinearity

The MAP estimate is given by:

\[
\text{Max} \sum_{x \in \chi} \sum_{ij \in L_i} \left\{ \alpha x_{ij} + \beta \sum_{kl} \theta_{kl} R(x_{ij}, x_{kl'}) - \frac{y_{ij} - C(K_0(\phi) + K_1(\phi)) x_{ij}'}{2\sigma_u^2}\right\}
\]

(5.12)

\[
\left.\left(1 - \frac{y_{ij}}{C(K_0(\phi) + K_1(\phi)) x_{ij}'}\right)^2\right]
\]

(5.13)
The posterior marginal probability at each pixel is maximized by:

\[
\text{Max}_{x_{ij} \in x} \left\{ \alpha \ x_{ij} + \beta \sum_{kl} \theta_{kl} \ R(x_{ij}, x_{kl}) - \frac{\left( 1 - \frac{y_{ij}}{C \ (K_o(\phi) + K_1(\phi) \ x_{ij})} \right)^2}{2\sigma^2_u} \right\}
\]

It should be noted that using Eqns. (5.12) and (5.14) avoids direct maximization which is called for if one uses instead Eqns. (5.11) or (5.13); therefore, the functions needed to be maximized are given by:

\[
V_a(x_{ij}) = \alpha \ x_{ij} + \beta \sum_{kl} \theta_{kl} \ R(x_{ij}, x_{kl}) - \frac{\left[ y_{ij} - C \ (K_o(\phi) + K_1(\phi) \ x_{ij}) \right]^2}{2\sigma^2_w} \tag{5.15}
\]

and

\[
V_m(x_{ij}) = \alpha \ x_{ij} + \beta \sum_{kl} \theta_{kl} \ R(x_{ij}, x_{kl}) - \frac{\left( 1 - \frac{y_{ij}}{C \ (K_o(\phi) + K_1(\phi) \ x_{ij})} \right)^2}{2\sigma^2_u} \tag{5.16}
\]

\(V_a(x_{ij})\) and \(V_m(x_{ij})\) are the objective functions for additive and multiplicative noise respectively.

To restore the corrupted image, \(V_a\) expressed by Eq. (5.15) or \(V_m\) expressed by Eq. (5.16) should be maximized by employing any of the techniques described in chapter IV.

Eqns. (5.15) and (5.16) can be solved (in their most general form) using the brute force (direct search) optimization technique described in chapter IV; however, the quadratic potential function has been used for fast, reliable and efficient implementation. The following applications are cited.
5.2.2 Applications

1.a. Simple additive noise:

The estimate can be shown to be in the form:

\[
\hat{x}_{ij} = \frac{\alpha + \beta \sum_{kl} \theta_{kl} x_{ij} + \frac{\bar{y}_{ij} f_d}{\sigma_w^2}}{\beta \sum_{kl} \theta_{kl} + \left(\frac{f_d}{\sigma_w}\right)^2}
\]  

(5.17)

where: \((i,j) \in L_t\) and \(\bar{y}_{ij}\) is given in Eq. (3.10):

The problem is strictly convex and the solution is unique.

1.b. Additive noise with local blur: The solution can be found in the form:

\[
\hat{x}_{ij} = \frac{\alpha + \beta \sum_{kl} \theta_{kl} x_{ij} + \frac{\bar{y}_{ij} K_1(\phi)}{\sigma_w^2}}{\beta \sum_{kl} \theta_{kl} + \left(\frac{K_1(\phi)}{\sigma_w}\right)^2}
\]  

(5.18)

where:

\[
\bar{y}_{ij} = y_{ij} - K_0(\phi)
\]

and

\[(i,j) \in L_t\]

The problem is strictly convex and a unique solution can be obtained.

1.c. Additive noise with local blur and nonlinearity: The general problem can be solved using the brute force optimization procedures described in chapter IV; however, for some special cases, the solution can be found using the quadratic potential function efficiently.
This special case can be handled efficiently using the quadratic potential function as follows, upon substitution in Eqn. (5.15):

\[
V_a(x_{ij}) = \alpha x_{ij} + \beta \sum_{kl} \theta_{kl} \left\{ 1 - \frac{(x_{ij} - x_{kl})^2}{2} \right\} - \frac{(y_{ij} - C \sqrt{K_0(\phi) + K_1(\phi) x_{ij}})^2}{2\sigma_w^2} \tag{5.19}
\]

Differentiating (5.19) with respect to \(x_{ij}\) we get:

\[
\frac{\partial V_a(x_{ij})}{\partial x_{ij}} = \alpha + \beta \sum_{kl} \theta_{kl} x_{ij} - \frac{C^2 K_1(\phi)}{2\sigma_w^2} - \left( \beta \sum_{kl} \theta_{kl} \right) x_{ij} + \frac{C K_1(\phi) y_{ij}}{2\sigma_w^2 \sqrt{K_0(\phi) + K_1(\phi) x_{ij}}} \]

Let:

\[z_{ij} = K_0(\phi) + K_1(\phi) x_{ij} ; \quad t_{ij} = \sqrt{z_{ij}} > 0\]

It can be shown that:

\[
\frac{\partial V_a(x_{ij})}{\partial x_{ij}} = \frac{a}{t_{ij}} \left( t_{ij}^3 + pt_{ij} + q \right) \tag{5.20}
\]

\[
\frac{\partial^2 V_a(x_{ij})}{\partial x_{ij}^2} = -\frac{C K_1^2(\phi) y_{ij}}{4\sigma_w^2} \frac{1}{t_{ij}^3} - \beta \sum_{kl} \theta_{kl} < 0 \tag{5.21}
\]

Where:

\[
a = -\frac{\beta \sum_{kl} \theta_{kl}}{K_1(\phi)}
\]

\[
b = \alpha + \beta \sum_{kl} \theta_{kl} x_{kl} - \frac{C^2 K_1(\phi)}{2\sigma_w^2}
\]
\[ p = - \frac{b K_i(\phi) - K_0(\phi)}{\beta \sum_{kl} \theta_{kl}^3} \]
\[ q = - \frac{C K_i(\phi)^2 \nu_{ij}}{2\beta \sum_{kl} \theta_{kl} \sigma_w^2} \]  
(5.22)

Equating (5.20) to zero, results in a cubic Eqn. with one real solution (as a direct result of the convexity of the problem) and the required estimate is given as follows:

\[ t_{ij}^3 + p t_{ij} + q = 0 \]

The solution is given by:

\[ t_{ij} = \sqrt[3]{A} + \frac{-p}{3} \frac{1}{\sqrt[3]{A}} \]

Where:

\[ A = - \frac{q}{2} + \sqrt{\frac{q^2}{4} + \frac{p^3}{27}} \]

Thus:

\[ \tilde{t}_{ij} = t_{ij}^2 \]

and the required estimate is given by:

\[ \tilde{t}_{ij} = \frac{\tilde{t}_{ij} - K_0(\phi)}{K_1(\phi)} \]  
(5.23)

1.c.(ii) \( t = 1.5 \)

This special case can be handled efficiently using the quadratic potential function as follows:
\[ V_a(x_{ij}) = \alpha x_{ij} + \beta \sum_{kl} \theta_{kl} \left\{ 1 - \frac{(x_{ij} - x_{kl})^2}{2} \right\} - \frac{\left( \frac{y_{ij} - C \left[ K_0(\phi) + K_1(\phi) x_{ij} \right]^3}{2\sigma_w^2} \right)^2}{2\sigma_w^2} \]

Let:

\[ z_{ij} = K_0(\phi) + K_1(\phi) x_{ij} \quad ; \quad r_{ij} = \sqrt{z_{ij}} > 0 \]

It can be shown that:

\[ \frac{\partial V_a(x_{ij})}{\partial x_{ij}} = -b_0 \left( r_{ij}^4 + b_1 r_{ij}^2 + b_2 r_{ij} + b_3 \right) \]

Where:

\[ b_0 = \frac{3 C^2 K_1(\phi)}{2\sigma_w^2} > 0 \]

\[ b_1 = \frac{\left( \beta \sum_{kl} \theta_{kl} \right)}{b_0 K_1(\phi)} > 0 \]

\[ b_2 = -\frac{y_{ij}}{C} \leq 0 \]

\[ b_3 = \frac{\alpha + \beta \sum_{kl} \theta_{kl} x_{kl} + \beta \frac{K_0(\phi)}{K_1(\phi)} \sum_{kl} \theta_{kl}}{b_0} \]

The solution is obtained by finding the possible non-negative real roots of the fourth order polynomial given by:

\[ r_{ij}^4 + b_1 r_{ij}^2 + b_2 r_{ij} + b_3 = 0 \]
Fortunately, fourth order polynomials have a closed form solution and there is no need for root finding routines. However, due to the lack of convexity of this problem, the solution in this case is computationally more expensive than case 1.c.(i), but the solution is more economical than the brute force technique described in chapter IV. The feasible solution for Eqn. (5.25) is used to find the required estimate as follows:

\[ \hat{z}_{ij} = \tilde{f}_{ij} \]

and

\[ \hat{x}_{ij} = \frac{\hat{z}_{ij} - \bar{K}_0(\phi)}{\bar{K}_1(\phi)} \]  \hspace{1cm} (5.26)

1.c.(iii) \( t = -1 \)

This case is similar to the case of multiplicative noise with local blur given next.

2.b Multiplicative Noise With Local Blur

\[ V_m(x_{ij}) = \alpha x_{ij} + \beta \sum_{kl} \theta_{kl} \left\{ 1 - \frac{(x_{ij} - x_{kl})^2}{2} \right\} - \frac{\left( 1 - \frac{y_{ij}}{C(\bar{K}_0(\phi) + \bar{K}_1(\phi) x_{ij})} \right)^2}{2\sigma^2} \]  \hspace{1cm} (5.27)

for (i,j) \( \in \text{Lt} \), let:

\[ z_{ij} = K_0(\phi) + K_1(\phi) x_{ij} \quad x_{ij} > 0 \]

It can be shown that:

\[ \frac{\partial V_m(x_{ij})}{\partial x_{ij}} = a_0 \left( \frac{z_{ij}^4 + a_1 z_{ij}^3 + a_2 z_{ij} + a_3}{z_{ij}^3} \right) \]

\[ \frac{\partial^2 V_m(x_{ij})}{\partial x_{ij}^2} = a_0 \bar{K}_1(\phi) \left( 1 - \frac{2 a_2}{z_{ij}^3} - \frac{3 a_3}{z_{ij}^4} \right) \]  \hspace{1cm} (5.28)

Where:
\[
\begin{align*}
a_0 &= -\frac{\beta \sum_{kl} \theta_{kl}}{K_1(\phi)} \\
a_1 &= -K_1(\phi) \frac{\alpha + \beta \sum_{kl} \theta_{kl} x_{kl} + \beta \frac{K_0(\phi)}{K_1(\phi)} \sum_{kl} \theta_{kl}}{\beta \sum_{kl} \theta_{kl}} \\
a_2 &= \frac{\gamma_{ij} K_1^2(\phi)}{\sigma_a^2 \beta \sum_{kl} \theta_{kl}} \geq 0 \\
a_3 &= -a_2 y_{ij} \leq 0
\end{align*}
\] (5.29)

The solution is obtained by finding the non-negative real roots of the fourth order polynomial:

\[
\ddot{z}_{ij} + a_1 \dot{z}_{ij}^3 + a_2 \dot{z}_{ij}^2 + a_3 = 0
\] (5.30)

The solution is given by:

\[
\dot{x}_{ij} = \frac{\dot{z}_{ij} - K_0(\phi)}{K_1(\phi)}
\] (5.31)

2.c. Multiplicative noise with blur and nonlinearity

The general problem can be solved using the brute force optimization procedures described in chapter IV, however, for some special cases, the solution can be found using the quadratic potential function efficiently.

2.c.(i) \( t = 0.5 \)

This special case can be handled efficiently using the quadratic potential function as follows:
\[ V_m(x_{ij}) = \alpha x_{ij} + \beta \sum_{kl} \theta_{kl} \left\{ 1 - \frac{(x_{ij} - x_{kl})^2}{2} \right\} - \frac{\left( 1 - \frac{y_{ij}}{C} \sqrt{K_0(\phi) + K_1(\phi) x_{ij}} \right)^2}{2\sigma_x^2} \]  

(5.32)

Upon differentiation with respect to \( x_{ij} \) we get:

\[
\frac{\partial V_m(x_{ij})}{\partial x_{ij}} = \alpha + \beta \sum_{kl} \theta_{kl} x_{kl} - \frac{y_{ij}^2 K_1(\phi)}{2\sigma_x^2 C^2} - \left( \beta \sum_{kl} \theta_{kl} \right) x_{ij} + \frac{y_{ij} K_1(\phi)}{2\sigma_x^2 C \sqrt{K_0(\phi) + K_1(\phi) x_{ij}}} 
\]

Let:

\[ z_{ij} = K_0(\phi) + K_1(\phi) x_{ij} ; \quad t_{ij} = \sqrt{z_{ij}} > 0 \]

It can be shown that:

\[
\frac{\partial V_m(x_{ij})}{\partial x_{ij}} = \frac{a}{t_{ij}} \left( t_{ij}^2 + p t_{ij} + q \right) \]

(5.33)

\[
\frac{\partial^2 V_m(x_{ij})}{\partial x_{ij}^2} = -\frac{K_1^2(\phi) y_{ij}}{4\sigma_x^2 C} \left( \frac{1}{t_{ij}^2} - \beta \sum_{kl} \theta_{kl} < 0 \right) \]

(5.34)

Where:

\[ a = -\frac{\beta \sum_{kl} \theta_{kl}}{K_1(\phi)} \]

\[ b = \alpha + \beta \sum_{kl} \theta_{kl} \left( x_{kl} + \frac{K_0(\phi)}{K_1(\phi)} \right) - \frac{y_{ij}^2 K_1(\phi)}{2\sigma_x^2 C} \]

\[ p = -\frac{b K_1(\phi)}{\beta \sum_{kl} \theta_{kl}} \]
\[ q = - \frac{K_1(\phi)^3 y_{ij}}{2 C \sigma^2 \beta \sum_{kl} \theta_{kl}} \] (5.35)

the cubic Eqn. has only one real solution, and the required estimate is given by equating Eq. (5.33) to zero.

\[ f_{ij}^3 + pf_{ij} + q = 0 \]

The solution is given by:

\[ f_{ij} = \frac{3}{\sqrt[3]{A}} + \frac{-p}{3} \frac{1}{3\sqrt[3]{A}} \]

Where:

\[ A = -\frac{q}{2} + \sqrt{\frac{q^2}{4} + \frac{p^3}{27}} \]

Thus:

\[ \hat{f}_{ij} = f_{ij}^2 \]

and the required estimate is given by:

\[ \hat{x}_{ij} = \frac{\hat{f}_{ij} - K_0(\phi)}{K_1(\phi)} \] (5.36)

2.c.(ii) \( t = -1 \)

This case is similar to the case of additive noise with local blur

A unique global solution can be found in the form:
\[
\hat{x}_{ij} = \frac{\alpha + \beta \sum_{kl} \theta_{kl} x_{kl} + \frac{y_{ij} K_1(\phi)}{\sigma_u^2 C} \left(1 - \frac{y_{ij} K_0(\phi)}{C}\right)}{\beta \sum_{kl} \theta_{kl} + \left(\frac{y_{ij} K_1(\phi)}{C \sigma_u}\right)^2}
\]

2.c.(iii) \( t = 1.5 \)

This case is similar to case 1.c.(ii) given above.

**REMARK:**

The quadratic pairwise interactions MRF as a special class of scene models are used in the restoration of images corrupted by a variety of degradation sources. However, the cost is minimal in the case of additive noise with or without blur, and in some cases of nonlinearity. The implementation of the multiplicative noise with or without blur requires the solution of fourth degree polynomial. The technique is quite efficient if the number of grey levels is moderate to high (in comparison to direct search optimization).

**5.3 Restoration of Images Corrupted By Compound Noise With Blur and Nonlinearity Using MRF Scene Models**

In order to eliminate the problems associated with the implementation of the multiplicative noise algorithm given in section 5.2, the application of an optimal linear model approximation first proposed by J.Lee [11] will be presented. Lee [11] used simple additive and/or multiplicative noise in his model. The objective is to apply this linear model to the environment of stochastic estimation using MRF scene models to a more general degradation phenomena. This section is
organized as follows: In section 5.3.1 the optimal linear model for multiplicative noise with blur and nonlinearity will be given. Subsection 5.3.2 is an extension of 5.3.1 in the case of compound noise with blur and nonlinearity.
5.3.1 Optimal Linear Model and Restoration of Images Degraded by
Multiplicative Noise With Blur and Nonlinearity Using MRF Scene model

Neglecting the effect of the additive noise term in Eq. (2.4) result in:

\[ Y = C \left[ H * F(X) \right] U \]

i.e.

\[ y_{ij} = C \left[ K_o(\phi) + K_i(\phi) x_{ij} \right] u_{ij} \]

Where \( K_o(\phi) \) and \( K_i(\phi) \) are given in Eq. (5.3) and Eq. (5.4)

The model can be represented by Fig. 5.1

![Diagram of multiplicative noise with blur and nonlinearity degradation model]

**Figure 5.1**: Multiplicative noise with blur and nonlinearity degradation model

Let:

\[ C \left[ K_o(\phi) + K_i(\phi) x_{ij} \right] = g(x_{ij}) \]

The linear approximation is given by:
\[ \hat{y}_{ij} = A_1 u_{ij} + A_2 g(x_{ij}) + A_3 \]

Where \( A_1, A_2, \) and \( A_3 \) are constants to be determined by a standard criterion.

1. \( E\{Y_{ij}\} = E\{\hat{Y}_{ij}\} \quad (5.39) \)

2. \( E\{(Y_{ij} - \hat{Y}_{ij})^2\} \) be minimized. \( (5.40) \)

\( E \) is the expectation operator.

Eq. (5.39) guarantees an unbiased estimate while Eq. (5.4) minimizes the mean square error (MMSE). The mathematical derivation of \( A_1, A_2, \) and \( A_3 \) is given in appendix A. It is not difficult to see that the linear model is identical to the linear term of the taylor series expansion around the means \( \bar{C}(H F(x_{ij})^T, \bar{u}_{ij}). \)

Let:

\[ \lambda_{ij} = \bar{g}(x_{ij}) = \frac{\bar{y}_{ij}}{\bar{u}_{ij}} \quad (5.41) \]

Thus:

\[ \hat{y}_{ij} = \bar{u}_{ij} g(x_{ij}) + \lambda_{ij} (u_{ij} - \bar{u}_{ij}) \quad (5.42) \]

\( \lambda_{ij} \) can be found (estimated) by calculating the local means of the degraded image using a window with an appropriate size (typically \( 7 \times 7 \) [11]).

Since each pixel can be processed independently, the implementation of this step can be done in real time using parallel processor capabilities. Eqn. (5.42) gives:

\[ u_{ij} = \frac{y_{ij} - \bar{u}_{ij} g(x_{ij}) + \lambda_{ij} \bar{u}_{ij}}{\lambda_{ij}} \]

\[ P(Y_{ij} \mid X_{ij}) = P \left( U_{ij} = \frac{y_{ij} - \bar{u}_{ij} g(x_{ij}) + \lambda_{ij} \bar{u}_{ij}}{\lambda_{ij}} \mid X_{ij} \right) \]
Since X and U are independent stochastic processes, then:

\[ P(Y_{ij} \mid X_{ij}) = P(U_{ij} = u_{ij}) \]

\[ = \frac{1}{\sqrt{2\pi\sigma_u^2}} e^{-\frac{(y_{ij} - \mu_i - g(x_j))^2}{2\sigma_u^2}} \quad (5.43) \]

Where:

\[ \sigma_u^2 = \lambda_{ij}^2 \sigma_u^2 \quad (5.44) \]

Maximization of the posterior marginal probability at each pixel is obtained by maximizing \( V_m(x_{ij}) \): for all \((i,j) \in L_t \)

Where:

\[ V_m(x_{ij}) = \alpha \cdot x_{ij} + \beta \sum_{kl} \theta_{kl} \cdot R(x_{ij}, x_{kl}) - \frac{(y_{ij} - \mu_i - g(x_j))^2}{2\sigma_u^2} \quad (5.45) \]

Using the quadratic pairwise interaction MRF potential function we get:

\[ V_m(x_{ij}) = \alpha \cdot x_{ij} + \beta \sum_{kl} \theta_{kl} \left\{ 1 - \frac{(x_{ij} - x_{kl})^2}{2} \right\} - \frac{(y_{ij} - \mu_i - g(x_j))^2}{2\sigma_u^2} \quad (5.46) \]
Applications

1- Multiplicative Noise With or Without Blur

The implementation of the multiplicative noise with or without blur is made very simple by using the optimal linear model and the resulting estimate can be shown to be:

\[
\hat{x}_{ij} = \frac{\alpha + \beta \sum_{kl} \theta_{kl} x_{kl} + \frac{\bar{u}_{ij} K_{i}(\phi) \bar{y}_{ij}}{\sigma_{u}^2}}{\beta \sum_{kl} \theta_{kl} + \left( \frac{\bar{u}_{ij} K_{i}(\phi)}{\sigma_{u}} \right)^2}
\]  

(5.47)

where:

\[
\bar{y}_{ij} = y_{ij} - \bar{u}_{ij} K_{0}(\phi)
\]  

(5.48)

2- Multiplicative Noise With Blur and Nonlinearity

Consider the case \( t = 0.5 \)

This value was used by Geman and Geman [1], and the solution was obtained using stochastic relaxation. The linear model approximation gives a solution that is similar to case 1.c.(i) described in section 5.2. In comparison with [1], the proposed technique is computationally more efficient.

5.3.2 Restoration of Images Degraded by Combined Additive and Multiplicative Noise With Blur and Nonlinearity Using MRF scene Models

The degradation model is given in the most general form in Eq. (2.4)

\[
Y = C \left[ H \ast F(X) \right]' \cdot U + W
\]

or

\[
y_{ij} = C \left[ \hat{K}_{0}(\phi) + K_{i}(\phi) x_{ij} \right]' u_{ij} + w_{ij}
\]  

(5.49)
The model is shown below:

\[ F(x_{ij}) \rightarrow H \rightarrow C_{ij} \rightarrow u_{ij} \rightarrow w_{ij} \rightarrow y_{ij} \]

**Figure 5.2:** Compound noise with blur and nonlinearity degradation model

Let:

\[ C \left[ K_0(\phi) + K_t(\phi) x_{ij} \right] = g(x_{ij}) \]

and

\[ \delta_{ij} = \bar{g}(x_{ij}) = \frac{\bar{y}_{ij} - \bar{w}_{ij}}{\bar{u}_{ij}} \]  \hspace{1cm} (5.50)

It can be shown that:

\[ \bar{y}_{ij} = \bar{u}_{ij} g(x_{ij}) + \delta_{ij} (u_{ij} - \bar{u}_{ij}) + w_{ij} \]  \hspace{1cm} (5.51)

\( \delta_{ij} \) can be found (estimated) by calculating the local means of the degraded image using a window with an appropriate size (typically 7 x 7 [11]).

Eq. (5.51) can be written in the form:

\[ \delta_{ij} u_{ij} + w_{ij} = \bar{y}_{ij} - \bar{u}_{ij} g(x_{ij}) + \delta_{ij} \bar{u}_{ij} \]  \hspace{1cm} (5.52)

Consider a stochastic process \( V \), which is defined as follows:
\[ V = \{ V_{ij}, (i,j) \in \Omega \} \]

Where the associated random variables are related by:

\[ V_{ij} = \delta_{ij} U_{ij} + W_{ij} \]

Eq. (5.52) gives:

\[ P(Y_{ij} | X_{\zeta_{ij}}) = P(V_{ij} = y_{ij} - \overline{u}_{ij} g(x_{ij}) - \delta_{ij} \overline{u}_{ij} | X_{\zeta_{ij}}) \] (5.53)

It can be shown that (details are given in Appendix B), the distribution of \( V \) is also Gaussian

\[ V_{ij} = N(\overline{v}_{ij}, \sigma^2) \]

Where

\[ \overline{v}_{ij} = \overline{u}_{ij} + \overline{w}_{ij} = \delta_{ij} \overline{u}_{ij} + \overline{w}_{ij} \] (5.54)

\[ \sigma^2 = \sigma_u^2 + \sigma_w^2 = \delta_{ij} \sigma_u^2 + \sigma_w^2 \] (5.55)

It is clear that \( V \) and \( X \) are independent stochastic processes and (5.53) gives:

\[ P(Y_{ij} | X_{\zeta_{ij}}) = P(V_{ij} = y_{ij} - \overline{u}_{ij} g(x_{ij}) + \delta_{ij} \overline{u}_{ij}) \]

\[ = \frac{1}{\sqrt{2\pi\sigma_v^2}} e^{-rac{(y_{ij} - \overline{u}_{ij} g(x_{ij}) - \delta_{ij} \overline{u}_{ij})^2}{2\sigma_v^2}} \] (5.56)

the objective function needed to be maximized can be written in the form:

\[ V_c(x_{ij}) = \alpha \cdot x_{ij} + \beta \sum_{kl} \theta_{kl} R(x_{ij}, x_{kl}) - \frac{(y_{ij} - \overline{u}_{ij} g(x_{ij}) - \overline{w}_{ij})^2}{2\sigma_v^2} \] (5.57)

Using the quadratic potential function we get:

\[ V_c(x_{ij}) = \alpha \cdot x_{ij} + \beta \sum_{kl} \theta_{kl} \left\{ 1 - \frac{(x_{ij} - x_{kl})^2}{2} \right\} - \frac{(y_{ij} - \overline{u}_{ij} g(x_{ij}) - \overline{w}_{ij})^2}{2\sigma_v^2} \] (5.58)
5.3.3 Applications

1. Combined additive and multiplicative noise with or without blur:

Using Eqn. (5.58) with $\bar{u}_i = 1$ and $\bar{w}_i = 0$, the required estimate can be shown to be:

$$\hat{x}_{ij} = \frac{\alpha + \beta \sum_{kl} \theta_{kl} x_{kl} + \frac{K_i(\phi) \bar{y}_{ij}}{\sigma_v^2}}{\beta \sum_{kl} \theta_{kl} + \left(\frac{K_i(\phi)}{\sigma_v^2}\right)}$$ \hspace{1cm} (5.59)

where:

$$\bar{y}_{ij} = y_{ij} - K_i(\phi)$$ \hspace{1cm} (5.60)

Where:

2. Combined additive and multiplicative noise with or without blur and nonlinearity:

The general nonlinearity problem can be handled using the brute force optimization techniques described in chapter IV. The function needed to be maximized can be expressed as:

$$V_c(x_{ij}) = \alpha x_{ij} + \beta \sum_{kl} \theta_{kl} \left\{ 1 - \frac{|x_{ij} - x_{kl}|}{2^{-1}} \right\} - \frac{(y_{ij} - g(x_{ij}))^2}{2\sigma_v^2}$$

where $r$ is a tuning constant, which can be chosen arbitrarily. The special cases of nonlinearity discussed in subsection 5.1.c can be implemented using the quadratic potential function ($r = 2$) with minimal computational task.

**REMARK:** The optimal linear approximation of the multiplicative noise results in a merge of the two independent stochastic processes representing the additive and multiplicative noises into an equivalent stochastic process represented by $V$ whose variance is pixel dependent (see Appendix B).
5.4 **Sensitivity Analysis**

In this section the sensitivity of the restoration algorithms with respect to the blurring angle $\phi$, the nonlinearity constant $t$ and the signal to noise ratio $\rho$, will be discussed.

1. **Sensitivity with respect to $\phi$.**

The original image is corrupted by additive noise with blur. The actual blurring angle was forty five degrees, the restoration algorithm was used to restore the corrupted image. The blurring angle was allowed to vary in the range zero to ninety degrees. The quality of restoration shows that the results were very insensitive to inaccurate estimation of the blurring angle $\phi$ for almost the entire range as shown in Fig. 5.3.

2. **Sensitivity with respect to $t$.**

The original image was corrupted by additive noise with nonlinearity. The actual value of $t$ used was 0.5, the restoration algorithm was used to restore the corrupted image. $t$ was allowed to vary between 0.2 to 1. The quality of the restored image shows that the results (restored images) were quite sensitive to the nonlinearity parameter $t$ as shown in Fig. 5.4.

3. **Sensitivity with respect to $\rho$.**

The original image was corrupted by simple additive noise. The actual signal to noise ratio was 0.54, the restoration algorithm was used to restore the corrupted image. The used SNR was allowed to vary between 0.1 and 1.2. The quality of the restored images shows that the results were not affected by inaccurate estimation of the signal to noise ratio as shown in Fig. 5.5. Figure 5.7 shows the sensitivity of the example given in Figure 5.6. (note that in this case the parameter $\beta$ was chosen on an ad. hoc. basis).
Figure 5.6: Restoration at high level of noise.
three level image. SNR = 0.55, $\sigma = 90.0$
1 = original, 2 = Noisy, 3 = Restored
Q/Q_0 = 10.56
Figure 5.7: Sensitivity of Fig. 5.6 w.r.t SNR.
restorations at: 1=SNR .2, 2=SNR .4, 3=SNR .9, 4=SNR 1.2
Q(1)/Q^0 = 10.47, Q(2)/Q^0 = 10.52, Q(3)/Q^0 = 10.49, Q(4)/Q^0 = 10.38
Figure 5.8 shows the original scene and its degraded version. The original image was corrupted by a high level of additive noise, focus blur, and positive nonlinearity. The algorithm presented in section 5.2 (the gradient based algorithm using the quadratic potential function) was used to restore the image. The restored image is shown in Fig. 5.9. An example of multiplicative noise with nonlinearity for binary image is given in Fig. 5.10 and Fig. 5.11.

The performance of the multiplicative noise with nonlinearity at different levels of noise is given in Fig. 5.12. The effect of changing the amount of focus blur (size and angle) is given in Fig. 5.13. Fig. 5.14 shows continuous intensity scene and its degraded version. The degradation is due to multiplicative noise with negative non-linearity. The restored image is shown in Fig. 5.15. Figure 5.16 shows an example of compound noise with blur and nonlinearity at two levels of degradation, where the restored image was post processed. Figure 5.17 shows an original image and Fig. 5.18 shows the effect of adding compound noise to it. Figure 5.19 shows a current realization after five iterations. Finally, Fig. 5.20 shows the restored image after 10 iterations. Figure 5.21 gives an example of simple compound noise for eight level scene. Figure 5.22 shows an example of scene corrupted by compound noise. The additive noise was dominant in this combination. The restoration was performed in two ways. In the first, the compound noise algorithm was applied and the restored image was obtained. In the second, the noise was diagnosed as pure additive, and the restored image was obtained. The results show little difference between them. Figure 5.23 shows the case where the two noise components were equal. The additive noise algorithm could not restore the image as effective as the compound noise algorithm. While in Fig. 5.24 the multiplicative noise was dominant in the combination. The additive noise algorithm failed as shown in Fig. 5.24.
Figure 5.8: Additive noise with blur and nonlinearity.
three level image.  1=original, 2=noisy
SNR = 1.0, $\phi = 45$, $\sigma = 75.0$, and $t = 0.5$
Figure 5.9: Restored image of Fig. 5.8. $Q/Q^0 = 47.23$
Figure 5.10: Multiplicative noise with nonlinearity.

Binary image, 1=original, 2=noisy $\frac{\sigma_n}{\sigma_i} = 0.05$ and $t = -0.5$. 
Figure 5.11: Restored image of Fig. 5.10.
(Q/Q^0 = 8.241)
Figure 5.12: Performance of the multiplicative noise with non-linearity at different levels of noise.
Figure 5.13: Effect of changing blur extent on the quality of restoration.
Figure 5.14: Continuous intensity scene corrupted by multiplicative noise with nonlinearity. SNR=1.0, t=-0.5, 256 level image. 1=original, 2=noisy
Figure 5.15: Continuous intensity restored image..
(Q/Q^0 = 8.241)
Figure 5.16: Compound noise with blur and nonlinearity. (a) SNR = 1.0, t = 0.5, φ = 45,
(b) SNR = 2.0, t = 0.8, φ = 30, 1 = original, 2 = noisy, 3 = restored, 4 = post processed
Q(a)/Q^0 = (7.54, 7.91), Q(b)/Q^0 = (13.17, 14.15)
Figure 5.17: Original image
Figure 5.18: Degraded image of Fig. 5.17.
Figure 5.19: Scene realization after 5 iterations. $Q/Q^0 = 5.12$
Figure 5.20: Restored image (after 10 iterations). $Q/Q^0 = 8.76$
Figure 5.21: Simple compound noise. SNR = 2.0, $\frac{\sigma_2}{\sigma_1} = 0.05$. Eight level image

1=original, 2=noisy, 3=restored

$Q/Q_0 = 11.65$
Figure 5.22: Simple compound noise with dominant additive component. 1=original, 2=noisy, 3=restored (additive), 4=restored (compound)
Q(3)/Q^0 = 10.05, Q(4)/Q^0 = 11.15
Figure 5.23: Simple compound noise with equal components. 1=original, 2=noisy, 3=restored (additive), 4=restored (compound) $Q(3)/Q^0 = 7.86, Q(4)/Q^0 = 13.68$
Figure 5.24: Simple compound noise with dominant multiplicative component.
1=original, 2=noisy, 3=restored (additive), 4=restored (compound)
Q(3)/Q^0 = 1.15, Q(4)/Q^0 = 15.18
Chapter VI

SPECIAL PURPOSE ALGORITHM FOR RESTORATION OF IMAGES DEGRADED BY EXCESSIVE BLUR AND NOISE USING MRF SCENE MODELS

6.1 Overview

In chapter V an algorithm for restoration of images corrupted by focus blur and additive noise was developed. The algorithm was quite efficient in removing local blur and additive noise when the blur level was not too high. It was noted that when the local blur is excessive the algorithm performance was not good enough to remove the degradation. In the present chapter a special purpose algorithm for restoration of images degraded by excessive blur and additive noise is proposed.

In the previous chapter it was assumed that, given \( x \) the random variables \( Y_{ij} \), \( (i,j) \in L \) are conditionally independent and each \( Y_{ij} \) depends only on the scene values inside a window centered at \( (i,j) \). Fig. 6.1 shows such dependence given by an association between \( Y_{ij} \) and \( X_{i,j} \).

In this chapter this assumption will be maintained, but all the random variables in the degraded image whose blur extent includes the current pixel in the current realization of the scene random field will be considered. This will generate an expanded graph. Fig. 6.2 shows the resulting dependence among the random fields given by an association between \( Y_{i,j} \) and \( X_{i,j} \).
Figure 6.1: The dependance of the random variable $Y_{ij}$ on the random field $X$

If the original graph (blurring window) is of dimension $W$ and $h$, then the expanded graph would be a larger window of dimensions $\tilde{W}$ and $\tilde{h}$, where it can be shown that:

$$\tilde{W} = 2W - 1$$  \hspace{1cm} (6.1)

and

$$\tilde{h} = 2h - 1$$  \hspace{1cm} (6.2)
6.2 Mathematical Foundation

The state of the current pixel is to be updated using the available information. This information includes the original records and the current realization elsewhere. Parallel implementation of the restoration algorithm using SIMD machine architecture can be performed.
using a finite number of synchronous updates (clock cycles). This number of clock cycles depends on the size of the generated graph, and it represents the time required to process the whole image. Let \(|K\) represent the time (clock cycle) at the present state of the current realization, then \(|K + 1|\) would represent the time of the updated state. The posterior marginal probability at each pixel can be written in the form:

\[
P(X_{ij}^{[K+1]} | Y, (X \setminus X_{ij})^{[K]} ) \propto P(Y_{ij} | X_{ij}^{[K+1]}, X_{\zeta_{ij}}^{[K]} ) \cdot P_{ij}(X_{ij}^{[K+1]} | X_{\eta_{ij}}^{[K]} ) \prod_{m,n \in \Psi_{ij}} P(y_{mn} | X_{ij}^{[K+1]}, X_{\zeta_{mn}}^{[K]} )
\]

(6.3)

Where:

\[
\Psi_{ij} = \{ mn : (i,j) \in \zeta_{mn} \}
\]

(6.4)

\[
\Psi_{mn} = \{ i, j \}
\]

(6.5)

Eq. (6.3) can be simplified as:

\[
P(X_{ij} | Y, (X \setminus X_{ij}) ) \propto P(Y_{ij} | X_{\zeta_{ij}}^{=}) \cdot P_{ij}(X_{ij} | X_{\eta_{ij}}^{=}) \prod_{m,n \in \Psi_{ij}} P(y_{mn} | X_{\zeta_{mn}}^{=})
\]

(6.6)

Where:

\[
P(Y_{ij} | X_{\zeta_{ij}}^{=}) = P( W_{ij} = y_{ij} - K_0(\phi) - K_1(\phi) x_{ij} )
\]

\[
= \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(y_{ij} - K_0(\phi) - K_1(\phi) x_{ij})^2}{2\sigma^2}}
\]

(6.7)

The loglikelihood term is given by:

\[
I = \ln P(Y_{ij} | X_{\zeta_{ij}}^{=}) + \sum_{m,n \in \Psi_{ij}} \ln P(y_{mn} | X_{\zeta_{mn}}^{=})
\]

(6.8)

After neglecting constant terms in Eq. (6.8) we obtain:
\[
I = \frac{-1}{2\sigma^2} \sum_{\mu\nu} \left( y_{\mu\nu} - \sum_{k=-l_s}^{l_s} \sum_{l=-l_w}^{l_w} H(k, l) F_x(k + \mu, l + \nu) \right)^2
\]

Eqn. (6.9) can be written in the form (proof is given in Appendix C)

\[
I \approx \frac{1}{\sigma^2} K'(\phi) x_{ij} - \frac{1}{2\sigma^2} (n_{\infty} K_2^2(\phi) + K_1^2(\phi)) x_{ij}^2
\]

Where:

\[
K_1(\phi) = f_d \cos^2 \phi
\]

\[
K_2(\phi) = \frac{f_d \sin^2 \phi}{n_{\infty}}
\]

\[
K'_y(\phi) = K_2(\phi) \sum_{l_i=-l_s}^{l_s} \tilde{y}(i + l_i, j + l) + K_1(\phi) \tilde{y}(i, j)
\]

\[
\tilde{y}(i + l_i, j + l) = \tilde{y}(i + l_i, j + l) - K_2(\phi) \tilde{S}(i + l_i, j + l) - K_1(\phi) x(i + l_i, j + l)
\]

for all \((i, j) \in L_i\), and \((l_i, l) \in \xi_{ij}\)

\[
\tilde{y}(i, j) = y_{ij} - f_m
\]

\[
\tilde{S}(i + l_i, j + l) = \left\{ \sum_{k=-l_s}^{l_s} \sum_{l=-l_w}^{l_w} x(i + l_i + k, j + l + l) \right\} - x(i, j)
\]

\[
\tilde{S}(i + j) = \sum_{k=-l_s}^{l_s} \sum_{l=-l_w}^{l_w} x(i + k, j + l)
\]
The function needed to be maximized is given by:

\[
V_b(x) = \sum_{ij \in L} \left\{ \alpha x_{ij} + \beta \sum_{kl} \theta_{kl} R(x_{ij}, x_{kl}) + \frac{1}{\sigma^2} K_y(\phi) x_{ij} \right. \\
\left. - \frac{1}{2\sigma^2} \left( n_w K_2^2(\phi) + K_1^2(\phi) \right) x_{ij}^2 \right\}
\]

(6.18)

Where the natural logarithm of Eq. (6.4) was used. Maximization of Eq. (6.14) can be done in many different ways. Geman and Geman [1] proposed MAP estimation using stochastic relaxation. Their technique is powerful but computationally demanding. Elliott et al. [40] made an approximation to the MAP estimate by processing the image in strips of finite width recursively using a dynamic programming algorithm, or it can be maximized using brute force optimization given in chapter IV. These are just examples of non-economical solution to the restoration problem, however, the proposed gradient based optimization using the quadratic pairwise interaction potential function is given by:

\[
V_b(x) = \sum_{ij \in L} \left\{ \alpha x_{ij} + \beta \sum_{kl} \theta_{kl} \left\{ 1 - \frac{(x_{ij} - x_{kl})^2}{2} \right\} + \frac{1}{\sigma^2} K_y(\phi) x_{ij} \right. \\
\left. - \frac{1}{2\sigma^2} \left( n_w K_2^2(\phi) + K_1^2(\phi) \right) x_{ij}^2 \right\}
\]

(6.19)

can be maximized by differentiation and equating the derivative to zero. This gives an analytical formulation of the desired estimate. The problem is strictly convex resulting in a unique solution given by:
\[
\hat{x}_{ij} = \frac{\alpha + \beta \sum_{kl} \theta_{kl} x_{kl} + \frac{K_1(\phi)}{\sigma^2}}{\beta \sum_{kl} \theta_{kl} + \frac{\eta_0 K_2(\phi) + K_1(\phi)}{\sigma^2}}
\]  \hspace{1cm} (6.20)

6.3 Results and Conclusion

In this section some examples at different levels of blur are presented and a comparison with the restoration algorithm given in 5.2.2 will be given. If blur is not excessive the performance of the restoration algorithm is shown to be slightly better than that proposed in 5.2.2. Figure 6.3 shows a four-level image and Fig. 6.4 is the degraded version of Fig. 6.3. The restoration algorithm presented in 5.2.2 results in Fig. 6.5, while Fig. 6.6 is the result of applying Eqn. (6.20) above. The window was 5x3 and the blurring angle was equal to forty five degrees which is not excessive. A comparison between the two restoration techniques given by Eqns. (5.18) and (6.20) when the blurring angle is equal to ninety degrees is given in Figure 6.7. The restored images are given in Fig. 6.8. It is clear that the algorithm given by Eqn. (5.18) fails at this level of blur. Figure 6.9 shows a comparison between the two restoration algorithms for the same original as given in Fig. 6.8, at blurring angles sixty and seventy five degrees respectively. In conclusion, if the blurring angle is less than forty five degrees it is recommended that the general purpose algorithm given by Eqn. (5.18) be used, since it is more economical than the algorithm given in the present chapter while if the blurring angle exceeds forty five degrees, Eq. (6.20) should be used in order to achieve better restoration.
Figure 6.3: Original image.
Figure 6.4: Corrupted image by blur (angle = 45) and additive noise.
Figure 6.5: Restored image using Eqn. (5.18). $Q/Q^0 = 5.13$
Figure 6.6: Restored image using Eqn. (6.20). $Q/Q^0 = 7.41$
Figure 6.7: A Comparison between the special purpose algorithm, and the general purpose algorithm at high level of blur.
Figure 6.8: The restored images of Fig. 6.7.
1 = original, 2 = noisy,
3 = Results using Eq. (5.18),
4 = Results using Eq. (6.20),
$Q(3)/Q^0 = 1.02$, $Q(4)/Q^0 = 3.12$
Figure 6.9: Comparison between the two algorithms for the original given in Fig. 6.8 at two levels of blur.
1.2 = Results using Eq. (5.18).
3.4 = Results using Eq. (6.20).
Chapter VII
GIBBS FIELD MODELS RESTORATION ALGORITHMS
HARDWARE FEASIBILITY ON SIMD MACHINE ARCHITECTURE

7.1 Overview

Since the Cellular array processor was proposed in 1958 [47], many studies have been carried out on the use of two-dimensional mesh-connected arrays of simple processors for processing images.

The general principle behind the Single Instruction Stream, Multiple Data Stream (SIMD) arrays is that each elementary processor in the mesh, usually referred to as a processing element (PE), is assigned to a single pixel (or possibly to addressed multiple pixels) in the image to be processed. The processing elements can perform arithmetic or logic operations on their own pixels, or by the appropriate connection between the neighbouring processors carry out neighbourhood operations on small windows of the image. The standard window size in the Cellular Logic Image Processor (CLIP) family [48] is 3 x 3. Larger neighbourhoods and/or blurring window sizes are constructed by combination of 3 x 3 operations interspersed with shifting.

SIMD processors are characterized by broadcasting instructions to every PE in the array and every PE executes the same instruction at the same instant. Only one controller and one instruction store is required. In contrast, in the MIMD processors each PE exercises local autonomy to the extent of fetching data from a locally determined address and making connections to its own selections of neighbouring processing elements, and each PE is able to
execute its own program. SIMD/MIMD arrays permit parallel transfer of data between processing elements and their neighbours.

Hierarchically ordered neighbourhood systems can be implemented on a parallel SIMD architecture. The potential speeds up over an equivalent serial processor is equal to the number of array processors divided by the number of synchronous updates (clock cycles). A method of increasing the efficiency of the developed hardware is considered.

The parallel structure of the restoration algorithms is made possible since the estimate at each location in the image array depends only on the original record and on the state of pixels in the vicinity of the current pixel in the image array. The parallel structure requires small communication among processing elements. The Distributed Array Processor (DAP) [49] is designed for performing a parallel structure of the first order neighbourhood systems, an example of which is given in [50] where the image array is processed in two synchronous updates (clock cycles).

Section 7.2 describes the serial implementation of the restoration algorithms (to explain what parallelism is available and how it is exploited). The parallel implementation is presented in section 7.3. Comments and conclusion are given in section 7.4.

7.2 Serial Implementation

The restoration algorithms for any of the techniques used in this research, starts with an initial guess of \( x \) (the clean scene). The proper choice is the original records (quantized to the scene grey levels).

The maximum likelihood (ML) classifier is used to set the initial value of the parameter \( \beta \), for an univariate record, \( \beta = 0 \) is used as the appropriate initial value. Then \( \beta \) is incremented from iteration to iteration by a value which depends on the noise level (see section 4.4). For low level
of noise larger increments of $\beta$ speeds up the restoration while for high noise level small
increments of $\beta$ must be used.

Serial implementation is always done using a raster scan. It is helpful to change the scan
from cycle to cycle in order to reduce the directional effects which might otherwise occur.

7.3 Parallel Implementation

In this section, feasibility of the implementation of the restoration algorithms will be
discussed on SIMD machine architecture.

Among existing array processors that used successfully in image processing applications are:
Illiac IV [51], Distributed Array Processor (DAP) [49], Massively Parallel Processor (MPP) [52],
Adaptive Array Processor AAP [53], [54], and the Cellular Logic Image Processor (CLIP) family
[48].

The Adaptive Array Processor AAP2 consists of a square lattice of 256 x 256 processors,
each with its own 8K of RAM, and each processing element is connected to its eight neighbours.
Thus the one to one mapping of PE onto pixel and a second order neighbourhood graph to a
second order neighbourhood interaction in the MRF sense is quite clear and obvious.

The processing steps at each pixel location are identical, an attempt to process the image
array simultaneously would not work since the neighbouring pixels will not be in stable state to
calculate the required estimate of the center pixel. The solution to this problem is to hold the state
of the neighbouring pixels while calculating the state of the center one.

A block diagram of AAP2-LSI [54] is shown in Fig. 7.1. The processor consists of 256 x
256 PE array units, an interface unit, which is equipped with a data buffer memory (DM), an
instruction memory (IM), and a scalar processor (SP).
Figure 7.1: The ADAPTIVE ARRAY PROCESSOR AAP2
The AAP2 system is interconnected via the interface unit with a host processor. The host processor sends objective data and a set of microinstruction data to control all the processing elements as well as the array control unit. When the array processor has completed the processing then the results are returned to the host processor.

There are three types of instructions for AAP2-PE. The first is for data transfer operations between the processing elements and the array control unit. The second is for PE operations related to the expanded memory. The third is for operations in the PE array. Twenty bits are provided in the instruction for the external memory address. More information about AAP2 can be found in [54].

Geman and Geman in [1] addressed the parallel implementation of their algorithm. They proposed MIMD architecture using asynchronous updating scheme. In this environment each processor is being updated if and only if, the neighbouring pixels are not changing state at a given instant. The nature of the stochastic relaxation algorithm proposed by Geman and Geman [1] requires this type of machine hardware (the convergence properties of the Gibbs sampler and the annealing schedule necessitates asynchronous updating scheme). In comparison with serial implementation, the resulting speed is approximately equal to the number of processors. Unfortunately, the cost of implementing MIMD machine architecture is too expensive to be of any practical use.

In the analysis to be presented below, we will assume second and higher order neighbourhood system; however, first order neighbourhood system is found to be unrealistic for most practical purposes.

Synchronous updating scheme using SIMD machine architecture is used for the parallel implementation of the proposed restoration algorithms. It is mandatory in this architecture that pixel processors that change simultaneously are not neighbours. Fig. 7.2(a) shows the connection
of the processing element at location \((i,j)\), while Fig. 7.2(b) shows the distribution of the clock for parallel processing of the pixels.

For second order neighbourhood system, with blur extent does not exceed 3x3 window size, four clock cycles are required and all pixels in the image array can be visited in just four synchronous updates. It is possible to prevent broadcast commands reaching certain processors using a logical mask. The mask for the first cycle is shown in Fig. 7.3(a) and that for the next update is a right shift of the current mask and so on. Under this process the maximum speed up over equivalent serial computer is 0.25 times the number of processors in the array or 16,384 for 256x256 array processors.

The number of processing elements required to implement the proposed restoration algorithms depend on the specific hardware structure adopted. The fastest processing is achieved if a processing element is associated with each pixel location. However, an economy of scale can be handled in two different ways. The \textit{FIRST} is achieved if one is allowed to trade speed for a reduction in the number of processors. In this scheme illustrated in Fig. 7.3(b) PE will be located only at pixel location labeled '1'. At the first clock cycle the pixel values at location '1' will be updated, followed by global shift of pixels in the image array to the left by one column. The pixel update algorithm followed by a shift to the left is again initiated. The procedure is performed four times to complete the processing. Since all computation is done 'in place', the scheme essentially incorporates acceleration to speed up convergence. The \textit{SECOND} point of importance is how to handle larger images using array processor of smaller size efficiently (with minimum overhead). In a 512x512 image (TV picture) and AAP2 system with 256x256 processing elements it is possible to update spatially different parts of the image simultaneously, making use of the full power of the given array (the processors are allowed to run at their individual rates). Such a scheme would require four pixels 'sharing' each processor and an addressed logical mask is also
Figure 7.2: Parallel structure for second order neighbourhood system. (a) Cell Processor. (b) Clock Signals Distribution
Figure 7.3: Parallel structure for second order neighbourhood system...
(a) Logical Mask, ...
(b) Modified Structure for pixel Processing
required. Fig. 7.4 and Fig. 7.5 shows four synchronous updates (clock cycles) of the array processor where each processor would be in active use and the potential speed up over a serial processor would be approximately equal to the number of processors (some time would be lost in addressing overheads [50]).

Although many researchers are currently using second order neighbourhood systems with satisfactory results, it generally would seem desirable to adopt a larger neighbourhood to allow for curvature effects in scenes, through coefficients which depend on relative positions of neighbours.

Cross and Jain [42] used the 'coding' techniques to estimate model parameters associated with ordered neighbourhoods. However, they used codes of size nine in implementing third order neighbourhood systems. The nine codes result in slower and unnecessarily more complicated hardware than the present proposed hardware architecture. Third order neighbourhood systems has been successfully implemented using only six synchronous updates (clock cycles). Fig. 7.7(a) shows the PE connections to the neighbouring processing elements. While Fig. 7.8(a) shows the distributions of the clock signals for parallel processing of the image array. Implementation of fourth and fifth order neighbourhood systems are shown in figures 7.7(b) and 7.8(b).
Figure 7.4: Processing Big image using array processor efficiently. (a) First clock cycle, (b) Second clock cycle
Figure 7.5:  Processing Big image .... (a) Third clock cycle, ... (b) forth clock cycle
Processing Element connections in Third, Fourth, and fifth order neighbourhood systems. (a) Third order neighbourhood system, ... (b) Fourth and Fifth order neighbourhood systems.
Figure 7.7: Clock Signal Distribution. (a) Third order neighbourhood system, ... (b) forth and fifth order neighbourhood systems
7.4 Comments and Conclusions

1. A parallel implementation of the restoration algorithm is always possible whatever the order of the neighbourhood system and/or the extent of the blur.

2. The number of synchronous updates depends upon the blur extent and/or the neighbourhood system order, i.e. the degree of parallelism of the hardware implementation depends on these factors.

3. The proposed hardware structure of the third order neighbourhood system is much more efficient than that proposed by Cross and Jain [42].
Chapter VIII

CONCLUSION

8.1 Summary

1. The problem of restoration of degraded images using Gibbs Field scene models was considered.

2. An algorithm for restoration of binary images was derived and this binary restoration algorithm was used for restoration of multilevel images. A comparison with existing similar techniques was conducted.

3. The degradation model was made more general by introducing parameters for blur, additive noise, multiplicative noise and nonlinearity effects.

4. One step approach to image restoration using deterministic relaxation type of algorithms was presented using a particular class of scene models.

5. Quadratic pairwise interactions MRF potential function was proposed. And this particular scene model was used in the derivation of restoration algorithms with different sources of degradations, including additive and/or multiplicative noise(s) with blur and nonlinearity.

6. An algorithm for restoration of images corrupted by blur and additive noise was proposed. The algorithm has been shown to be very efficient even when the blur is excessive.

7. A parallel structure for the restoration algorithms (using SIMD machine architecture) was proposed.
8.2 Discussion

This section is divided into two parts. The first part cites the advantages of the proposed restoration algorithms and the second part details the limitations of the algorithms and the models.

8.2.1 Advantages

1. The degradation model is made more general and it allows for different types of noise sources.

2. The use of Partial Binary Restoration (PBR) in the application of binary algorithms in multilevel restorations is much more efficient in comparison with existing similar techniques.

3. The computational cost was tremendously reduced in the proposed restoration algorithms as a direct result of using the pairwise interaction quadratic potential function in a gradient based algorithm.

4. The restoration algorithms are very effective even at very low signal to noise ratios.

5. The algorithms performance is quite insensitive to inaccurate estimation of model and/or noise parameters.

6. The linear model approximation allows for efficient implementation of the multiplicative noise and the compound noise cases.

7. The special purpose algorithm for restoration of images corrupted by excessive blur and additive noise is effective.

8. Algorithms derived on the basis of realistic models of the scene and the degradation phenomena have been shown to perform satisfactorily on a variety of images.

9. Efficient parallel structure for third order neighbourhood systems has been proposed.
10. The proposed algorithms provide the user with the flexibility of controlling the resolution needed in the segmented image.

8.2.2 Limitations

1. Exact boundary shapes using the assumed image model cannot be enforced.
2. The parallel implementation of the special purpose algorithm for restoration of images corrupted by blur and additive noise using SIMD machine architecture when blur extent (window size) is large might not be very effective since the expanded window would require large number of synchronous updates affecting the resulting speed.
3. The level of the multiplicative noise should not be too high otherwise the linear model would not be a valid approximation.

8.3 SUGGESTIONS FOR FUTURE WORK

1. Development of a neural network for implementation of the restoration algorithms.
2. Simultaneous processing of images in non-overlapping segments.
3. Extending the restoration algorithm to noisy textured images.
Appendix A

MATHEMATICAL DERIVATION OF $A_1$, $A_2$, AND $A_3$ IN 5.3.1

$$y_{ij} = C\left( K_0(\phi) + \dot{K}_1(\phi) x_{ij} \right)^t u_{ij}$$

Let:

$$C\left( K_0(\phi) + K_1(\phi) x_{ij} \right)^t = g(x_{ij})$$

Thus:

$$y_{ij} = g(x_{ij}) u_{ij}$$

The linear model is given by:

$$\hat{y}_{ij} = A_1 u_{ij} + A_2 g(x_{ij}) + A_3$$

Where $A_1, A_2, \text{and } A_3$ are non-random and to be determined. For an unbiased estimate, the following is required:

$$E\{Y_{ij}\} = E\{\hat{Y}_{ij}\}$$

Since $U$ and $X$ are independent stochastic processes then,

$$E\{U_{ij} g(X)\} = E\{U_{ij}\} E\{g(X_{ij})\}$$

$$= \bar{u}_{ij} \bar{g}(x_{ij})$$

From Eqns. (A.3),(A.4) and (A.5) the result is:

$$\bar{u}_{ij} \bar{g}(x_{ij}) = A_1 \bar{u}_{ij} + A_2 \bar{g}(x_{ij}) + A_3$$

i.e.

$$A_3 = \bar{u}_{ij} \bar{g}(x_{ij}) - A_1 \bar{u}_{ij} - A_2 \bar{g}(x_{ij})$$

Substituting (A.7) into (A.4) results in:
\[ y_{ij} = A_1(u_{ij} - \bar{u}_{ij}) + A_2(g(x_{ij}) - \bar{g}(x_{ij})) + \bar{u}_{ij} \bar{g}(x_{ij}) \] 

(A.8)

\( A_1 \) and \( A_2 \) are determined as follows:

Let

\[ f = E \{ Y_{ij} - \hat{Y}_{ij} \}^2 \]

\[ f = E \{ A_1(u_{ij} - \bar{u}_{ij}) + A_2(g(x_{ij}) - \bar{g}(x_{ij})) - (u_{ij} g(x_{ij}) - \bar{u}_{ij} \bar{g}(x_{ij})) \}^2 \]

\[ = E \{ A_1^2(u_{ij} - \bar{u}_{ij}) + A_2^2(g(x_{ij}) - \bar{g}(x_{ij})) + 2A_1A_2(u_{ij} - \bar{u}_{ij})(g(x_{ij}) - \bar{g}(x_{ij})) \} \]

\[ - 2A_1(u_{ij} - \bar{u}_{ij})(u_{ij} g(x_{ij}) - \bar{u}_{ij} \bar{g}(x_{ij})) - 2A_2(g(x_{ij}) - \bar{g}(x_{ij}))(u_{ij} g(x_{ij}) - \bar{u}_{ij} \bar{g}(x_{ij})) \]

\[ \frac{\partial f}{\partial A_1} = 0 \rightarrow \]

\[ A_1 \text{ var } \{ U_{ij} \} = \bar{g}(x_{ij}) \text{ var } \{ U_{ij} \} \]

Thus:

\[ A_1 = \bar{g}(x_{ij}) \] 

(A.10)

Similarly by equating \( \frac{\partial f}{\partial A_2} \) to zero then:

\[ A_2 = \bar{u}_{ij} \] 

(A.11)

Substituting (A.10) and (A.11) into (A.8) yields:

\[ \hat{y}_{ij} = \bar{u}_{ij} g(x_{ij}) + \bar{g}(x_{ij})(u_{ij} - \bar{u}_{ij}) \] 

(A.12)

for all \((i,j) \in L_t\)

It is clear that the same result could be obtained by expanding \( y_{ij} = u_{ij} g(x_{ij}) \)

around the mean \((\bar{u}_{ij}, \bar{g}(x_{ij}))\) using Taylor theorem.
Appendix B

DERIVATION OF EQNS. (5.54) AND (5.55)

Given:

\[ V_{ij} = S_{ij} U_{ij} + W_{ij} \]  \hspace{1cm} (B.1)

Find the probability density function (PDF) of \( V_{ij} \) given that \( U_{ij} \) and \( W_{ij} \) are independent and Gaussian.

Solution:

Let

\[ \tilde{U}_{ij} = S_{ij} U_{ij} \]  \hspace{1cm} (B.2)

The probability density function of \( \tilde{U}_{ij} \) can be found in terms of the probability density function of \( U_{ij} \).

Let:

\[ f_{\tilde{u}_{ij}} \text{ be the PDF of } \tilde{U}_{ij} \]

and

\[ f_{u_{ij}} \text{ be the PDF of } U_{ij} \]

It can be shown that:

\[ f_{\tilde{u}_{ij}} = \frac{1}{S_{ij}} f_{u_{ij}} \left( \frac{u_{ij}}{S_{ij}} \right) \]  \hspace{1cm} (B.3)
\[
N(S_{ij}, \bar{u}_{ij}, S_{ij}^2, \sigma^2_u) \quad (B.4)
\]

Let:
\[
\sigma_{\bar{u}_{ij}} = S_{ij} \sigma_{u_{ij}} \quad (B.5)
\]

and
\[
\bar{u}_{ij} = S_{ij} \bar{u}_{ij} \quad (B.6)
\]

Thus:
\[
f_{\bar{u}_{ij}}(u_{ij}) = N(\bar{u}, \sigma^2_u) \quad (B.7)
\]

While:
\[
f_{w_{ij}}(w_{ij}) = N(\bar{w}, \sigma^2_w) \quad (B.8)
\]

\(\bar{U}\) and \(W\) are independent stochastic processes. Thus,
\[
f_v(v) = \int_{\mathbb{R}^2} f_{\bar{u}}(v-w) f_w(w) \, dw \quad (B.9)
\]

For clarity the suffix \(ij\) will be dropped throughout this derivation. Substituting (B.7) and (B.8) in (B.9) gives:
\[
f_v(v) = \frac{1}{2\pi\sigma_{\bar{u}} \sigma_w} \int_{\mathbb{R}^2} e^{-\frac{(v-w-\bar{u})^2}{2\sigma_{\bar{u}}^2}} e^{-\frac{(w-\bar{w})^2}{2\sigma_w^2}} \, dw
\]

\[
\Phi = \frac{1}{2\pi\sigma_{\bar{u}} \sigma_w} \int_{\mathbb{R}^2} e^{-\frac{(v-w-\bar{u})^2}{2\sigma_{\bar{u}}^2} + \frac{(w-\bar{w})^2}{2\sigma_w^2}} \, dw \quad (B.10)
\]

Let
\[
I = \left\{ \frac{(v-w-\bar{u})^2}{2\sigma_{\bar{u}}^2} + \frac{(w-\bar{w})^2}{2\sigma_w^2} \right\} \quad (B.11)
\]
Writing:

\[ a^2 = \sigma_u^2 + \sigma_w^2 \]

and

\[ b = \frac{(\sigma_u^2 \bar{w} + \sigma_w^2 (v - \bar{u}))}{\sqrt{\sigma_u^2 + \sigma_w^2}} \]

and

\[ c = \pm \sigma_u \sigma_w \frac{(v - \bar{u} - \bar{w})}{\sqrt{\sigma_u^2 + \sigma_w^2}} \]  \hspace{1cm} (B.12)

and using (B.12), (B.11) can be written using as:

\[ f = \left\{ \frac{(aw - b)^2 + c^2}{2\sigma_u^2 \sigma_w^2} \right\} \]

\[ = \frac{(aw - b)^2}{2\sigma_u^2 \sigma_w^2} + \frac{c^2}{\sigma_u^2 + \sigma_w^2} \]  \hspace{1cm} (B.13)

Where:

\[ c^2 = \frac{(v - \bar{u} - \bar{w})^2}{2 (\sigma_u^2 + \sigma_w^2)} \]

Upon substituting (B.12) into (B.10), then upon carrying out the necessary mathematical procedures yields:

\[ f_{v_{ij}}(v_i) = N(\bar{v}_{ij}, \sigma_{v_{ij}}^2) \]

Where:

\[ \bar{v}_{ij} = \bar{u}_{ij} + \bar{w}_{ij} = S_{ij} \bar{u}_{ij} + \bar{w}_{ij} \]  \hspace{1cm} (B.14)

\[ \sigma_v^2 = \sigma_u^2 + \sigma_w^2 = S_{ij} \sigma_u^2 + \sigma_w^2 \]  \hspace{1cm} (B.15)
Appendix C

DERIVATION OF EQN. (6.10)

\[
I = \frac{-1}{2\sigma^2} \left( \sum_{u=i-l}^{j+l} \sum_{v=j-l}^{j+l} \left( y_{uv} - \sum_{k=d}^{l} \sum_{l=d}^{l} H_\phi(k, l) F(x(k + u, l + v)) \right) \right)^2
\]  
(C.1)

Let:

\[
I_2 = \sum_{k=d}^{l} \sum_{l=d}^{l} H_\phi(k, l) F(x(k + u, l + v))
\]

Then:

\[
I_2 = f_m + \frac{f_d\sin^2\phi}{n_w} \sum_{k=d}^{l} \sum_{l=d}^{l} x(k + u, l + v) + \cos^2\phi f_d x_{uv}
\]

\[
I_2 = f_m + K_2(\phi) \sum_{k \in \zeta_u} x(k + u, l + v) + K_1(\phi) x_{uv}
\]  
(C.2)
Where:

\[ K_1(\phi) = f_d \cos^2 \phi \]

\[ K_2(\phi) = \frac{f_d \sin^2 \phi}{n_w} \]

Substituting (C.2), (C.3), and (C.4) into (C.1) gives:

\[
I = \frac{-1}{2\sigma^2} \sum_{u=-l_u}^{i+l_u} \sum_{v=-l_v}^{j+l_v} \left\{ \bar{y}_{uv} - K_2(\phi) \left[ \sum_{k=-l_k}^{l_k} \sum_{l=-l_l}^{l_l} x(k+u, l+v) - K_1(\phi) x_{uv} \right] \right\}^2
\]

Where:

\[ \bar{y}_{uv} = y_{uv} - f_m \]

Assuming a 3 x 3 window, then generalization to any window size will follow. Let:

\[
S(u, v) = \sum_{k' \in \Omega} x(k+u, l+v)
\]

\[
I = \frac{-1}{2\sigma^2} \sum_{u=-i-1}^{i+1} \sum_{v=-j-1}^{j+1} \left\{ \bar{y}_{uv} - K_2(\phi) S(u, v) - K_1(\phi) x(u, v) \right\}^2
\]

\[
I = \frac{-1}{2\sigma^2} \left\{ \bar{y}_{i-1,j-1} - K_2(\phi) S(i-1, j-1) - K_1(\phi) x(i-1, j-1) \right\}^2
\]

\[ + \left[ \bar{y}(i-1,j) - K_2(\phi) S(i-1, j) - K_1(\phi) x(i-1, j) \right]^2
\]

\[ + \left[ \bar{y}(i-1,j+1) - K_2(\phi) S(i-1, j+1) - K_1(\phi) x(i-1, j+1) \right]^2
\]

\[ + \left[ \bar{y}(i,j-1) - K_2(\phi) S(i, j-1) - K_1(\phi) x(i, j-1) \right]^2
\]

\[ + \left[ \bar{y}(i,j) - K_2(\phi) S(i, j) - K_1(\phi) x(i, j) \right]^2
\]
\begin{align*}
+ \left[ \bar{v}(i, j + 1) - K_2(\phi) \ S(i, j + 1) - K_1(\phi) \ x(i, j + 1) \right]^2 \\
+ \left[ \bar{v}(i + 1, j - 1) - K_2(\phi) \ s(i + 1, j - 1) - K_1(\phi) \ x(i + 1, j - 1) \right]^2 \\
+ \left[ \bar{v}(i + 1, j) - K_2(\phi) \ s(i + 1, j) - K_1(\phi) \ x(i + 1, j) \right]^2 \\
+ \left[ \bar{v}(i + 1, j + 1) - K_2(\phi) \ s(i + 1, j + 1) - K_1(\phi) \ x(i + 1, j + 1) \right]^2
\end{align*}

\text{(C.7)}

The procedure is to isolate \( x(i, j) \) from the remaining terms given in Eqn. (C.7) as follows:

\begin{align*}
S(i - 1, j - 1) &= x(i, j) + \sum_{kl \in \xi_{i,j}} x(k, l) \\
&= x(i, j) + \bar{S}(i - 1, j - 1) \tag{C.8}
\end{align*}

and the same is done to the other terms. The general term is defined as follows:

\( S(i-p,j-q) \) represents the sum of the scene values inside a window centered at \((i-p,j-q)\) excluding \( x(i-p,j-q) \) and \( x(i,j) \). Substituting \( \text{(C.8)} \) and the similar terms in Eq. \( \text{(C.7)} \) it can be shown that:

\begin{align*}
I &= \frac{-1}{2\sigma^2} \sum_{i', j' \in \xi} \left[ \bar{y}(i + l_i, j + l_j) - K_2(\phi) \ x(i, j) \right]^2 - \frac{1}{2\sigma^2} - \bar{y}(i, j) - K_1(\phi) \ x(i, j) \tag{C.9}
\end{align*}

Where:

\[ \bar{y}(i + l_i, j + l_j) = \bar{y}(i + l_i, j + l_j) - K_2(\phi) \ \bar{S}(i + l_i, j + l_j) - K_1(\phi) \ x(i + l_i, j + l_j) \tag{C.10} \]

Where:

\((l_i, l_j) \in \xi_{ij}\)

and

\[ \bar{y}(i, j) = \bar{y}(i, j) - K_2(\phi) \ \bar{S}(i, j) \tag{C.11} \]

Neglecting constant terms Eqn. \( \text{(C.8)} \) gives:
\[ I = \frac{1}{\sigma^2} \left\{ K_2(\phi) \sum_{l_i = -l_k}^{l_k} \tilde{y}(i + l_i, j + l_k) + K_1(\phi) \tilde{y}(i, j) \right\} x(i, j) \]

\[ - \frac{1}{2\sigma^2} \left( n \cdot K_2(\phi) + K_1(\phi) \right) x^2(i, j) \]

Let:

\[ K_y(\phi) = K_2(\phi) \sum_{l_i = -l_k}^{l_k} \tilde{y}(i + l_i, j + l_k) + K_1(\phi) \tilde{y}(i, j) \]

Thus Eqn. (C-13) can be written in the form:

\[ I = \frac{1}{\sigma^2} K_y(\phi) x(i, j) - \frac{1}{2\sigma^2} \left( n \cdot K_2(\phi) + K_1(\phi) \right) x^2(i, j) \]

\[ \tilde{S}(p, q) \] can be written in the general form as follows:

\[ \tilde{S}(i + l_i, j + l_k) = \left[ \sum_{k = -l_k}^{l_k} \sum_{l_i = -l_k}^{l_k} x(i + l_i + k, j + l_k) \right] - x(i, j) \]

\[ (l_i, l_k) \neq (0, 0) \]

While:

\[ \tilde{S}(i + j) = \left[ \sum_{k = -l_k}^{l_k} \sum_{l_i = -l_k}^{l_k} x(i + k, j + l) \right] - x(i, j) \]
REFERENCES


38. R. Kinderman and J. L. Snell, 'Markov Random Fields and Their Applications'.


40. H. Elliott et al., 'Modelling Segmentation of Noisy and Textured Images Using Gibbs
   Report ECE-UMASSE84-15*.

41. R. Cristi, 'Application of Markov Random Fields to Smoothing and Segmentation of

42. G. C. Cross and A. K. Jain, 'Markov Random Field Texture Models', *IEEE Transaction

43. J. Besag 'Efficiency of Pseudolikelihood Estimation for Simple Gaussian Fields'.
   *Biometrika*, 64, 616-618.


   Academic, 1982.

46. F. S. Cohen et al., 'Simple Parallel Hierarchical and Relaxation Algorithms for

   1744-1750, 1958.

48. T. J. Fountain et al., 'The CLIP7A Image Processor', *IEEE Transaction on Pattern
   Analysis Machine Intelligence* Vol. PAMI-10, NO. 3, May 1988

- 153 -


VITA AUCTORIS:

1979/84 Lab. Instructor at Kuwait University, ECE DEPT, Kuwait.
1975/1979 Graduate Assistant at Al-Azhar University, Cairo, Egypt.
1972/1974 B.Sc in Pure and Applied Mathematics with first class Hons. from Al-Azhar University, Cairo, Egypt.
1966/1971 B.A.Sc in Electrical Engineering from Cairo University, Cairo, Egypt.
1966 Matriculated from Al-Malek El-Kamel High School, Almansoura, Dakahlia Egypt.

PUBLICATIONS

1. "Restoration of Noisy Images Using Markov Random Fields With Gibbs Distributions"