Sub-band coding for visual communications

This item was submitted to Loughborough University's Institutional Repository by the/an author.

Additional Information:

• A Master’s Thesis. Submitted in partial fulfilment of the requirements for the award of Master of Philosophy at Loughborough University.

Metadata Record: https://dspace.lboro.ac.uk/2134/20926

Publisher: © Wai Yiu Ng

Rights: This work is made available according to the conditions of the Creative Commons Attribution-NonCommercial-NoDerivatives 2.5 Generic (CC BY-NC-ND 2.5) licence. Full details of this licence are available at: http://creativecommons.org/licenses/by-nc-nd/2.5/

Please cite the published version.
This item was submitted to Loughborough University as an MPhil thesis by the author and is made available in the Institutional Repository (https://dspace.lboro.ac.uk/) under the following Creative Commons Licence conditions.

![Creative Commons License](https://creativecommons.org/licenses/by-nc-nd/2.5/)

**Attribution-NonCommercial-NoDerivs 2.5**

You are free:

- to copy, distribute, display, and perform the work

Under the following conditions:

- **Attribution.** You must attribute the work in the manner specified by the author or licensor.

- **Noncommercial.** You may not use this work for commercial purposes.

- **No Derivative Works.** You may not alter, transform, or build upon this work.

- For any reuse or distribution, you must make clear to others the license terms of this work.

- Any of these conditions can be waived if you get permission from the copyright holder.

Your fair use and other rights are in no way affected by the above.

This is a human-readable summary of the [Legal Code](http://creativecommons.org/licenses/by-nc-nd/2.5/).
To God
I would like to dedicate this thesis to my grandmother who passed away in July, 1995.
SUBBAND CODING
FOR
VISUAL COMMUNICATIONS

by

Wai Yiu Ng

Supervisor

Professor H. Gharavi

A Masters Thesis submitted in partial fulfilment of the requirements for the award of the degree of Master of Philosophy of the Loughborough University

April, 1997

Department of Electronic and Electrical Engineering
Loughborough University
United Kingdom

© by Wai Yiu Ng, April 1997
Abstract

This thesis looks into the problems arising in video image transmission when transmitting over a wireless environment. The main problems faced by future visual communication systems are multi-resolution video transmission, error resilience and compatibility with existing coding standards. Two main areas were investigated in this thesis. The first area focused on subband decompostion, where a comprehensive comparison of various subband filters were carried out. The other area centred around subband coding and modelling, where various subband models were implemented.

The initial work on this project started with the study of still and video image compression algorithms. This led to an investigation of subband coding as an alternative to standard image coding techniques. Various subband filters were investigated and implemented as part of a subband still image system. The results show that the performance of QMF filters is the best of all the filters tested. The other filters were either inefficient or too sensitive to the truncation errors.

The next stage of the project involved using standard compression algorithms on subband coding. In order to maintain compatibility to the existing video coding standards, motion compensation, block transformation, quantisation and entropy coding were adopted from the ITU-T H.263 standard. This work concentrated on the area of half pixel motion compensation and 3D VLC entropy coding algorithms. These techniques were used to enhance the previous developed subband models. The Hybrid DPCM/Subband Model and Subband Multi-Resolution Transmission System were implemented using these new techniques. Finally, the loop filter technique was also applied to improve the picture quality in a Subband Mutli-Resolution Transmission System.
## Contents

Abstract ..................................................................................................................... i
Table of Contents ................................................................................................... ii
List of Figures and Tables ...................................................................................... v
Glossary Acronyms and Abbreviations .............................................................. ix
Acknowledgments ................................................................................................. xi

### Chapter 1 Introduction
1.1 General ........................................................................................................... 1
1.2 Research Objectives .................................................................................... 2
1.3 Scope of Thesis ............................................................................................ 3

### Chapter 2 Review Of Still Image Coding Systems
2.1 Introduction ................................................................................................... 5
2.2 Predictive Coding System .......................................................................... 6
2.3 Vector Quantisation System ....................................................................... 9
2.4 Transform Coding System ......................................................................... 11
   2.4.1 Orthonormal Transforms ........................................................................ 13
   2.4.2 Quantisation of Transform Coefficients ............................................... 16
   2.4.3 Entropy Coding of Transform Coefficients ........................................... 20
2.5 International Standard Still Image Coding Systems .................................. 22
   2.5.1 JPEG Still Image Coding System ......................................................... 22
   2.5.2 Other Developments in the Standard Bodies ....................................... 27
2.6 Subband Coding System ............................................................................ 28
   2.6.1 Quantisation of Subbands .................................................................. 31
   2.6.2 Run Length of Subbands .................................................................... 34

### Chapter 3 Subband Still Image Coding System
3.1 Introduction .................................................................................................. 37
3.2 Subband Filtering Theory .......................................................................... 38
   3.2.1 Quadrature Mirror Filtering ................................................................ 44
   3.2.2 Lattice Structure Filtering .................................................................. 45
3.2.3 Infinite Impulse Response Filtering .................................. 48
3.3 Subband Filtering Performance Evaluation ............................ 50
3.4 Proposed Subband Still Image Coding System ....................... 60
3.5 Proposed Still Image Coding Scheme Simulation Results .......... 63

Chapter 4 Review Of Video Image Coding Systems
4.1 Introduction .................................................................................67
4.2 Interframe Predictive Coding System ....................................... 68
4.2.1 Motion Estimation Technique ........................................... 71
4.3 Hybrid DPCM/DCT Video Image Coding System ..................... 75
4.4 International Standard Video Image Coding Systems ............... 76
4.4.1 ITU-T H.263 Video Image Coding Algorithms .................... 76
4.4.2 Other Developments in the Standard Bodies ....................... 81
4.5 Hybrid DPCM/Subband Video Image Coding System ............... 83
4.5.1 Subband Model I System ................................................... 83
4.5.2 Subband Model II System .................................................. 85

Chapter 5 Proposed Subband Video Coding System
5.1 Introduction .................................................................................86
5.2 Proposed Subband Video Coding System .................................. 87
5.3 Proposed Video Coding Scheme Simulation Results ............... 89

Chapter 6 Subband Multi-Resolution Transmission System
6.1 Introduction ................................................................................97
6.2 Multi-Resolution Transmission System .................................... 98
6.2.1 Subband Multi-Resolution Transmission System ............... 99
6.3 Subband Multi-Resolution Scheme Simulation Results .......... 104
6.4 Subband Scheme with Loop Filter ........................................... 111
6.5 Subband Scheme with Loop Filter Simulation Results .......... 113
Chapter 7 Conclusion And Further Work

7.1 Conclusion ....................................................................................117

7.2 Further Work and Recommendations ..........................................121

References ..........................................................................................123

Appendices
Appendix A MPEG-2 Specification ....................................................130
Appendix B Subband Filtering Coefficients .......................................131
# List of Figures and Tables

<table>
<thead>
<tr>
<th>Figure</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.2.1</td>
<td>Predictive Coding System</td>
</tr>
<tr>
<td>2.2.2</td>
<td>An example of Predictive Coding</td>
</tr>
<tr>
<td>2.2.3</td>
<td>DPCM Coding System</td>
</tr>
<tr>
<td>2.3.1</td>
<td>Vector Quantisation Coding System</td>
</tr>
<tr>
<td>2.4.1</td>
<td>Transform Coding System</td>
</tr>
<tr>
<td>2.4.2</td>
<td>Transform Coefficients Distribution</td>
</tr>
<tr>
<td>2.4.3</td>
<td>Uniform (midtread) Quantiser</td>
</tr>
<tr>
<td>2.4.4</td>
<td>An example of Zonal Mask</td>
</tr>
<tr>
<td>2.4.5</td>
<td>An example of Quantisation Matrix</td>
</tr>
<tr>
<td>2.4.6</td>
<td>An example of Run Length Coding</td>
</tr>
<tr>
<td>2.5.1</td>
<td>Sequential DCT Coding System</td>
</tr>
<tr>
<td>2.5.2</td>
<td>Procedure for DCT Scanning</td>
</tr>
<tr>
<td>2.5.3</td>
<td>An example of Neighbourhood Prediction</td>
</tr>
<tr>
<td>2.6.1</td>
<td>Subband Coding System</td>
</tr>
<tr>
<td>2.6.2</td>
<td>4 Bands Decomposition Structure</td>
</tr>
<tr>
<td>2.6.3</td>
<td>An example of 4 Bands Decomposition Output</td>
</tr>
<tr>
<td>2.6.4</td>
<td>Octave 7 Bands Decomposition Structure</td>
</tr>
<tr>
<td>2.6.5</td>
<td>DPCM of All Frequency Bands</td>
</tr>
<tr>
<td>2.6.6</td>
<td>DPCM of the First Band</td>
</tr>
<tr>
<td>2.6.7</td>
<td>Dead Zone Quantisation</td>
</tr>
<tr>
<td>2.6.8</td>
<td>Procedure for Block by Block Scanning</td>
</tr>
<tr>
<td>3.2.1</td>
<td>Subband Filter Bank</td>
</tr>
<tr>
<td>3.2.2</td>
<td>Decimation by a factor of 2</td>
</tr>
<tr>
<td>3.2.3</td>
<td>Interpolation by a factor of 2</td>
</tr>
<tr>
<td>3.2.4</td>
<td>Quadrature Mirror Filter</td>
</tr>
<tr>
<td>3.2.5</td>
<td>PRQMF Lattice Structure Filter Coding System</td>
</tr>
<tr>
<td>3.2.6</td>
<td>PRLPF Lattice Structure Filter Coding System</td>
</tr>
<tr>
<td>3.2.7</td>
<td>IIR Filter Coding System</td>
</tr>
<tr>
<td>Figure</td>
<td>Description</td>
</tr>
<tr>
<td>--------</td>
<td>-------------</td>
</tr>
<tr>
<td>3.2.7</td>
<td>IIR Filter Structure</td>
</tr>
<tr>
<td>3.2.8</td>
<td>FIR Filter Structure</td>
</tr>
<tr>
<td>3.3.1</td>
<td>Subband Decomposition and Reconstruction</td>
</tr>
<tr>
<td>3.3.2</td>
<td>Test Sample Images</td>
</tr>
<tr>
<td>3.3.3</td>
<td>QMF 8A Filter Simulation Results</td>
</tr>
<tr>
<td>3.3.4</td>
<td>QMF 16A Filter Simulation Results</td>
</tr>
<tr>
<td>3.3.5</td>
<td>PRQMF Filter Simulation Results</td>
</tr>
<tr>
<td>3.3.6</td>
<td>PRLPF Filter Simulation Results</td>
</tr>
<tr>
<td>3.3.7</td>
<td>IIR Filter Simulation Results</td>
</tr>
<tr>
<td>3.4.1</td>
<td>Proposed Subband Still Image Coding System</td>
</tr>
<tr>
<td>3.5.1</td>
<td>Subband Still Image Coding Scheme Bits Comparison</td>
</tr>
<tr>
<td>3.5.2</td>
<td>Subband Still Image Coding Scheme PSNR_Y Comparison</td>
</tr>
<tr>
<td>4.2.1</td>
<td>Interframe Prediction Coding System</td>
</tr>
<tr>
<td>4.2.2</td>
<td>Interframe DPCM Predictor Error Image</td>
</tr>
<tr>
<td>4.2.3</td>
<td>Block Matching Method</td>
</tr>
<tr>
<td>4.2.4</td>
<td>Half Pixel Prediction by Bilinear Interpolation</td>
</tr>
<tr>
<td>4.3.1</td>
<td>Basic Hybrid DPCM/DCT Video Image Coding System</td>
</tr>
<tr>
<td>4.4.1</td>
<td>ITU-T H.263 Video Encoder</td>
</tr>
<tr>
<td>4.4.2</td>
<td>MacroBlock (MB) Structure</td>
</tr>
<tr>
<td>4.4.3</td>
<td>PB Frame Arrangement</td>
</tr>
<tr>
<td>4.5.1</td>
<td>Subband Model I System</td>
</tr>
<tr>
<td>4.5.2</td>
<td>Subband Model II System</td>
</tr>
<tr>
<td>5.2.1</td>
<td>Proposed Subband Video Image Coding System</td>
</tr>
<tr>
<td>5.3.1</td>
<td>Proposed Subband Scheme for Carphone Comparison</td>
</tr>
<tr>
<td>5.3.2</td>
<td>Proposed Subband Scheme for Salesman Comparison</td>
</tr>
<tr>
<td>5.3.3</td>
<td>Proposed Subband Scheme for Suzie Comparison</td>
</tr>
<tr>
<td>5.3.4</td>
<td>Subband Scheme Versus ITU-T H.263 for Carphone Comparison</td>
</tr>
<tr>
<td>5.3.5</td>
<td>Subband Scheme Versus ITU-T H.263 for Salesman Comparison</td>
</tr>
<tr>
<td>5.3.6</td>
<td>Subband Scheme Versus ITU-T H.263 for Suzie Comparison</td>
</tr>
<tr>
<td>6.2.1</td>
<td>Laplacian Pyramid Coding System</td>
</tr>
<tr>
<td>Figure</td>
<td>Description</td>
</tr>
<tr>
<td>--------</td>
<td>-------------</td>
</tr>
<tr>
<td>6.2.2</td>
<td>Proposed Subband Multi-Resolution Model</td>
</tr>
<tr>
<td>6.2.3</td>
<td>Five Layer Subband Multi-Resolution Transmission System</td>
</tr>
<tr>
<td>6.3.1</td>
<td>Proposed Multi-Resolution Scheme for Salesman Comparison</td>
</tr>
<tr>
<td>6.3.2</td>
<td>Reconstructed Salesman Frame 57 Comparison</td>
</tr>
<tr>
<td>6.3.3</td>
<td>Enlargement Reconstructed Salesman Frame 57</td>
</tr>
<tr>
<td>6.3.4</td>
<td>Reconstructed Salesman Frame 87 Comparison</td>
</tr>
<tr>
<td>6.3.5</td>
<td>Enlargement Reconstructed Salesman Frame 87</td>
</tr>
<tr>
<td>6.4.1</td>
<td>Subband Multi-Resolution Scheme with Loop Filter</td>
</tr>
<tr>
<td>6.5.1</td>
<td>Carphone Comparison with Loop Filter</td>
</tr>
<tr>
<td>6.5.2</td>
<td>Salesman Comparison with Loop Filter</td>
</tr>
<tr>
<td>6.5.3</td>
<td>Suzie Comparison with Loop Filter</td>
</tr>
<tr>
<td>B.1.1</td>
<td>QMF 8A Filter Response</td>
</tr>
<tr>
<td>B.2.1</td>
<td>QMF 16A Filter Response</td>
</tr>
<tr>
<td>B.3.1</td>
<td>PRQMF Lattice 1 Filter Response</td>
</tr>
<tr>
<td>B.3.2</td>
<td>PRQMF Lattice 2 Filter Response</td>
</tr>
<tr>
<td>B.3.3</td>
<td>PRQMF Lattice 3 Filter Response</td>
</tr>
<tr>
<td>B.4.1</td>
<td>PRLPF Lattice 1 Filter Response</td>
</tr>
<tr>
<td>B.4.2</td>
<td>PRLPF Lattice 2 Filter Response</td>
</tr>
<tr>
<td>B.4.3</td>
<td>PRLPF Lattice 3 Filter Response</td>
</tr>
<tr>
<td>B.5.1</td>
<td>IIR Filter Response</td>
</tr>
<tr>
<td>Table</td>
<td>Page</td>
</tr>
<tr>
<td>-------</td>
<td>------</td>
</tr>
<tr>
<td>2.5.1 Table for Loseless Mode Prediction</td>
<td>25</td>
</tr>
<tr>
<td>2.6.1 An example of B codes</td>
<td>35</td>
</tr>
<tr>
<td>4.4.1 ITU-T H.263 Picture Formats</td>
<td>77</td>
</tr>
<tr>
<td>4.4.2 Group of Block and MacroBlock Arrangement</td>
<td>78</td>
</tr>
<tr>
<td>5.3.1 ITU-T H.263 Simulation Results</td>
<td>89</td>
</tr>
<tr>
<td>A.1 MPEG-2 Profile Coding Features</td>
<td>130</td>
</tr>
<tr>
<td>A.2 MPEG-2 Upper Bounds for Level Parameter</td>
<td>130</td>
</tr>
<tr>
<td>B.1.1 QMF 8A Filter Coefficients</td>
<td>131</td>
</tr>
<tr>
<td>B.2.1 QMF 16A Filter Coefficients</td>
<td>132</td>
</tr>
<tr>
<td>B.3.1 PRQMF Filter Coefficients</td>
<td>133</td>
</tr>
<tr>
<td>B.4.1 PRLPF Filter Coefficients</td>
<td>136</td>
</tr>
<tr>
<td>B.5.1 IIR Filter Coefficients</td>
<td>139</td>
</tr>
</tbody>
</table>
## Glossary Acronyms and Abbreviations

<table>
<thead>
<tr>
<th>Acronym</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>ATM</td>
<td>Asynchronous Transfer Mode</td>
</tr>
<tr>
<td>B-ISDN</td>
<td>Broadband Integrated Services Digital Networks</td>
</tr>
<tr>
<td>CATV</td>
<td>Cable Television</td>
</tr>
<tr>
<td>CCIR</td>
<td>International Radio Consultative Committee</td>
</tr>
<tr>
<td>CCITT</td>
<td>International Telegraph and Telecommunications Consultative Committee (see ITU)</td>
</tr>
<tr>
<td>CD-ROM</td>
<td>Compact Disc Read-Only Memory</td>
</tr>
<tr>
<td>CIF</td>
<td>Common Intermediate Format</td>
</tr>
<tr>
<td>CODEC</td>
<td>Coder-decoder</td>
</tr>
<tr>
<td>DAT</td>
<td>Digital Audio Tape</td>
</tr>
<tr>
<td>DCT</td>
<td>Discrete Cosine Transform</td>
</tr>
<tr>
<td>DPCM</td>
<td>Differential Pulse Code Modulation</td>
</tr>
<tr>
<td>EOB</td>
<td>End Of Block</td>
</tr>
<tr>
<td>FM</td>
<td>Frame Memory</td>
</tr>
<tr>
<td>FSBA</td>
<td>First Stage Bit Accuracy</td>
</tr>
<tr>
<td>HDTV</td>
<td>High Definition Television</td>
</tr>
<tr>
<td>IEC</td>
<td>International Electrotechnical Commission</td>
</tr>
<tr>
<td>IIR</td>
<td>Infinite Impulse Response</td>
</tr>
<tr>
<td>ISDN</td>
<td>Integrated Services Digital Networks</td>
</tr>
<tr>
<td>ISO</td>
<td>International Standard Organisation</td>
</tr>
<tr>
<td>ITU</td>
<td>International Telecommunication Union</td>
</tr>
<tr>
<td>ITU-T</td>
<td>International Telecommunication Union Telecommunication Standard Sector</td>
</tr>
<tr>
<td>KLT</td>
<td>Karhunen-Loeve Transform</td>
</tr>
<tr>
<td>LF</td>
<td>Loop Filter</td>
</tr>
<tr>
<td>LPF</td>
<td>Low Pass Filter</td>
</tr>
<tr>
<td>MC</td>
<td>Motion Compensation</td>
</tr>
<tr>
<td>ME</td>
<td>Motion Estimation</td>
</tr>
<tr>
<td>Acronym</td>
<td>Description</td>
</tr>
<tr>
<td>----------</td>
<td>--------------------------------------------------</td>
</tr>
<tr>
<td>MPEG</td>
<td>Motion Picture Expert Group</td>
</tr>
<tr>
<td>NTSC</td>
<td>National Television System Committee</td>
</tr>
<tr>
<td>PAL</td>
<td>Phase Alternation Line</td>
</tr>
<tr>
<td>PCM</td>
<td>Pulse Code Modulation</td>
</tr>
<tr>
<td>PCN</td>
<td>Personal Communication Network</td>
</tr>
<tr>
<td>PDC</td>
<td>Pel Difference Classification</td>
</tr>
<tr>
<td>PRLPF</td>
<td>Perfectly Reconstruction Linear Phase Filter</td>
</tr>
<tr>
<td>PRQMF</td>
<td>Perfectly Reconstruction Quadrature Mirror Filter</td>
</tr>
<tr>
<td>QCIF</td>
<td>Quarter Common Intermediate Format</td>
</tr>
<tr>
<td>QM</td>
<td>Quantisation Matrix</td>
</tr>
<tr>
<td>QMF</td>
<td>Quadrature Mirror Filter</td>
</tr>
<tr>
<td>QP</td>
<td>Quantisation Parameter</td>
</tr>
<tr>
<td>SAD</td>
<td>Sum Absolute Difference</td>
</tr>
<tr>
<td>SG</td>
<td>Study Group</td>
</tr>
<tr>
<td>SQCIF</td>
<td>Sub-Quarter Common Intermediate Format</td>
</tr>
<tr>
<td>SSBA</td>
<td>Second Stage Bit Accuracy</td>
</tr>
<tr>
<td>SSD</td>
<td>Sum Squared Difference</td>
</tr>
<tr>
<td>VLC</td>
<td>Variable Length Code or Variable Length Coding</td>
</tr>
</tbody>
</table>
Acknowledgments

I am greatly indebted to my parents, brothers and sisters for their support and constant encouragement, which made this research possible.

I wish to express my sincere thanks to my supervisor Professor H. Gharavi for his invaluable guidance, patience, devotion, constructive criticisms and practical advice throughout the research period.

Special thanks is also extended to R. Maniccasagar, S. C. Lim, V. Thomson, P. Connelly, L. S. Tan, W. K. Lo, K. T. Ho, H. W. Au Yeung, W. K. Chung (Bill) and all my friends for their help and support. Without the moral encouragement and trust, especially during the difficult times of my study, this work would not have been possible. I will cherish these greatly.

I would also like to express my gratitude to J. Steveson for her suggestions and advice on the art of writing reports.

Last but not least, I wish to thank Dr. M. J. J. Holt, Dr. W. K. Chung (Alexis), Dr. W. T. Chung, Dr. K. H. Chong, Dr. C. F. Fung, Dr. Y. K. Liu, Dr. R. Naimimohasses, Dr. H. Sari and Dr. B. C. Hou for all their helpful advice and discussion.

Finally, I also wish to thank my colleagues in Visual Communications Research Group, S. Dart, C. I. Richards, H. Y. Fan and Y. L. Song for all their help and encouragement.
Chapter 1

Introduction

1.1 General

In the past two decades, digital video technology has gone through tremendous improvement in both compression and transmission. With the new development of digital communications technology, visual communication based services are expected to be supported by cable operators and telephone companies [1]. New digital transmission channels such as Broadband Integrated Services Digital Networks (B-ISDN) and Asynchronous Transfer Mode (ATM) networks can provide a large channel capacity to subscribers. At the same time, the demand for the high quality video services such as High Definition Television (HDTV) for broadcasting and video conferencing is rapidly increasing. Therefore, the development of a single video compression technique to deal with various grades of video is becoming essential.

For instance, the future services are expected to support video applications such as entertainment, personal communications, multi-point video conferencing and distance learning. In order to support all these services, the new digital visual systems should cover a wide range of video quality from low resolutions to high resolutions which require different bit rates. A single compressed digital video signal is also expected to be transmitted via a variety of media ranging from copper wire, coaxial cable, optical fibre and wireless radio channels. As a result of this, a bottleneck situation will occur between the transfer of data from high bandwidth channels to low bandwidth channels such as optical fibre to copper wire. As video communications continue to advance, the integration and distribution of various video services over narrow channels will be the new challenge for future video telecommunications techniques. Therefore, it is essential to consider multi-layer and multi-resolution transmission systems to support future demand.
Another expanding field in communications is wireless Personal Communication Networks (PCN). This is due to the advances in digital communications, portable computers and semiconductor techniques. Although there is not yet any international standard for PCN, visual communication services such as wireless videophone and remote surveillance are expected to grow rapidly. Currently, a major challenge for portable and mobile video transmission research is dealing with multi-path fading interference of sensitive signals. This is because existing transmission channels are narrow band and the video signal is highly compressed, making the signal extremely sensitive to channel errors. With emergence of wireless visual communications services in the future, an effective robust video coding system is essential [2].

1.2 Research Objectives

Visual communication coding schemes in the future will need to deal with issues such as, multi-resolution, wireless transmission and compatibility. Subband coding techniques can provide a unique flexibility to achieve the above expectations [3]. The main objective of this research is to investigate various subband coding techniques and to develop an efficient video coding scheme.

This research work is concerned with two main aspects. The first aspect is subband filtering, where a comprehensive comparison of various subband filters will be carried out. The second part is concerned with subband coding and modelling aspects for image and video compression.

The first stage of this research is to look at the different still image and video image coding techniques. This is followed by investigation of new video image coding to find a flexible coding technique. Finally, video coding for multi-resolution transmission systems will be implemented.
1.3 Scope of Thesis

This thesis deals with the development of subband coding for visual communications, which includes issues regarding multi-resolution transmission and compatibility. These topics are arranged as follows.

Chapter 2 provides an overview of the basic still image compression techniques. A number of video coding techniques such as predictive coding, vector quantization, transform coding, subband coding and entropy coding are discussed. This also includes a brief description of standard compression algorithm like JPEG.

Chapter 3 concentrates on subband still image coding by considering different subband filters such as QMF, Lattice Structure Filters and IIR Filters. This is followed by our proposed subband coding system backed by a series of simulations.

Chapter 4 presents an overview of basic video image coding techniques by discussing the principle of interframe Differential Pulse Code Modulation (DPCM) including motion estimation techniques. The most popular video image coding systems such as Hybrid DPCM/DCT and Hybrid DPCM/Subband will be discussed. In addition, the ITU-T H.263 Standard will be briefly described.

Chapter 5 presents a subband video coding scheme which is based on a Hybrid DPCM/Subband model and the latest International Standard ITU-T H.263. Results from the simulation are presented and conclusions are drawn about the bit rate and picture quality.

Chapter 6 deals with the multi-resolution aspects of video transmission. A Subband Multi-Resolution Coding Scheme is proposed. This proposed coding scheme is then simulated using computer simulation. To improve the picture quality DPCM loop filtering is then investigated for the subband multi-resolution scheme.
In Chapter 7, conclusions will be drawn on this work. It will also outline further avenues of research.
Chapter 2

Review Of Still Image Coding Systems

2.1 Introduction

Research on image compression techniques has been actively carried out in the past two decades. The main goal in image compression is to reduce the amount of data carried by in the image. This will minimize the memory required for storage and the bandwidth needed for transmission. Image compression techniques can be divided into two main classes, namely still image coding and image sequence (video) coding. The former requires the removal of redundancies in the spatial domain (within a frame) and the latter also takes advantage of the temporal correlation (frame-to-frame).

For still image compression techniques, there are numerous methods which have been developed in the past [4-5]. For instance, most well established techniques are predictive coding, vector quantisation, transform coding, and subband coding. In addition, work in recent years has focused on evaluating suitable image coding systems in terms of performance, computation and compatibility.

A number of the above techniques have been adopted by industries and individual groups of people as proprietary standards for their particular use. However, it is important to have an international standard for retrieving and receiving encoded image data in a world wide digital data exchange. For this reason standards such as digital facsimile and JPEG (Joint Photographic Experts Group) have been developed for still image coding system.
2.2 Predictive Coding System

Predictive coding [5] is a popular technique for image compression. This technique removes the spatial redundancies, which normally exist between the neighbouring pixels within a two-dimensional image array. In this approach a pixel is predicted from its previous coded pixels. The prediction error signal is then formed by subtracting a given pixel value from its predicted value. The error signal is subsequently entropy coded and sent to the channel for transmission. On the receiver side, the original pixel is reconstructed by adding the prediction error to the predicted value.

The predictive coding technique is relatively simple and easy to implement. However, the drawback of this technique is its sensitivity to channel errors. A block diagram of a simple predictive coding system is illustrated in Figure 2.2.1.

![Predictive Coding System Diagram](image_url)

Figure 2.2.1 Predictive Coding System
The predictor can be based on a one-dimensional or two-dimensional prediction. For a one-dimensional prediction, only pixels along the same line are considered. However, two-dimensional predictors utilize pixels on the same line as well as on the previous line. An example of how various pixels can be used for predictive image coding is shown in Figure 2.2.2.

For instance, the one-dimensional predictor uses $S_{i-1,j}$ to estimate $S_{i,j}$. On the other hand, the two-dimensional predictor uses $S_{i-1,j}$ (from current line) and $S_{i,j-1}$, $S_{i,j-1}$ and $S_{i+1,j-1}$ (from previous line) to estimate the incoming pixel.

The above predictive coding method is normally considered for the lossless transmission of visual information. For lossy coding another predictive method known as Differential Pulse Code Modulation (DPCM) [5] should be considered. This method employs a quantiser to reduce the number of bits for each pixel. DPCM has been widely used for video compression to exploit frame to frame correlation. A simple block diagram of DPCM coding technique is illustrated in Figure 2.2.3. Interframe DPCM coding will be discussed in section 4.2.
Chapter 2
Review Of Still Image Coding Systems

Figure 2.2.3 DPCM Coding System
2.3 Vector Quantisation Coding System

Another alternative technique to achieve image compression is vector quantisation [6] which is based on learning and using a training vector set. This approach is also known as block quantisation or pattern matching quantisation. The vector quantisation can be defined as a mapping function \( Q \) (or an L-level K-dimensional quantiser) which maps input vector \( X \) into an output vector (or codebook) \( Y \). Thus,

\[
Q : X \rightarrow Y
\]

(Eqn. 2.3.1)

where \( X = \{x_i, i = 1, 2, \ldots, K\} \) is the input vector and \( Y = \{y_i, i = 1, 2, \ldots, L\} \) is the codebook vector.

In vector quantisation, an image is first divided into small blocks (e.g. \( 8 \times 8 \)). Each of these blocks is compared with the codebook entries. For the closest match of these entries, an index \( y_i \) corresponding to the appropriate entry is transmitted to the receiver. The receiver side uses the same codebook entry for reconstruction. The optimization procedure to design a codebook was introduced by Linde, Buzo and Gray [7], and is also known as the LBG algorithm. This algorithm provides an optimum codebook by minimizing the distortion.

The main drawback of vector quantisation is the complexity of training the codebook and also matching the input image with the codebook. In addition, the codebook is dependent on the input image. The simple vector quantisation coding system is illustrated in Figure 2.3.1.
Chapter 2 Review Of Still Image Coding Systems

- Form Block Vector
- Closest Mapping
- Codebook \( y_i \), \( i = 1, 2, \ldots, L \)

(a) Encoder

- From channel or storage
- Codebook Look Up
- Reconstructed Image
- Codebook \( y_i \), \( i = 1, 2, \ldots, L \)

(b) Decoder

Figure 2.3.1 Vector Quantisation Coding System
2.4 Transform Coding System

Transform coding has been developed for more than two decades and has proven to be one of the most effective image compression techniques [8-9]. The fundamental principle of transform coding is to transform the input image into a new domain represented by transform coefficients. Its aim is to decorrelate the input image into uncorrelated transform coefficients. In the transform domain, most of the energy is concentrated within the first few coefficients. Here, the transform coefficients with low energy can be removed without causing any significant degradation on the reconstructed image.

In transform coding, the input image is partitioned into $N \times N$ (e.g. $8 \times 8$ or $16 \times 16$) blocks (sub-images) using an orthonormal transform. A block size $8 \times 8$ has been adopted for most video coding standards mainly to reduce the transformation complexity as well as better exploitation of image redundancies between the neighbouring blocks. The coefficients are then quantised before they can be entropy encoded. Due to the energy compaction in the transform domain, one possibility to achieve compression is to use zonal coding. This method employs a mask matrix to cover the low frequency coefficients for quantisation and encoding. The remaining uncovered coefficients are discarded (set to zero). Considerable compression can be achieved depending on the size of the mask using this method.

The only problem with the zonal coding approach is the blurring effect as a result of eliminating higher frequency components. Another scheme to encode the transform coefficients is to perform thresholding on each transform coefficient. When a transform coefficient is below a threshold value, the coefficient is set to zero. The remaining non-zero coefficients after quantisation together with their address information are entropy coded. This can be achieved by a combination of Variable Length Coding (VLC) and runlength coding. For better subjective image quality, the quantiser in both cases should be designed to optimize the reconstructed image quality for a given number of bits. Finally, the encoded image will be transmitted through the channel (or stored). An
inverse operation is performed at the receiver side. A simple block diagram for a transform coding system is shown in Figure 2.4.1.

*Figure 2.4.1 Transform Coding System*
2.4.1 Orthonormal Transforms

The efficiency of a transform coding depends on the energy compacting ability of its transform method. There are various transform techniques such as Walsh-Hadamard, Harr, Slant, Cosine, Sine, Karhunen-Loeve ... etc. that have been considered for image compression [8-9]. Basically, Walsh-Hadamard, Harr and Slant transforms are easy to implement and similar in performance. However, Cosine transform due to its superb performance has been widely used for pictorial coding. Its performance is very close to the extremely complex but optimum Karhunen-Loeve transform.

The forward transform can be expressed as:

\[ Y = AX \]  
(Eqn. 2.4.1)

where \( X \) is the vector block of input image, \( A \) is the transform matrix and \( Y \) is the output transform coefficient vector.

The inverse transform can be expressed as:

\[
X' = A^{-1}Y \\
= X
\]  
(Eqn 2.4.2)

where \( X' \) is the reconstructed images, and \( A^{-1} \) is the inverse transform matrix.
2.4.1.1 Karhunen-Loeve Transform Technique

The Karhunen-Loeve Transform (KLT) [8-9] is an optimal transformation for decorrelating the image data. It also maximizes the energy compacted into the lower order coefficients. The KLT matrix depends on the image data and requires the estimation of the auto-covariance matrix of the image block and the determination of its eigenvectors (basis vectors) and eigenvalues (coefficients). This means that the transform matrix can be different for each image block. Thus, the KLT transform matrices have to be transmitted and stored along with the coded data.

Unfortunately, the computational complexity and large storage requirement make the implementation of KLT very difficult and therefore is seldom used. However, it is employed in theoretical studies of image coding. In practice, the KLT is usually substituted by the sub-optimum unitary transforms.

2.4.1.2 Discrete Cosine Transform Technique

The Discrete Cosine Transform (DCT) [8-9] is the most widely used transformation for image coding. It is an orthogonal transform which has a fixed set of base vector functions. These basic functions closely resemble the basic function of KLT. It also has correlation reduction capability, good energy compaction and fast computational properties [10].

The most popular block sizes for transform coding are $8 \times 8$ and $16 \times 16$. The two-dimensional $8 \times 8$ DCT has become a universal standard for image and data compression application. It can be written in terms of the pixel value $x(i, j)$, and the frequency domain transform coefficients $X(u, v)$. The transfer function of the forward DCT transform is given by:

$$X(u, v) = \frac{1}{4} \sum_{i=0}^{7} \sum_{j=0}^{7} C(u)C(v)x(i, j)\cos\left(\pi(2i+1)\frac{u}{16}\right)\cos\left(\pi(2j+1)\frac{v}{16}\right)$$

(Eqn. 2.4.3)
The transfer function of the inverse DCT transform is given by:

\[
x(i, j) = \frac{1}{4} \sum_{u=0}^{7} \sum_{v=0}^{7} C(u)C(v)X(u, v) \cos\left[\pi(2i + 1)\frac{u}{16}\right] \cos\left[\pi(2j + 1)\frac{v}{16}\right]
\]

(Eqn. 2.4.4)

where \(i\) and \(j\) are the spatial coordinates in the pixel domain, \(u\) and \(v\) are the coordinates in the transform domain, \(C(u) = 1/\sqrt{2}\) for \(u = 0\), otherwise 1, \(C(v) = 1/\sqrt{2}\) for \(v = 0\), otherwise 1.

with \(u, v, i, j = 0, 1, 2, ..., 7\)

Note

- Within the block being transformed, \(i = 0\) and \(j = 0\) refer to the pixel nearest to the top left corner of the picture.

For example, the transform coefficient distribution is illustrated in Figure 2.4.2. The DC coefficient contains the average energy of the block and the AC coefficients contain the high frequency components of the block [8-9].

![Figure 2.4.2 Transform Coefficients Distribution](image-url)
2.4.2 Quantisation of Transform Coefficients

This main concept of quantisation is to identify which coefficients should be retained for entropy coding. Each retained coefficient should be quantised in order to minimize the overall mean square quantisation noise. Quantisation can be performed by a uniform quantiser where all the quantisation intervals are equally spaced. Although this method is not very efficient, under certain conditions it can be effective to quantise the transform coefficients.

2.4.2.1 Uniform Quantisation Technique

After the DCT transformation, each block of $8 \times 8$ coefficients is quantised by a uniform quantiser. A simple block diagram of a uniform (midtread) quantiser is shown in Figure 2.4.3 (see reference [5]). In this figure, the quantisation step size is determined by the Quantisation Parameter ($QP$) which may differ from one block to the next. In addition, except for the DC coefficient, all the remaining coefficients use the same QP.

![Uniform (midtread) Quantiser Diagram](image.png)

**Figure 2.4.3 Uniform (midtread) Quantiser**
For example:

For d.c. coefficient,

\[ y(n) = \text{int}\{x(n)/8\} \quad \text{(Eqn. 2.4.5)} \]

For a.c. coefficient,

\[ y(n) = \text{int}\{x(n)/2 \times QP\} \quad \text{(Eqn. 2.4.6)} \]

The basic dequantisation equation is given by:

For d.c. coefficient,

\[
\begin{align*}
x'(n) &= 0 & \text{if} & \quad y(n) = 0 \\
x'(n) &= y(n) \times 8 & \text{if} & \quad y(n) \neq 0
\end{align*}
\quad \text{(Eqn. 2.4.7)}
\]

For a.c. coefficient,

\[
\begin{align*}
x'(n) &= 0 & \text{if} & \quad y(n) = 0 \\
x'(n) &= y(n) \times 2 \times QP & \text{if} & \quad y(n) \neq 0 , \ QP \ is \ odd \\
x'(n) &= y(n) \times 2 \times QP - 1 & \text{if} & \quad y(n) \neq 0 , \ QP \ is \ even
\end{align*}
\]

where

- \( x(n) \) DCT transform coefficient (input signal)
- \( x'(n) \) dequantised DCT transform coefficient (reconstructed signal)
- \( y(n) \) quantised signal
- \( QP \) quantisation parameter

The decision in selecting the Quantisation Parameter \( QP \) is based on a compromise between picture quality and bit rate requirements. The advantage of using this method is simplicity and ease of implementation.
2.4.2.2 Zonal Coding Quantisation Technique

Zonal Coding [9] is based on the fundamental idea of retaining certain transform coefficients. The locations of the coefficients with the highest energy are indicated by a zonal mask. This zonal mask is shown in Figure 2.4.4 and signifies the retained coefficients by ‘1’ and the removed (low information) coefficients by ‘0’. The same zonal mask is normally used for all blocks within the image. However, its drawback is loss of image sharpness as a result of discarding the higher order transform coefficients. On the other hand, the first transform coefficient which contains the average energy of the entire block should be encoded with greater accuracy to avoid blocking artifacts.

![Figure 2.4.4 An example of Zonal Mask](image)

2.4.2.3 Threshold Quantisation Technique

An alternative method to quantisation of transform coefficients is based on the spectral and statistical characteristics of each block. This would provide better image quality at a given bit rate as well as maintaining the significant higher order transform coefficients. In this approach, each block of transform coefficient is quantised by a Quantisation Matrix (QM). Each QM has different quantisation intervals and retains the coefficient in different locations. Threshold coding is a better method where only those coefficients
whose magnitudes are above a threshold are retained within each block [9]. In practice, thresholding and quantising can be combined by means of Quantisation Matrix as

\[ S'(i, j) = \text{int}\left[ \frac{S(i, j)}{T(i, j)} \right] \]  

(Eqn. 2.4.9)

where \( S'(i, j) \) is the threshold and quantised approximation of \( S(i, j) \) and \( T(i, j) \) is the element of the Quantisation Matrix with respect to coordinates \( i \) and \( j \). Coarser quantisation is used for higher frequency coefficients. A typical Quantisation Matrix for luminance images is illustrated in Figure 2.4.5 (extracted from reference [53]).

![Quantisation Matrix](image)

**Figure 2.4.5** An example of Quantisation Matrix
2.4.3 Entropy Coding of Transform Coefficients

The last stage in a transform coding system as shown in Figure 2.4.1 is entropy coding. The idea behind entropy coding is to assign shorter codewords to the more frequent symbols and thus minimize the average length of the binary representation of the input information [11]. If there are $N$ input symbols $s_1, s_2, s_3, \ldots, s_N$ with probabilities $p(s_1), p(s_2), p(s_3), \ldots, p(s_N)$, then the average information rate is given by entropy (measured in bits)

$$H(S) = -\sum_{i=1}^{N} p(s_i) \log_2 p(s_i) \quad \text{(Eqn. 2.4.10)}$$

The entropy $H(S)$ for $N$ input symbols can range from 0 to $\log_2 N$. The performance of the code can be measured with respect to its length average word length. For example, the input symbols $s_1, s_2, s_3, \ldots, s_N$ are entropy encoded by codewords $c_1, c_2, c_3, \ldots, c_N$ with word length $l_1, l_2, l_3, \ldots, l_N$. This gives the average number of bits required by the codeword set $R(S)$ (also known as average codeword length),

$$R(S) = \sum_{i=1}^{N} l_i p(s_i) \quad \text{(Eqn. 2.4.11)}$$

Run Length Coding (RLC) was first considered for black and white images. This was achieved by considering each scan line as a consecutive black and white run. The run length is found by counting the number of consecutive black and white pixels along each line. An example of a horizontal line along an image is illustrated in Figure 2.4.6. It is coded as 5 black-run, 4 white-run, 7 black-run, 5 white-run \ldots etc., where 0 and 1 represents black and white pixels respectively. The codebook is designed in accordance with the run length probability distribution of the bilevel signal. The limitation of this method is that can only be applied to black and white image coding.
This runlength coding method has been further developed as two-dimensional Variable Length Coding (2D VLC) [4] so that colour images can be encoded. The image is encoded as an *EVENT*. Each *EVENT* contains *RUN* and *LEVEL*.

\[ \text{EVENT} = (\text{RUN}, \text{LEVEL}) \]

where  
- *RUN* is the number of successive zeros preceding the quantised coefficient  
- *LEVEL* is the non zero value for the quantised coefficient

Finally, 3D VLC [45] is developed to improve the coding efficiency. In this approach, each *EVENT* contains *LAST*, *RUN*, *LEVEL*. The *LAST* event is represented by the *End of Block (EOB)* which indicates that no more zero coefficients are encoded for this block.

\[ \text{EVENT} = (\text{LAST}, \text{RUN}, \text{LEVEL}) \]

where  
- *LAST* = 0  there are more non zero coefficients in this block  
- *LAST* = 1  this is the last non zero coefficient in this block  
- *RUN*  is the number of successive zeros preceding the quantised coefficient  
- *LEVEL*  is the non zero value of the quantised coefficient

The codebook design is based on the probability distribution of the 3D *EVENT*. The limitation of this method is the complexity of constructing the codebook. However, it is very efficient in terms of coding and has been adopted as part of the ITU-T H.263 Coding Standard [45].

---

**Figure 2.4.6** An example of Runlength Coding

\[
\begin{array}{cccccccccccccccc}
0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & \ldots \end{array}
\]

5 Black-Run 7 Black-Run 4 White-Run 5 White-Run
Chapter 2 Review

2.5 International Standard Still Image Coding Systems

During the tremendous growth in the computer and media industries, many new digital image data compression techniques have been developed. In order to be able to communicate internationally, it is vital that standardized coding schemes and transmission formats are developed.

Recognizing this need, several international organizations have been participating in the development of international still image standards. These organizations are the International Standard Organisation (ISO), the International Telecommunication Union (ITU) (formerly International Telegraph and Telephone Consultative Committee (CCITT)) and the International Electrotechnical Commission (IEC).

Expert groups have been formed and chartered by ISO, IEC and ITU (formerly CCITT) and various proposals submitted from companies, universities and research laboratories. The international standards have been selected from these submitted proposals based on image quality, compression performance and practical constraints.

2.5.1 JPEG Still Image Coding Algorithms

The Joint Photographic Experts Group (JPEG) [12-14] was formed in 1986 to establish a standard for continuous tone (grey scale and colour) still images. The JPEG committee could not achieve an algorithm for every still image compression application. Therefore, the committee proposed four different modes of operation, Sequential DCT-based mode, Progressive DCT-based mode, Loseless mode and Hierarchical mode. This work was published in 1992 as CCITT Recommendation T.81. A text identical to this was also published as ISO/IEC International Standard 10918-1.

1. Sequential DCT-based mode offers very good compression ratios, whilst maintaining excellent image quality. A simplified diagram of a sequential DCT-based Coding System is illustrated in Figure 2.6.1. All DCT-based JPEG coders begin the coding process by partitioning the input image into non-overlapping 8 x 8 blocks of pixels. This
block of 64 pixels is transformed to a block of 64 coefficients by the forward discrete cosine transform (DCT). The coefficients are then quantised and entropy coded. In order to increase the compression efficiency, a simple predictive method is used to entropy code the DC coefficient. The average intensity of each of component block has a similar DC coefficient value and it is advantageous to encode the difference between DC coefficient of adjacent blocks rather than their values.

Figure 2.5.1 Sequential DCT Coding System
The AC coefficients after zig-zag scanning (see Figure 2.5.2) and quantisation are entropy coded. The entropy coder assigns codewords representing the number of zeros (RUN) since the last nonzero coefficient, and the value of the non-zero coefficient (LEVEL). These codewords are also called two dimensional variable length codes (2D VLC).

![Zigzag scan ordering of coefficients](image)

**Figure 2.5.2 Procedure for DCT Scanning**

2. **Progressive DCT-based mode** has been designed to satisfy the need for a fast picture decoding process. This process supports those with low bandwidth channel transmission. By partially encoding the quantised DCT coefficients in multiple passes, only a portion of these coefficients are transmitted in each pass. The image is reconstructed progressively from a coarse level to a high quality picture. Either spectral selection, successive approximation, or a combination of the two, is used to code the quantised coefficients.

In the spectral selection method, the quantised DCT coefficients of a block are first partitioned into non-overlapping spectral bands along the zig-zag block scan. The lowest frequency bands are encoded and sent first. This yields a rather blurred image at the receiver. The reconstructed image continues to improve as more AC coefficients are
received. When all the DCT coefficients are eventually decoded, the image quality will be the same as that of the sequential mode.

In the successive approximation method, the precision of the coefficients is successively increased during multiple passes. The DC coefficient of each block is sent first with full precision to avoid mean level mismatch. The AC coefficients are transmitted starting with the most significant bit plane. Successive approximation usually gives better quality images than spectral selection at lower bit rates.

3. **Loseless mode** was defined for applications where the output images from a decoder were identical to the input images of the encoder. That means no information is lost in this process and the image is perfectly reconstructed. In this mode, the compression ratios are much smaller than other modes. No DCT transform is required, instead a predictive coding method is used. A predictor with a selection of seven choices is employed as shown in Table 2.5.1. Figure 2.5.3 illustrates the prediction neighbourhood of the sample pixel \( x \) with the neighbour pixel \( a, b \) and \( c \). Entries 1 to 3 in Table 2.5.1 are used for one-dimensional predictive coding, and 4 to 7 for two-dimensional predictors. There is no prediction for Entry 0.

<table>
<thead>
<tr>
<th>Selection value</th>
<th>Prediction</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>No prediction</td>
</tr>
<tr>
<td>1</td>
<td>( a )</td>
</tr>
<tr>
<td>2</td>
<td>( b )</td>
</tr>
<tr>
<td>3</td>
<td>( c )</td>
</tr>
<tr>
<td>4</td>
<td>( a + b - c )</td>
</tr>
<tr>
<td>5</td>
<td>( a + ((b - c) / 2) )</td>
</tr>
<tr>
<td>6</td>
<td>( b + ((a - c) / 2) )</td>
</tr>
<tr>
<td>7</td>
<td>( (a + b) / 2 )</td>
</tr>
</tbody>
</table>

*Table 2.5.1 Table for Loseless Mode Prediction*
Chapter 2 Review Of Still Image Coding Systems

Figure 2.5.3 An example of Neighbourhood Prediction

4. Hierarchical mode offers a higher quality alternative to the previously described progressive method. The input image is coded as a succession of increasingly higher resolution frames, also known as pyramidal coding. However, it is more expensive to implement. It also allows decoders with a different resolution image to use the same compressed stream.

In the first stage, the lowest resolution image is encoded using one of the sequential or progressive modes. The output of each hierarchical stage is then upsampled (interpolated) and is taken as the prediction for the next stage. The upsampling filters double the resolution horizontally and vertically. In the next stage, the difference between the actual second layer in the pyramid and the upsampled first layer is encoded and transmitted. This process continues until the decoded image has the same resolution as the full resolution input image. After that, one or more full resolution difference images may be coded. A hierarchical decoder may abort the decoding process when the frame attains the desired resolution.

The hierarchical encoding and decoding processes are not symmetrical. Indeed, a hierarchical encoder must also include the greater part of a decoder. However, a hierarchical decoder is more complex than a nonhierarchical decoder because it must provide a way to upsample the decoding layer. This increased complexity may be justified because of the flexibility afforded in matching the decoder to a specific application.
In these JPEG mode operations, either Huffman or arithmetic coding is employed for entropy coding. A baseline coder can only use Huffman coding. JPEG does not have a standard Huffman code table. Therefore, the encoders are required to transmit their relevant Huffman code tables according to the appropriate application. The output of an arithmetic coder is a single real number for each colour component image. Unlike a Huffman coder, an arithmetic coder does not require an integral number of bits to represent an input symbol. As a result, arithmetic coders are usually more efficient than Huffman coders.

2.5.2 Other Developments in the Standard Bodies

The most well known image compression standards are the ITU (formerly CCITT) digital facsimile Group 3 (G3) and Group 4 (G4) coding schemes [15-16] for bi-level images. The ITU G3 coding scheme is used primarily for transmitting an A4 size (210 mm x 297 mm) document over a Public Switched Telephone Network (PSTN). It employs the Modified Huffman code and also an optional Modified READ code (Relative Element Address Designate) for encoding documents and still images [15].

The ITU (former CCITT) G4 coding scheme was designed for more general images by incorporating four possible resolutions (200, 240, 300 and 400 pels per inch). This provides for a mixed mode operation containing symbols and graphics. Those parts of the image containing alpha-numerics are transmitted as text formats using the ASCII code. The other parts of the image containing non-character information such as drawing and handwriting are coded by the basic G3 coding schemes. Typical G4 compression ratios has been found to be 15:1 on representation test documents [16].

The Joint Bi-level Image Group (JBIG) [17] developed an alternative bi-level image compression standard to improve the performance of G3 and G4. The JBIG algorithm is more complex than that of Group 3 and Group 4 facsimile algorithms. This offers two advantages that compensate each other. These advantages are that they provide better compression for text or line-art images and allow progressive coding.
2.6 Subband Coding System

Subband coding was introduced by Crochier [18-20] in 1976 and initially used for speech coding applications. Since the introduction of a perfect reconstruction subband filter by Croisier [21], subband filtering techniques have been applied for image coding applications. Its advantage over other coding schemes is that the generated noise within that particular subband will not spread to other bands. The other advantage of subband coding is the absence of blocking artifacts compared with DCT [22].

The basic idea behind subband coding is to partition the original signal spectrum into several bands (subbands). Each subband is then separately encoded and transmitted through the channel individually. At the receiving end, each subband is decoded and combined together to reconstruct the original signal. A simple block diagram for a subband coding system is illustrated in Figure 2.6.1.

![Subband Coding System Diagram](image)

(a) Encoder

(b) Decoder

Figure 2.6.1 Subband Coding System
In 1984, Vetterli [23] first introduced the concept of extending a one dimensional subband to a two dimensional separable subband filter. This is achieved by applying a one-dimensional subband filter along both the horizontal and vertical directions of the image, which produces four basic subbands as illustrated in Figure 2.6.2. An example of 4 Band decomposition of the image spectrum is shown in Figure 2.6.3. These basic band structures may be further extended in a tree structure manner to obtain more subbands as illustrated in Figure 2.6.4. More details of Subband Filtering techniques will be discussed in Chapter 3.

![Subband Decomposition Structure](image1)

(a) Subband Decomposition Structure

![Decomposition Bands](image2)

(b) Decomposition Bands

Figure 2.6.2 4 Bands Decomposition Spectrum
Figure 2.6.3 An example of 4 Bands Decomposition Output

(a) Subband Decomposition Structure

(b) Decomposition Bands

Figure 2.6.4 Octave 7 Bands Decomposition Spectrum
2.6.1 Quantisation of Subbands

After the introduction of a separable subband filtering technique, a number of subband image coding systems were developed. Quantisation techniques were first considered for the subband image coding system. DPCM (Differential Pulse Code Modulation) technique was one of the early developments applied to subband coding [24]. The input image is first decomposed into four subbands. Each of these subbands will be quantised by DPCM. The number of bits for each pixel depends on the quantisation level. This method is very easy to implement. It also takes advantage of the correlation of the image. A simple block diagram of this model is illustrated in Figure 2.6.5.

![Diagram of DPCM of All Frequency Bands]

**Figure 2.6.5** DPCM of All Frequency Bands
However, the main drawback of the above model is its lack of efficiency because of encoding the higher frequency bands with DPCM. An alternative model utilizes Dead Zone Quantisation to improve the efficiency [25-26]. First the DPCM is quantised for the lowest frequency bands. However, the rest of the higher frequency bands are quantised by the dead zone quantisation. This method takes advantage of the characteristics of subband. After subband decomposing, the lowest band contains most of the energy and the higher frequency bands contain lower energy. Dead Zone Quantisation was designed using this consideration. Finally, each quantised subband is encoded by a combination of variable length and runlength coding. A simple block diagram of this model is illustrated in Figure 2.6.6.

**Figure 2.6.6 DPCM of the First Band**
2.6.1.1 Dead Zone Quantisation

An effective method of quantising higher frequency subband signals is to use a quantiser with a dead zone [25]. Figure 2.6.7 (extracted from reference [25]) shows the dead zone quantiser with the following characteristics. First, the quantiser has a centre dead zone ‘d’ which is used to eliminate the picture noise. Second, the quantiser range is fixed by the lower and upper limit thresholds ‘±t’. Third, the signal falling within the active range is uniformly quantised into L levels. Finally, any signal above the threshold ‘+t’ or below threshold ‘-t’ is mapped to saturation values +y and -y, respectively. Thus the total number of quantisation levels is given by L+3.

![Dead Zone Quantisation Diagram]

**Figure 2.6.7 Dead Zone Quantisation (extracted from ref. [25])**

The decision in selecting parameters d, t, y, and L is based on a compromise between picture quality and bit rates requirements. These parameters can be changed according to the size of the buffer. For example, a large dead zone value d can be used to prevent the high bit rate [25].
2.6.2 Run Length Coding of Subbands

The last stage of the subband coding system (shown in Figure 2.4.4) is binary encoding. After quantisation, the locations of non-zero values are encoded by B codes [26]. The reason for using B codes is that the probability density functions of images (after quantisation) are very similar to the distribution of B codes [27]. It is also very simple to implement this method.

The basic idea behind B codes involves using continuation (or colour) bits and information bits to represent each codeword. The basic B codes can be extended to give \( B_1, B_2 \) and \( B_3 \) codes. These basic codewords are shown in Table 2.6.1. The B codes are nearly optimum for the probability density function of the inputs which obey the power law (Equation 2.6.1).

\[
p(s_i) = i^{-\gamma} \quad \text{(Eqn. 2.6.1)}
\]

where \( p(s_i) \) is the probability distribution of run length \( s_i \), \( i \) is the number of run length, \( \gamma \) is any positive constant value.

An example of a \( B_1 \) code is presented on the next page Table 2.6.1. Half the number of bits in each codeword are continuation bits (or colour bits \( C \)) and the other half are information bits. The information bits are represented by a binary number which increases in length. The continuation bit is either a '0' or '1'. For a facsimile image, each pixel is black or white therefore the continuation bit can be set equal to the gray level (\( C=0 \) for black and \( C=1 \) for white). Note that the code is not instantaneous because the decoder must look ahead to the next continuation bit in order to determine whether or not the present reception or transmission of the codeword has ended.

The implementation of a \( B_1 \) code is very simple. It uses an up-counter to count one for each identical coloured bit until the end of that run. At the end of each run the counter is reset to zero and the continuation bit is flipped (colour bit, \( C \), is flipped from 0 to 1 or vice versa). This indicates the colour of the next run. Decoding can be accomplished by
presetting a down-counter and performing count down until the continuation bit changes state. This process continues in accordance with the received information. Higher order B codes can also be implemented in a similar way. A $B_n$ code uses $n$ information bits for each continuation bit. An example of B codes is shown in table 2.6.1.

<table>
<thead>
<tr>
<th>Run</th>
<th>B1 Code</th>
<th>B2 Code</th>
<th>B3 Code</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>C0</td>
<td>C00</td>
<td>C000</td>
</tr>
<tr>
<td>2</td>
<td>C1</td>
<td>C01</td>
<td>C001</td>
</tr>
<tr>
<td>3</td>
<td>C0C0</td>
<td>C10</td>
<td>C010</td>
</tr>
<tr>
<td>4</td>
<td>C0C1</td>
<td>C11</td>
<td>C011</td>
</tr>
<tr>
<td>5</td>
<td>C1C0</td>
<td>C00C00</td>
<td>C100</td>
</tr>
<tr>
<td>6</td>
<td>C1C1</td>
<td>C00C01</td>
<td>C101</td>
</tr>
<tr>
<td>7</td>
<td>C0C0C0</td>
<td>C00C10</td>
<td>C110</td>
</tr>
<tr>
<td>8</td>
<td>C0C0C1</td>
<td>C00C11</td>
<td>C111</td>
</tr>
<tr>
<td>9</td>
<td>C0C1C0</td>
<td>C01C00</td>
<td>C000C00</td>
</tr>
<tr>
<td>10</td>
<td>C0C1C1</td>
<td>C01C01</td>
<td>C000C01</td>
</tr>
<tr>
<td>11</td>
<td>C1C0C0</td>
<td>C01C10</td>
<td>C000C10</td>
</tr>
<tr>
<td>12</td>
<td>C1C0C1</td>
<td>C01C11</td>
<td>C000C11</td>
</tr>
<tr>
<td>13</td>
<td>C1C1C0</td>
<td>C10C00</td>
<td>C000C100</td>
</tr>
<tr>
<td>14</td>
<td>C1C1C1</td>
<td>C10C01</td>
<td>C000C101</td>
</tr>
<tr>
<td>15</td>
<td>C0C0C0</td>
<td>C10C10</td>
<td>C000C110</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>etc.</td>
<td>etc.</td>
<td>etc.</td>
<td>etc.</td>
</tr>
</tbody>
</table>

C is the colour of the run

Table 2.6.1 An example of B codes

Runlength coding efficiency can be further enhanced by a scanning process [40]. Each band is first partitioned into non-overlapping blocks. The scanning is done on a block by block basis starting from the first block on the upper most left and continuing in a horizontal direction until the last block (upper most right ) is scanned together with non-zero coded values. The same process is followed until the last strip of block in each sub-image is scanned and coded. Three different scanning techniques are used and they are all based on the direction of the actual scanning process. These are the horizontal (HS),
vertical (VS) and diagonal (DS) scans. The scanning process is shown in Figure 2.6.8 (extracted from reference [40]).

Figure 2.6.8 Procedure for Block by Block Scanning (extracted from ref. [40])
Chapter 3

Subband Still Image Coding System

3.1 Introduction

A Subband coding technique has the inherent advantage of providing multi-resolution output. This is because the lowest frequency band always carries the most important information. The resolution of the four decomposed subbands is a quarter of the original image. This provides an option for the receiver to decode either just the lowest frequency or all the subbands. Therefore, the receiver can reconstruct low or high resolution images. The other advantage of subband coding compared with DCT is being free of blocking artifacts.

The objective of this research is to develop an efficient subband-based coding scheme compatible with existing standard. This chapter discusses a number of different subband filtering techniques. These filters include Quadrature Mirror Filters (QMF), Lattice Structure Filters and the Infinite Impulse Response (IIR) Filters. A comprehensive evaluation of these filters will be carried out. This is then followed by presenting our proposed subband still image coding scheme. Finally, the performance of our scheme will be further evaluated.
3.2 Subband Filtering Theory

The basic principle behind subband coding [28-29] is the partitioning of the signal spectrum into several bands so that they can be encoded separately. This method is particularly suited for applications in image coding because natural images tend to have a non-uniform frequency spectrum. The basic 1-D subband filter bank is illustrated in Figure 3.2.1. Here, the input signal is divided into low and high frequency bands each of which contain half the input signal information.

![Subband Filter Bank Diagram](image)

(a) Analysis Filter Bank

(b) Synthesis Filter Bank

Figure 3.2.1 Subband Filter Bank
The input signal \( x(n) \) is filtered by a low pass filter \( F_0(z) \) and a high pass filter \( F_1(z) \) which are also called *analysis* filters. As a result, two different filtered signals \( Y_0(z) \) and \( Y_1(z) \) are obtained. Each of these filtered signals has a bandwidth that is half the size of the original input signal \( x(n) \). The resulting signals can be described in the frequency domain as:

\[
Y_0(z) = F_0(z)X(z) \\
Y_1(z) = F_1(z)X(z)
\]  
(Eqn. 3.2.1)

where \( X(z) \) is the Z-transform of the input signal \( x(n) \), \( Y_0(z) \) and \( Y_1(z) \) are the low-pass and high-pass filtered signals respectively.

These two filtered signals \( Y_0(z) \) and \( Y_1(z) \) are then subsampled by an integer factor of 2 which gives the subband signals \( X_0(z) \) and \( X_1(z) \). The process of downsampling (or decimation) consists of taking every other sample from the filtered signal \( y_0(n) \) and \( y_1(n) \) (i.e. \( Y_0(z) \) and \( Y_1(z) \) in Z-domain) and discarding the rest. The decimated output signals \( x_0(m) \) and \( x_1(m) \) are given by:

\[
x_0(m) = y_0(n) \\
x_1(m) = y_1(n)
\} \text{ with } \begin{align*}
m &= 1,2,3,4,\ldots \\
n &= 2,4,6,8,\ldots
\end{align*}
\]  
(Eqn. 3.2.2)

where \( x_0(m) \) and \( x_1(m) \) are the decimated output signals. The number of samples is now half that of the filtered signal \( y_0(n) \) and \( y_1(n) \). The above equation 3.2.2 when transformed into Z-domain is equivalent to the given equation 3.2.3.

\[
X_0(z) = \frac{1}{2} \left[ Y_0(z^{1/2}) + Y_0(-z^{1/2}) \right] \\
X_1(z) = \frac{1}{2} \left[ Y_1(z^{1/2}) + Y_1(-z^{1/2}) \right]
\]  
(Eqn. 3.2.3)

where \( Y_0(z) \) and \( Y_1(z) \) are the Z-transforms of the filtered signals \( y_0(n) \) and \( y_1(n) \), \( X_0(z) \) and \( X_1(z) \) are the Z-transforms of the decimated signal \( x_0(m) \) and \( x_1(m) \), respectively.
The decimation process for the filtered signal $Y_d(z)$ in the frequency domain is shown in Figure 3.2.2. The downsampled processes for signal $Y_d(z)$ and $Y_1(z)$ are similar therefore only $Y_d(z)$ is used as an example. It can be seen that compression in the time domain (decimation) results in expansion in the frequency domain. The over-lapping areas of the bottom part of Figure 3.2.2 indicate the aliasing errors due to subsampling. This effect can also be deduced from equation 3.2.3, when the signals $Y_d(z^{1/2})$ and $Y_d(-z^{1/2})$ are added. Consequently, if the filtered signal has a bandwidth of only $[-\pi/2, \pi/2]$, no aliasing errors will occur.

Figure 3.2.2 Decimation by a factor of 2
Combining equation 3.2.1 and 3.2.3 yields the subband signals

\[
X_0(z) = \frac{1}{2} \left[ F_0(z^2)X(z^2) + F_0(-z^2)X(-z^2) \right]
\]
\[
X_1(z) = \frac{1}{2} \left[ F_1(z^2)X(z^2) + F_1(-z^2)X(-z^2) \right]
\]

(Eqn. 3.2.4)

Note that the total number of samples has not increased after splitting into \(X_0(z)\) and \(X_1(z)\) subbands [30]. Then the resulting subband signals \(X_0(z)\) and \(X_1(z)\) are encoded, transmitted through the channel (refer to Figure 3.2.1 (a)).

In Figure 3.2.1 (b), the subband signals \(X'_0(z)\) and \(X'_1(z)\) are first decoded by the receiver and then upsampled (or interpolated) to the full bandwidth signals \(Y'_0(z)\) and \(Y'_1(z)\). The upsampling process for both \(X'_0(z)\) and \(X'_1(z)\) are similar therefore only \(X'_0(z)\) is used as an example. Upsampling or interpolation of the subband signal with an integer factor of 2 is done by inserting a zero between every sample of the signal \(x'_0(m)\) (i.e. \(X'_0(z)\) in Z-domain).

\[
y'_0(n) = x'_0(m), \text{ where } m = 2, 4, 6, \ldots \ldots
\]
\[
y'_0(n) = 0, \text{ otherwise}
\]

(Eqn. 3.2.5)

where \(y'_0(n)\) is the interpolated output signal, \(n\) is an integer (i.e. \(n=1, 2, 3, \ldots\)).

The above equation 3.2.5 is transformed into Z-domain and given by

\[
Y'_0(z) = X'_0(z^2)
\]

(Eqn. 3.2.6)
An expansion in the time domain results in compression in the frequency domain as illustrated in Figure 3.2.3. Since $Y'_0(z)$ has a replica (or an image) of the spectrum $X'_0(z)$, the interpolator is said to cause an imaging effect. To remove this effect, the interpolation stage is usually followed by band-pass filtering. It can be seen from Figure 3.2.1 (b) that at the subband reconstruction stage the imaging effect is cancelled by using suitable band-pass filtering techniques.

![Diagram](image-url)

(a) Before Interpolation

![Diagram](image-url)

(b) After Interpolation

Figure 3.2.3 Interpolation by a factor of 2
Referring to equation 3.2.6, the interpolated (upsampled) subband signals $Y'_0(z)$ and $Y'_1(z)$ are given by:

$$
Y'_0(z) = X'_9(z^2)
$$

$$
Y'_1(z) = X'_1(z^2)
$$

(Eqn. 3.2.7)

In order to put each (decoded) subband signal at its original position in the signal spectrum, the interpolated signals $Y'_0(z)$ and $Y'_1(z)$ are then band-pass filtered using a low-pass filter $G_0(z)$ and a high-pass filter $G_1(z)$ respectively. These are also called *synthesis* filters.

Finally, the filtered results are added to obtain the reconstructed signal $X'(z)$:

$$
X'(z) = G_0(z)Y'_0(z) + G_1(z)Y'_1(z)
$$

(Eqn. 3.2.8)

Assuming there are no quantisation and transmission errors, the input and output relations can be obtained from equation 3.2.4 and 3.2.8.

$$
X'(z) = \frac{1}{2} G_0(z)[F_0(z)X(z) + F_0(-z)X(-z)] + \frac{1}{2} G_1(z)[F_1(z)X(z) + F_1(-z)X(-z)]
$$

(Eqn. 3.2.9)

This can be rewritten by grouping separately the parts consisting of the original signal $X(z)$, and the parts consisting of the aliasing error effect $X(-z)$.

$$
X'(z) = \frac{1}{2} [G_0(z)F_0(z) + G_1(z)F_1(z)]X(z)
$$

$$
+ \frac{1}{2} [G_0(z)F_0(-z) + G_1(z)F_1(-z)]X(-z)
$$

(Eqn. 3.2.10)

The first term in equation 3.2.10 is the desired reconstructed signal $x'(n)$ while the second term is the aliased component.
3.2.1 Quadrature Mirror Filter

To eliminate the aliasing problem in the previous section, Quadrature Mirror Filtering was introduced as a special class of low-pass and high-pass filters [30]. These filters are designed to have a frequency spectrum that has a mirror image symmetry at the centre frequency of $\pi/2$ as in Figure 3.2.4.

![Figure 3.2.4 Quadrature Mirror Filter](image)

The aliased components can be eliminated by setting the synthesis filter to:

\[
G_o(z) = 2F_1(-z) \\
G_1(z) = -2F_0(-z)
\]  
(Eqn. 3.2.11)

For convenience, a factor of 2 was introduced here to remove the constant $(1/2)$ in the first term on the right hand side of equation 3.2.10. Once the aliasing is cancelled, the alias-free reconstructed signal becomes:

\[
X'(z) = [F_0(z)F_1(-z) - F_0(-z)F_1(z)]X(z)
\]  
(Eqn. 3.2.12)
The two channel filter banks can be written as a linear and shift-invariant system which is described by the transfer function:

\[ T(z) = \frac{X'(z)}{X(z)} = \left[ F_0(z)F_1(-z) - F_1(z)F_0(-z) \right] \]  
(Eqn. 3.2.13)

Various QMF filters based on FIR filter structure have been developed which are known as Johnston 8A, 16A, 24B, 32C and 64E filters [31]. In this thesis, only the Johnston QMF 8A and 16A are considered because of the small number of filter coefficient taps used.

### 3.2.2 Lattice Structure Filtering

Subband filtering is carried out primarily using Finite Impulse Response (FIR) filters which normally have a large number of filter coefficient taps. Lattice structure filters can be used to reduce the computational time [32]. Orthogonal Perfect Reconstruction Quadrature Mirror Filter (PRQMF) and bi-orthogonal Perfect Reconstruction Linear Phase Filter (PRLPF) are examples which are based on this structure [33-34].

The transfer functions of these filters are given by:

\[
    G_0(z) = -\gamma z^{-2(D-L)} F_1(-z) \\
    G_1(z) = \gamma z^{-2(D-L)} F_0(-z) 
\]  
(Eqn. 3.2.14)

where

- \( \gamma \) is constant gain,
- \( D \) is delay factor,
- \( L \) is the number of filter tap coefficients,
- \( F_0(z) \) is low-pass analysis filter,
- \( F_1(z) \) is high-pass analysis filter,
- \( G_0(z) \) is low-pass synthesis filter,
- \( G_1(z) \) is high-pass synthesis filter.
For equation 3.2.14 to have perfect reconstruction the necessary condition is $D \geq L$. In order to simplify and keep the generality of equation 3.2.14, $D$ is set to be equal to $L$. Therefore $G_0(z)$ and $G_1(z)$ become:

$$
G_0(z) = -\gamma \, F_1(-z) \quad (\text{Eqn. 3.2.15})
$$

$$
G_1(z) = \gamma \, F_0(-z)
$$

For a PRQMF lattice filter, $\gamma$ can be expressed as follows in terms of the lattice coefficient, $\rho_i$:

$$
\gamma = -\frac{1}{2^{N+1}(1 + \rho_i^2)} \quad \text{with} \quad 0 < \gamma < 1 \quad (\text{Eqn. 3.2.16})
$$

For a PRLPF lattice filter, $\gamma$ can be expressed as follows in terms of the lattice coefficient, $\rho_i$:

$$
\gamma = -\frac{1}{2^{N+1}(1 - \rho_i^2)} \quad \text{with} \quad 1 - \rho_i^2 \neq 1 \quad (\text{Eqn. 3.2.17})
$$

The PRQMF and PRLPF filters are illustrated in Figures 3.2.5 and 3.2.6 respectively. Both of these filters have a filtering length of $2(N+1)$, where $(N+1)$ is the number of lattice coefficients $\rho_i$. As shown in Figure 3.2.5 and 3.2.6, the value of $\eta$ and $\lambda$ are obtained by splitting the gain constant $\gamma$ into analysis and synthesis gain constants, where $\gamma = \eta \times \lambda$ with $\eta \neq 0$ and $\lambda \neq 0$. 

46
Figure 3.2.5 PRQMF Lattice Structure Filter Coding System

Figure 3.2.6 PRLPF Lattice Structure Filter Coding System
3.2.3 Infinite Impulse Response Filtering

An alternative to the previous approaches is using an Infinite Impulse Response (IIR) filter. The advantage of using the IIR Filter is fewer computational operations because of the small number of filter coefficient taps. Figure 3.2.7 shows an example of employing an IIR filter structure for bandwidth decomposition [35].

The input signal is decomposed into subbands using a polyphase filter unit, which includes an IIR all pass filter. To reconstruct the original image, the IIR all pass filter is approximated by a Finite Impulse Response (FIR) filter. Both filter coefficients are in terms of powers two. Thus, no multiplication is required to implement these filters. This also minimizes the amount of processing required by each filter.

![Diagram of IIR Filter Coding System](image)

(a) Encoder

(b) Decoder

Figure 3.2.7 IIR Filter Coding System
The transfer function of the IIR filter is given by:

\[ T(z) = \frac{1 + z^{-1}}{\frac{8}{1 + \frac{1}{8} z^{-1}}} \]  
(Eqn. 3.2.18)

The implementation of this transfer function is shown in Figure 3.2.8.

\[ x(n) \rightarrow + \rightarrow 1/8 \rightarrow + \rightarrow y(n) \]

\[ \text{Z}^{-1} \]

\[ \frac{1}{8} \]

\[ \text{Figure 3.2.8 IIR Filter Structure} \]

The FIR filter is used to reconstruct the original image. The transfer function of this is given by:

\[ T(z) = \frac{1}{8} + \left(1 - \frac{1}{64}\right) \left(z^{-1} - \frac{1}{8} z^{-2} + \frac{1}{64} z^{-3}\right) \]  
(Eqn. 3.2.19)

The implementation of this transfer function is shown in Figure 3.2.9.

\[ x(n) \rightarrow Z^{-1} \rightarrow Z^{-1} \rightarrow Z^{-1} \rightarrow + \rightarrow + \rightarrow + \rightarrow + \rightarrow y(n) \]

\[ \frac{1}{64} \]

\[ -\frac{1}{8} \]

\[ \frac{1}{8} \]

\[ \text{Figure 3.2.9 FIR Filter Structure} \]
3.3 Subband Filtering Performance Evaluation

A comprehensive comparison between the QMF filter, Lattice Structure Filter and IIR filter is carried by decomposing and reconstructing a number of test images. The QMF filters are Johnston's 8 & 16 taps which are classified as type A. The lattice structure filters under this evaluation are PRQMF and PRLPF. The aim of this evaluation is to look at the sensitivity to truncation error with respect to these five different filters.

Filter simulations are carried out by decomposing an image into four subbands using two dimensional analysis filter banks. The image is subsequently reconstructed using the appropriate synthesis filter. The 2D decomposition and reconstruction process consists of two stages as shown in Figure 3.3.1.

As mentioned in section 2.6, subband decomposition and reconstruction is achieved by applying a 1D subband filter along the horizontal and vertical direction of the image. Therefore, each of the 2D decomposition and reconstruction processes is divided into two stages. The first stage consists of one analysis filter whereas the second stage consists of two analysis filters (see Figure 2.6.2). The first stage of the decomposition results in two bands (horizontal decomposition) and then four bands after the second stage (vertical decomposition). At the end of each stage, the subband output is truncated into an integer number.

Let's assume that the First Stage Bit Accuracy (FSBA) and Second Stage Bit Accuracy (SSBA) correspond to the output bit accuracy of the first and second stages of the decomposition processes, respectively. In practice, each pixel of the image is represented by an integer number and therefore, truncation has to be applied to each of the decomposed subband outputs.
Figure 3.3.1 Subband Decomposition and Reconstruction

Assuming a lossless compression, the image is reconstructed by the synthesis filter. This ensures that the reconstructed image is free of quantisation and channel noise.
The evaluation of these filters is based on their Peak Signal-to-Noise Ratio (PSNR) for the luminance component (Y). The PSNR is represented by:

\[
PSNR \ (in \ dB) = 10 \log_{10} \frac{\sum_{i=0}^{m-1} \sum_{j=0}^{n-1} (255)^2}{\sum_{i=0}^{m-1} \sum_{j=0}^{n-1} [s(i,j) - s'(i,j)]^2} \tag{Eqn \ 3.3.1}
\]

where \( m \) and \( n \) are the width and the height of the image respectively, \( s(i,j) \) is the original image, and \( s'(i,j) \) is the reconstructed image.

In this section, the performance evaluation is based on the luminance component and is referred to as PSNR\_Y. Three different test images called “Carphone”, “Salesman” and “Suzie” are employed here. These images are shown in Figure 3.3.2 where each image has a spatial resolution of 352 \( \times \) 288 (known as Common Intermediate Format (CIF)).

Figure 3.3.3 presents the simulation results for the QMF 8A filter. As can be seen, the PSNR\_Y starts at about 43 dB. There is only a small change in the PSNR\_Y values when both First Stage Bit Accuracy (FSBA) and Second Stage Bit Accuracy (SSBA) are increased to 9 bits. As shown in this Figure, there is no further improvement when these bit accuracies exceed 9 bits. However, the QMF 16A filter behaves differently from the QMF 8A. In this case, the PSNR\_Y values start at about 50 dB. As the FSBA value exceeds 11 bits, the PSNR\_Y value also increases rapidly until it reaches infinity. QMF 16A filters show perfect reconstruction properties when the FSBA value reaches 12 and SSBA values exceed 13 bits (see Figure 3.3.4). This is due to the longer filter taps of the QMF 16A filter.

In the case of PRQMF and PRLPF filters, they appear to be more sensitive to the truncation errors than any of the above filters. In addition, the dynamic range of these filters is 9 bits per pixel. (The range between the minimum and maximum values of a pixel is known as the dynamic range.) Figure 3.3.5 shows that the PSNR\_Y values of
PRQMF increase rapidly as the FSBA values exceed 9 bits. For instance, “Carphone” and “Salesman” images can be perfectly reconstructed when the FSBA and the SSBA values are greater than 11 and 12, respectively. For the “Suzie” image, both FSBA and SSBA values must be greater than 10 and 11 respectively. For PRLPF filter, the PSNR Y values increase slowly as the FSBA values increase above 10 bits. In this case, the image is perfectly reconstructed when the FSBA and SSBA values exceed 11 and 13, respectively (see Figure 3.3.6).

Figure 3.3.7 shows the simulation results for the IIR filter which produces a better PSNR Y values than QMF 8A filter but more sensitive to the bit truncation errors than QMF 8A filter. The PSNR Y values starts at about 50 dB with a small improvement as the FSBA increases from 8 to 11 bits. However, there are no changes when the SSBA values are greater than 12. This is because the IIR filter has a very short filter taps.

Finally, it can be concluded that although PRQMF and PRLPF have perfect reconstruction properties, they are most sensitive to truncation errors. In addition, they have a higher dynamic range than the other filters. For short delay applications, the IIR filter is more suitable than the other filters because of its shorter filter taps. Overall, the QMF 16A filter has a better performance than the other filters. This is mainly due to the fact that this filter is more tolerant to truncation errors compared with PRQMF and PRLPF filters.
Figure 3.3.2 Test Samples Images

(a) Sample Image “Carphone”

(b) Sample Image “Salesman”

(c) Sample Image “Suzie”
Figure 3.3.3 QMF 8A Filter Simulation Results
Figure 3.3.4 QMF 16A Filter Simulation Results
Figure 3.3.5 PRQMF Filter Simulation Results
Figure 3.3.6 PRLPF Filter Simulation Results
Figure 3.3.7 IIR Filter Simulation Results
3.4 Proposed Subband Still Image Coding System

In the previous chapter, the design of the early subband still image coding system was described. From the early development, it was clear that separating the quantisation and encoding between the lowest frequency band and the other higher frequency bands was essential. This is because each of the individual subbands have different properties (see reference [26]). In [26], *B1 code* was considered to encode the position of non-zero quantised values.

However, according to [26], the main reasons behind selecting the B1 codes was ease of implementation as well as dealing with the multi-resolution nature of hierarchical subband decomposition (i.e. 7 bands, 10 bands, etc.). Under these conditions it is necessary to design a number of codebooks in accordance with the spatial resolution (i.e. horizontal resolution) of each band which is determined by the maximum runlength. For instance, let us consider a 7 bands subband coding system with an input spatial resolution of $352 \times 288$ undergoing two stages of 2-D decomposition. The output resolution of the first stage and second stage will be $176 \times 144$ and $88 \times 72$ respectively. Thus, to encode the positional information of non-zero quantised values on a line by line basis requires two codebooks with maximum runs of 176 and 88.

One solution to this problem would be to partition the individual bands into non-overlapping blocks and design a Huffman codebook with a fixed maximum run in accordance with the length of the block. Unfortunately, the runlength statistical order of the quantised bands tends to change significantly from block to block. This would make it impossible to design a near optimum codebook to encode the locations of the non-zero values for all blocks.

To overcome this limitation, DCT transformation was used prior to quantising each individual band. However, there is no doubt that the employment of DCT can be very effective in exploiting the spatial correlation which normally exists for the lowest frequency band, (note that Hybrid DPCM/DCT is also based on the same principles).
As far as the higher frequency bands are concerned, DCT may not provide a significant gain in terms of coding efficiency (compared with the lowest band) to justify the added complexity. This is due to the fact that the spatial correlation within these bands is moderately low. However, in our investigation the motivation for considering DCT (see Figure 3.4.1) for these bands is mainly to improve the behaviour of the runlength statistics which can be more systematic in the DCT domain.

To further enhance the overall coding performance the use of 3D VLC has been considered instead of B codes. As mention before (see section 2.4.3), the 3D VLC is formed by combining the three main events in the quantised DCT block which are: LEVEL (non-zero quantised values), RUN (the length of zeros between two consecutive levels, and EOB (End Of Block).

Figure 3.4.1 (see next page) shows the proposed subband still image coding scheme incorporating all the above modifications. The input image is first decomposed into four subband outputs using a 2D subband filter (SB). Each of these subbands are then transformed and quantised. The quantisation technique used here is a simple uniform quantisation method. It uses Quantisation Parameter (QP) to select an appropriate uniform quantisation step size in order to control the bit rate and picture quality. Then the quantised subbands are entropy coded using a 3D VLC codebook. For our investigation we have used the same 3D VLC codebook recommended by the recent ITU-T recommendation for very low bit rate video compression [45]. Therefore, any other international standards systems can receive and decode these subband encoded images. The same idea can be employed for still image compression when using JPEG encoding techniques.
SB = Subband Analysis Filter
DCT = Discrete Cosine Transform
Q = Quantiser
3D VLC = 3D Variable Length Encoder

Figure 3.4.1 Proposed Subband Still Image Coding System
3.5 Proposed Still Image Coding Scheme Simulation Results

In this section, the aim is to compare the efficiencies and picture qualities of the subband still image coding system using; QMF 8A & QMF 16A filters, PRQMF & PRLPF filters, and the IIR filter. The same test images have been used for these experiments (refer to Figure 3.3.2). The luminance and chrominance components of these images have been encoded using the proposed scheme as shown in Figure 3.4.1. From the previous results, it was noticed how the truncation error could affect the reconstructed image quality. To eliminate the truncation errors, the FSBA (First Stage Bit Accuracy) must be set to values greater than 14. Once the truncation error is eliminated, the quantisation error will entirely depend on the quantisation step size. This is controlled by the QP (Quantisation Parameter) which can have integer values of 2 to 10 (refer to section 2.4.2.1). The simulation results in terms of average bits per pixel versus QP are illustrated in Figure 3.5.1 to 3.5.2.

From Figure 3.5.1, the results show that the PRQMF filter requires the highest value of bits per pixel whereas the PRLPF achieves the lowest amongst all the filters. The number of bits per pixel for QMF 8A and QMF 16A filters are very close to each other. The IIR filter produces higher bits per pixel values than the QMF 8A and QMF 16A filters. For very low quality image coding, all the filters tend to produce the same number of bits per pixel, except for the PRLPF filter which provides the lowest value. At the same time, as shown in Figure 3.5.2, the PRLPF filter provides the worst PSNR_\text{Y} value. This is because the PRQMF tends to lose more information due to the quantisation (i.e. forcing more lower amplitude values to zero) which would consequently result in lower bits per pixel but higher PSNR_\text{Y}.

The PRQMF filter however, tends to have higher bits per pixel and higher PSNR_\text{Y} values when the QP value is less than 4. When the value of QP is greater than 4, the PSNR_\text{Y} values of the PRQMF tend to be lower than the rest which indicates its sensitivity to the coding noise.
Overall, the results of the QMF 16A filter indicate the highest PSNR_Y value and the second best bits per pixel results after the PRLPF filter. Both the QMF 8A and IIR filters produce similar PSNR_Y values except when the QP is less than 4. However, the QMF 8A filter performs better for those images that contain high frequency components such as the "Salesman" image.

Finally, it can be concluded that the PRQMF and PRLPF despite their perfect reconstruction properties, cannot perform well in the presence of quantisation errors. As far as the IIR filter is concerned, its performance in terms of bits per pixel does not appear to be as good as the QMF 8A filter even though it produces a better PSNR_Y value. Overall, the QMF 16A filter produces the best results followed by the QMF 8A filter particularly for images containing high frequency components.
Figure 3.5.1 Subband Still Image Coding Scheme Bits Comparison
Figure 3.5.2 Subband Still Image Coding Scheme PSNR\_Y Comparison
Chapter 4

Review Of Video Image Coding Systems

4.1 Introduction

A video signal is commonly defined as a time sequence of still images. Therefore, most of the still image compression ideas can be applied to video image signals. Predictive Coding for instance is considered one of the most popular techniques which can be easily applied to video coding applications [36-37].

In this case, the predictive coding technique is applied along the temporal direction and therefore is known as interframe prediction. During the video coding process, the first frame is encoded in the spatial domain which is known as intraframe coding. The intraframe coded first frame will then be used as the reference to encode the next frame in the sequence. The basic principle of Differential Pulse Code Modulation (DPCM) using interframe prediction will be described in the following section.

In addition, this chapter also looks into Hybrid DPCM/DCT video coding system. Then it will discuss the International Standard for video coding. Finally, Hybrid DPCM/Subband video coding system will be discussed. The early development of subband models are highlighted in the last section.
4.2 Interframe Predictive Coding System

Video image sequences contain high correlations in a temporal domain, particularly between current and adjacent previous frames. In this approach the prediction error is formed by taking the difference between the current frame and the reconstructed previous frame. The basic structure of the interframe predictive coding and decoding system [36] is illustrated in Figure 4.2.1.

Figure 4.2.1 Interframe Predictive Coding System
The basic equations describing DPCM are,

\[ e(n) = x(n) - x'(n) \]  \hspace{1cm} (Eqn. 4.2.1)
\[ u(n) = e(n) + q(n) \]  \hspace{1cm} (Eqn. 4.2.2)
\[ v(n) = u(n) + w(n) \]  \hspace{1cm} (Eqn. 4.2.3)
\[ y(n) = v(n) + x'(n) \]  \hspace{1cm} (Eqn. 4.2.4)

where

- \( e(n) \) is the prediction error
- \( q(n) \) is the quantisation error
- \( u(n) \) is the quantised prediction error
- \( v(n) \) is the received prediction error
- \( w(n) \) is the channel noise
- \( x(n) \) is the input image
- \( x'(n) \) is the predicted image
- \( y(n) \) is the reconstructed image (output image)

The output \( y(n) \) is approximately equal to input \( x(n) \) provided that the quantisation error \( q(n) \) is very small and the channel is no noise (i.e. \( w(n) = 0 \)). The output \( y(n) \) is approximately equal to:

\[
\begin{align*}
y(n) &= v(n) + x'(n) \\
&= u(n) + w(n) + x'(n) \\
&= e(n) + q(n) + w(n) + x'(n) \\
&= x(n) - x'(n) + q(n) + w(n) + x'(n) \\
&= x(n) + q(n) + w(n) \\
&= x(n)
\end{align*}
\]  \hspace{1cm} (Eqn 4.2.5)

The performance of interframe DPCM can be improved significantly by interpreting the displacement of a moving object. Figure 4.2.2 shows the difference between the prediction error\(^1\) with and without motion compensation. These show that motion

\(^1\) Prediction error was offset by 128 for display.
estimation techniques can further reduce the temporal redundancy for motion pictures. In practice, most of the video image sequences contain motion which are due to the translational movement, camera panning and zooming. Therefore, it is very important to have a good predictor within the DPCM loop so as to precisely predict the current frame. The overall performance of a DPCM coding system also depends on the quantiser design and entropy encoder for transmitting the prediction error signal. A detailed discussion about motion estimation, quantisation, and entropy coding will be covered in later sections.

![Figure 4.2.2 Interframe DPCM Prediction Error Image](image)

(a) Previous frame  
(b) Current frame

(c) Prediction error without motion estimation  
(d) Prediction error with motion estimation

Figure 4.2.2 Interframe DPCM Prediction Error Image
4.2.1 Motion Estimation Technique

Motion estimation involves measuring the temporal motion activities in the video sequences. These activities are due to translations, rotations, zooming and panning. There are number of motion estimation techniques such as the Pel Recursive [41] and Block Motion Estimation [42]. The most widely used motion estimation technique is the Block Matching Motion Estimation which will be discussed here.

4.2.1.1 Block Matching Method

In block matching motion estimation the coding (current) frame is partitioned into small non-overlapping blocks of size $m \times n$. It is assumed that all the pixels within this non-overlapped block have the same displacement vector and the motion is purely translational. This motion vector is estimated by searching through a larger block (search window of size $m+2u \times n+2v$) which is centred at the same location on the previous frame (see Figure 4.2.3). The best matching block will give the minimum error and the motion vector is therefore taken from this location. The matching of the blocks can be quantified according to various criteria including Sum Absolute Difference (SAD), Sum Squared Difference (SSD), and Pel Difference Classification (PDC) [43].

(a) Select the search window
These criteria are outlined as followed:

**Sum Absolute Difference (SAD)**

\[
SAD(x, y) = \sum_{x=0}^{m-1} \sum_{y=0}^{n-1} |s(i, j, k) - s(i-x, j-y, k-1)|
\]  
(Eqn. 4.2.1)

**Sum Squared Difference (SSD)**

\[
SSD(x, y) = \sum_{x=0}^{m-1} \sum_{y=0}^{n-1} [s(i, j, k) - s(i-x, j-y, k-1)]^2
\]  
(Eqn. 4.2.2)
**Pel Difference Classification (PDC)**

In the Pel Difference Classification method [43], each pixel in the block is classified as one of two states:

1. matching pixel
2. mismatching pixel

A threshold $t$ is selected to perform the above classification:

$$T(i, j, x, y) = 1, \quad \text{if} \quad |s(i, j, k) - s(i - x, j - y, k - 1)| \leq t$$

$$= 0, \quad \text{otherwise} \quad \text{(Eqn. 4.2.3)}$$

$T(i, j, x, y)$ is the binary representation of pixel difference and its value of either one or zero, corresponds to a matching or mismatching pixel, respectively.

The number of matching pixels are given by $G(x, y)$, which can be defined as follows:

$$G(x, y) = \sum_{x=0}^{\infty-1} \sum_{y=0}^{\infty-1} T(i, j, x, y) \quad \text{(Eqn. 4.2.4)}$$

The value $G(x, y)$ represents the number of matching pixels that exist between the current block and the block on the previous reference frame that was shifted by $i$ pixels and $j$ lines. The largest value of $G(x, y)$ was found by searching through a search window. This gives the best match. Thus

$$G_m(d_x, d_y) = \max_{i, j} \{G(x, y)\} \quad \text{(Eqn. 4.2.5)}$$

where

- $i, j$ are the spatial coordinates,
- $x, y$ are the motion vector spatial coordinates,
- $d_x, d_y$ are the components of the best estimated displacement vector,
is the time reference for the current frame
\( k-J \) is the time reference for the previous frame
\( s(i, j, k) \) is the intensity of the current frame,
\( s(i, j, k-1) \) is the intensity of the previous frame.

The performance of PDC is better than the other matching methods as mentioned above. In this method, the matching process is reduced to a binary level which consequently simplifies the computational complexity, as described by Gharavi [43] in 1990. However, the SAD method has been adopted as an international standard because of its simplicity.

4.2.1.2 Half Pixel Block Matching Method

A half pixel searching window is created by a bilinear interpolation technique [44-45] as shown in Figure 4.2.4. Matching is now done by first using the integer searching window and then using half pixel searching to find the best block match. This method has the advantage of producing more accurate prediction than the integer pixel block matching method. However, this method requires extra computational complexities to create the half pixel searching window. Therefore, for each of the reference blocks the search begins with an integer pixel block first. Then the motion vector for the best match is used to carry out further half pixel searching. This searching will carry on until the best match block is found. The half pixel block matching technique has been considered here to carry out the experiments.

![Figure 4.2.4](image)

Figure 4.2.4 Half Pixel Prediction by Bilinear Interpolation
4.3 Hybrid DPCM/DCT Video Image Coding System

The concept of Hybrid DPCM/Transform Coding was first introduced by Habibi [38] for still image coding. This concept was later extended by combining interframe coding and DCT coding to increase coding efficiency. Figure 4.3.1 shows a simple block diagram of a Hybrid DPCM/DCT video coding system.

As shown in the above figure, the prediction error which is the difference between the input signal $x(n)$ and the predicted signal $x'(n)$, is obtained from the interframe prediction $P$. This predictor performs motion estimation and compensation using the previous reconstructed frame.

The predicted error $e(n)$ is then transformed by a DCT transform matrix $T$ and quantized in a zig-zag order by a quantiser $Q$. The quantised coefficients $u(n)$ are subsequently entropy coded and transmitted through the channel. At the receiver end, the entropy decoded signal is inverse DCT transformed and dequantised to the prediction error $v(n)$. The reconstructed input signal $y(n)$ is obtained by adding the prediction error and the predicted signal together. It was based on this model that all the existing International Standards (i.e. H.261, MPEG-1, MPEG-2, H.263) for video compression were established.

Figure 4.3.1 Basic Hybrid DPCM/DCT Video Coding System
4.4 International Standard Video Image Coding Systems

The first video coding standard was the ITU-T H.261 [39]. After completing the recommendation work for ITU-T H.261, the Expert Group began to consider video coding for Broadband ISDN (B-ISDN). Conscious of the fact that MPEG-1 had turned out to be similar to ITU-T H.261 but not compatible with it at the bit level, many concluded that there would be beneficial if the next ISO and ITU standards were identical. This was formally agreed and the MPEG Video meeting became a joint session to produce a common text. The ITU-T Study Group XV has submitted the ITU-T Recommendation H.262 in 1993. The identical text is also published as MPEG-2. Recently, ITU-T H.263 was developed for low bit rate video applications. A new MPEG-4 Standard has been proposed at the MPEG meeting and has been scheduled to produce a draft specification in 1997.

4.4.1 ITU-T H.263 Video Image Coding Algorithm

ITU-T Recommendation H.263 defines a video codec for low bit rate communications. The video codec test model, TMN-5 (Test Model Near-term) was prepared in 1993. This recommendation specified a coded representation that can be used for compressing the moving picture components of audio-visual services at low bit rates. The draft ITU-T Recommendation H.263 was published in 1995 [44-45].

ITU-T H.263 algorithms are also based on hybrid DPCM, motion compensation, quantisation, discrete cosine transform and entropy coding techniques. A block diagram of the ITU-T H.263 encoder is shown in Figure 4.5.1.
The source coder operates on a similar video input format as ITU-T H.261. Furthermore, it can also support sub-QCIF, 4CIF, and 16CIF. For each of these picture formats, the luminance sampling structure is \( x \) pixels per line, \( y \) lines per picture in an orthogonal arrangement. A summary of all the picture format for ITU-T H.263 is shown in Table 4.4.1.

<table>
<thead>
<tr>
<th>Picture Format</th>
<th>number of pixels for luminance ((x))</th>
<th>number of pixels for luminance ((y))</th>
<th>number of pixels for chrominance ((x))</th>
<th>number of pixels for chrominance ((y))</th>
</tr>
</thead>
<tbody>
<tr>
<td>sub-QCIF</td>
<td>128</td>
<td>96</td>
<td>64</td>
<td>48</td>
</tr>
<tr>
<td>QCIF</td>
<td>176</td>
<td>144</td>
<td>88</td>
<td>72</td>
</tr>
<tr>
<td>CIF</td>
<td>352</td>
<td>288</td>
<td>176</td>
<td>144</td>
</tr>
<tr>
<td>4CIF</td>
<td>704</td>
<td>576</td>
<td>352</td>
<td>288</td>
</tr>
<tr>
<td>16CIF</td>
<td>1408</td>
<td>1152</td>
<td>704</td>
<td>576</td>
</tr>
</tbody>
</table>

Table 4.4.1 ITU-T H.263 Picture Formats
The compressed ITU-T H.263 video bit stream contains four layers which is the same as ITU-T H.261. Each picture frame is partitioned into 8 pixels by 8 lines image block. Each MacroBlock (MB) consists of 4 luminance blocks (Y), 2 chrominance blocks (Cb & Cr) at the same location as shown in Figure 4.4.2.

![Figure 4.4.2 MacroBlock Structure](image)

However, the Group Of Block (GOB) arrangement for the picture formats are different from ITU-T H.261. The summary of the GOB and MB arrangements for the picture formats is shown in Table 4.4.2.

<table>
<thead>
<tr>
<th>Picture Format</th>
<th>number of group of block (GOB) for a picture</th>
<th>number of macroblock (MB) for a group of block (GOB)</th>
</tr>
</thead>
<tbody>
<tr>
<td>sub-QCIF</td>
<td>6</td>
<td>8</td>
</tr>
<tr>
<td>QCIF</td>
<td>9</td>
<td>11</td>
</tr>
<tr>
<td>CIF</td>
<td>18</td>
<td>22</td>
</tr>
<tr>
<td>4CIF</td>
<td>18</td>
<td>88</td>
</tr>
<tr>
<td>16CIF</td>
<td>18</td>
<td>352</td>
</tr>
</tbody>
</table>

Table 4.4.2 Group of Block and Macroblock Arrangement
The ITU-T H.263 encoder structure is still based upon the Hybrid DPCM/DCT system. However, half pixel accuracy motion compensation is employed in this system for motion estimation and the prediction block is formed from the previously transmitted macroblock or block. Each $8 \times 8$ block contained in the macroblock is DCT transformed and clipped as a 12 bit data with the range $-2048$ to $2047$. These 64 pixels blocks of quantised and transformed coefficients are reordered according to the zig-zag scanning pattern. Three-dimensional variable code length (3D VLC) is adopted to encode the transformed coefficient block. There are four other options available to improve this system. These are the Advanced prediction mode, Unrestricted motion mode, Syntax-based arithmetic coding mode and PB-frames mode.

1. **Advanced prediction mode** is one of the options for P-frames (Predicted frame) encoding. Here, instead of using one motion vector per MB, four more motion vectors are used. This provides a better prediction and thus reduces the number of bits that are required to encode the transformed coefficients. Nevertheless, this will increase the bits required to encode the motion vectors.

2. **Unrestricted motion vector mode** allows the motion vector to point outside the picture. In this mode, the pixels on the edge are used to predict the non existing pixels which are outside the edge. This will provide a virtual search window for the current block to search outside the picture. In this mode a significant gain is achieved if there is a movement involving an object entering or leaving the picture and also panning situations.

3. **Syntax-based arithmetic coding mode** employs an arithmetic coding process to replace the Variable Length Code (VLC). Syntax-based Arithmetic Coding (SAC) is an algorithm which encodes the symbols into a fractional number [59]. This is different from the VLC process where each symbol is encoded as a integer number. The SAC method of coding can lead to a reduction in the bit rate. The disadvantages of SAC are the complexity in encoding the data and a low tolerance for errors.
4. **PB-frames mode** allows the two pictures to be coded as one unit (see Figure 4.4.3). These two pictures (frames) are the forward predicted frame (P-frame) and the bi-directional predicted frame (B-frame). B-frame is predicted from both the previous P-frame and the P-frame currently being decoded. This effectively doubles the temporal resolution of the picture without greatly increasing the bit rate. However, PB-frame does not work as well as B-frame in MPEG because there are no separate bidirectional vectors in ITU-T H.263. The advantage of ITU-T H.263 over MPEG is that it requires much less overhead which is useful in low bit rate transmission.

![Figure 4.4.3 PB Frame Arrangement](image-url)
4.4.2 Other Developments in the Standard Bodies

The first videophone standard available is the ITU-T H.261 [46-50]. It is primarily intended for bit rates in the range of 40 kbits/s to 2 Mbits/s range. ITU-T Recommendation H.261 defines a video coding scheme for digital audiovisual services by the ITU-T Study Group XV. This video transmission standard was established for ISDN H0-channel (384 kbits/s) and later for B-channel (64 kbits/s). The development of ITU-T H.261 went through several stages. At the beginning, the target transmission rate was from 384 kbits/s up to 1920 kbits/s. Later, the transmission rate was reduced to the range of 64 kbits/s and 320 kbits/s. However, by late 1989, the final CCITT recommendations were made for the range of 64 kbits/s up to 1920 kbits/s. Therefore, ITU-T H.261 is also known as $p \times 64$ codec, where $p$ is between 1 and 30.

The Motion Picture Expert Group (MPEG) was formed in 1988 to establish a standard for the compression of digital audio and video transmissions. The MPEG-I [51-52] is the first phase video compression standard aimed primarily at digital storage media and transmission media, at bit rates of 1 to 1.5 Mbits/s. This was chartered by the ISO/IEC JTC1/SC29/WG11 to standardize a coding scheme. This scheme is well suited for a wide range of applications such as, Compact Disk Read-Only Memory (CD-ROM), Digital Audio Tape (DAT), Cable Television (CATV), telecommunication networks, and digital video broadcasting. The MPEG-I standard was published in 1993 as ISO/IEC 11172 (Coding of moving pictures and associated for digital storage up to about 1.5 Mbits/s). Part 1 of this standard describes the system, which includes information about the synchronization and multiplexing of video and audio streams. Parts 2, 3 and 4 describe video, audio and conformance testing respectively.

The MPEG-2 [53] is the second phase video compression standard which is aimed at coding above 2 Mbits/s. Preparation of the MPEG-2 standard started in 1991 and provides a solution for applications that are not successfully covered by MPEG-1. Due to the wide range of applications covered by MPEG-2, work on MPEG-3 was dropped in July 1992. To achieve this wide range of applications, the syntax is stipulated by the
means of profile and level\textsuperscript{2}. A text identical to that of MPEG-2 was published as ITU-T Recommendation H.262. Recently, the MPEG-2 standard has been approved by the Advanced Television System Committee (ATSC) as a Digital High Definition Television (HDTV) [54-55] Standard in the United States.

Formulation of a new MPEG-4 [56] Standard was begun at the MPEG meeting in Brussels in September, 1993. It is scheduled to produce a draft specification in 1997. The primary target of this standard is very low bit rate applications. The MPEG-4 standard supports a wide range of applications such as videophone over analogue telephone lines, sign language captioning, mobile audiovisual communications and interactive multimedia communications.

\textsuperscript{2} A summary of the profile and level features are illustrated in appendix A.
4.5 Hybrid DPCM/Subband Video Image Coding System

An alternative video coding system to the previous approach is the Hybrid DPCM/Subband video coding system [40]. Two distinct interframe subband models were developed based on the interframe predictive coding technique. In the first model (Subband Model I), the subband decomposition is performed on the motion compensated prediction error signal (see Figure 4.5.1). In this case, the subband filter is located inside the DPCM loop. In the second model (Subband Model II) each video frame is first split into four subbands before they can be encoded (see Figure 4.5.2).

The development of these two models is based on the interframe predictive technique. Both of these models have a multi-layer structure. However, the limitation of Subband Model I is that it is not suitable for multi-resolution video transmission. Therefore, Subband Model II was designed to support a multi-resolution transmission application.

4.5.1 Subband Model I System

The basic block diagram for Subband Model I System [40] is shown in Figure 4.5.1. This coding system is based on interframe predictive DPCM loop. Each given frame is predicted by employing the motion estimation technique. Here, the prediction error is represented by the difference between the current frame and the previous reconstructed frame. The prediction error is decomposed into four different bands and then quantised. The quantised subbands are then encoded and transmitted.
Figure 4.5.1 Subband Model I System

SB = Subband Analysis Filter
SB\(^{-1}\) = Subband Synthesis Filter
Q = Quantiser
Q\(^{-1}\) = Inverse Quantiser
MC = Motion Compensation
ME = Motion Estimation
FM = Frame Memory
4.5.2 Subband Model II System

The basic Subband Model II System [40] is also based on the interframe predictive DPCM method which is illustrated in Figure 4.5.2. In this case, the video input signal is first decomposed into four subbands. Each of the decomposed subbands is then applied to a separate DPCM loop. The motion vector is found from the lowest frequency band. The other three subbands can use the same motion vector to perform motion compensation. The advantage of this model is its flexibility for multi-resolution video transmission.

Figure 4.5.2 Subband Model II System
Chapter 5

Proposed Subband Video Coding System

5.1 Introduction

The Subband video coding approach has many important features such as scalability, compatibility, and inherent multi-resolution transmission capability which makes it suitable for modern communication applications.

This chapter looks into the development of the subband video coding system to exploit the above advantages. The subband video coding system introduced here is based on the previously mentioned Subband Model I. In this model, the subband decomposition is performed on the motion compensated prediction error signal. The individual bands are then encoded using dead zone quantisation and a combination of VLC and B1 runlength coding.

As mentioned in section 3.4 in this thesis, a different but more efficient approach has been considered to encode the decomposed prediction error signals. This proposed scheme will be discussed in the following section.
5.2 Proposed Subband Video Coding System

In these investigations the main objective has been to devise a new approach to enhance the coding performance of the interframe hybrid subband model I technique. The proposed subband scheme can be divided into five modules. These modules are interframe predictive DPCM loop (with motion compensation), subband filtering, DCT transformation, Quantisation and Entropy Coding. Different types of subband filters such as QMF 8A, QMF 16A, PRQMF, PRLPF and IIR have been considered. For the ease of implementation, the same DCT transformation, Quantisation and Entropy Coding are adopted from the ITU-T H.263 video coding system.

The advantage of this model is that important information such as the lowest frequency band can be transmitted through a high priority channel with better error protection. This multi-channel feature is suitable for mobile communication applications [3]. The basic block diagram of the proposed Subband Video Coding System is illustrated in Figure 5.2.1. In this scheme, the video input signal is first predicted by a motion estimator using a half pixel block matching motion estimation technique. The prediction error signal is then formed by subtracting the predicted signal from the video input signal. This signal is subsequently decomposed into four different bands using subband filtering. Each band after DCT transformation is quantized by a uniform quantiser. The quantised transform coefficients are scanned (i.e. zig-zag) and then entropy coded using 3D VLC coding.
Figure 5.2.1 Proposed Subband Video Coding System
5.3 Proposed Subband Video Coding System Simulation Results

The proposed Subband Video Coding System was designed and simulated. The aim of this simulation is to compare the subband system using QMF 8A, QMF 16A, PRQMF, PRLPF and IIR filters. Three different video sequences are employed in this simulation. They are the "Carphone", "Salesman" and "Suzie" sequences (refer to section 3.4). Each of them consists of 120 frames.

The ITU-T H.263 Standard Encoder (refer to section 4.4.1) was first simulated using a half pixel motion estimation as a reference comparison. The size of search windows was set to ±31. Neither PB frame nor the syntax-based arithmetic coding method were used for P-frame transmission. No intraframe blocks are used for interframe encoding. The encoder was set to operate in the variable bit rate mode. Thus buffer control mode was switched off. In addition, the Quantisation Parameter (QP) was set to 10 (default setting in ITU-T H.263). The simulation results are shown in Table 5.3.1.

<table>
<thead>
<tr>
<th>Sequences</th>
<th>Average Bit Rate (kbits/s)</th>
<th>PSNR_Y (in dB)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Carphone</td>
<td>167.33</td>
<td>37.75</td>
</tr>
<tr>
<td>Salesman</td>
<td>91.40</td>
<td>35.79</td>
</tr>
<tr>
<td>Suzie</td>
<td>150.37</td>
<td>38.23</td>
</tr>
</tbody>
</table>

Table 5.3.1 ITU-T H.263 Simulation Results

To make a fair comparison, it was necessary to operate under identical conditions. Therefore, the parameters for the proposed subband scheme was set to produce the same PSNR_Y value as the ITU-T H.263 Encoder. However, it was difficult to obtain identical PSNR_Y values for the ITU-T H.263 Standard Encoder and the proposed systems. Nevertheless Quantisation Parameters were adjusted to produce PSNR_Y values as close as possible to each other. The simulated results are shown in Figures 5.3.1 to 5.3.3.
Figures 5.3.1 (a) to 5.3.3 (a) show that the PRLPF filter has the highest bit rate amongst all the other filters. This value was almost double that of the QMF 16A filter. Both PRQMF and PRLPF filters have slightly better PSNR\textsubscript{Y} values than the rest but at the expense of much higher bit rates. Therefore, these two filters will not be further considered.

As shown in Figure 5.3.1 (b) to 5.3.3 (b) QMF 16A and QMF 8A filters provide the lowest bit rates. The results of the QMF 16A filter show a better PSNR\textsubscript{Y} performance than the QMF 8A filter. However, for a highly active sequence such as “Salesman”, the QMF 8A filter has the advantage over the QMF 16A filter.

From these results, the performance of the IIR filter is not as good as the QMF 8A filter. The IIR filter not only has higher bit rates but also has lower PSNR\textsubscript{Y} values than the QMF 8A filter.

In the case of the QMF 8A and QMF 16A filters, they perform better than the other filters. A comparison between these QMF filters and ITU-T H.263 is made and shown in Figure 5.3.4 to 5.3.6. The results show that both QMF filters performed as well as ITU-T H.263. In addition, they have a lower bit rate than ITU-T H.263.

It can be concluded that the Lattice Structure filter is not as suitable as originally anticipated. Overall, the QMF 16A filter has the best performance followed by the QMF 8A filter. However, in the terms of subjective evaluation, QMF 8A filter has the best picture quality. Both QMF 8A and QMF 16A filters will be used to carry out the investigation and implementation in the next chapter.
Chapter 5  
Proposed Subband Video Coding System

Figure 5.3.1 Proposed Subband Scheme for Carphone Comparison
Figure 5.3.3 Proposed Subband Scheme for Suzie Comparison
Figure 5.3.4 Subband Scheme Versus ITU-T H.263 for Carphone Comparison
Figure 5.3.6 Subband Scheme Versus ITU-T H.263 for Suzie Comparison
Chapter 6

Subband Multi-Resolution Transmission System

6.1 Introduction

With the emergence of the new digital transmission technologies, visual communication services are expected to move toward high quality video applications. In addition, the demand for mobile communications is rapidly increasing. Therefore, it is beneficial to design a universal coding method which can provide a wide range of video quality from low spatial resolution to high resolution video representation.

In order to achieve this, a subband coding technique has been considered here. Although a simple subband model in the previous chapter has been successfully applied to multi-channel transmission, features such as multi-resolution transmission (i.e. spatial scalability) and compatibility with existing coding standards cannot be incorporated [2]. Therefore, subband multi-channel coding schemes with multi-resolution representation may prove to be the best alternative as the demand for higher quality multimedia increases.

A simple multi-resolution video transmission system will be discussed in the following section. A more efficient approach has been considered to overcome the scalability problem. This proposed scheme will be discussed and finally its performance will be evaluated.
6.2 Multi-Resolution Transmission System

The basic concept of multi-resolution was originally designed for progressive image transmission and is based on the Laplacian Pyramid Coding System [57]. Each input image is represented by a low resolution image (i.e., a quarter size of the input image) and a prediction error image. This low resolution image $X_l$ is obtained from the output of a low pass filter (LPF) and a decimator. The prediction error image $E_o$ is the difference between the original input image $X_0$ and a prediction based on the low resolution image. At the receiver side, the low resolution image $X_l$ is decoded and interpolated. The reconstructed image is obtained by adding the prediction error $E_o$ and interpolated image $X_l$ together. A simple block diagram of Laplacian Pyramid Coding System is illustrated in Figure 6.2.1.

\[ \text{(a) Encoder} \quad \text{(b) Decoder} \]

Figure 6.2.1 Laplacian Pyramid Coding System

The advantage of this approach is that it provides an independent multi-resolution output in a multi-layer fashion. This approach not only can transmit high resolution video but also low resolution video signals. However, the main drawback of this system is its inefficiency because of the increased number of pixels required to encode the image (i.e., the total of the high and low resolution signals is increased by 50%).
6.2.1 Subband Multi-Resolution Transmission System

As mentioned in chapter 4, Subband Model II coding scheme has been shown to be suitable for multi-resolution. However, the disadvantage is that its reconstructed picture quality is much worse than Subband Model I (refer to section 4.5.1). In addition, extra DPCM loops are required to encode the other higher frequency bands. These problems and a new multi-resolution structure have been discussed in reference [58].

In this chapter a new coding approach based on the new multi-resolution Subband Model I is proposed. The advantage of this method is due to its superior performance compared with the Model II and the Pyramid Coding System. In addition, the proposed scheme provides full compatibility with ITU-T H.263 standard at its lower resolution.

The Structure of this Subband Multi-Resolution System consists of only two layers. This system accepts video input in CIF format. For the first layer the CIF input is low pass filtered (LPF) to reduce the spatial resolution by a factor of 4 (i.e. QCIF Format). The QCIF format image is then encoded by an ITU-T H.263 encoder which produces a B1 stream (see Figure 6.2.2) that can be received and decoded independently by ITU-T H.263 decoder. The video input in CIF format is also sent to the Hybrid DPCM/DCT loop which produces the motion compensated prediction error. This prediction error is then decomposed into four bands and sent to encoders (except for the lowest frequency band). At the local decoder the first band is replaced by the difference between the decoded ITU-T H.263 signal and the prediction signal after low pass filtering. The other three bands will be DCT transformed and quantised using uniform quantisers. These quantised transform coefficients will be encoded by 3D VLC codebooks. A block diagram of this system is shown in Figure 6.2.2.
Figure 6.2.2 Proposed Subband Multi-Resolution Scheme
The above approach can be further extended to support any number of different resolutions for video distributions. Figure 6.2.3 shows an example of a five layer Subband Multi-Resolution System.

![Diagram of a five layer Subband Multi-Resolution Transmission System]

Figure 6.2.3 Five Layer Subband Multi-Resolution Transmission System
This approach splits the high quality input video signal into spatial layers, $B_1$, $B_2$, $B_3$, ..., $B_k$, by means of subband filtering. The decomposed layers can be grouped together with an appropriate proportion to form the basic layer $L_1$ and the contribution layers $C_1$, $C_2$, $C_3$, ..., $C_{k-1}$, where $L_1$, $L_1 + C_1$, $L_1 + C_1 + C_2$, ..., $L_1 + C_1 + C_2 + C_3 + ... + C_{k-1}$ represent quality grades $Q_1$, $Q_2$, $Q_3$, ..., $Q_k$, respectively. After being individually coded, the layers are made available to the network for independent distribution.

In Figure 6.2.3 the five level video are represented in terms of their spatial resolutions by $SQ_1$, $SQ_2$, $SQ_3$, $SQ_4$, and $SQ_5$, which in turn correspond to quality grades $Q_1$, $Q_2$, $Q_3$, $Q_4$, and $Q_5$ respectively. The relationships between $SQ_1$, $SQ_2$, $SQ_3$, $SQ_4$ and $SQ_5$ are given by:

$$SQ_1 = \frac{SQ_2}{2^2} = \frac{SQ_3}{4^2} = \frac{SQ_4}{16^2} = \frac{SQ_5}{256^2}$$

(4.1.1)

where,

$SQ_1 = 88 \times 72$

$SQ_2 = 176 \times 144$

$SQ_3 = 352 \times 288$

$SQ_4 = 704 \times 576$

$SQ_5 = 1408 \times 1152$

In Figure 6.2.3, the highest grade video $Q_5$ is encoded by the fifth layer encoder (DPCM/DCT loop). Inside this loop the prediction error (motion compensation) signal is decomposed into four bands so that each band has the same resolution as the fourth grade video $Q_4$. The lowest frequency of the fifth layer is not transmitted. Instead it will be replaced by the decoded fourth layer signal $Q_4$. The replacement is done by transmitting the difference between the decoded fourth layer signal $Q_4$ and the lowest band prediction error signal from the fifth layer. The rest of the other three higher band signals are encoded and they form the contribution layer signal $C_4$. 

102
The input signal grade $Q_4$ of the fourth layer encoder (refer to Figure 6.2.3) is obtained from the LPF filter. The result of this filtering will reduce the spatial resolution by a factor of 4 from the $Q_5$ grade signal. In the same manner, the lowest frequency of the fourth layer encoder is also not transmitted but it is replaced by the decoded third layer signal. The input signal of grades $Q_3$ and $Q_2$ are used in the same manner to reduce the spatial resolution. The first layer signal $Q_1$, is obtained after $Q_2$ is passed through the lowest frequency subband filter. Finally, the $Q_1$ grade signal is encoded by the first layer encoder which is a hybrid DCT/DPCM for compatibility with ITU-T H.263 Standard.

Each coded layer is then sent to the channel and the receiver is allowed to select the individual layers. Once the selection is made the receiver will decode and reconstruct the video signal with different grades. Except for the first layer encoded signal $L_1$, all other layer signals are multiplexed together to form contribution layers $C_1$, $C_2$, $C_3$, $C_4$ and $C_5$ with their bit rates at $BC_1$, $BC_2$, $BC_3$, $BC_4$ and $BC_5$, respectively.
6.3 Subband Multi-Resolution Scheme Simulation Results

The aim of this simulation is to compare the ITU-T H.263 encoder with the proposed model. Frame 57 to 87 of the "Salesman" video sequence were used to carry out this experiment. The reason for choosing this part of the video sequence is because it contains a large number of movements. This will provide the worst encoding scenario for the video coding system.

The same parameters that were used in section 5.3, will be used here for the ITU-T H.263. However, the set up for the proposed model will be slightly different because it has a local ITU-T H.263 encoder in the basic layer. Therefore, the search window for the first layer and second layer must be set to ±15 and ±31 respectively.

It is difficult to have the same PSNR_Y value for comparison because each system has its own unique characteristics. The QP values for the basic layer and second layer of the subband multi-resolution scheme (uses QMF 8A and QMF 16A filters) were adjusted manually until the PSNR_Y value was close to the ITU-T H.263 system. For intraframe, the QP value for the basic layer and second layer was found to be equal to 4 and 7 for the QMF 8A respectively. Also the QP value for both the basic layer and second layer was set to 5 for QMF 16A. However, for interframe the QP value for both the basic and second layer was set to 6 for QMF 8A and QMF 16A.

Figure 6.3.1 shows that both QMF 8A and QMF 16A filters perform better than ITU-T H.263 for intraframe coding. Both the ITU-T H.263 encoder and the multi-resolution scheme have similar PSNR_Y values but the multi-resolution scheme has a lower bit rate than the ITU-T H.263. Figure 6.3.2 shows the reconstructed intraframe image. Figure 6.3.3 shows the enlarged block of ITU-T H.263 and the multi-resolution scheme. It was noted that with ITU-T H.263 blocking effect was more obvious than in the multi-resolution scheme. Furthermore, the results show that multi-resolution using QMF 16A filter performs better than QMF 8A.
For interframe coding, the ITU-T H.263 model has a slightly better performance than the multi-resolution which uses QMF 8A and QMF 16A filters. Figure 6.3.1 (b), shows that their bit rates are very similar. Figure 6.3.2 (c) shows that the QMF 8A has a better PSNR_Y value than QMF 16A. Figure 6.3.4 (b) & (c) shows the patches and mosquito effect. The enlargement of this effect is shown in Figure 6.3.5 (b) & (c). In the next section, the loop filter from ITU-T H.261 will be employed into the multi-resolution scheme to eliminate the patches and mosquito effect.
Figure 6.3.1 Proposed Multi-Resolution Scheme for Salesman Comparison
Figure 6.3.2 Reconstructed Salesman Frame 57 Comparison
Figure 6.3.3 Enlargement of Reconstructed Salesman Frame 57
Figure 6.3.4 Reconstructed Salesman Frame 87 Comparison
Figure 6.3.5  Enlargement of Reconstructed Salesman Frame 87
6.4 Subband Scheme with Loop Filter

The Subband Multi-Resolution Transmission System has been constructed and investigated in the previous section. However, the video image output shows some patches and mosquito effect during the reconstruction. Therefore, the ITU-T H.261 [47] loop filter has been implemented in this subband multi-layer transmission system. This loop filter is placed at locations LF1, LF2 and LF3 as shown in the Figure 6.4.1.

![Figure 6.4.1 Subband Multi-Resolution Scheme with Loop Filter](image_url)
ITU-T H.261 loop filter is a modified version of a two dimensional spatial filter which operates on pixels within a predicted 8 by 8 image block. The loop filter is separable into one-dimensional horizontal and vertical functions. Both functions are non-recursive with coefficients of $1/4$, $1/2$, $1/4$, except at block edges where one of the tap coefficients will fall outside the block. In such cases the 1-D filter coefficients are changed to 0, 1, 0. Full arithmetic precision is retained by rounding to 8 bit integer values at the 2-D filter output.
6.5 Subband Scheme with Loop Filter Simulation Results

The aim of this simulation is to evaluate the effect of the loop filter inside the Subband Multi-Resolution Transmission System when using the QMF 8A filter. A Loop Filter is used to filter the whole image. It does not have a switch like that of the ITU-T H.261, to filter a specific macroblock. The ITU-T H.261 loop filter is placed at three different locations inside the multi-resolution scheme (refer to figure 6.4.1). The parameters used here are identical to those in section 5.3. Three different video sequences of test samples are employed and they are the “Carphone”, “Salesman” and “Suzie” sequences. Each of them consists of 60 frames. The simulation results are shown in Figures 6.5.1 to 6.5.3.

The Loop Filter at location 1 (LF1 in Figure 6.4.1) has the worst performance compared to the others in terms of PSNR_\_Y values mainly because the prediction error is filtered by the loop filter. This results in a lower prediction error signal and therefore requires a less bit rate. The Loop Filter at location 2 produces the highest PSNR_\_Y values with a bit rate lower than LF3. The Loop Filter at location 3 has the highest bit rate but a PSNR_\_Y value as low as Loop Filter at location 1.

It can be concluded that the best position for a loop filter is at location 2. In the subjective test, it was noted that the patches and mosquito effect had been reduced. However, the loop filter implemented for the multi-resolution scheme reduces the sharpness of the background image. This reduction in sharpness can be improved in future work, possibly by adding adaptive switching into the system.
Figure 6.5.1 Carphone Comparison with Loop Filter
Figure 6.5.2 Salesman Comparison with Loop Filter
Figure 6.5.3 Suzie Comparison with Loop Filter
Chapter 7

Conclusion And Further Work

7.1 Conclusion

The research work presented in this thesis is concerned with the investigation of image and video compression using subband coding techniques. Subband coding was considered due to many of its features including multi-resolution transmission, compatibility with existing coding standards, and multi-layer video transmission with differing error protection against channel noise. These important features have been fully investigated throughout this thesis.

As subband coding consists of two major elements, filtering and coding, this investigation has concentrated on both. For image applications, subband filtering has undergone major investigation in the past few years. For example, in addition to the well established near-perfect reconstruction QMF filters (i.e. Johnston filters), there exists a variety of other filtering techniques that have been developed for image coding applications. These include perfect reconstruction filters, infinite impulse response filters, and filters based on wavelet optimization, etc. Although the emphasis has been aimed at the perfect reconstruction property of these filters, their relative performances in the presence of quantisation noise as well as bit truncation accuracy, remain unclear.

Therefore, the initial objective of this research presented in chapter 3 was to compare these filters in the presence of noise. A number of aliased free reconstruction filters were used for these studies. The filters considered in these experiments were near-perfect Quadrature Mirror Filter (QMF) banks, Lattice Structure Filters and Infinite Impulse Response (IIR) Filters. For near-perfect, QMF banks such as QMF 8A & QMF 16A were considered. Perfect Reconstruction Lattice Structures filters such as PRQMF and
PRLPF were investigated. These investigations also included Infinite Impulse Response (IIR) Filters with Finite Impulse Response (FIR) reconstruction properties.

A major consideration in this thesis was to evaluate the performance of these filters in the presence of bit truncation noise. Reducing the bit accuracy at the output of the decomposed image is essential for intraframe and interframe coding (i.e. 8 and 9 bits accuracy are required respectively). In addition, since two dimensional filtering needed to be employed (i.e. in a tree structural manner), the bit truncation was applied at two stages; i) after first stage decomposition in the horizontal direction (i.e. First Stage Bit Accuracy, FSBA) and, ii) after the vertical decomposition (i.e. Second Stage of Bit Accuracy, SSBA). The bit truncation also needed to be applied in the synthesis process.

These filters were then evaluated under the same conditions for image applications. The image was reconstructed by synthesis filtering assuming a lossless compression. This ensured that the reconstructed image was free of quantisation and channel noise. Therefore, comparison could be made under the same conditions. It was concluded that although PRQMF and PRLPF have perfect reconstruction properties, they were most sensitive to truncation errors. For short delay applications, the IIR filter was more suitable than the other filters because of its shorter filter taps. Overall, the QMF 16A filter had a better performance than the other filters for bit accuracy with lower values (i.e. 8 to 9 bits accuracy). In the remainder of Chapter 3, the coding aspect for subband still image compression was considered. A new coding method was then proposed and employed to encode individual decomposed bands. The proposed coding scheme was based on utilizing the DCT to improve the runlength statistics. In addition, an adaptive quantisation scheme similar to the ITU-T H.263 quantisation method was deployed to quantise each DCT transformed subband. To further improve the subband coding efficiency a three dimensional VLC was considered which is a combination of VLC for levels, runlength coding for consecutive zeros and end of block codes. Subsequently, the performance of all the above filters were compared in the presence of quantisation noise. Based on these experiments it was concluded that the QMF 16A filter bank can still produce the best results amongst the other filters.
For video applications, Chapter 4 provided an overview of the most popular video compression techniques such as Hybrid DPCM/DCT as well as information on motion estimation and compensation techniques for interframe video coding. This included various motion estimation criteria for block matching techniques. In addition, this chapter provided a summary of the most recent video compression coding standards such as ITU-T H.263 recommended for a very low bit rate video applications. Furthermore, since the investigations in this thesis were mainly concerned with subband video coding, the remainder of this chapter was dedicated to this approach by describing the state of the art techniques covering various interframe subband coding models (Model I and Model II).

In chapter 5, the investigation concentrated on what is known as the Subband Model I approach. In this model the subband decomposition is performed inside the interframe DPCM loop. This model has been recognized to provide a better compression efficiency than Subband Model II which performs subband decomposition before interframe encoding of each decomposed band separately.

With regards to the coding aspect of the Model I approach, the decomposed prediction error signals were encoded using the proposed approach which is a combination of DCT, quantisation and 3D VLC. It should be emphasized that the state of the art techniques use only a dead zone quantisation and a runlength coding based on the B1 code.

The proposed subband coding scheme was simulated using various subband filter banks. The performance of the subband scheme was then compared with the recently recommended ITU-T H.263 which is based on Hybrid DCT/DPCM. Both schemes used the same test video sequences which were in CIF format. The computer results showed that although ITU-T H.263 produced a slightly better PSNR_Y, the subband approach provided a lower bit rate. These results have been very encouraging mainly due to the fact that the subband scheme has a unique property and that is its inherent multi-layering
structure which can be directly utilised for transmission of video over ATM or wireless environments at differing priority levels. In order to impose channel priorities in hybrid coding, additional processing at the expense of allocating more bits would be required. It should be noted however, that subband model I despite its superior performance compared with subband Model II, is incapable of producing multi-resolution video. Due to the importance of multi-resolution video transmission for modern visual communications as well as compatibility with the existing coding techniques, the investigations were then concentrated on these important aspects.

Thus Chapter 6 presented the new results using another interframe subband model. The new model was simulated using the same proposed subband coding scheme to encode the individual decomposed bands. In addition, the coding model was designed to provide full compatibility with ITU-T H.263 Standard at its lower resolution. A two-layer structure to produce video at two levels of resolution was then implemented. Simulations were then carried out to compare the multi-resolution approach with the ITU-T H.263 scheme; both receiving video in accordance to the CIF spatial resolutions. Due to the compatibility constraint imposed on the multi-resolution subband, the first layer had to be encoded at QCIF format. The QCIF frame was then encoded by an ITU-T H.263 encoder and, as a result, the encoded first layer could be received and decoded independently by the ITU-T H.263 decoder. For the second layer, the proposed coding scheme was considered to encode the decomposed bands which are represented by the second layer.

The computer simulation results showed that the subband scheme performed better than ITU-T H.263 for intraframe coding. In addition, the subband scheme had a lower bit rate value than the ITU-T H.263. For interframe coding, the ITU-T H.263 scheme had a slightly better performance than the subband scheme. During interframe coding, the subjective results indicated that the subband scheme suffers from certain artifacts which appeared as small patches and slight mosquito effect. It was concluded that this was the result of using independently estimated motion vectors for each layer (i.e. basic layer
and second layer). To reduce these effects, a loop filtering technique was proposed and investigated.

The loop filter, which operates on a block by block basis, was considered and applied over the entire frame to overcome patches and mosquito effect. The loop filter was placed in three different locations and its best location was found to be inside the subband multi-resolution model. However, the sharpness of the background image was reduced by this loop filter. This reduction in sharpness can be improved in future work.

7.2 Further Work and Recommendations

The side effect introduced by the loop filter can be eliminated by applying an adaptive technique. The loop filter will only be switched on if the image block detected as a match block contains distortion. This can improve the sharpness of the background image. By comparing the structure of the subband multi-channel system with the subband multi-layer system, it can be seen that only one motion estimation is used in the subband multi-channel system. However, two motion estimations are used in the simulation for the subband multi-resolution system (i.e. basic layer and second layer). Due to the difference of these two motion estimations, the patches and mosquito effect appear in the reconstructed image. Therefore, it is essential to modify the system to share the same motion vector between them (i.e. basic layer and second layer).

An upward motion estimation approach can be used to improve the system. In this approach, the motion vectors are estimated from the first layer and then scaled up by a factor of 2. These motion vectors will be used in the second layer for motion prediction and compensation. An alternative approach to this is downward motion prediction, where the motion vectors are first estimated in the second layer. Each motion vector is then scaled down by a factor of two. This has the advantage of having less computational complexity in finding the best matching block.
Further work needs to be carried out on the motion estimation technique mentioned above. It is also important to investigate new entropy codebooks and temporal coding techniques to further reduce the bit rate and improve the picture quality. The Pel Difference Classification (PDC) motion estimation and Pel Recursive motion estimation can be considered to further improve the motion estimation. The new entropy codebook can be used in the second layer to enhance the coding efficiency. In addition, arithmetic coding can also be considered. Finally, temporal subband coding can be considered to provide further layering for multi-resolution video transmission.
References


[16] CCITT Recommendation T.6, “Facsimile Coding Schemes and Coding Control Functions for Group 4 Facsimile Apparatus”


[51] ISO/IEC IS 11172-2, "Information Technology - Coding of Moving Pictures and Associated Audio for Digital Storage media at up to about 1.5 Mbits/s - Part 2 : Video", MPEG-1, 1993


Appendix A

MPEG 2 Specification

<table>
<thead>
<tr>
<th>Profile</th>
<th>Coding Features</th>
</tr>
</thead>
<tbody>
<tr>
<td>Simple</td>
<td>Includes all functionality provided by main level but</td>
</tr>
<tr>
<td></td>
<td>• does not support B-frame prediction modes</td>
</tr>
<tr>
<td></td>
<td>• 4:2:0 format</td>
</tr>
<tr>
<td>Main</td>
<td>Nonscalable coding algorithm supporting functionality for</td>
</tr>
<tr>
<td></td>
<td>• coding interlaced video</td>
</tr>
<tr>
<td></td>
<td>• random access</td>
</tr>
<tr>
<td></td>
<td>• B-frame prediction modes</td>
</tr>
<tr>
<td></td>
<td>• 4:2:0 format</td>
</tr>
<tr>
<td>SNR Scalable</td>
<td>Supports all functionality provided by low and main level for</td>
</tr>
<tr>
<td></td>
<td>• SNR scalable coding (2 layer allowed)</td>
</tr>
<tr>
<td></td>
<td>• 4:2:0 format</td>
</tr>
<tr>
<td>Spatial Scalable</td>
<td>Supports all functionality provided by the high 1440 level for</td>
</tr>
<tr>
<td></td>
<td>• Spatial scalable coding (3 layer allowed)</td>
</tr>
<tr>
<td></td>
<td>• 4:2:0 format</td>
</tr>
<tr>
<td>High</td>
<td>Supports all functionality provided by the main, high 1440, and high level for</td>
</tr>
<tr>
<td></td>
<td>• 3 layers with the SNR and Spatial</td>
</tr>
<tr>
<td></td>
<td>• 4:2:0 or 4:2:2 format</td>
</tr>
</tbody>
</table>

Table A.1 Profile Coding Features

<table>
<thead>
<tr>
<th>Level</th>
<th>Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>Low</td>
<td>352 pixels/line</td>
</tr>
<tr>
<td></td>
<td>288 lines/frame</td>
</tr>
<tr>
<td></td>
<td>30 frames/s</td>
</tr>
<tr>
<td></td>
<td>4 Mb/s</td>
</tr>
<tr>
<td>Main</td>
<td>720 pixels/line</td>
</tr>
<tr>
<td></td>
<td>576 lines/frame</td>
</tr>
<tr>
<td></td>
<td>30 frames/s</td>
</tr>
<tr>
<td></td>
<td>15 Mb/s</td>
</tr>
<tr>
<td>High 1440</td>
<td>1440 pixels/line</td>
</tr>
<tr>
<td></td>
<td>1152 lines/frame</td>
</tr>
<tr>
<td></td>
<td>60 frames/s</td>
</tr>
<tr>
<td></td>
<td>60 Mb/s</td>
</tr>
<tr>
<td>High</td>
<td>1920 pixels/line</td>
</tr>
<tr>
<td></td>
<td>1152 lines/frame</td>
</tr>
<tr>
<td></td>
<td>60 frames/s</td>
</tr>
<tr>
<td></td>
<td>80 Mb/s</td>
</tr>
</tbody>
</table>

Table A.2 Upper Bound for Level Parameter
Appendix B

B.1 QMF Filter Coefficients
The QMF 8A filter coefficients is denoted in Table B.1.1. The filter response is shown in Figure B.1.1

<table>
<thead>
<tr>
<th>Coefficient</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>h₀</td>
<td>+0.9387 1500 E-02</td>
</tr>
<tr>
<td>h₁</td>
<td>-0.7065 1830 E-01</td>
</tr>
<tr>
<td>h₂</td>
<td>+0.6942 8270 E-01</td>
</tr>
<tr>
<td>h₃</td>
<td>+0.4899 8080 E+00</td>
</tr>
<tr>
<td>h₄</td>
<td>+0.4899 8080 E+00</td>
</tr>
<tr>
<td>h₅</td>
<td>+0.6942 8270 E-01</td>
</tr>
<tr>
<td>h₆</td>
<td>-0.7065 1830 E-01</td>
</tr>
<tr>
<td>h₇</td>
<td>-0.9387 1500 E-02</td>
</tr>
</tbody>
</table>

Table B.1.1 QMF 8A Filter Coefficients

Figure B.1 QMF 8A Filter Response
B.2 QMF Filter Coefficients
The QMF 16A filter coefficients is denoted in Table B.2.1. The filter response is shown in Figure B.2.1.

<table>
<thead>
<tr>
<th>QMF 16 A</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>h₀ +0.1050 1670 E-02</td>
<td>h₈ +0.4810 2840 E+00</td>
</tr>
<tr>
<td>h₁ -0.5054 5260 E-02</td>
<td>h₉ +0.9779 8170 E-01</td>
</tr>
<tr>
<td>h₂ -0.2589 7560 E-02</td>
<td>h₁₀ -0.9039 2230 E-01</td>
</tr>
<tr>
<td>h₃ +0.2764 1400 E-01</td>
<td>h₁₁ -0.9666 3760 E-02</td>
</tr>
<tr>
<td>h₄ -0.9666 3760 E-02</td>
<td>h₁₂ +0.2764 1400 E-01</td>
</tr>
<tr>
<td>h₅ -0.9039 2230 E-01</td>
<td>h₁₃ -0.2589 7560 E-02</td>
</tr>
<tr>
<td>h₆ +0.9779 8170 E-01</td>
<td>h₁₄ -0.5054 5260 E-02</td>
</tr>
<tr>
<td>h₇ +0.4810 2840 E+00</td>
<td>h₁₅ +0.1050 1670 E-02</td>
</tr>
</tbody>
</table>

Table B.2.1 QMF 16A Filter Coefficients

Figure B.2.1 QMF 16A Filter Response
## B.3 PRQMF Filter Coefficients

Table B.3.1 below shows the filter coefficient of PRQMF and filter response is illustrated in Figure B.3.1 to B.3.3.

<table>
<thead>
<tr>
<th>Lattice 1</th>
<th>Stage 1 (--)</th>
<th>Stage 2 (• -•)</th>
<th>Stage 3 (• •)</th>
</tr>
</thead>
<tbody>
<tr>
<td>( p_0 )</td>
<td>+1.000 000 E +00</td>
<td>+1.000 000 E +00</td>
<td>+1.000 000 E +00</td>
</tr>
<tr>
<td>( p_1 )</td>
<td>+6.000 000 E+00</td>
<td>-6.000 000 E+00</td>
<td>-6.000 000 E+00</td>
</tr>
<tr>
<td>( p_2 )</td>
<td>-2.343 750 E-02</td>
<td>+1.562 500 E-02</td>
<td>+1.562 500 E-02</td>
</tr>
<tr>
<td>( p_3 )</td>
<td>+1.875 000 E-01</td>
<td>-1.875 000 E-01</td>
<td>-1.875 000 E-01</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Lattice 2</th>
<th>Stage 1 (--)</th>
<th>Stage 2 (• -•)</th>
<th>Stage 3 (• •)</th>
</tr>
</thead>
<tbody>
<tr>
<td>( p_0 )</td>
<td>-1.000 000 E+00</td>
<td>-1.000 000 E+00</td>
<td>-1.000 000 E+00</td>
</tr>
<tr>
<td>( p_1 )</td>
<td>-6.000 000 E+00</td>
<td>-6.000 000 E+00</td>
<td>-6.000 000 E+00</td>
</tr>
<tr>
<td>( p_2 )</td>
<td>+2.343 500E-02</td>
<td>+2.343 750 E -02</td>
<td>+2.343 750 +E02</td>
</tr>
<tr>
<td>( p_3 )</td>
<td>+1.875 000 E-01</td>
<td>-1.875 000 E-01</td>
<td>-1.875 000 E-01</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Lattice 3</th>
<th>Stage 1 (--)</th>
<th>Stage 2 (• -•)</th>
<th>Stage 3 (• •)</th>
</tr>
</thead>
<tbody>
<tr>
<td>( p_0 )</td>
<td>-1.000 000 E-01</td>
<td>-1.000 000 E+00</td>
<td>-7.812 500E-03</td>
</tr>
<tr>
<td>( p_1 )</td>
<td>-6.000 000 E+00</td>
<td>-6.000 000 E+00</td>
<td>-4.000 000 E+00</td>
</tr>
<tr>
<td>( p_2 )</td>
<td>1.562 000 E-02</td>
<td>+1.171 875 E-02</td>
<td>+7.500 000E-01</td>
</tr>
<tr>
<td>( p_3 )</td>
<td>-1.875 000E-01</td>
<td>-1.875 000 E-01</td>
<td>+3.750 000 E-01</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>( p_0 )</th>
<th>( p_1 )</th>
<th>( p_2 )</th>
<th>( p_3 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( p_0 )</td>
<td>+1.000 000 E +00</td>
<td>+1.000 000 E +00</td>
<td>+1.000 000 E +00</td>
</tr>
<tr>
<td>( p_1 )</td>
<td>+6.000 000 E+00</td>
<td>-6.000 000 E+00</td>
<td>-6.000 000 E+00</td>
</tr>
<tr>
<td>( p_2 )</td>
<td>-2.343 750 E-02</td>
<td>+1.562 500 E-02</td>
<td>+1.562 500 E-02</td>
</tr>
<tr>
<td>( p_3 )</td>
<td>+1.875 000 E-01</td>
<td>-1.875 000 E-01</td>
<td>-1.875 000 E-01</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>( p_0 )</th>
<th>( p_1 )</th>
<th>( p_2 )</th>
<th>( p_3 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( p_0 )</td>
<td>-1.000 000 E+00</td>
<td>-1.000 000 E+00</td>
<td>-1.000 000 E+00</td>
</tr>
<tr>
<td>( p_1 )</td>
<td>-6.000 000 E+00</td>
<td>-6.000 000 E+00</td>
<td>-6.000 000 E+00</td>
</tr>
<tr>
<td>( p_2 )</td>
<td>+2.343 500E-02</td>
<td>+2.343 750 E -02</td>
<td>+2.343 750 +E02</td>
</tr>
<tr>
<td>( p_3 )</td>
<td>+1.875 000 E-01</td>
<td>-1.875 000 E-01</td>
<td>-1.875 000 E-01</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>( p_0 )</th>
<th>( p_1 )</th>
<th>( p_2 )</th>
<th>( p_3 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( p_0 )</td>
<td>-1.000 000 E-01</td>
<td>-1.000 000 E+00</td>
<td>-7.812 500E-03</td>
</tr>
<tr>
<td>( p_1 )</td>
<td>-6.000 000 E+00</td>
<td>-6.000 000 E+00</td>
<td>-4.000 000 E+00</td>
</tr>
<tr>
<td>( p_2 )</td>
<td>1.562 000 E-02</td>
<td>+1.171 875 E-02</td>
<td>+7.500 000E-01</td>
</tr>
<tr>
<td>( p_3 )</td>
<td>-1.875 000E-01</td>
<td>-1.875 000 E-01</td>
<td>+3.750 000 E-01</td>
</tr>
</tbody>
</table>

**Table B.3.1 PRQMF Filter Coefficients**
Figure B.3.1 PRQMF Lattice 1 Filter Response

Figure B.3.2 PRQMF Lattice 2 Filter Response
Figure B.3.3 PRQMF Lattice 3 Filter Response
### B.4 PRLPF Filter Coefficients

Table B.4.1 below shows the filter coefficient of PRLPF and filter response is illustrated in Figure B.4.1 to B.4.3.

<table>
<thead>
<tr>
<th>Lattice 1</th>
<th>Stage 1 (---)</th>
<th>Stage 2 (---)</th>
<th>Stage 3 (---)</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \rho_0 )</td>
<td>1.200 000 E+01</td>
<td>-1.200 000 E+01</td>
<td>-1.200 000 E+01</td>
</tr>
<tr>
<td>( \rho_1 )</td>
<td>6.000 000 E+00</td>
<td>-6.250 000 E-02</td>
<td>-6.250 000 E-02</td>
</tr>
<tr>
<td>( \rho_2 )</td>
<td>-2.400 000 E+01</td>
<td>+1.280 000 E+02</td>
<td>+2.457 000 E+04</td>
</tr>
<tr>
<td>( \rho_3 )</td>
<td>+1.875 000 E+01</td>
<td>+6.000 000 E+00</td>
<td>+6.000 000 E+00</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Lattice 2</th>
<th>Stage 1 (---)</th>
<th>Stage 2 (---)</th>
<th>Stage 3 (---)</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \rho_0 )</td>
<td>-3.000 000 E+00</td>
<td>+3.000 000 E+00</td>
<td>+6.000 000 E+00</td>
</tr>
<tr>
<td>( \rho_1 )</td>
<td>-3.750 000 E+01</td>
<td>+4.000 000 E+00</td>
<td>+6.000 000 E+00</td>
</tr>
<tr>
<td>( \rho_2 )</td>
<td>-8.000 000 E+00</td>
<td>-3.000 000 E+00</td>
<td>-3.000 000 E+00</td>
</tr>
<tr>
<td>( \rho_3 )</td>
<td>+3.750 000 E+01</td>
<td>-3.125 000 E+02</td>
<td>-1.200 000 E+01</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Lattice 3</th>
<th>Stage 1 (---)</th>
<th>Stage 2 (---)</th>
<th>Stage 3 (---)</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \rho_0 )</td>
<td>-6.000 000 E+00</td>
<td>-1.200 000 E+01</td>
<td>-3.000 000 E+00</td>
</tr>
<tr>
<td>( \rho_1 )</td>
<td>-3.750 000 E-01</td>
<td>-6.000 000 E+00</td>
<td>-4.000 000 E+00</td>
</tr>
<tr>
<td>( \rho_2 )</td>
<td>-2.400 000 E+01</td>
<td>+8.000 000 E+00</td>
<td>+4.000 000 E+00</td>
</tr>
<tr>
<td>( \rho_3 )</td>
<td>+3.750 000 E-01</td>
<td>+3.750 000 E-01</td>
<td>+3.750 000 E-01</td>
</tr>
</tbody>
</table>

Table B.4.1 PRLPF Filter Coefficients
Figure B.4.3 PRLPF Lattice 3 Filter Response
B.5 IIR Filter Coefficients
The IIR filter coefficients is denoted in Table B.5.1. The filter response is shown in Figure B.5.2.

<table>
<thead>
<tr>
<th>IIR Filter</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>( a_0 )</td>
<td>+1/8</td>
</tr>
<tr>
<td>( a_1 )</td>
<td>+1</td>
</tr>
<tr>
<td>( b_0 )</td>
<td>+1</td>
</tr>
<tr>
<td>( b_1 )</td>
<td>+1/8</td>
</tr>
</tbody>
</table>

Table B.5.1 IIR Filter

Figure B.5.1 IIR Filter Response