Inter-frame subband video coding

Maliti Kelvin Chipango

University of Wollongong

Follow this and additional works at: https://ro.uow.edu.au/theses

Copyright Warning

You may print or download ONE copy of this document for the purpose of your own research or study. The University does not authorise you to copy, communicate or otherwise make available electronically to any other person any copyright material contained on this site.

You are reminded of the following: This work is copyright. Apart from any use permitted under the Copyright Act 1968, no part of this work may be reproduced by any process, nor may any other exclusive right be exercised, without the permission of the author. Copyright owners are entitled to take legal action against persons who infringe their copyright. A reproduction of material that is protected by copyright may be a copyright infringement. A court may impose penalties and award damages in relation to offences and infringements relating to copyright material.

Higher penalties may apply, and higher damages may be awarded, for offences and infringements involving the conversion of material into digital or electronic form.

Unless otherwise indicated, the views expressed in this thesis are those of the author and do not necessarily represent the views of the University of Wollongong.

Recommended Citation

Chipango, Maliti Kelvin, Inter-frame subband video coding, Master of Engineering (Hons.) thesis, Department of Electrical and Computer Engineering, University of Wollongong, 1993.

https://ro.uow.edu.au/theses/2458

Research Online is the open access institutional repository for the University of Wollongong. For further information contact the UOW Library: research-pubs@uow.edu.au
INTER-FRAME SUBBAND VIDEO CODING

A thesis submitted in partial fulfilment of the requirements for the award of the degree

MASTER OF ENGINEERING

in

TELECOMMUNICATIONS ENGINEERING (Honours)

from

The University of Wollongong

by

Maliti Kelvin Chipango, BE (Honours)

Department of Electrical and Computer Engineering

October 1993
Three inter-frame subband video coding models are studied. The three models, referred to as models I, II and III, are studied by employing three different inter-frame predictive coding methods together with subband analysis/synthesis using separable two-dimensional quadrature mirror filter (QMF) banks. Models I and II use block matching and pel-recursive motion compensation whereas model III is based on a simple subband inter-frame difference coding scheme. Model I applies motion compensation to the full band image followed by subband analysis of the motion compensated prediction error. Model II applies subband analysis to the full band image and then uses motion compensated prediction in the resulting subbands. Uniform symmetrical quantisers are employed in each of the models to quantise the prediction error.

The simulation results show good performance of the models, in terms of the peak signal-to-noise ratio (PSNR) and the entropy of the prediction error. Model I shows no significant difference in performance with either block matching or pel-recursive motion compensation. The results show that, for the test image sequence used, the performance of model II is significantly better than model I and III. This effect is noted in simulations using both the block matching and pel-recursive motion compensation schemes. The results further indicate that Model II can be simplified by limiting the motion estimation to one subband and making available the same motion vectors to the other subbands, without compromising its performance. However, the PSNR results also suggest that the pel-recursive motion estimation, which is sensitive to large spatial gradients, works better in model II when the motion estimates are derived from the lowpass subbands. Model III, which encodes the subband frame differences, with no motion compensation, is shown to perform significantly better than model I on a test TV sequence where the movement is small.
ACKNOWLEDGMENTS

I would like to sincerely thank my supervisor, Dr. Phillip Ogunbona for his guidance, support and encouragement throughout the period of this thesis.

I would also like to thank Mr Jim Andrew for providing very helpful suggestions, and opinions on different topics of this thesis.

My special thanks go to the Australian International Assistance Bureau (AIDAB) for awarding me the scholarship, through my home government, to further my education.

Finally, I wish to convey my heartfelt gratitude to my father (MHRIP) and mother, my wife Susan and my son Gregory, whose moral support has contributed immensely to this thesis.
CONTENTS

1. INTRODUCTION 1
   1.1 Background 1
   1.2 Project overview 3
      1.2.1 Elements of inter-frame subband video coding 3
      1.2.2 System set-up 3
   1.3 Organisation of the thesis 4

2. INTER-FRAME SUBBAND CODING - A REVIEW 6
   2.1 Introduction 6
   2.2 Inter-frame Video coding 8
      2.2.1 General 8
      2.2.2 Conditional replenishment 9
      2.2.3 Predictive Video coding and Motion Compensation 12
   2.3 Subband Video Coding 19
      2.3.1 General 19
      2.3.2 Analysis and synthesis systems for subband Video coding 21
      2.3.3 QMF banks and their reconstruction properties 22
   2.4 Inter-frame subband Video coding 26
   2.5 Contribution of the thesis 27

3. INTER-FRAME SUBBAND VIDEO CODING MODELS 29
   3.1 Introduction 29
   3.2 Model I 29
   3.3 Model II 32
   3.4 Model III 34
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.5</td>
<td>Model analysis</td>
<td>36</td>
</tr>
<tr>
<td>3.5.1</td>
<td>General</td>
<td>36</td>
</tr>
<tr>
<td>3.5.2</td>
<td>Preliminary model assessment</td>
<td>37</td>
</tr>
<tr>
<td>3.6</td>
<td>Summary</td>
<td>40</td>
</tr>
<tr>
<td>4.</td>
<td>ISSUES IN INTER-FRAME SUBBAND VIDEO CODING</td>
<td>41</td>
</tr>
<tr>
<td>4.1</td>
<td>Introduction</td>
<td>41</td>
</tr>
<tr>
<td>4.2</td>
<td>Motion detection and Image segmentation</td>
<td>41</td>
</tr>
<tr>
<td>4.3</td>
<td>Motion estimation and Motion Compensation</td>
<td>45</td>
</tr>
<tr>
<td>4.3.1</td>
<td>General</td>
<td>45</td>
</tr>
<tr>
<td>4.3.2</td>
<td>Fourier Method</td>
<td>45</td>
</tr>
<tr>
<td>4.3.3</td>
<td>Spatio-temporal Gradient Method</td>
<td>47</td>
</tr>
<tr>
<td>4.3.4</td>
<td>Block Matching Method</td>
<td>48</td>
</tr>
<tr>
<td>4.3.5</td>
<td>Block Matching motion estimation test results</td>
<td>54</td>
</tr>
<tr>
<td>4.3.6</td>
<td>Pel-recursive Method</td>
<td>56</td>
</tr>
<tr>
<td>4.3.7</td>
<td>Pel-recursive motion estimation test results</td>
<td>59</td>
</tr>
<tr>
<td>4.4</td>
<td>QMF bank design example</td>
<td>63</td>
</tr>
<tr>
<td>4.5</td>
<td>Inter-frame signal Quantisation</td>
<td>67</td>
</tr>
<tr>
<td>4.6</td>
<td>Summary</td>
<td>68</td>
</tr>
<tr>
<td>5.</td>
<td>MODEL PERFORMANCE EVALUATION</td>
<td>70</td>
</tr>
<tr>
<td>5.1</td>
<td>Introduction</td>
<td>70</td>
</tr>
<tr>
<td>5.2</td>
<td>Experimental set-up</td>
<td>70</td>
</tr>
<tr>
<td>5.3</td>
<td>Simulation results</td>
<td>74</td>
</tr>
<tr>
<td>5.3.1</td>
<td>Model I, using block matching motion compensation</td>
<td>74</td>
</tr>
<tr>
<td>5.3.2</td>
<td>Model II, using block matching motion compensation</td>
<td>75</td>
</tr>
<tr>
<td>5.3.3</td>
<td>Model I, using pel-recursive motion compensation</td>
<td>78</td>
</tr>
<tr>
<td>5.3.4</td>
<td>Model II, using pel-recursive motion compensation</td>
<td>79</td>
</tr>
<tr>
<td>5.3.5</td>
<td>Model III, coding subband inter-frame differences</td>
<td>81</td>
</tr>
</tbody>
</table>
5.4 Model Comparisons 82
5.5 Conclusion 88

6. CONCLUSIONS 90

BIBLIOGRAPHY 93

APPENDICES 103

A.1 Synthetic test image generation 103
A.2 Two dimensional gradient estimation 103
A.3 Two dimensional linear interpolation 105
B FIR filter coefficients 106
# LIST OF FIGURES

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>Frames from the test image sequences</td>
<td>4</td>
</tr>
<tr>
<td>2.1</td>
<td>Basic schemes for inter-frame subband coding</td>
<td>7</td>
</tr>
<tr>
<td>2.2</td>
<td>A simple conditional replenishment coder</td>
<td>10</td>
</tr>
<tr>
<td>2.3</td>
<td>Block diagram of a basic DPCM system</td>
<td>12</td>
</tr>
<tr>
<td>2.4</td>
<td>One-dimensional partitioning of input spectrum</td>
<td>19</td>
</tr>
<tr>
<td>2.5</td>
<td>A general subband coding scheme</td>
<td>20</td>
</tr>
<tr>
<td>2.6</td>
<td>Classical two-band analysis/synthesis filter bank</td>
<td>21</td>
</tr>
<tr>
<td>2.7</td>
<td>A basic 2-D QMF bank structure</td>
<td>24</td>
</tr>
<tr>
<td>3.1</td>
<td>Inter-frame subband coder Model I</td>
<td>31</td>
</tr>
<tr>
<td>3.2</td>
<td>Inter-frame subband coder model II</td>
<td>33</td>
</tr>
<tr>
<td>3.3</td>
<td>Inter-frame subband coder model III</td>
<td>35</td>
</tr>
<tr>
<td>4.1</td>
<td>A typical two-stage image segmentation scheme</td>
<td>43</td>
</tr>
<tr>
<td>4.2</td>
<td>Example of image segmentation</td>
<td>44</td>
</tr>
<tr>
<td>4.3</td>
<td>Block matching geometry for calculation of cost function</td>
<td>49</td>
</tr>
<tr>
<td>4.4</td>
<td>Block matching - orthogonal search algorithm</td>
<td>53</td>
</tr>
<tr>
<td>4.5</td>
<td>Block matching motion estimation result</td>
<td>55</td>
</tr>
<tr>
<td>4.6</td>
<td>Entropy of the MC prediction error (using BM)</td>
<td>55</td>
</tr>
<tr>
<td>4.7</td>
<td>Convergence of the pel-recursive method</td>
<td>60</td>
</tr>
<tr>
<td>4.8</td>
<td>Pel-recursive motion estimation</td>
<td>61</td>
</tr>
<tr>
<td>4.9</td>
<td>Entropy of the MC prediction error using the pelrecursive method</td>
<td>61</td>
</tr>
<tr>
<td>4.10</td>
<td>Entropy comparison of the MC prediction error for block matching and pelrecursive methods</td>
<td>62</td>
</tr>
<tr>
<td>4.11</td>
<td>Frequency responses of the lowpass and highpass FIR filters</td>
<td>65</td>
</tr>
<tr>
<td>4.12</td>
<td>The impulse response of the prototype lowpass FIR filter</td>
<td>66</td>
</tr>
<tr>
<td>4.13</td>
<td>Overall QMF bank frequency response</td>
<td>66</td>
</tr>
<tr>
<td>Section</td>
<td>Description</td>
<td>Page</td>
</tr>
<tr>
<td>---------</td>
<td>-------------</td>
<td>------</td>
</tr>
<tr>
<td>5.1</td>
<td>Experimental set-up</td>
<td>71</td>
</tr>
<tr>
<td>5.2</td>
<td>PSNR result of modes I and II, using block matching MC</td>
<td>83</td>
</tr>
<tr>
<td>5.3</td>
<td>PSNR result of modes I and II, using pel-recursive MC</td>
<td>83</td>
</tr>
<tr>
<td>5.4</td>
<td>PSNR result of model I, using pel-recursive and block matching motion compensation</td>
<td>84</td>
</tr>
<tr>
<td>5.5</td>
<td>PSNR result of model II, using pel-recursive and block matching motion compensation</td>
<td>85</td>
</tr>
<tr>
<td>5.6</td>
<td>Comparison of PSNR for Models I, II and III</td>
<td>86</td>
</tr>
<tr>
<td>5.7</td>
<td>Comparison of input and output images</td>
<td>87</td>
</tr>
</tbody>
</table>
## LIST OF TABLES

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.1</td>
<td>Design specifications for the prototype FIR filter</td>
<td>65</td>
</tr>
<tr>
<td>5.1</td>
<td>Model I result of using block matching motion compensation</td>
<td>75</td>
</tr>
<tr>
<td>5.2</td>
<td>Model II result of using separate block matching motion estimation in each subband</td>
<td>76</td>
</tr>
<tr>
<td>5.3</td>
<td>Model II result of using block matching motion compensation, where motion vectors are derived from the LL subband</td>
<td>76</td>
</tr>
<tr>
<td>5.4</td>
<td>Model II result of using block matching motion compensation, where motion vectors are derived from the LH subband</td>
<td>77</td>
</tr>
<tr>
<td>5.5</td>
<td>Model II result of using block matching motion compensation, where motion vectors are derived from the HH subband</td>
<td>77</td>
</tr>
<tr>
<td>5.6</td>
<td>Model II result of using block matching motion compensation, where motion vectors are derived from the HL subband</td>
<td>78</td>
</tr>
<tr>
<td>5.7</td>
<td>Model I result of using pel-recursive motion compensation</td>
<td>79</td>
</tr>
<tr>
<td>5.8</td>
<td>Model II result of using separate pel-recursive motion compensation in each subband</td>
<td>80</td>
</tr>
<tr>
<td>5.9</td>
<td>Model II result of using pel-recursive motion compensation, where the motion vectors are derived from the LL subband</td>
<td>80</td>
</tr>
<tr>
<td>5.10</td>
<td>Model II result of using pel-recursive motion compensation, where motion vectors are derived from the HH subband</td>
<td>81</td>
</tr>
<tr>
<td>5.11</td>
<td>Model III result of entropy and PSNR</td>
<td>82</td>
</tr>
<tr>
<td>5.12</td>
<td>Average entropy and average PSNR of models I and II</td>
<td>84</td>
</tr>
<tr>
<td>5.13</td>
<td>Average entropy and average PSNR of models I, II and III</td>
<td>86</td>
</tr>
</tbody>
</table>
1.1 BACKGROUND

Inter-frame subband video coding is a new method of bandwidth compression of image sequences which may well supplant conventional coding strategies. It is a coding technique resulting from a culmination of advances in both inter-frame predictive video coding (Musmann et al., 1985), which has been in use over the last decade, and the more recent subband image coding methods (Woods, 1991) to form what is clearly the backbone of many present and future image sequence coding applications. Inter-frame subband video coding will play a key role in applications such as video conferencing, medical archiving, videophones and high definition television (HDTV).

Many digital image sequences in transmission and storage applications are generated by scanning a scene several times per second. As a result there are a significant amount of both temporal and spatial redundancies in the image sequences. It is desirable for an image sequence coding scheme to reduce the redundancy in both the temporal and spatial dimensions to suit a given bandwidth requirement and ultimately to limit the distortion in the output image to a level that is perceptually tolerable.

The problem of reducing the spatial redundancy in image sequences has been well investigated by many authors and has resulted in many still-image compression techniques such as differential pulse code modulation (DPCM) and transform coding (in particular, the discrete cosine transform (DCT)). In the last 8 years subband image coding has emerged as an extremely efficient coding method (in a rate-distortion sense)
(Pearlman, cited in Woods, 1991) and also as a method that is able to exploit the human perception of image distortion (Sakrison, 1979; Johnsen et al., 1990; Kim et al., 1993).

The temporal redundancy, on the other hand, has been exploited for the last two decades by using inter-frame predictive coding. The initial inter-frame predictive coding schemes, were limited by the available digital signal processing (DSP) technology, and could only achieve modest compression ratios. With the advent of more powerful DSP technology more efficient inter-frame coding algorithms have been developed, that use motion compensated prediction to reduce the temporal redundancy.

Most present methods of coding image sequences employ a combination of the block-based transform coding and DPCM, the so-called hybrid coding, in order to achieve the necessary compression ratios. However, these methods have now been found to be inadequate for high compression video applications such as HDTV (Gharavi, cited in Woods, 1991). The perceptual quality of the output from these schemes possess some undesirable block artefacts. Inter-frame subband video coding has emerged as the best alternative for image sequence coding that integrates the superior performances of both subband and inter-frame video coding (Woods and Naveen, 1989).

Although inter-frame subband video coding is a recent technique of coding image sequences, experience has already been accumulated in various investigations on the existing methods of inter-frame and subband coding. The recent investigations on inter-frame subband coding of image sequences suggest that these can be classified into three types of coding schemes - loosely called 'models'. From these investigations and other literature the author has not found any comprehensive performance comparison of these models with regard to different inter-frame predictive strategies.

It is the goal of this thesis, therefore, to study and compare the performance of these models, thus giving an in-depth performance analysis of the emerging class of image
sequence coding schemes. In this attempt the thesis also presents a review of the subsystems that form these models.

1.2 PROJECT OVERVIEW

1.2.1 ELEMENTS OF INTER-FRAME SUBBAND VIDEO CODING

Part of the motivation for conducting this study arose from the knowledge that an inter-frame subband video coding system consists of a few basic subsystems which can be separately analysed and designed before the system is assembled. An inter-frame subband coding consists of the following main components;

(i) a subband analysis/synthesis scheme
(ii) an inter-frame predictive scheme
(iii) a compression/decompression scheme.

This investigation dwells on three inter-frame subband video coding models (Woods and Naveen, 1989; Gharavi, 1990). The key to most video sequence coding applications is motion compensated prediction which exploits the temporal redundancy. These models, two of which employ motion compensation, are simulated and compared.

1.2.2 SYSTEM SETUP

All the model simulations in this project were performed on a Sun SPARC station IPC computer in a UNIX environment. The software programs were written in C++. Two image sequences were used in the simulations. Single frames from the two image sequences are shown in Figure 1.1. The input to the models is a ten-frame greyscale, digitised non-interlaced TV sequence, "Newscaster", of size 512 x 512 pixels. The sequence shows a man actively speaking in a TV conversation. The sequence known as "Rings" is a synthetically generated (Netravali and Robbins, 1978) greyscale image of size 512 x 512 pixels. This image is used in testing the accuracy of the motion estimation algorithms.
1.3 ORGANISATION OF THE THESIS

The topics in this thesis are grouped into four sections, namely, review of inter-frame subband coding, introduction of the image coding models, model design issues, model testing and performance evaluation and the conclusion.

The review of subband and inter-frame video coding, in chapter 2, provides independent backgrounds on inter-frame and subband video coding. Then, a discussion is presented on some recent development of inter-frame subband video coding, highlighting the key role played by both predictive inter-frame video coding and intra-frame subband video coding.

Chapter 3 presents three inter-frame subband video coding models and discusses their various components and operations. Also presented in chapter 3 is a preliminary assessment of the models to illustrate the expected performance of the models.
Chapter 4 considers the design aspect of the different components of the models. These include the algorithms for motion estimation, image segmentation, motion detection and the design of the subband analysis/synthesis filters. Preliminary test results from different components of the models are discussed.

The performance of the models is evaluated in Chapter 5. Simulation results from each model are presented and compared.

Finally, chapter 6 presents the conclusion of the thesis and discusses the lessons learned from the whole investigation and also presents some suggestions for possible improvement of this work.
CHAPTER 2
INTER-FRAME SUBBAND CODING - A REVIEW

2.1 INTRODUCTION

Inter-frame subband coding is gaining considerable attention in the area of compression of image sequences. Its potential has been demonstrated in such application areas as high compression video (Gharavi, cited in Woods, 1991), high definition television (HDTV) (Kazunari and Kishimoto, 1990; Ansari and Le Gall, cited in Woods, 1991) and video conferencing. This chapter reviews the development of inter-frame subband video coding. The review starts with the separate developments of inter-frame video coding and subband video coding. Then, some recent studies and applications of inter-frame subband video coding are presented that highlight the benefits of this technique in coding image sequences.

Figures 2.1 (a) and (b) show two basic schemes for inter-frame subband video coding. In the scheme depicted in Figure 2.1(a) the input video signal is initially partitioned into M frequency subbands by using a set of analysis filters. The subbands are decimated, typically by a factor M. The temporal redundancy in each subband is removed by the inter-frame processor, resulting in a subband inter-frame error signal which is encoded before transmission. The inter-frame error signal in each subband is basically a pixel-by-pixel difference between the current subband and its prediction. Each subband inter-frame error signal can be encoded independently with a certain number of bits depending on the perceptual significance of the specific subband. At the receiver the transmitted information is decoded. The subband signals are recovered by
another inter-frame processor corresponding to the one at the transmitter. The full band image is then reconstructed by the subband synthesis filters.

Figure 2.1. Basic schemes for inter-frame subband coding.

In the system shown in Figure 2.1(b) the input image is applied to the inter-frame processor to remove the temporal redundancy and form an inter-frame error signal. This inter-frame processor is similar to the scheme in Figure 2.1(a), except that it operates on the full band image. The inter-frame processor uses the current frame and a prediction of the current frame to generate an inter-frame error signal, which is partitioned into M subbands by the analysis filters. The subbands are then independently encoded before being transmitted. At the receiver, the transmitted information is decoded, and the resulting M subband signals are interpolated by the synthesis filter banks and are summed to obtain an approximation of the inter-frame
error signal. The inter-frame processor at the receiver uses the previously received frames and the reconstructed inter-frame error signal to approximate the input image frame.

From Figure 2.1, it can be noted that inter-frame subband coding consists of two main functions, namely; inter-frame processing and subband analysis/synthesis. In this chapter the focus is on the inter-frame process because it plays a very important role in reducing the temporal redundancy in image sequences (Ansari and Le Gall, cited in Woods, 1991). An extensive literature exists on the optimal design solutions of the analysis/synthesis filter banks (Crochiere and Rabiner, 1983; Smith and Barnwell, 1986; Smith and Eddins, 1990). The study presented in this chapter is limited to the near perfect reconstruction solution provided through the quadrature mirror filter (QMF) banks.

2.2 INTER-FRAME VIDEO CODING

2.2.1 GENERAL

Image sequences are generated by scanning a scene several times per second. In certain scenes the difference between consecutive image frames may be very little. Inter-frame video coding is intended to reduce the frame-to-frame redundancy that exists in image sequences (Seyler, 1965; Connor and Limb, 1974). In inter-frame coding, redundancy can be characterised in the following ways (Haskell el al., 1972):

(i) the similarity that exists between consecutive frames of an image sequence,

(ii) the differing resolution requirements in the display of stationary and moving objects. Studies on the human visual response system indicate (Dubois et al., 1980, cited in Huang, 1981) that viewers require less resolution in rapidly changing areas of the image than in the slowly changing areas. This implies that
rapidly changing areas of the image can be coded with less resolution with no significant degradation in quality.

The next sections present several methods that have been used in coding image sequences.

2.2.2 CONDITIONAL REPLENISHMENT

Conditional replenishment (CR) was first demonstrated by Mounts, 1969, and later considered by Pease and Limb, 1970. It was shown that bandwidth savings could be achieved if only those areas that change from frame to frame are coded and transmitted. In this type of conditional replenishment, only the amplitudes of pixels that have changed in the ith frame since the (i-1)th frame are coded and transmitted along with their addresses. Coding only the changed areas requires a less number of bits than coding the full frames. However, very little savings in bandwidth are achieved if there is motion in the scene since most of the pixels are changing. Figure 2.2 shows a block diagram of a basic conditional replenishment coder. At the transmitter the motion detector compares the ith frame to the (i-1)th frame in memory, and generates address information for all the pixels that have changed. The changed pixel intensities and their addresses are encoded and transmitted to the receiver. At the receiver the information is decoded. The address information is used to reconstruct the ith frame. Pixels that have changed are reconstructed from the decoded intensities, whereas those that have not changed are simply replaced by the corresponding pixels in the (i-1)th frame. Further reduction in the bit rate can be achieved by subsampling. Subsampling can be performed in the spatial domain or the temporal domain. However, in scenes with motion above about 1 pixel per frame, the temporal subsampling is reported to cause some undesirable artefacts in the output image (Pease and Limb, 1970)
Figure 2.2. A simple conditional replenishment coder.

Candy et al., 1971, proposed an extension to the coder presented by Mounts, 1969, to transmit coded frame differences in the moving areas of the image, instead of pixel amplitudes. They suggested that the degradation caused by subsampling was not objectionable in this case because viewers could tolerate small impairments in moving areas. However, this method could not remove the objectionable impairments in the stationary areas of the image. It was then suggested that by transmitting full amplitudes regularly, the so-called forced updating, the impairments could be reduced. The frame differences in the moving areas were then coded using variable length coding. Though this was an improvement over the existing methods it resulted in more uneven data streams requiring larger buffers at both the receiver and transmitter and also increases the overhead of transmitting the side information.

Subsequent works on CR include that by Candy et al., 1972, and also Haskell et al., 1972. The most notable improvement over the earlier methods was the reduction in the
overhead information, achieved by addressing the changed pixels in groups rather than individually. A similar scheme was proposed by Connor et al., 1972, where it was further suggested to have the coder switch between different operating modes. In scenes with low activity, such as picturephone, clusters of moving pixels are coded as frame differences. When the activity is high, spatial subsampling is performed in the moving areas, using interpolation at the receiver to reconstruct the intervening pixels. During more rapid movement, frames are repeated at the receiver. The improvement in quality of the picture in these methods over the other methods is reported to be minimal and occurs only in scenes with small amounts of motion.

Limb et al., 1973, considered the problems of motion detection and buffer management in a CR coder. They suggested that significant improvement in bit-rate reduction could be achieved by coding the intra-frame pixel differences in the moving areas. However, the benefit of coding intra-frame pixel differences is only realised when motion increases, resulting in camera integration. The camera integration forces an increase in intra-frame pixel correlation in the moving areas, thus, reducing the intra-frame pixel difference.

CR coding is noted for its low complexity. It also reduces the temporal redundancy substantially at reasonable image quality for scenes with little motion. However CR has the following disadvantages:

(i) During more rapid motion the buffers tend to overflow, thus requiring buffer management to control the bit rate.

(ii) A CR coder based on frame differences or intra-frame differences is more efficient than plain CR, however it is very sensitive to noise unless an efficient segmentation scheme is devised.

(iii) CR causes some undesirable degradation in active scenes due to the poor tracking of the changes from frame to frame.
Based on the conjecture that linear prediction, adapted to the motion in the image sequence could give better performance than CR, it was realised (Haskell, 1975) that the entropy of coders based on adaptive linear prediction were significantly lower than those based on the techniques discussed above. It was also clear that adaptive frame-differential coding, similar to differential pulse code modulation (DPCM), with movement compensation was more efficient. This resulted in the focus on motion compensation and other predictive coding strategies.

2.2.3 PREDICTIVE VIDEO CODING AND MOTION COMPENSATION

One of the most widely used methods to reduce the transmission bit-rate is DPCM (Rao and Yip, 1990). Figure 2.3 shows a block diagram of a basic DPCM system. In this figure a prediction, \( f' \) of the input signal \( f \) is made based on the previously transmitted and decoded information. The difference, \( e \), between the current sample, \( f \), and its prediction, \( f' \) is coded and transmitted. After decoding of the transmitted codewords the receiver reconstructs the sample by adding the prediction value to the quantised

![Block diagram of a basic DPCM system.](image)

**Figure 2.3.** Block diagram of a basic DPCM system.
prediction error. In order to have the same predicted value at both the receiver and transmitter, the transmitter must base its prediction on the reconstructed samples.

Because of the high correlation between the samples, the range of the prediction error, $e$, is less than that of the original pixels. Thus a fewer set of amplitudes (or quantisation levels) are needed, thereby achieving data compression. The quantisation process inherently results in distortion in the output at the receiver. An optimum quantiser for a DPCM system should yield a minimum number of quantisation levels and reduce the distortion. The distortion measure can be defined by subjective criteria, such as the visibility threshold (Musmann, 1985) or by some objective function like minimum mean-squared error (MMSE). Let $M$ be the total number of quantisation levels and $r_i$ the representation of the prediction error at the $i$th level. Using the MMSE as a measure, the distortion function

$$D_{MMSE} = \sum_{i=0}^{M-1} \int_{d_{i-1}}^{d_i} (e - r_i)^2 p(e) de$$

has to be minimised, where $d_0 < d_1 < \ldots < d_{M-1}$ are the decision levels, $e$ is the input prediction error and $p(e)$ is the probability density function of the prediction error. Hence, by fitting the prediction error to a probability distribution, an optimum quantiser can be designed. One popular quantiser design method is the Lloyd-Max algorithm (Lloyd, 1982). The Laplacian probability distribution is widely used as the model for the prediction error signals (Westerink et al., cited in Woods, 1991).

An inter-frame predictor is similar to the conventional DPCM scheme described above. In an inter-frame predictor, the prediction of the present pixel in the $i$th frame is based on previously transmitted pixels in the $(i-1)$th frame.
An inter-frame predictive scheme can be significantly improved by taking into account the frame-to-frame motion of objects. Let \( f(x,y,i-1) \) be the pixel intensity at the location \((x,y)\) in the \((i-1)\)th frame. If the pixel \( f(x,y,i) \), in the \(i\)th frame is moving with horizontal and vertical displacements, \( \Delta x \) and \( \Delta y \), respectively, then it can be estimated from the previous frame by the following equation

\[
 f(x,y,i) = f(x-\Delta x, y-\Delta y, i-1),
\]

(2.2)

The motion compensated (MC) prediction error is given by

\[
 e(x,y,i) = f(x,y,i) - f(x-\Delta x, y-\Delta y, i-1)
\]

(2.3)

By using a similar coding scheme as that shown in Figure 2.3, the motion compensated prediction error for the \(i\)th frame is encoded and transmitted. At the receiver the predictor uses the decoded prediction error to approximate the \(i\)th frame. The prediction algorithm can be either forward or backward (Musmann, 1985). In a forward prediction scheme the displacement information is transmitted along with the prediction error. A backward prediction scheme, on the other hand, bases the prediction on the previously decoded frames, and hence the transmission of displacement information is unnecessary.

If the motion information about the pixels in the \(i\)th frame is known, then a prediction of the moving pixels in the \(i\)th frame can be obtained from the \((i-1)\)th frame by using the motion information of each pixel. The MC prediction error obtained has a low entropy, thus the transmission bit rate is reduced. The success of motion compensation depends on the ability to estimate the motion of the moving objects.

Motion estimation, in general has been discussed by various authors, including Huang, 1981, and also Musmann, 1985. Because of the computational complexity and the
required real-time processing, motion estimation has been limited to estimation of the translational component of motion.

Some of the earliest methods for measuring displacement in image sequences were presented by Limb and Murphy, 1975, and by Cafforio and Rocca, 1976. However these algorithms were only useful for measuring small displacements.

Netravali and Robbins, 1978, presented several motion estimation methods. By far the most successful method was based on a recursive algorithm, the so-called pel-recursive motion estimation method. In pel-recursive motion estimation the displacement at each moving pixel is recursively estimated using a steepest descent algorithm that minimises the prediction error. The performance of the pel-recursive algorithm in terms of bit rate reduction was shown to be up to 50 percent better than using simple frame differences. The prediction error was quantised using a symmetrical quantiser. Since the motion is calculated independently at every pixel, this method is better at modelling the displacements in natural scenes where the motion is non-uniform. Its drawback is that it has a slow convergence, and if the image has high spatial gradients, exemplified by edges and lines, the algorithm may fail to converge.

In an extension of their earlier work Netravali and Robbins, 1980, reported that improvement in the accuracy of the motion estimates could be achieved by averaging the prediction over several pixels. The improvement was most notable in image scenes where the motion was uniform. However there was little gain in performance in scenes containing non-uniform displacement.

Paquin and Dubois, 1982, suggested a motion estimation scheme based on a function relating spatial gradients and temporal directional derivatives. Displacement estimation for each pixel was calculated from three-dimensional blocks in the image sequence. The advantage of this scheme is that it has temporal recursion and therefore performs
better even for rapidly moving scenes and can be robust to noise. However it is more complex unless simplifications are made. The temporal recursion also requires some additional storage to keep the previous frames.

Moorhead et al., 1987, analysed the performance of the pel-recursive motion estimation and compensation. Through Taylor series expansion, the conditions and the rate of convergence for the pel-recursive method were illustrated. It was argued that implementing the analytical model for the motion estimation would result in significant bit rate reduction. They further suggested that a better strategy would be to project motion estimates along the direction of motion, instead of the spatial direction or the temporal direction as required in the existing methods. Due to its computational complexity this scheme is less attractive for real-time implementation.

Efstratiadis and Katsaggelos, 1990, suggested that a model based pel-recursive motion estimation method for motion compensation could perform better than the earlier methods. Their method utilises spatio-temporal correlation in image sequences in an auto regressive (AR) model to estimate motion at each pixel. This method is shown to be robust in terms of noise immunity but is computationally very intensive. In addition, it requires more memory than the pel-recursive method due to its temporal nature.

In order to improve the accuracy of the displacement estimates Bergeron and Dubois, 1991, extended the method of Paquin and Dubois, 1982. A maximum likelihood (ML) and a maximum a priori probability (MAP) motion estimation schemes were suggested. In this method the motion estimates are calculated for blocks of pixels instead of individual pixels. However, the MAP estimation scheme tends to increase the computational complexity of the method.

The amount of computation required in the pel-recursive motion estimation method can be reduced if the calculation is based on blocks, however this is at the expense of the
accuracy of the estimates. A recently proposed block-based recursive motion estimation and compensation scheme (Orchard, 1993) was claimed to achieve displacements with significantly higher accuracy and lower bit rates than standard methods, resulting in up to 30 percent savings in the bit rate. This scheme, however, requires some additional side information to be transmitted to the receiver and appears to be complex.

Block matching (BM) is by far the most widely used motion estimation method (Jain and Jain, 1981; Puri et al., 1987; Rao and Yip, 1990; Huang and Tsai, cited in Huang, 1981). In BM the $i$th frame is first divided into small blocks. By searching for these blocks in the $(i-1)$th frame, the displacement of the moving objects can be approximated within pixel accuracy. BM is relatively simpler to implement than the pel-recursive method. The difficulty with BM is that, unlike the pel-recursive method, each pixel within a block is assumed to undergo the same translation, and as a result it may lead to block artefacts appearing in the reconstructed image. The literature cited above discusses full search as well as fast search BM algorithms. However, fast search BM algorithms are often preferred due to their computational efficiency. In a block matching forward predictive scheme (Musmann, et al., 1985) the displacement estimates need to be coded and transmitted to the receiver. In order to reduce the overhead of the side information, most coding schemes use integer displacements. If sub-pixel accuracy is desired, two-dimensional interpolation is required. Rao and Yip, 1990, have discussed a number of efficient BM schemes, that have been implemented in a number of application specific integrated circuits (ASICs).

Other predictive schemes exist that do not rely on motion estimation (Dubois et al., 1980; Musmann et al., 1985). These schemes include the conventional DPCM system previously discussed. There are two main categories of prediction;

i) linear predictive schemes and

ii) non linear predictive schemes.
Consider the set, \( \{ f(m-k, n-l); k, l \in Q \} \) representing some previously transmitted image pixels in the neighbourhood of the pixel at the location \((m,n)\), where \( Q \) is a causal quarter plane (QP) \textit{region of support} (Maragos et al., 1984; Nam and O'neill, 1987). A linear prediction of \( f(m,n) \) based on the previously transmitted pixels can be expressed in the following form:

\[
\hat{f}(m,n) = \sum_{k,l \in Q} \alpha(m,n,k,l) f(m-k, n-l)
\]

where \( \alpha(.) \) are the predictor coefficients. An optimum predictor can be obtained by choosing the coefficients, \( \alpha(m,n,k,l) \) such that the mean-squared error, \( E[\hat{f}(m,n) - f(m,n)]^2 \), is minimised. It can be shown that the coefficients are given by the solution to the following normal equations

\[
\sum_{m,n \in Q} \alpha(m,n,k,l) R(m-k, n-l) = R(k,l), \quad k, l \in Q
\]

where \( R(k,l) \) represents the correlation coefficients. A discussion on non linear inter-frame predictors may be found in Dubois et al., 1981.

In the review of inter-frame predictive video coding presented by Musmann et al., 1985, it is shown that for low bit-rate applications such as video conferencing, motion compensated predictive coding is more effective than the other forms of predictive coding.
2.3 SUBBAND VIDEO CODING

2.3.1 GENERAL

Subband coding has been extensively used for bandwidth compression of speech signals since the introduction of quadrature mirror filters (QMF) by Esteban and Galand, 1977. The application of subband coding to images was first considered by Woods and O’Neil, 1986, and later by Gharavi and Tabatabai, 1986.

The idea of subband coding is to partition the input signal spectrum into a number, M, of frequency subbands and to encode each of these subbands separately. Figure 2.4 illustrates a basic one dimensional subband partition (M = 4).

![Figure 2.4](image.png)

**Figure 2.4.** One-dimensional partitioning of input spectrum. M = 4 subbands.

The motivation for subband coding is that, by partitioning a signal spectrum as in Figure 2.4, any subband that is considered less significant to the receiver or interpreter of the signal can be conveniently discarded with minimal degradation in quality. In particular, subband image coding is motivated by the nature of the human visual response mechanism (Dubois et al., 1981; Ansari and Le Gall, cited in Woods, 1991). It is well known that in typical natural scenes, the signal energy is not uniformly distributed in frequency. Thus, when the image is partitioned into subbands the quantisation error can be effectively controlled by quantising the subbands according to their perceptual significance.
Subband analysis also makes it possible to represent images according to scale and orientation (Simoncelli et al., cited in Woods, 1991). Since each frequency range corresponds to a particular scale and orientation, subband analysis enables the extraction of image structures such as edges and lines.

The general subband coding approach starts by passing the input signal through a bank of analysis filters, as shown in Figure 2.5. If all the M subbands have the same bandwidth then each subband occupies only one-Mth of the bandwidth of the input signal. Clearly each subband can be re-sampled at the new Nyquist frequency. The subbands are therefore sub-sampled by a factor of M. Sub sampling here implies that only every Mth sample is kept, while the rest are discarded.

The down-sampled subband signals are then encoded and transmitted. It must be noted that the creation of subband signals does not achieve any compression in itself, since the total number of samples in the input and the subband signals is the same. At the receiver the subband signals are decoded, up-sampled, interpolated by the synthesis filter bank and summed to form an approximation of the input signal. Up sampling is achieved by inserting M-1 zeros between each input sample and filtering.
From the foregoing, it should be noted that a subband codec consists of three distinct parts:

(i) Analysis filter bank that partitions the input image into a number of subbands
(ii) A coder/decoder which codes/decodes each subband separately.
(iii) Interpolation and synthesis filter bank.

2.3.2 ANALYSIS/SYNTHESIS SYSTEMS FOR SUBBAND CODING

Figure 2.6 depicts a basic one-dimensional analysis/synthesis system. In order to partition the signal into two subbands, the analysis filter bank consist of a lowpass and a highpass filter set, $H_0(z)$ and $H_1(z)$, respectively. The synthesis filter bank consists of the lowpass/highpass filter set, $G_0(z)$ and $G_1(z)$, respectively. The outputs from the synthesis filters are summed to form a reconstruction of the input signal.

\[
X'(z) = \frac{1}{2} X(z) \left[ H_0(z)G_0(z) + H_1(z)G_1(z) \right]
\]
\[+ \frac{1}{2} X(-z) \left[ H_0(-z)G_0(z) + H_1(-z)G_1(z) \right] \]

Figure 2.6. Classical two-band analysis/synthesis filter bank.
From equation (2.6) it is noted that some aliasing distortion is introduced in the output signal, $X'(z)$. Other distortions that affect the output signal include amplitude distortion and the phase distortion (Vaidyanathan, 1990; Smith and Eddins, 1990). Consequently the analysis/synthesis filters should be designed so that the distortion is eliminated or reduced to a minimum.

A class of filters, that has received attention in literature (Smith, cited in Woods, 1991), and which can achieve distortion-free reconstruction in the absence of coding errors are the quadrature mirror filters (QMF). The next section introduces the concept of quadrature mirror filters and presents a review of quadrature mirror filter (QMF) banks for subband image coding.

2.3.3 QUADRATURE MIRROR FILTER BANKS AND THEIR RECONSTRUCTION PROPERTIES

Quadrature mirror filter (QMF) banks were originally proposed by Croisier et al., 1976, for speech coding. The application of QMF to image coding was first considered by Woods and O'Neil, 1986. Quadrature mirror filters are so called due to their characteristic mirror symmetry about $\pi/2$. Most QMF banks are designed from finite impulse response (FIR) filters. However, QMF banks based on infinite impulse response (IIR) filters have been reported (Smith, cited in Woods, 1991). The advantage of using FIR filters is that they can easily be designed to have linear phase. Their disadvantage is that they are computationally more complex than IIR filters.

In the QMF solution by Croisier et al., 1976, it was suggested that the aliasing component in equation (2.6) can be eliminated by choosing the filters as follows:
\[ H_1(z) = H_0(-z) , \]

\[ G_0(z) = H_0(z) \quad (2.7) \]

\[ G_1(z) = -H_1(z) = -H_0(-z) \]

And in order to have a perfect reconstruction the following condition must be satisfied:

\[ |H_0^2(z)| + |H_0^2(-z)| = 2 \quad (2.8) \]

where \( H_0(z) \) is the z-transform of a linear phase FIR filter.

It is well known that condition (2.8) cannot be met exactly in practice. However, it can be approximated by optimisation techniques (Johnston, 1980; Parks and Mclellan, 1972), leading to the so-called near perfect reconstruction QMF banks.

A technique for designing perfect reconstruction filters is given by Smith and Barnwell, 1986. Perfect reconstruction is achieved by choosing the filters in equation (2.6) as follows:

\[ G_0(z) = H_1(-z) \quad (2.9) \]

\[ G_1(z) = -H_0(-z) \]

and in addition

\[ H_1(z) = -H_0(-z^{-1})z^{-N} \quad (2.10) \]
The design of the prototype filter $H_0(z)$ is generally more complex than the near perfect solution. However, several design techniques exist (Simoncelli and Adelson, cited in Woods, 1991; Nayebi et al., 1992). Using the two-band QMF bank principle discussed above, the input signal can be divided into more subbands of smaller widths by cascading such two-band QMF banks in a tree structure.

In two dimensions the subband analysis/synthesis process consists of two or more stages. For a uniform analysis/synthesis QMF bank, each subband is successively partitioned into four subbands until the desired number of subbands is reached. One commonly used partition, the so-called octave band partition, can be obtained by successively partitioning the low frequency subband. The 1-D QMF can be extended to

![Diagram](image-url)

Figure 2.7. A basic 2-D QMF bank structure.
2-D by using separable filtering (Vetterli, 1984). This is achieved by applying 1-D QMF along the rows and then along the columns, resulting in the four basic subbands, LL, LH, HH and HL, as shown in Figure 2.7.

Two-dimensional QMF banks based on non-separable filter realisations have also been reported (Simoncelli and Adelson, 1990). Separable filters are commonly used in two-dimensional filtering applications because they are computationally simpler to implement than the non-separable filters, since 1-D properties can be applied directly in each direction of filtering.

Tree structured QMF banks can be efficiently implemented by exploiting the similarities of the filter coefficients (Crochiere and Rabiner, 1983). However, they introduce more system delay and also accumulate the truncation errors for the filters as well as the reconstruction noise. The complexity of the QMF banks is generally dependent on the length of the filters employed. Parallel structured analysis/synthesis filter banks (Crochiere and Rabiner, 1983), on the other hand, have a simpler structure, they introduce less system delay and do not accumulate the distortions. The design of parallel analysis/synthesis filter banks is, however, more complex than tree structured analysis/synthesis filter banks. Methods for designing parallel analysis/synthesis filter banks have been explored by several authors (Galand and Nassbaumer, 1984; Chu, 1985; Smith and Barnwell, 1986; Vaidyanathan, 1987; and Nayebi et al., 1992).

The encoding of intra-frame subband signals has been extensively investigated (Woods, 1991). The process consists of two distinct functions, namely; quantisation and entropy coding. Different methods for encoding intra-frame subband signals have also been discussed by; Jayant and Noll, 1984; Biemond et al., 1990; and Kim et al., 1993.
2.4 INTER-FRAME SUBBAND VIDEO CODING

This section presents some recent techniques that have been used, and others that have been proposed for inter-frame subband coding of image sequences. As the name suggests, these recent techniques incorporate both the inter-frame and subband analysis/synthesis techniques that have been discussed thus far in a single coder.

Current low bit rate image coding methods such as hybrid DPCM/DCT perform block-based processing on the motion compensated error image. These schemes result in output images with edge degradation and visible block structures. Ahmad and Dennis, 1990, proposed a coder which uses both subband analysis/synthesis and inter-frame prediction. In their scheme the motion compensated prediction error is partitioned into several subbands by a set of analysis filters. A sample selection procedure was used to vector quantise the subbands. An interesting result of this scheme is that the reconstructed images have less block artefacts, and less edge degradation. The sample selection scheme was used to identify subband areas that are significant to be vector quantised. Run-length coding was then used for coding the selected subband areas.

Podilchuk et al., 1990, developed a three dimensional subband coding scheme for full motion video. In this scheme the input image was partitioned into several spatio-temporal frequency subbands. Perceptual criteria were used to code each subband. The high frequency subbands with relatively low energy content were quantised using a low bit rate vector quantiser (VQ). The remaining subbands were coded with higher quality using a scalar quantiser. The results indicate that this method has potential for low bit rate application. A similar coding technique was also suggested by Akansu and Kadur, 1990. An adaptive VQ was used to code the motion compensated prediction error. It was noted that for the same visual quality of the reconstructed images, this method required 30% less bits than a conventional adaptive coding technique.
Irie and Kishimoto, 1990, presented an adaptive subband coder for HDTV. QMF banks were used to partition the input image into four subbands, LL, LH, HH and HL. In the low frequency subband (LL), motion compensation was employed followed by an adaptive DCT scheme. In the LH subband direct frame differences were coded using DPCM. The rest of the subbands were simply quantised with a symmetrical quantiser. The simulation results show that the adaptive technique, which uses motion compensation, results in lower values of entropy and higher SNR than non adaptive techniques.

Gharavi, cited in Woods, 1991, discussed two prominent inter-frame subband models. Block matching was used for motion estimation and motion compensation. Results in terms of SNR show that the inter-frame subband coding was better than the commonly used adaptive hybrid DPCM/DCT scheme.

An edge-based vector quantisation method to code the upper subbands of the MC prediction error signal was suggested by Mohsenian and Nasrabadi, 1992. QMF banks were employed for the analysis/synthesis of the input video signal. Block matching motion estimation and compensation was employed. Perceptually significant subband areas were selected by using a Laplacian of a Gaussian function to detect edge information. The compression ratios and the visual quality of the reconstructed images were reported to be significantly better than DCT results.

### 2.5 CONTRIBUTION OF THE THESIS

Recent studies clearly suggest that inter-frame subband coding will play a key role in image sequence coding applications. From these studies two basic coding models have emerged as the most serious candidates for inter-frame subband coding, notably those discussed by Gharavi, 1990. The performance of the two models under different motion
compensation schemes has so far only been presented on an ad hoc basis (Gharavi, cited in Woods, 1991; Woods and Naveen, 1989).

This thesis provides a comprehensive performance assessment of the two models, using different motion compensation schemes. Two most popular motion estimation and compensation strategies, namely, block matching and the pel-recursive methods are incorporated in the two models, resulting in four alternative coding schemes. These schemes are simulated, and their performance is evaluated.

A third coding scheme, based on subband inter-frame differences is also proposed, which is not only simpler than the other two models but is also expected to be very useful in coding certain types of sequences where there is little or no displacement at all. In such scenes motion compensation becomes unnecessarily complex since the same result can be achieved by coding subband inter-frame differences at a much lower computational cost.

In total five experimental coding schemes have been simulated on a UNIX SPARC station. A performance assessment of the models is presented, based on test results from a greyscale non-interlaced TV image sequence. Suggestions and possible improvements are also presented. These inter-frame subband models are introduced in chapter 3.
CHAPTER 3
INTER-FRAME SUBBAND VIDEO CODING MODELS

3.1 INTRODUCTION

In this chapter three inter-frame subband coding models are presented. These coding models are denoted models I, II and III, respectively. The operations of each of the models is discussed. The general structure of models I and II are based on previous suggestions by Gharavi, cited in Woods, 1991. Similar models can also been found in Mohsenian and Nasrabadi, 1992; Rao and Yip, 1990. Model III is proposed here as a simpler alternative for application in special types of scenes where motion compensation is unnecessary. A preliminary analysis of models I, II and III, which considers the prediction error in each model is presented, showing some constraints that are likely to influence the performance of the models. In particular, models I and II are shown to be equivalent under certain conditions imposed on the motion estimation process. In all the models QMF banks have been adopted for subband analysis and synthesis.

3.2 MODEL I

Figure 3.1 shows the block diagram of model I. This model consists of three main processes, namely, the motion compensation process, the subband analysis/synthesis process and the quantisation/de-quantisation process. At the transmitter, motion compensation is applied to the full band image to predict the ith input frame from a reference frame. As shown in Figure 3.1, the reference frame is a sum of both the previously predicted \((i-1)th\) image and the reconstructed version of the displaced frame.
difference ($DFD$). The DFD is formed by subtracting the predicted image from the input image. The DFD is then partitioned into $N$ subbands by using the analysis QMF banks. The motion compensation process at the transmitter obtains the quantised version of the DFD ($DFD'$) by using the synthesis QMF banks, as shown in Figure 3.1.

It should be noted that it is necessary for the transmitter to use the quantised version of the DFD in order to match the prediction at the receiver, where only the quantised version of the DFD is available.

The motion detector at the transmitter uses the previously predicted image to detect the moving areas in the input image. The input image is then segmented into moving areas and stationary background. Motion estimation and compensation is only employed in the moving areas. In the stationary background, the predicted pixels assume the corresponding values in the previous frame. The motion estimator together with the image segmentor generate the motion information. In a forward MC prediction scheme, such as block matching, the motion information consists of the addresses of the moving blocks of pixels and their motion estimates. In a backward motion MC prediction scheme, such as the pel-recursive method, only the addresses of the moving pixels are transmitted to the receiver.

The motion compensation process begins with the second frame. The first frame is coded with DPCM without motion compensation. After every $i$th prediction, the frame memory is replenished with the predicted frame. The $N$ subbands from the DFD signal are separately quantised and coded before being transmitted, along with the coded motion information.
Figure 3.1. Inter-frame subband coder model I.
At the receiver, the N subbands for the DFD, and the motion information are decoded. By using the synthesis QMF banks the DFD subbands are interpolated and summed to approximate the DFD at the transmitter. The output image is then constructed by adding the MC predictor output and the reconstructed DFD. The prediction of the $i$th frame is based on the reconstructed $(i-1)$th image output. It should be noted that the prediction can either be in the forward or backward direction. In the case of backward prediction, the received motion information consists of only the addresses of the moving areas. Motion compensation is only employed to predict the moving areas. In all areas that are stationary, the pixels assume the intensities of the corresponding locations in the previously reconstructed frame.

3.2 MODEL II

Figure 3.2 shows a general block diagram of model II. Like model I, model II also consists of three basic processes, namely, motion compensation, subband analysis/synthesis and quantisation/de quantisation. In model II the input image is initially divided into N subbands by the analysis QMF banks. The subbands, each of which is one-$N$th of the size of the full band image, are then applied to the $N$ motion compensation schemes. Each predicted subband is subtracted from the input subband to produce a displaced frame difference (DFD) subband signal (see Figure 3.2). The DFD signals are separately quantised and encoded, along with the motion information corresponding to each subband, before being transmitted to the receiver. When block matching motion compensation is employed, the motion information consists of the motion estimates and the motion address information. On the other hand, if the pel-recursive motion compensation is used, no motion information is transmitted, since it is based on a backward predictive scheme.
Figure 3.2. Inter-frame subband coder model II.

\( Q = \text{Quantiser} \)

\( \text{DFD} = \text{Motion compensated prediction error} \)
At the receiver, the N motion compensated prediction error subband signals and the corresponding motion information are decoded. By using motion compensation, the N subbands are reconstructed by adding a prediction of each subband to the corresponding MC prediction error. The output image is then obtained by interpolating the N subbands, using the synthesis QMF banks, and summing them.

In both models I and II the MC prediction error is best quantised using a non-uniform quantiser that is matched to the probability density function of the error signal. The MC prediction error signal is generally assumed to have a Laplacian probability density function (Westerink et al., cited in Woods, 1991; Mohsenian and Nasrabadi, 1992). However it has been observed (Gharavi, cited in Woods, 1991) that such a quantiser is unsuitable for the subband error signals. Vector quantisation is reported to be more suitable for coding the MC prediction subband-error signal (Akansu and Kadur, 1990; Mohsenian and Nasrabadi, 1992). A simple but equally effective approach is to use a symmetrical quantiser (Gharavi and Tabatabai, 1986; Ansari and Le Gall, cited in Woods, 1991).

### 3.3 MODEL III

Figure 3.3 shows a block diagram of model III. Unlike models I and II, no motion compensation is required in Model III. While motion compensation is undoubtedly the most effective way to reduce the temporal redundancy, it is noticed that for scenes with little or no displacement at all, motion compensation may be unnecessary, and a simpler model should be used to achieve comparable performance. For this reason model III is proposed as a suitable candidate. Model III integrates frame-to-frame difference coding (Haskell, 1972) with subband analysis. From Figure 3.3, the input image is first partitioned into N subbands by the analysis QMF banks. Each subband image is segmented into moving areas and non-moving areas by the segmentor. The segmentation is achieved by comparing the frame differences against a pre-defined
threshold, T. All areas where the magnitude of the frame difference is below the threshold are considered to be stationary and the FD is set equal to zero. The frame differences are then quantised and transmitted to the receiver. At the receiver the quantised frame differences for each subband are decoded. For every $i$th image frame the $N$ subbands are predicted by adding the decoded subband frame differences to their respective frame memories containing the $N(i-1)$th subbands. In the stationary areas of

![Diagram of inter-frame subband coder model III](image)

**Transmitter**

**Receiver**

$Q = \text{quantiser}$

**Figure 3.3.** Inter-frame subband coder model III (Subband frame difference coder).
the image each subband simply assumes the pixel values of the previously constructed subband. The subbands are interpolated by using the synthesis QMF banks and are summed to generate the $i$th image frame.

Model III is noted for its low complexity as compared to models I and II. However its performance in scenes with moderate to high activity is expected to be poor, based on previous results from full band frame difference coding (Haskell et al., 1972; Haskell et al., 1977; Musmann et al., 1985).

### 3.4 MODEL ANALYSIS

#### 3.4.1 GENERAL

The models introduced in this chapter provide a general guideline on how inter-frame subband coding may be implemented. It is apparent that a variety in performance of the models can be achieved by varying the subband analysis/synthesis scheme as well as the inter-frame predictive strategy. In view of this diversity, the approach here is to adopt one subband analysis/synthesis strategy and then investigate the performance of the models under different motion compensation schemes. This is not seen as a limitation on the model assessment since, as noted in chapter 2, inter-frame prediction is the most effective way to exploit the temporal redundancy that exists in image sequences.

An insight into the performance of the models can be gained by looking at the quantised prediction error signals (denoted DFD in Figures 3.1 - 3.2, and FD in Figure 3.3) and the output image. The quantised prediction error signals are considered because, from their entropy one is able to compare the effectiveness of the predictive schemes in each model. It should be noted that if the total entropy is desired, one should account not only for the quantised prediction error signals but also for the bits allocated to the motion information and any other side information. For discussions of methods to
encode the motion information, the reader is referred to Haskell, 1976, and also Rao and Yip, 1990. The models can also be compared in terms of the peak signal-to-noise ratios (PSNR) as well as the visual quality of the reconstructed images.

3.4.2 PRELIMINARY MODEL ASSESSMENT

Models I and II

The performance of the two models can be compared by considering the effect of both subband analysis and motion compensation on the quantised DFD.

Let \( f(x,y,t-T) \) be the pixel intensity in the previous frame, where \( T \) represents the time interval between frames. If the pixel at \((x,y,t)\) is moving with pure translation, then it can be predicted from the previous frame by the following equation

\[
f(x,y,t) = f(x - \Delta x, y - \Delta y, t - T), \tag{3.1}
\]

where \( \Delta x \) and \( \Delta y \) are, respectively, the horizontal and the vertical components of the motion during the frame time interval \( T \). If instead, the translation is approximated by the components \( \Delta \hat{x} \) and \( \Delta \hat{y} \) then, from equation (3.1), the predicted image in model I can be expressed as follows;

\[
\hat{f}(x,y,t) = f(x - \Delta \hat{x}, y - \Delta \hat{y}, t - T), \tag{3.2}
\]

According to model I, the displaced frame difference (DFD) is given by;

\[
DFD(x,y,t) = f(x,y,t) - \hat{f}(x,y,t) \tag{3.3}
\]

The DFD is partitioned by the analysis QMF banks into \( N \) subbands, which are each sub sampled and quantised independently. If the quantisation noise is assumed to be additive, it can be shown (Gharavi, cited in Woods, 1991) from equations (3.2) and
(3.3) that the quantised version of each subband of the MC prediction error, DFD, has the following Fourier transform;

\[
S'_m(\omega_1, \omega_2, t) = \frac{1}{N} \sum_{k=0}^{N-1} \sum_{l=0}^{N-1} H_m \left[ \frac{\omega_1 + 2k\pi}{n}, \frac{\omega_2 + 2l\pi}{n} \right] \times F \left[ \frac{\omega_1 + 2k\pi}{n}, \frac{\omega_2 + 2l\pi}{n} \right] \left[ 1 - \exp \left( j(\omega_1 \Delta x + \omega_2 \Delta y)T \right) \right] + Q_m(\omega_1, \omega_2, t)
\]

\[m = 0, 1, \ldots, n-1.\]

where \( n \) is the decimation factor (\( N = n^2 \)), \( \omega \) represents the spatial frequency and \( Q_m(\omega_1, \omega_2, t) \) is the Fourier transform of the quantisation noise.

By using a similar analysis to the one discussed above, the quantised version of the DFD in each subband of model II can be expressed in the Fourier domain as follows (Gharavi, cited in Woods, 1991);

\[
S''_m(\omega_1, \omega_2, t) = \frac{1}{N} \sum_{k=0}^{N-1} \sum_{l=0}^{N-1} H_m \left[ \frac{\omega_1 + 2k\pi}{n}, \frac{\omega_2 + 2l\pi}{n} \right] \times F \left[ \frac{\omega_1 + 2k\pi}{n}, \frac{\omega_2 + 2l\pi}{n} \right] \left[ 1 - \exp \left( j(\omega_1 \Delta x'' + \omega_2 \Delta y'')T \right) \right] + Q''_m(\omega_1, \omega_2, t)
\]

\[m = 0, 1, \ldots, N-1.\]

where \( \Delta x'' \) and \( \Delta y'' \) are the displacement estimates calculated in the subband domain.

It should be recalled that in model I the motion estimation is employed on the full band image whereas in model II the motion is estimated in the individual subbands. This imposes some constraints on the performance of the models. From equations (3.4) and (3.5) it can be deduced that if the same motion compensation scheme is employed in each case, the two models will perform equally in terms of the entropy of the quantised prediction error and, possibly, the PSNR, as long as the following conditions are satisfied;
\[
\begin{align*}
\Delta \hat{x} &= n\Delta x'' \\
\Delta \hat{y} &= n\Delta y'' 
\end{align*}
\] 

(3.6)

and

\[Q_m(\omega_1, \omega_2, t) = Q''_m(\omega_1, \omega_2, t)\] 

(3.7)

where \(n\) is the decimation factor in the image rows and columns.

Condition (3.6) also implies that the subbands in model II should yield the same displacement estimates. This is unlikely, given that different subbands have different orientations of the edges, which are likely to cause the motion estimates to vary, and hence influence the model performance. Conditions (3.6) and (3.7) provide some additional incentive to investigate the following:

(i) the performance of models I and II under different motion compensation schemes.

(ii) the performance of model II, when the motion estimation is computed from one subband and the motion vectors from this subband are shared by all the other subbands.

**Model III**

Model III is more easily analysed by considering the effect of subband analysis on the inter-frame differences. It is expected that the entropy of the frame differences will be influenced by both the amount of motion in the image and the local image characteristics in the different subbands. For an image with little motion (which implies more temporal correlation) the frame difference in the lowpass subbands, which have most of the spatial correlation as well, are expected to have lower entropy. However, as
the motion increases the temporal correlation decreases, thus, increasing the entropy of the inter-frame difference. The highpass subbands are expected to have higher entropy due to the edges that are likely to occur in certain scenes.

With a fixed quantiser, the resulting output image will be highly degraded because of the expected entropy changes. An adaptive quantiser is more likely to give a better performance.

3.5 SUMMARY

This chapter has presented models I, II and III. The basic operations of the models have been discussed. A preliminary analysis of the models has been presented, providing the necessary basis on which the models can be compared. Another motivation has been gained to assess models I and II in a new perspective. One interesting experiment is to investigate how the motion estimation in the different subbands influences the performance of model II. Another approach is to determine how models I and II perform under different motion compensation methods. Model III has been introduced as a simple and special alternative coding strategy to models I and II. It is expected to be ideal for coding image sequences where there is little or no motion.
CHAPTER 4
ISSUES IN INTER-FRAME SUBBAND VIDEO CODING

4.1 INTRODUCTION

In order to realise the coding models introduced in chapter 3, namely, models I, II and III, a hierarchical approach is adopted in this chapter to design the models. It can be noted from chapter 3 that these models consist of a few basic components. In order to facilitate the design process this chapter discusses these components and presents a number of algorithms for motion estimation and image segmentation. Emphasis is placed on the algorithms that were used in the simulation of the models. A design example of the QMF banks based on Chebyshev approximation is also presented. Simulation results obtained from block matching and the pel-recursive motion estimation algorithms are also presented and discussed. Finally, the quantisation of inter-frame signals is considered and a symmetrical quantiser is adopted, which is used in the model simulations.

4.2 MOTION DETECTION AND IMAGE SEGMENTATION

Motion detection for image sequence coding applications involves the identification of areas in the present image frame that have changed as a result of motion since the previous frame. A basic image segmentation scheme involves classification of pixels or groups of pixels into stationary background and moving areas. Motion detection and image segmentation serve two other purposes in MC predictive coding applications;

(i) they reduce the amount of computation so that the motion estimation is only carried out in areas of the image that are detected as moving.
(ii) they improve the accuracy of the prediction in motion compensation by basing the motion estimates only on the moving areas of the image.

Motion detection is generally based on frame differences (FD) between the current and the previous frame(s). In scenes undergoing displacement, neglecting changes caused by camera noise, the FD in the stationary background is zero, whereas the moving areas have non-zero FD. Therefore, all pixels where the frame difference is non-zero can be classified as moving and the rest as stationary background. Although this scheme is simple, it does not yield satisfactory results due to the effects of camera noise on images (Haskell, 1972). Several improvements have been suggested to alleviate this problem (Pease and Limb, 1971; Cafforio and Rocca, 1976; Netravali and Robbins, 1978; Driessen et al., 1989; Rao and Yip, 1990)

Figure 4.1 shows a block diagram of a practical two-stage segmentation scheme (Netravali and Robbins, 1978; and also Driessen et al., 1989). In this scheme, detector 1 compares the input frame difference (FD) at each pixel location to a threshold, T1. All pixels where the FD is below T1 are classified as stationary. Pixels whose FD is above T1 are applied to the motion estimation/compensation scheme. Detector 2 compares the displaced frame difference (DFD) at each moving pixel to another threshold, T2. If the DFD is less than T2 the pixel is classified as predictable, to indicate that the motion estimation has been successful, otherwise it is classified as unpredictable. Unpredictable pixels represent areas in the image where the motion estimation algorithm may have failed to produce accurate motion estimates. In recursive motion estimation methods, the accuracy of the motion estimates is further improved through a feedback to the motion estimation algorithm (see Figure 4.1). All unpredictable pixels are then excluded from any future iterations of the motion estimation algorithm.
Figure 4.1. A typical two-stage image segmentation scheme.

The disadvantage of the scheme described above is that the thresholds, $T_1$ and $T_2$ cannot be known in advance, and they are heuristically determined. In view of this observation, the following segmentation algorithm was proposed;

First the threshold, $T_1$, was redefined to vary according to the image characteristics. This was achieved by averaging the FD over the whole image and assigning the result to $T_1$. The threshold, $T_2$, was redefined to be a variable integer. All pixels in the current frame where the magnitude of the FD is less than $T_1$ are considered stationary. At the remaining pixels (ie the moving pixels), the number of pixels where the FD is greater than $T_2$ is calculated, within a square of 5x5 pixels (the pixel under consideration is at the centre). If the number is less than $T_2$ the pixel is considered stationary, otherwise it is considered moving.

Figure 4.2 shows some segmentation results, from the sequences "Newscaster" and "Rings", obtained by using the modified segmentation scheme. It should be noted that the threshold, $T_2$, is variable while the threshold, $T_1$, is no longer heuristically determined, instead it is determined internally by the segmentation algorithm.
Figure 4.2. Example of image segmentation. (a) - (b) Image "Rings", for \( T2 = 1 \) and \( T2 = 15 \), respectively; (c) - (d) Image "Newscaster", for \( T2 = 15 \) and \( T2 = 20 \), respectively. Light areas indicate the moving pixels.

From Figure 4.2 it can be noted that the size of the moving area changes by varying the threshold, \( T2 \). In Figure 4.2 (c) and (d), it is noted that the size of the moving area is reduced when the threshold, \( T2 \), is increased from 15 to 20. A disadvantage of this method is that the detection is not optimal (in a maximum likelihood (ML) detection sense) and the moving areas are less contiguous, as displayed in the real-life image "Newscaster". This has the effect of increasing the amount of side information required to address the moving areas. More optimal, but complex segmentation algorithms have been discussed by Cafforio and Rocca, 1976; and also Driessen et al., 1989.
4.3 MOTION ESTIMATION AND MOTION COMPENSATION

4.3.1 GENERAL

Most motion estimation schemes presented in literature consider the translational component of motion due to real-time computational constraints (Musmann et al., 1985). For this reason, motion will, for the rest of this chapter, refer to translation. This section discusses four main classes of motion estimation algorithms that have appeared in literature, namely, the Fourier method, block matching method, the pel-recursive method and the spatio-temporal gradient method. In addition results of applying block matching and the pel-recursive methods to test image sequences are presented.

4.3.2 FOURIER METHOD

The Fourier method is based on the displacement property of the Fourier transform. If a function \( f(x,y) \), with a Fourier transform \( F(u,v) \), is related to another function \( q(x,y) \) with a Fourier transform \( Q(u,v) \), through the following equation;

\[
q(x, y) = f(x \pm \Delta x, y \pm \Delta y),
\]

then,

\[
Q(u, v) = F(u, v) \exp[\pm j2\pi(u\Delta x + v\Delta y)] \quad (4.1)
\]

where \((x,y)\) denote spatial coordinates, \((u,v)\) represent spatial frequencies, and \((\Delta x, \Delta y)\) denote two dimensional translation. Further, the difference between the phase angles of the two Fourier transforms is given by the equation;

\[
\Delta \varphi(u, v) = \arg{Q(u,v)} - \arg{F(u,v)} = \pm 2\pi(u\Delta x + v\Delta y) \quad (4.2)
\]

where \(\arg{.}\) denotes the phase angle.
Hence, the displacements, $\Delta x$ and $\Delta y$ can be calculated by evaluating equation (4.2) at two frequency points.

Consider any two successive image frames, $f$ and $q$, and further suppose that the image plane is projected onto the x and y axes, by summing across the image rows and columns respectively. This operation results in the one-dimensional arrays, $f_1(x), f_1(y), q_1(x)$ and $q_1(y)$. If the Fourier transforms of the arrays are denoted by $F_1(u), F_1(v), Q_1(u)$ and $Q_1(v)$, then it can be shown using equation (4.2) that

$$
\Delta \phi_1(u) = \arg\{Q_1(u)\} - \arg\{F_1(u)\} = \pm 2\pi u \Delta x
$$

$$
\Delta \phi_1(v) = \arg\{Q_1(v)\} - \arg\{F_1(v)\} = \pm 2\pi v \Delta y
$$

(4.3)

Therefore the displacement, $\Delta x$ can be determined by calculating $\Delta \phi_1$ at only one value of the frequency, $u$. Similarly, $\Delta y$ can be determined by working with the y-projections of $f$ and $q$.

The advantage of the Fourier method over other methods is that it works in the frequency domain and therefore, translation, rotation and scaling can be treated separately (Huang and Tsai, cited in Huang, 1981).

The Fourier method is considered to be unsuitable in this application due to the following reasons:

(i) The phase angle calculated from equations (4.3) falls in the range $-\pi$ to $\pi$, where there is an ambiguity of $2\pi$. If principal values of $\arg\{Q_1(u)\}$ and $\arg\{F_1(u)\}$ are used then the resulting $\Delta \phi_1$ may be wrong.

(ii) The method assumes that the image consists of isolated objects moving on a uniform background, which is not realistic for typical image sequences. The method breaks down if the background is not uniform.
Solutions to some of the problems presented above have been discussed by Huang and Tsai, 1981, cited in Huang, 1981; Persoon and Fu, 1977; Wallace and Wintz, 1980; Wang and Clarke, 1991. However, the results of schemes using the Fourier method for motion estimation are less impressive compared to other methods, such as block matching.

4.3.3 SPATIO-TEMPORAL GRADIENT METHOD

The first motion estimation scheme, based on image gradients was proposed by Limb and Murphy, 1975 and later by Cafforio and Rocca, 1976. These methods were limited to measuring very small displacements (Musmann et al., 1985). Gradient methods in general are based on the constraint equation (Paquin and Dubois, 1982) relating the spatial gradient to the temporal derivative:

\[ \mathbf{v} \cdot \nabla f + \frac{\partial f}{\partial t} = 0 \]  

(4.4)

where \( \cdot \) denotes the inner product, \( f \) is the image intensity, \( \mathbf{v} \) is the object velocity at a given pixel location and time, \( \nabla f \) and \( \frac{\partial f}{\partial t} \) are, respectively, the spatial gradient and temporal derivative at the pixel location and time \((x,t)\), where the vector, \( x \), represents a two-dimensional spatial coordinate variable.

The velocity, \( \mathbf{v} \) in equation (4.4) can be obtained by approximating \( \nabla f \) and \( \frac{\partial f}{\partial t} \) with finite differences and selecting \( \mathbf{v} \) such that equation (4.4) is satisfied over a given area. Once the velocity \( \mathbf{v} \) is determined, and the time between successive frames, \( T \), is known, the displacement estimate is simply equal to \( \mathbf{v}T \).

By using finite differences, and minimising the mean squared error in equation (4.4) Paquin and Dubois, 1982, obtained the following recursive algorithm
\[ \hat{d}(x,t) = \tilde{d}(x,t) - \varepsilon \left( \sum_{x' \in \text{vol}} \nabla_x(x',t) \nabla_x(x',t)^T \right)^{-1} \left( \sum_{x' \in \text{vol}} DFD(x',t,\tilde{d}(x',t)) \nabla_x(x',t) \right) \]

(4.5)

where \( \hat{d} \) is the displacement estimate, \( \tilde{d} \) is the displacement estimate at the same point in the previous frame, \( \varepsilon \) is a constant, \( x \) is a two-dimensional spatial coordinate, \( \nabla_x(.) \) stands for the spatial gradient, and \( t \) represents the time. The volume in equation (4.5) is evaluated over a number of frames. After the computation in (4.5), the block is slid to the next pixel and the computation is repeated.

For scenes with large displacements, the temporal recursion tends to improve the accuracy of the motion estimates, since the temporal derivative, \( \frac{\partial f}{\partial t} \), is better approximated. However, the temporal recursion requires additional storage for the previous frames. The temporal recursion also causes the gradients at the edges of images to increase over time and may cause the algorithm to diverge (Moorhead et al., 1987). Due to the reasons cited above the spatio-temporal gradient method was not considered for implementation in the proposed models.

### 4.3.4 BLOCK MATCHING METHOD

The basic idea of block matching is to compare a block of pixels of size \((M \times N)\) in frame \( k \) with a corresponding block within a search area of size \((M+2p \times N+2p)\) in frame \( k-1 \) as shown in Figure 4.3. The best match is determined based on a cost function such as the mean squared error (MSE) or maximum cross-correlation (CC). Consider the template \( f \) (refer to Figure 4.3) in the \( k \)th frame, with a centre pixel located at \((m,n)\) and further, assume that a matching block is located, within the search area \( f_r \), in the \((k-1)\)th frame with a centre pixel located at \((m+i,n+j)\). For a given search criterion the objective is to determine the displacements \( i \) and \( j \), such that the cost function is minimum (maximum for CC). The basic assumptions in the algorithm are.
that all pixels within each block have the same displacement, and the maximum
displacement for any pixel within the block is limited to $\pm p$ pixels.

![Diagram](image)

**Figure 4.3.** Block matching geometry for calculation of cost function.

The accuracy of the estimation depends on the search criterion applied. The following
are some of the search criteria that can be employed (note that $f(k,l)$ represents the pixel
intensity at the location $k,l$):

**(a) Cross-correlation function**

$$M(i, j) = \frac{\sum_{m=1}^{M} \sum_{n=1}^{N} f(m, n) f_r(m+i, n+j)}{\sqrt{\sum_{m=1}^{M} \sum_{n=1}^{N} f^2(m, n) \sum_{m=1}^{M} \sum_{n=1}^{N} f_r^2(m+i, n+j)}}$$  \hspace{1cm} (4.6)

$-p \leq i, j \leq p$

The displacement estimates are obtained from the position $(i,j)$ such that the value of
the function $M(i,j)$ is maximum.

**(b) Normalised MSE**
The displacement estimates are obtained from the position \((i,j)\) such that the value of the function \(M(i,j)\) is minimum.

\[ M(i, j) = \frac{\sum_{m=1}^{M} \sum_{n=1}^{N} [f(m,n) - fr(m+i,n+j)]^2}{\sum_{m=1}^{M} \sum_{n=1}^{N} f^2(m,n)}, \quad -p \leq i, j \leq p \]  

\[ (4.7) \]

\(c\) **Mean absolute difference (MAD)**

\[ M(i, j) = \frac{1}{MN} \sum_{m=1}^{M} \sum_{n=1}^{N} |f(m,n) - fr(m+i,n+j)|, \quad -p \leq i, j \leq p \]  

\[ (4.8) \]

where the displacement estimates are obtained from the position \((i,j)\) such that the value of the function \(M(i,j)\) is minimum.

The disadvantage of the cross-correlation function is that it is less accurate for small block sizes and when the blocks are not undergoing pure translation (Jain and Jain, 1981). In order to improve the accuracy, images usually need to be sharpened by high-pass filtering prior to motion estimation. This, however, increases the complexity of the scheme. Other functions such as the MSE are more accurate but they also increase the complexity of the motion estimation. The MAD function, on the other hand, is simple to implement and is considered to be suitable for low bit rate video applications (Gharavi, cited in Woods, 1991).

From the geometry depicted in Figure 4.3, it can be shown that a full search of the area, \(fr\), (considering all horizontal and vertical shifts) requires as many as \((2p + 1)^2\) computations of the function \(M(i,j)\). This problem has led to the design of more efficient search methods.
An efficient search algorithm can be obtained by assuming that a distortion function monotonically increases as the search is directed away from the direction of its minimum (Jain and Jain, 1981). Suppose \( O = (x_o, y_o) \) is the optimum search point and \( A = (x_A, y_A) \) is any other point in a search region (SR). A distortion function \( M(x) \) is **quadrant monotonic** (Jain and Jain, 1981) if \( M(x) < M(A) \) for any \( X = (x_X, y_X) \in SR \) that satisfies the following conditions:

(i) \( X \) and \( A \) belong to the same quadrant with respect to \( O \), ie, \( x_x - x_o \) (and \( y_x - y_o \)) have the same sign as \( x_A - x_o \) (and \( y_A - y_o \)) and

(ii) either

\[
|x_x - x_o| < |x_A - x_o| \text{ and } |y_x - y_o| \leq |y_A - x_o|,
\]

or

\[
|x_x - x_o| \leq |x_A - x_o| \text{ and } |y_x - y_o| < |y_A - x_o|.
\]

If it is assumed, using the above model, that \( O = (x_o, y_o) \) is the minimum point and, further, if two distinct points are placed in SR such that \( M(A) > M(B) \), the following properties hold:

(a) If \( y_A = y_B \) and \( x_A = x_B \), then \( O \) cannot exist in the half plane defined by

\[
\{(x, y) \in SR | x \geq x_A \}.
\]

(b) If \( x_A > x_B \) and \( y_A > y_B \), then \( O \) cannot exist in the quadrant defined by

\[
\{(x, y) \in SR | x > x_A \text{ and } y > y_A \}.
\]

Thus, by examining every pair of points in the search area hierarchically, the regions that prohibit the minimum point can be eliminated and as a result the amount of computation is reduced.
Several algorithms have been proposed, based on the model described above, that are computationally more efficient than the full search algorithm (Rao and Yip, 1990; Musmann et al., 1985). One of these algorithms is the orthogonal search, which is described below.

Orthogonal search algorithm (Puri et al., 1987)

The search region (SR) is set equal to \((M + 2p) \times (N + 2p)\) pixels.

Set step number \(i = 1\).

Initial step size \(l = \lceil p/2 \rceil\) where \(\lceil x \rceil\) indicates the smallest integer greater than \(x\).

**Step 1:** Three search points \((1.1, 1.2, 1.3)\) are placed horizontally in the centre of SR as shown in Figure 4.4. The distance between every two neighbouring points equals the step size. The cost function is calculated at the three search points. The minimum point is selected as the centre of the next step.

Set step \(i = i + 1\);

**Step 2:** Two more search points \((2.1, 2.2)\) are placed vertically around the minimum point \((1.2)\) from the previous step. The distance between every two neighbouring points equals the step size, as shown in Figure 4.4.

**Step 3:** The remaining region of uncertainty now has an area of \(4(l - 1) \times (l - 1)\). If \(l = 1\), stop. Otherwise, \(l = \lceil l/2 \rceil\).

Set step \(i = i + 1\), and continue.

**Step 4:** Two more search points are placed horizontally around the minimum from the vertical step. The minimum point is set as the centre for the next step.

Set step \(i = i + 1\), and go back to step 2.
The orthogonal search method requires only $4\lceil \log_2 p \rceil + 1$ computations of the cost function, $M(i,j)$, per block compared to $(2p+1)^2$ for the full search method. However, the so-called fast-search algorithms, which include the orthogonal search method, are in some scenes less accurate than the full search algorithm. This is mainly due to the assumption that the distortion function is quadrant monotonic. It is assumed ('keeping fingers crossed') that as one moves away, in any direction, from the minimum point the function monotonically increases. While most images satisfy this condition (Jain and Jain, 1981) there is no guarantee that the quadrant monotonicity condition is always satisfied.

Block matching, in general, is simple to implement and was selected as one of the motion compensation schemes to be used in the simulations presented in this work. The algorithms described above measure displacements only up to an integer number of pixels per frame. Sub pixel accuracy can be obtained by two-dimensional interpolation (Gonzalez and Wintz, 1987). This, however, increases the complexity and the memory requirements. The choice of the block sizes is a compromise between accuracy of the motion estimates and the overhead of the side information. If larger block sizes are
selected then the assumption of uniform motion for all pixels in a block will breakdown. On the other hand, the overhead due to the motion information increases if smaller block sizes are selected.

### 4.3.5 BLOCK MATCHING MOTION ESTIMATION TEST RESULTS

A full search block matching algorithm was selected for implementation for reliability reasons discussed above. This algorithm was implemented in a program called BMA. The inputs to the program are; the image sequence name, the size of each image, the maximum allowed displacement, p, and the dimensions of the blocks, M and N.

The motion estimation was applied to the synthetically generated image sequence "Rings". Appendix A.1 provides a description of the generation of the concentric rings pattern in this image. The use of this image allows a known displacement to be estimated by the algorithm, thus providing a measure of accuracy. The image was displaced by known amounts corresponding to 1, 2, 3, 4, 5 and 6 pixels per frame. Each image was divided into 8x8 blocks and the mean absolute difference (MAD) in equation (4.8) was used as the cost function. The maximum allowed displacement in both vertical and horizontal directions was limited to ±6 pixels. Only integral displacements were considered in the motion estimation in order to reduce the amount of computation. Figure 4.5 shows the average estimated displacement against the true displacement.
Figure 4.5. Block matching motion estimation result. From the image "Rings".

It can be observed from Figure 4.5 that the block matching motion estimation algorithm performs very well on the synthetic image, "Rings". This is not surprising as the image is noise free, but the results indicate that the underlying model is accurate under ideal conditions. The performance of the motion estimation algorithm can also be considered by observing the entropy of the quantised MC prediction error. Figure 4.6 compares the entropy of the MC prediction error with the entropy of direct frame differences.

Figure 4.6. Entropy of the MC prediction error (using BM). From the sequence "Rings".
It is apparent from Figure 4.7 that block matching motion compensation offers a significant reduction in entropy over simple frame differences. This result also gives an indication of the coding gain that can be achieved by using motion compensation.

4.3.6 PEL-RECURSIVE METHOD

Netravali and Robbins, 1978, published the first pel-recursive motion estimation algorithm. In their pel-recursive motion estimation algorithm it is assumed that an initial displacement estimate $\hat{D}^i$ is found to produce a new improved estimate $\hat{D}^{i+1}$ according to the following recursion:

$$\hat{D}^{i+1} = \hat{D}^i + U^i$$  \hspace{1cm} (4.9)

where $U^i$ is an update term of the iteration. The recursion is executed for each pixel at consecutive locations along the image rows or along the columns.

Consider the intensities of two consecutive image frames, $f(x,t-T)$ and $f(x,t)$, as functions of the two-dimensional spatial variable, $x$, and time $t$, where $T$ is the time interval between frames. If an object is moving in translation then, neglecting the uncovered background, the current frame, $f(x,t)$, can be predicted from the previous frame by:

$$f(x,t) = f(x-D,t-T)$$ \hspace{1cm} (4.10)

where $D$ is the translation by which the object has moved.

If $D$ is instead estimated by $\hat{D}$, then the displaced frame difference (DFD) in the estimate is given by

$$DFD(x,\hat{D}^i) = f(x,t) - f(x-\hat{D}^i,t-T)$$ \hspace{1cm} (4.11)
where $\hat{D}^i$ is an $i$th estimate of the displacement vector $D$. As $\hat{D}^i$ converges to the true displacement, $D$, DFD approaches zero. An iterative equation to estimate $D$, using the DFD function (equation 4.11) can be developed using the steepest descent algorithm (Netravali and Robbins, 1980; Moorhead et al., 1987) giving the following equation

$$\hat{D}^{i+1} = \hat{D}^i - \frac{\varepsilon}{2} \nabla_D[DFD(x, \hat{D}^i)]^2$$

(4.12)

where $\nabla[.]$ is the two-dimensional spatial gradient operator. The positive scalar coefficient, $\varepsilon$, governs the convergence of the recursion. Using the definition of DFD in equation (4.11) and evaluating the gradient, equation (4.12) becomes

$$\hat{D}^{i+1} = \hat{D}^i - \varepsilon DFD(x, \hat{D}^i) \nabla f(x - \hat{D}^i, t - T)$$

(4.13)

where $\nabla = \nabla_x$ is the gradient with respect to horizontal and vertical coordinates of the vector $x$. The displacement estimate $\hat{D}$ is not restricted to integral values. Therefore, two-dimensional interpolation of $f(x - \hat{D}^i, t - T)$ is necessary for non integral values of $\hat{D}$, in order to obtain the gradients in equation (4.13). The reader is referred to Appendices A.2 and A.3 for discussions on two dimensional interpolation and calculation of the gradients. From equation (4.13) it can be noted that at every $i$th iteration the previous estimate of the displacement is updated by a magnitude that is proportional to the prediction error, DFD. Thus, if a moving pixel at $x_a$ is predicted with displacement $\hat{D}^i$ and intensity $f(x - \hat{D}^i, t - T)$ resulting in the prediction error, $DFD(x_a - \hat{D}^i)$, the estimator tries to produce a new estimate, $\hat{D}^{i+1}$, such that

$$|DFD(x_a - \hat{D}^{i+1})| \leq |DFD(x_a - \hat{D}^i)|$$

(4.14)

This requires that the update vector of the displacement estimate is always directed towards and not opposite the actual displacement. It can be shown from equation (4.13)
that when the direction of displacement is perpendicular to the gradient $\nabla f(x - \hat{D}^i, t - T)$ the DFD is zero, giving a zero update even though there is displacement. Another difficulty with the pel-recursive algorithm is that it tends to diverge in image areas with large spatial gradients resulting from the presence of edges (Paquin and Dubois, 1982). Hence the convergence of $\hat{D}$ to the actual displacement, $D$, is not guaranteed under such conditions. The constant, $\varepsilon$, determines both the rate of convergence of the algorithm and the accuracy of the displacement estimate. A large value of $\varepsilon$ yields a quick convergence but the variance of the estimates increases (Moorhead et al., 1987). Due to the difficulty in analytically determining the optimum value of the constant, $\varepsilon$ (Netravali and Robbins, 1978), it is usually determined heuristically. The accuracy of the algorithm in equation (4.13) can be improved by averaging the calculation over several pixels in the calculation as follows

$$
\hat{D}^{i+1} = \hat{D}^i - \varepsilon \left\{ \sum_{j=0}^{P} W_j DFD(x, \hat{D}^j) \nabla f(x - \hat{D}^i, t - T) \right\}
$$

(4.15)

where $W_j$ are coefficients such that $W_j \geq 0$ and $\sum_{j=0}^{P} W_j = 1$.

The convergence rate can be increased and further improvement in accuracy of the algorithm described above can be achieved by replacing the constant term, $\varepsilon$, in equation (4.12) by a denominator term to give the following expression (Musmann et al., 1985)

$$
\hat{D}^{i+1} = \hat{D}^i - \frac{1}{2} \sum_{j=0}^{P} \left[ \nabla_D[DFD(x, \hat{D}^j)] \right]^2
\frac{1}{2} \sum_{j=0}^{P} \left[ \nabla_D f(x - \hat{D}^i, t - T) + \nabla f(x, t) \right] \nabla f(x, t)
$$

(4.16)
However this modification increases the computational complexity and memory requirements of the algorithm.

It should be noted that in this pel-recursive scheme, the displacement estimates are not transmitted to the receiver. Instead, only the quantised MC prediction error are coded and transmitted to the receiver as well as the motion addresses generated by the segmentation algorithm. At the receiver the displacement is derived from the previously transmitted pixels using equations (4.12) - (4.16). The present frame is calculated by adding the decoded MC prediction error to the predicted frame.

4.3.7 PEL-RECURSIVE MOTION ESTIMATION TEST RESULTS

In the first experiment the "Rings" sequence of size 512 x 512 pixels was translated by 2 pixels per frame. The convergence of the pel-recursive motion estimation algorithm was investigated by observing the normalised motion compensated (MC) prediction error over 20 iterations of equation (4.15) for $\varepsilon = 0.001$. The sequence was lowpass filtered and the same experiment was repeated for values of $\varepsilon = 0.001$ and $\varepsilon = 0.005$.

The result is shown in Figure 4.7. A result for a prediction based on direct frame differences is also included to highlight the benefit of motion compensation. Note the reduction in the normalised prediction error when the image is lowpass-filtered prior to motion compensation.
Figure 4.7. Convergence of the pel-recursive method. (a) Frame differences (b) Motion compensation before lowpass filtering ($\varepsilon = 0.001$). (c) Motion compensation after lowpass filtering ($\varepsilon = 0.001$). (d) Motion compensation after lowpass filtering ($\varepsilon = 0.005$).

Figure 4.7 also shows the reduction in the MC prediction error when the convergence constant, $\varepsilon$, is increased from 0.001 to 0.005. When the constant, $\varepsilon$ was increased to 0.007 the algorithm became unstable. This highlights the existence of an optimal value of $\varepsilon$ which can be determined analytically (Netravali and Robbins, 1978).

Figure 4.8 shows the motion estimation in the x-direction at each iteration of the recursive algorithm, with $\varepsilon = 0.001$. The "Rings" image was displaced by 1 pixel per frame in the x-direction. Fifty iterations were performed. Note how the estimation converges slowly towards the true displacement value 1.
In a separate experiment, 20 frames of the sequence “Rings” (translated by 2 pixels per frame) were generated. The pel-recursive motion compensation scheme was applied to the sequence and the entropy of the MC prediction error was noted. This procedure was repeated with a lowpass filtered sequence. Figure 4.9 shows the entropy of the motion compensated prediction error of the sequence. Note the reduction in entropy after the images are lowpass filtered.

**Figure 4.9.** Entropy of the MC prediction error using the pel-recursive method. \( \epsilon = 0.005 \), with and without lowpass filtering.
Figure 4.10 compares the entropy of the motion compensated prediction error obtained from the block matching algorithm and the pel-recursive motion estimation. The result for block matching show a significantly lower entropy than that for the pel-recursive method. Note, however, that over the period of twenty frames, the pel-recursive method result, with lowpass filtering approaches that for block matching. The oscillatory nature of the block matching result is attributed to the quantisation of the MC prediction error, coupled with the allocation of the integral motion estimates to blocks of pixels. Initially most of the MC prediction error is essentially due the frame differences of the image areas designated as stationary by the motion detector. However, in subsequent frames, the quantised MC prediction error which is used in the prediction introduces some variance in the reference image, which is a sum of the previously predicted frame and the quantised MC prediction error. As a result the reference image is no longer related to the input frame through simple integral displacements. The pel-recursive method result is relatively smooth since the motion estimates are assigned to individual pixels and they are not restricted to integer values.

![Figure 4.10. Entropy comparison of the MC prediction error for block matching and pel-recursive method (ε = 0.005) before and after lowpass filtering.](image-url)
4.4 QMF BANK DESIGN EXAMPLE

This section illustrates the methodology of designing the analysis/synthesis QMF banks. While the filters are one-dimensional, they are easily extended to two dimensions by adopting the 2-D separable filter model (Vetterli, 1984).

The basic input-output relation for the 1-D analysis/synthesis system is recalled from chapter 2:

\[ X'(z) = \frac{1}{2} X(z) [H_0(z) G_0(z) + H_1(z) G_1(z)] \]

\[ + \frac{1}{2} X(-z) [H_0(-z) G_0(z) + H_1(-z) G_1(z)] \]  \hspace{1cm} (4.17)

For simplicity reasons, the approach here is to adopt the near perfect reconstruction QMF solution (Croisier et al., 1976), given in equation (2.7), where the alias term in equation 4.17 is cancelled. It is recalled that the z-transforms, \( H_I(z) \), \( G_0(z) \) and \( G_I(z) \) are related to \( H_0(z) \) according to equations (2.7). The corresponding impulse responses for the filters \( h_I[n] \), \( g_0[n] \), \( g_I[n] \) can all be readily derived from \( h_0[n] \), by using the z-transform relation in equations (2.7), to give the following equations;

\[ h_I[n] = (-1)^n h_0[n] \] \hspace{1cm} (4.18.a)

and

\[ g_0[n] = h_0[n] \] \hspace{1cm} (4.18.b)

\[ g_I[n] = -(-1)^n h_0[n] \] \hspace{1cm} (4.18.c)

As discussed in chapter 2, FIR filters are found to be most desirable for the QMF banks because they can easily be designed to have a linear phase. This requires that the prototype filter, \( h_0[n] \), must be chosen such that it is an FIR filter with a linear phase. It can be shown (Rabiner and Gold, 1975; Crochiere and Rabiner, 1983) that a useful
solution satisfying this linear phase condition is obtained when $h_0[n]$ is chosen to be a positive symmetric FIR filter of even length, satisfying the following equation:

$$h_0[n] = h_0[N-1-n], \quad n = 0, 1, \ldots, N-1. \quad (4.19)$$

The task is then to design the filters given in equations (4.18) such that the reconstruction error in the output signal, $X'(z)$, is as small as possible. The accuracy of the reconstruction of $X'(z)$ is determined by the flatness of the following function (from equation (2.8));

$$|H_0^2(z)| + |H_0^2(-z)| = H_*(z) \quad (4.20)$$

where $H_*(z)$ represents the overall transfer function, which is desired to be a constant.

The filter design problem can be formulated as a non linear optimisation problem by defining an error function that takes into account the desired filter characteristics such as the overall magnitude distorting in equation (4.20), the stop band and pass band edges, the width of the transition band, the order of the filter (N), the energy in the stop band region, and the phase response. Optimisation techniques for filter design already exist (Parks and Mclellan, 1972, Johnston, 1980, Nayebi et al., 1992) that can be employed to design the prototype filter $h_0[n]$.

As an example, a weighted Chebyshev approximation procedure based on the Parks-McClellan algorithm (Oppenheim and Schafer, 1989) was implemented (in the program MINIMAX) to design the prototype lowpass FIR filter, $h_0[n]$. The design parameters for the filter are the pass band and stop band edges, $\omega_p$ and, $\omega_s$, the ratio between the pass band and stop band ripples, the order of the filter and the tolerance to indicate convergence of the algorithm. Figures 4.11 - 4.13 show the resulting filter characteristics of a Chebyshev design (for N = 32), obtained by using the design specifications listed in Table 4.1. As noted from Figure 4.13, the reconstruction
magnitude distortion is about $\pm 0.5$ dB. The largest magnitude distortion in the overall frequency response occurs in the transition band. For the same amount of transition and stop band ripple, the magnitude distortion can be reduced by increasing the length ($N$) of the filter. This, however, results in a longer system delay and increases the amount of computation incurred in filtering operations.

<table>
<thead>
<tr>
<th>Variable</th>
<th>Specification</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\omega_n$</td>
<td>0.23</td>
</tr>
<tr>
<td>$\omega_s$</td>
<td>0.33</td>
</tr>
<tr>
<td>K (ripple ratio)</td>
<td>40</td>
</tr>
<tr>
<td>$N$ (order of filter)</td>
<td>32</td>
</tr>
<tr>
<td>Tolerance</td>
<td>0.00001</td>
</tr>
</tbody>
</table>

Table 4.1. Design specifications for the prototype FIR filter $H_0(z)$.

![Diagram](image)

**Figure 4.11.** Frequency responses of the lowpass and highpass FIR filters, $H_0(z)$, and $H_f(z)$, respectively. Filter length, $N = 32$. 
Figure 4.12. The impulse response of the prototype lowpass FIR filter, $h_0[n]$ for $N = 32$.

Figure 4.13. Overall QMF bank frequency response.

The weighted Chebyshev approximation approach illustrated above, with the equiripple frequency response results in longer filters for a given stop band to pass band ripple ratio. A better design of the QMF with a smaller reconstruction error, for a given filter length, $N$, can be achieved by optimising an objective function of the following form (Johnston, 1980):
\[ E = \alpha \int_{\omega_s}^{\pi} |H_0(e^{j\omega})|^2 d\omega + 2 \int_{0}^{\pi/2} \left[ \left| H_0(e^{j\omega}) \right|^2 + \left| H_0(e^{j(\omega+\pi)}) \right|^2 - 1 \right] d\omega \]  

(4.21)

where the term \( E_s(\omega_s) \) describes the energy in the stop band region of the frequency response \( H_0(e^{j\omega}) \), and \( E_r \) represents the reconstruction error in equation (4.20). The weighting factor, \( \alpha \), determines the relative importance of the terms \( E_s(\omega_s) \) and \( E_r \), where \( \omega_s \) is the cut-off frequency.

Filter coefficients for the Chebyshev design example and the Johnston filter type-C (Johnston, 1980) are tabulated in Appendix B.

4.5 INTER-FRAME SIGNAL QUANTISATION

The function of the quantiser is to reduce the number of amplitude levels of the MC prediction error. This section discusses the design of the quantiser that was used to quantise the MC prediction error signal in this chapter and subsequently in the simulations of the inter-frame subband models.

The most optimal quantiser (in a minimum mean squared error sense) design approach is to have a non uniform quantiser that is matched to the probability density function (PDF) of the MC prediction error. The Laplacian distribution is widely accepted as the most suitable for modelling the MC prediction error signal (Podilchuk et al., 1990, Westerink et al., cited in Woods, 1991). However one recognises the need for the quantiser to perform equally well on subband MC prediction error signals since this is the intended application of the quantiser in this study. Previous investigations by Gharavi; and also by Ansari and Le Gall, both cited in Woods, 1991, showed that the
commonly used Laplacian PDF is subjectively unsuitable for encoding the subband MC error signal.

Coupled with this observation, and for simplicity reasons, a symmetrical quantiser was found to be adequate. A 32-level symmetrical quantiser was implemented with step size equal to 0.0625, where the normalised amplitude range was from -1.0 to 1.0.

4.6 SUMMARY

In this chapter the design theory for the different sub-systems that form the inter-frame subband image coding models I, II and III have been presented. More emphasis was placed on block matching and the pel-recursive motion estimation algorithms as well as motion detection/image segmentation and QMF bank design, since these are the main components of the models to be simulated. Other motion estimation schemes, namely, the Fourier method and the temporal gradient method, were also reviewed.

The general observation from the test results is that there is a definite reduction in entropy for motion compensation over simple inter-frame differences. For the test sequence used in the experiment, block matching appears to perform better than the pel-recursive method in terms of accuracy. The pel-recursive method shows a slow convergence rate. Its convergence is adversely affected by local image properties such as edges. It has been shown that the prediction error can be significantly reduced if the images are low pass-filtered prior to motion estimation. This essentially smoothenes the edges in the image. On the lowpass images the pel-recursive method shows a progressive improvement in entropy reduction. As more and more frames are received it is apparent that, for the test image used, there is improvement over block matching. Since block matching was only limited to integer motion estimation, its performance is expected to decrease for real-life scenes. If sub-pixel accuracy is accommodated in
block matching, then the complexity is expected to be comparable to the pel-recursive motion estimation.

A design example of the QMF banks, based on near perfect reconstruction filters has also been presented. The Johnston type-C \((N = 32)\) filters (Johnston, 1980) have been adopted for this application and will be employed in the model simulations in chapter 5.
CHAPTER 5
MODEL PERFORMANCE EVALUATION

5.1 INTRODUCTION

This chapter evaluates the performance of inter-frame subband image coding models I, II and III which were introduced in chapter 3. Simulation results, obtained from the greyscale, non-interlaced TV sequence "Newscaster", are presented. Models I and II are each simulated by separately employing block matching and the pel-recursive methods for motion compensation. Model III, which encodes the frame differences of the subbands, with no motion compensation is also simulated with the same TV sequence. In total, five image coding schemes are simulated. These schemes are evaluated in terms of the PSNR, the entropy of the subband motion compensated prediction error signals and the visual quality of the output image.

5.2 EXPERIMENTAL SET-UP

Figures 5.1 (a) - (c) show an overview of the systems set-up used in the computer simulations. All programs were implemented in C language and were run on a Sun SPARC computer. The input to the models are ten-frames of the greyscale non-interlaced TV sequence, "Newscaster", of size 512 x 512 pixels. The analysis/synthesis QMF banks employ 32-tap near perfect reconstruction filters of type-C by Johnston, 1980. The quantisers are the same as the 32-level symmetrical quantiser discussed in section 4.6. The TV sequence "Newscaster" was preferred as a more suitable test sequence because it possesses the "head and shoulder" type of features, which fit practical applications such as video conferencing and picturephone.
In the set-up of model I (shown in Figure 5.1 (a)), the motion compensated (MC) prediction error was partitioned into four subbands (LL, LH, HH and HL), each of size
256 x 256 pixels, by the analysis QMF banks. Each subband was quantised and the entropy was measured at the outputs of the quantisers.

The peak signal-to-noise ratio (PSNR) was calculated using the following equation (Gonzalez and Wintz, 1987):

\[
PSNR = 10 \log_{10} \left( \frac{255^2}{\frac{1}{N^2} \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} [f(x,y) - g(x,y)]^2} \right)
\]  

(5.1)

where \(f(x,y)\) and \(g(x,y)\) are, respectively, the input and output image intensities. \(N\) represent the sizes of the image rows and columns, where the image is assumed to be square.

Model II was set up as shown in Figure 5.1 (b). The input image was also partitioned into four subbands (LL, LH, HH, and HL) by using the analysis QMF banks. Each subband was applied to the motion compensation scheme to produce a MC prediction error. The MC prediction error of each subband was quantised using the same quantiser as in model I before being transmitted. At the receiver each of the subbands is reconstructed by motion compensation. The subbands are then interpolated and summed to form the output image. The entropy and the PSNR were measured as in model I.

In model III the transmitter and the receiver were connected in a back-to-back arrangement, as depicted in Figure 5.1 (c). The subband analysis and synthesis was implemented by using the same quadrature mirror filters as in models I and II. Every frame in the input sequence was partitioned into four subbands (LL, LH, HH and HL). The inter-frame differences of each subband were determined and all differences below
the threshold, $T_2$, were truncated to zero. The resulting subband frame difference signals were quantised using 32-level symmetrical quantisers (see section 4.5) and the entropy of each quantised subband was measured. At the receiver the decoded subbands were interpolated and summed to form the output image. The PSNR ratio was calculated using equation 5.1.

By using the systems arrangement shown in Figure 5.1 the following investigations were made:

(i) Comparison of the performance of models I, II and III, in terms of the entropy of the prediction error, the PSNR, the visual quality of the output image and the general complexity of the models.

(ii) Comparison of models I and II, with respect to the effect, on the entropy and the PSNR, of using block matching and the pel-recursive methods for motion compensation.

(iii) Assessment of the performance of model II, when the motion is estimated only from one subband and the motion vectors obtained are applied to all subbands. This is motivated by the fact that pixels at a particular location in each of the four subbands are derived from the same pixel in the full band image and, therefore, should have the same displacement.

Parts (i) and (ii) of the above investigations were conducted through the following model simulations;

1. Model I, using block matching motion compensation
2. Model II, using block matching motion compensation
3. Model I, using pel-recursive motion compensation
4. Model II, using pel-recursive motion compensation
5. Model III, coding subband inter-frame differences
Part (iii) of the investigation was performed by simulating model II under the following conditions:

(a) applying motion compensation on the LH, HH, and HL subbands by using motion vectors derived from the LL subband.
(b) applying motion compensation to the LL, HH, and HL subbands by using motion vectors derived from the LH subband.
(c) applying motion compensation to the LL, LH, and HL subbands by using motion vectors derived from the HH subband.
(d) applying motion compensation to the LL, LH, and HH subbands by using motion vectors derived from the HL subband.

5.3 SIMULATION RESULTS

5.3.1 MODEL I, USING BLOCK MATCHING MOTION COMPENSATION

Table 5.1 shows the simulation results of the entropy of the displaced frame difference (DFD) for the four subbands (LL, LH, HH, HL) and the PSNR of model I using block-matching. The operation of the motion compensation algorithm begins with frame 0 already in memory at both the receiver and transmitter. A block size of 8x8 pixels was selected and the maximum displacement, \( p \), was set equal to 4 pixels per frame. The results in Table 5.1 show that the average entropy of the LL and LH subbands is lower than that of the HH and HL subbands. The entropy is higher in the HH and HL subbands because these contain most of the edge information, which can be noted from the orientation of line details in the input image. The prediction error is generally high along the edges of the input image. The PSNR starts from a value of 41 dB at frame 1, and then gradually falls to 29 dB after 9 frames. At this point the DFD has increased slightly as a result of movement in the scene as well as due to quantisation of the prediction error. Frame 1 has a higher value of PSNR since the prediction at this stage is based essentially on frame 0, when the prediction error is still small.
### Table 5.1
Model I result of using block matching motion compensation. From image "Newscaster", Block size = 8x8, maximum displacement (p) = 4 pixels.

<table>
<thead>
<tr>
<th>Frame No.</th>
<th>Entropy of displaced frame difference (bits/pixel)</th>
<th>PSNR (dB)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>LL</td>
<td>LH</td>
</tr>
<tr>
<td>1</td>
<td>0.01</td>
<td>0.01</td>
</tr>
<tr>
<td>2</td>
<td>2.25</td>
<td>2.47</td>
</tr>
<tr>
<td>3</td>
<td>1.68</td>
<td>1.52</td>
</tr>
<tr>
<td>4</td>
<td>1.69</td>
<td>1.20</td>
</tr>
<tr>
<td>5</td>
<td>1.33</td>
<td>1.48</td>
</tr>
<tr>
<td>6</td>
<td>1.36</td>
<td>1.58</td>
</tr>
<tr>
<td>7</td>
<td>1.47</td>
<td>1.26</td>
</tr>
<tr>
<td>8</td>
<td>1.19</td>
<td>1.18</td>
</tr>
<tr>
<td>9</td>
<td>1.78</td>
<td>1.80</td>
</tr>
<tr>
<td>Average</td>
<td>1.42</td>
<td>1.39</td>
</tr>
</tbody>
</table>

5.3.2 MODEL II, USING BLOCK MATCHING MOTION COMPENSATION

Tables 5.2 - 5.6 show the entropy and the PSNR of model II using block-matching motion compensation. Table 5.2 shows the result of applying motion compensation to the four subbands, using independently estimated motion vectors from each subband. Tables 5.3-5.6 show the result of using common motion vectors derived from one subband. Motion vectors have been derived respectively from the LL, LH, HL, and HH subbands. From Table 5.2 it is noted that the average entropy for the LL and LH are, respectively, 1.36 and 1.44 bits/pixel which are lower than the entropy for the HH and HL subbands, where the average entropy in each subband is 1.98 bits/pixel. Tables 5.4 - 5.6, on the other hand, show no significant difference in the entropy and the PSNR. It is interesting and important to note that the results in Table 5.3, where the motion vectors are derived from the LL subband, are slightly better than those in Tables 5.4 - 5.6. This can be explained by the fact that the LL subband retains most of the features of the image sequence and, hence, should be more suitable for the block matching scheme. Tables 5.2 and 5.3, however, show no significant difference, in terms of the
entropy and PSNR, between estimating separate motion vectors in each subband and using motion vectors derived from the LL subband.

<table>
<thead>
<tr>
<th>Frame No.</th>
<th>Entropy of displaced frame difference (bits/pixel)</th>
<th>PSNR (dB)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>LL</td>
<td>LH</td>
</tr>
<tr>
<td>1</td>
<td>0.01</td>
<td>0.01</td>
</tr>
<tr>
<td>2</td>
<td>1.99</td>
<td>2.68</td>
</tr>
<tr>
<td>3</td>
<td>1.60</td>
<td>1.52</td>
</tr>
<tr>
<td>4</td>
<td>1.66</td>
<td>1.64</td>
</tr>
<tr>
<td>5</td>
<td>1.34</td>
<td>1.30</td>
</tr>
<tr>
<td>6</td>
<td>1.18</td>
<td>1.45</td>
</tr>
<tr>
<td>7</td>
<td>1.38</td>
<td>1.35</td>
</tr>
<tr>
<td>8</td>
<td>1.33</td>
<td>1.30</td>
</tr>
<tr>
<td>9</td>
<td>1.77</td>
<td>1.68</td>
</tr>
<tr>
<td>Average</td>
<td>1.36</td>
<td>1.44</td>
</tr>
</tbody>
</table>

Table 5.2. Model II result of using separate block matching motion estimation for each subband. From image "Newscaster", Block size = 8x8, maximum displacement (p) = 4 pixels.

<table>
<thead>
<tr>
<th>Frame No.</th>
<th>Entropy of displaced frame difference (bits/pixel)</th>
<th>PSNR (dB)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>LL</td>
<td>LH</td>
</tr>
<tr>
<td>1</td>
<td>0.01</td>
<td>0.01</td>
</tr>
<tr>
<td>2</td>
<td>1.99</td>
<td>2.50</td>
</tr>
<tr>
<td>3</td>
<td>1.60</td>
<td>1.55</td>
</tr>
<tr>
<td>4</td>
<td>1.66</td>
<td>1.65</td>
</tr>
<tr>
<td>5</td>
<td>1.34</td>
<td>1.36</td>
</tr>
<tr>
<td>6</td>
<td>1.18</td>
<td>1.47</td>
</tr>
<tr>
<td>7</td>
<td>1.38</td>
<td>1.34</td>
</tr>
<tr>
<td>8</td>
<td>1.33</td>
<td>1.28</td>
</tr>
<tr>
<td>9</td>
<td>1.77</td>
<td>1.69</td>
</tr>
<tr>
<td>Average</td>
<td>1.36</td>
<td>1.43</td>
</tr>
</tbody>
</table>

Table 5.3. Model II result of using block matching motion compensation, where motion vectors are derived from the LL subband. From image "Newscaster", Block size = 8x8, maximum displacement (p) = 4 pixels.
<table>
<thead>
<tr>
<th>Frame No.</th>
<th>Entropy of displaced frame difference (bits/pixel)</th>
<th>PSNR (dB)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>LL</td>
<td>LH</td>
</tr>
<tr>
<td>1</td>
<td>0.01</td>
<td>0.01</td>
</tr>
<tr>
<td>2</td>
<td>1.42</td>
<td>2.68</td>
</tr>
<tr>
<td>3</td>
<td>1.44</td>
<td>1.52</td>
</tr>
<tr>
<td>4</td>
<td>1.27</td>
<td>1.64</td>
</tr>
<tr>
<td>5</td>
<td>1.31</td>
<td>1.30</td>
</tr>
<tr>
<td>6</td>
<td>1.11</td>
<td>1.45</td>
</tr>
<tr>
<td>7</td>
<td>1.46</td>
<td>1.35</td>
</tr>
<tr>
<td>8</td>
<td>1.21</td>
<td>1.30</td>
</tr>
<tr>
<td>9</td>
<td>1.20</td>
<td>1.68</td>
</tr>
<tr>
<td>Average</td>
<td>1.16</td>
<td>1.44</td>
</tr>
</tbody>
</table>

**Table 5.4.** Model II result of using block matching motion compensation, where motion vectors are derived from the LH subband. From image "Newscaster", Block size = 8x8, maximum displacement (p) = 4 pixels.

<table>
<thead>
<tr>
<th>Frame No.</th>
<th>Entropy of displaced frame difference (bits/pixel)</th>
<th>PSNR (dB)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>LL</td>
<td>LH</td>
</tr>
<tr>
<td>1</td>
<td>0.01</td>
<td>0.01</td>
</tr>
<tr>
<td>2</td>
<td>2.20</td>
<td>2.56</td>
</tr>
<tr>
<td>3</td>
<td>1.63</td>
<td>1.51</td>
</tr>
<tr>
<td>4</td>
<td>1.64</td>
<td>1.63</td>
</tr>
<tr>
<td>5</td>
<td>1.37</td>
<td>1.26</td>
</tr>
<tr>
<td>6</td>
<td>1.26</td>
<td>1.44</td>
</tr>
<tr>
<td>7</td>
<td>1.41</td>
<td>1.40</td>
</tr>
<tr>
<td>8</td>
<td>1.30</td>
<td>1.30</td>
</tr>
<tr>
<td>9</td>
<td>1.70</td>
<td>1.72</td>
</tr>
<tr>
<td>Average</td>
<td>1.39</td>
<td>1.43</td>
</tr>
</tbody>
</table>

**Table 5.5.** Model II result of using block matching motion compensation, where motion vectors are derived from the HH subband. From image "Newscaster", Block size = 8x8, maximum displacement (p) = 4 pixels.
Table 5.6. Model II result of using block matching motion compensation, where motion vectors are derived from the HL subband. From image "Newscaster", Block size = 8x8, maximum displacement (p) = 4 pixels.

<table>
<thead>
<tr>
<th>Frame No.</th>
<th>Entropy of displaced frame difference(bits/pixel)</th>
<th>PSNR (dB)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>LL</td>
<td>LH</td>
</tr>
<tr>
<td>1</td>
<td>0.01</td>
<td>0.01</td>
</tr>
<tr>
<td>2</td>
<td>2.20</td>
<td>2.56</td>
</tr>
<tr>
<td>3</td>
<td>1.63</td>
<td>1.51</td>
</tr>
<tr>
<td>4</td>
<td>1.64</td>
<td>1.63</td>
</tr>
<tr>
<td>5</td>
<td>1.37</td>
<td>1.26</td>
</tr>
<tr>
<td>6</td>
<td>1.26</td>
<td>1.44</td>
</tr>
<tr>
<td>7</td>
<td>1.41</td>
<td>1.40</td>
</tr>
<tr>
<td>8</td>
<td>1.30</td>
<td>1.30</td>
</tr>
<tr>
<td>9</td>
<td>1.70</td>
<td>1.72</td>
</tr>
<tr>
<td>Average</td>
<td>1.39</td>
<td>1.43</td>
</tr>
</tbody>
</table>

5.3.3 MODEL I, USING PEL-RECURSIVE MOTION COMPENSATION

Table 5.7 shows the entropy of the DFD and the peak signal-to-noise ratio (PSNR) of model I using the pel-recursive motion compensation. The convergence constant, $\epsilon$, was set equal to 0.001 and the motion detection threshold, $T_2$, was selected as 15. It should be recalled from chapter 4 that the other threshold, $T_1$, is calculated internally within the motion compensation algorithm, and it is equal to the average of the inter-frame differences. From Table 5.7 it is noted that the entropy of the LL and LH subbands is significantly lower than that of the HH and HL subbands. This occurs mainly because, as discussed in sections 4.3.6 and 4.3.7, the DFD displays some higher values along the edges of the input image, caused by some inaccurate motion estimates along these areas. An interesting observation is that, as in the case of block matching, the PSNR gradually drops, over the ten frames, from 41 dB to 30 dB. This is attributed to the increase in amplitude of the DFD as a result of movement in the image and the quantisation, which further reduces the accuracy of the motion estimation algorithm.
<table>
<thead>
<tr>
<th>Frame No.</th>
<th>Entropy of displaced frame difference (bits/pixel)</th>
<th>PSNR (dB)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>LL</td>
<td>LH</td>
</tr>
<tr>
<td>1</td>
<td>0.01</td>
<td>0.01</td>
</tr>
<tr>
<td>2</td>
<td>2.20</td>
<td>2.56</td>
</tr>
<tr>
<td>3</td>
<td>1.63</td>
<td>1.51</td>
</tr>
<tr>
<td>4</td>
<td>1.64</td>
<td>1.64</td>
</tr>
<tr>
<td>5</td>
<td>1.39</td>
<td>1.26</td>
</tr>
<tr>
<td>6</td>
<td>1.27</td>
<td>1.44</td>
</tr>
<tr>
<td>7</td>
<td>1.45</td>
<td>1.39</td>
</tr>
<tr>
<td>8</td>
<td>1.31</td>
<td>1.34</td>
</tr>
<tr>
<td>9</td>
<td>1.71</td>
<td>1.87</td>
</tr>
<tr>
<td>Average</td>
<td>1.40</td>
<td>1.45</td>
</tr>
</tbody>
</table>

Table 5.7. Model I result of using pel-recursive motion compensation. From image "Newscaster", $e = 0.001$, $T2 = 15$.

5.3.4 MODEL II, USING PEL-RECURSIVE MOTION COMPENSATION

Tables 5.8 - 5.10 show the entropy of the DFD and the peak signal-to-noise ratio (PSNR) of model II using pel-recursive motion compensation. Table 5.8 shows the result of applying separate motion estimation to each subband, whereas Tables 5.9 and 5.10 show the results of applying motion compensation to one subband of the image sequence and allocating the same motion vectors to all the other subbands. In Tables 5.9 and 5.10 the motion vectors were derived, respectively, from the LL and HH subbands.

From Tables 5.8 - 5.10, it is noted that the result presented in Table 5.9, where the motion vectors are derived from the LL subband, has a slightly higher value of the average PSNR. Table 5.10, where the motion vectors were derived from the HH subband has the lowest average PSNR. This suggests that motion vectors derived from the LL subband yield a better PSNR than either using independent motion vectors (Table 5.8) or using motion vectors derived from the HH subband (Table 5.10). This is
in agreement with the discussion of the test results in section 4.3.7, where it was highlighted that the pel-recursive algorithm is likely to diverge in images, such as the

<table>
<thead>
<tr>
<th>Frame No.</th>
<th>Entropy of displaced frame difference(bits/pixel)</th>
<th>PSNR (dB)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>LL</td>
<td>LH</td>
</tr>
<tr>
<td>1</td>
<td>0.01</td>
<td>0.01</td>
</tr>
<tr>
<td>2</td>
<td>2.17</td>
<td>2.52</td>
</tr>
<tr>
<td>3</td>
<td>1.62</td>
<td>1.50</td>
</tr>
<tr>
<td>4</td>
<td>1.64</td>
<td>1.62</td>
</tr>
<tr>
<td>5</td>
<td>1.36</td>
<td>1.24</td>
</tr>
<tr>
<td>6</td>
<td>1.25</td>
<td>1.44</td>
</tr>
<tr>
<td>7</td>
<td>1.41</td>
<td>1.40</td>
</tr>
<tr>
<td>8</td>
<td>1.30</td>
<td>1.30</td>
</tr>
<tr>
<td>9</td>
<td>1.70</td>
<td>1.72</td>
</tr>
<tr>
<td>Average</td>
<td>1.38</td>
<td>1.43</td>
</tr>
</tbody>
</table>

Table 5.8. Model II result of using separate pel-recursive motion compensation for each subband. From image "Newscaster", $\varepsilon = 0.001$, $T2 = 15$.

<table>
<thead>
<tr>
<th>Frame No.</th>
<th>Entropy of displaced frame difference(bits/pixel)</th>
<th>PSNR (dB)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>LL</td>
<td>LH</td>
</tr>
<tr>
<td>1</td>
<td>0.01</td>
<td>0.01</td>
</tr>
<tr>
<td>2</td>
<td>2.17</td>
<td>2.56</td>
</tr>
<tr>
<td>3</td>
<td>1.62</td>
<td>1.51</td>
</tr>
<tr>
<td>4</td>
<td>1.64</td>
<td>1.63</td>
</tr>
<tr>
<td>5</td>
<td>1.36</td>
<td>1.26</td>
</tr>
<tr>
<td>6</td>
<td>1.25</td>
<td>1.44</td>
</tr>
<tr>
<td>7</td>
<td>1.41</td>
<td>1.40</td>
</tr>
<tr>
<td>8</td>
<td>1.30</td>
<td>1.30</td>
</tr>
<tr>
<td>9</td>
<td>1.70</td>
<td>1.72</td>
</tr>
<tr>
<td>Average</td>
<td>1.38</td>
<td>1.43</td>
</tr>
</tbody>
</table>

Table 5.9. Model II result of using pel-recursive motion compensation, where motion vectors are derived from the LL subband. From image "Newscaster", $\varepsilon = 0.001$, $T2 = 15$. 
HH subband, with large spatial gradients caused by edges. It was also shown in section 4.3.7, that the pel-recursive motion compensation scheme performed better on a lowpass filtered image.

<table>
<thead>
<tr>
<th>Frame No.</th>
<th>Entropy of displaced frame difference (bits/pixel)</th>
<th>PSNR (dB)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>LL</td>
<td>LH</td>
</tr>
<tr>
<td>1</td>
<td>0.01</td>
<td>0.02</td>
</tr>
<tr>
<td>2</td>
<td>2.21</td>
<td>2.56</td>
</tr>
<tr>
<td>3</td>
<td>1.64</td>
<td>1.51</td>
</tr>
<tr>
<td>4</td>
<td>1.60</td>
<td>1.63</td>
</tr>
<tr>
<td>5</td>
<td>1.37</td>
<td>1.24</td>
</tr>
<tr>
<td>6</td>
<td>1.21</td>
<td>1.44</td>
</tr>
<tr>
<td>7</td>
<td>1.37</td>
<td>1.24</td>
</tr>
<tr>
<td>8</td>
<td>1.26</td>
<td>1.12</td>
</tr>
<tr>
<td>9</td>
<td>1.79</td>
<td>1.73</td>
</tr>
<tr>
<td>Average</td>
<td>1.38</td>
<td>1.39</td>
</tr>
</tbody>
</table>

Table 5.10. Model II result of using pel-recursive motion compensation, where motion vectors are derived from the HH subband. From image "Newscaster", $\varepsilon = 0.001$, $T2 = 15$.

The above argument can be extended to the LH and the HL subbands which also possess edge information. The entropy results, on the other hand, are slightly high in Table 5.9, notably in the LH, HH and the HL subbands. However, this does not seem to affect the PSNR.

5.3.5 MODEL III, CODING SUBBAND INTER-FRAME DIFFERENCES

The results for this scheme are given in Table 5.11. The motion detection threshold, $T2$, was set equal to 10. From this table it can be noted that the entropy of the frame difference (FD) signals in the LL and the LH subbands are on average lower than the entropy in the HH and the HL subbands. The higher entropy in the HH and the HL
subbands is largely due to higher values of the FD signals which result from the edges.

Table 5.11 also shows a drop in PSNR from a value of 41 dB after the first frame to a value of 34 dB after the tenth frame, the average value being 36 dB. As more and more frames were received the FD increased due to changes in the image caused by motion. But since the bit rate was constrained by the symmetrical quantiser the PSNR decreased.

5.4 MODEL COMPARISONS

This section compares the performance of models I, II and III. Figures 5.2 to 5.6 show the PSNR for the models over the duration of 9 frames of the sequence "Newscaster".

Figure 5.2 shows the PSNR of models I and II with both coding schemes employing block matching motion compensation. Figure 5.3 compares the PSNR of models I and II, with each scheme using the pel-recursive motion compensation. It is observed from

<table>
<thead>
<tr>
<th>Frame No.</th>
<th>Entropy of frame difference (bits/pixel)</th>
<th>PSNR (dB)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>LL</td>
<td>LH</td>
</tr>
<tr>
<td>1</td>
<td>0.01</td>
<td>0.01</td>
</tr>
<tr>
<td>2</td>
<td>1.48</td>
<td>1.52</td>
</tr>
<tr>
<td>3</td>
<td>0.87</td>
<td>0.92</td>
</tr>
<tr>
<td>4</td>
<td>1.01</td>
<td>0.93</td>
</tr>
<tr>
<td>5</td>
<td>0.99</td>
<td>0.85</td>
</tr>
<tr>
<td>6</td>
<td>0.78</td>
<td>0.83</td>
</tr>
<tr>
<td>7</td>
<td>0.84</td>
<td>0.66</td>
</tr>
<tr>
<td>8</td>
<td>0.78</td>
<td>0.78</td>
</tr>
<tr>
<td>9</td>
<td>1.04</td>
<td>0.85</td>
</tr>
<tr>
<td>Average</td>
<td>0.87</td>
<td>0.82</td>
</tr>
</tbody>
</table>

Table 5.11. Model III (subband frame differences) result of entropy and PSNR. From image "Newscaster", \( T_2 = 10 \).
II, with each scheme using the pel-recursive motion compensation. It is observed from Figures 5.2 and 5.3 that with either block matching or the pel-recursive motion compensation schemes, model II performs significantly better than model I. The PSNR for Model I starts to fall well below that of model II just after the second frame, with the average difference in PSNR reaching 3.4 dB after frame 9. The advantage of model II over model I is signified by the average PSNR and the average entropy results in Table 5.12.

![Figure 5.2. PSNR result of modes I and II, using block matching motion compensation](image1)

![Figure 5.3. PSNR result of modes I and II, using pel-recursive motion compensation](image2)
Table 5.12. Average entropy and average PSNR of models I and II.

Figure 5.4 compares the PSNR of model I by separately employing block matching and the pel-recursive motion compensation. It can be seen from Figure 5.4 that the PSNR results are almost equivalent.

![Figure 5.4. PSNR result for model I, using pel-recursive and block matching motion compensation](image)

Figure 5.5 shows the PSNR for model II, separately using the pel-recursive and block matching motion compensation. The independent motion estimation result was used. It is interesting to note that initially both methods perform equally, however, as more frames are received the pel-recursive method falls 0.8 dB below the block matching result after 9 frames. This result is attributed to poor convergence of the pel-recursive motion estimation in some areas. It was noted in section 4.3.7 that the pel-recursive
method requires a large number of iterations to converge. In order to reduce computational delay only 20 iterations were allowed in the simulations.

Figure 5.5. PSNR result of model II, using pel-recursive and block matching motion compensation

Figure 5.6 compares the PSNR results for models I, II and III together. Only the block matching result is shown for models I and II. From Figure 5.6, it can be noted that model II has the best result of PSNR. An interesting observation is that initially the PSNR for model III is less than that of models I and II. However, after 3 frames the result of model I falls below that of model III by 2 dB. Table 5.13 summarises this result in terms of the average entropy of the prediction error and the average PSNR. From Table 5.13, it is noted that model III achieves a lower average entropy for the prediction error than either models I and II. This can be explained by recalling that in model III all image areas that are classified as stationary have their inter-frame differences truncated to zero. This implies that for the sequence “Newscaster”, which possesses little motion, most of the subband inter-frame differences are zero, hence the low value in the entropy. However this low entropy did not translate into a significant improvement in the PSNR because of the quantisation noise introduced by fixed quantisation and the error introduced by wrongly attributing zero displacement to areas that would otherwise have non-zero displacement. The effect of the last factor could be
alleviated by changing the threshold, \( T_2 \). Overall, the benefit of model III could be realised by using an adaptive or a non-uniform quantiser.

![Graph showing PSNR over number of frames](image)

**Figure 5.6.** Comparison of PSNR for Models I, II and III.

<table>
<thead>
<tr>
<th>Model</th>
<th>Average entropy of displaced frame difference (bits/pixel)</th>
<th>Avrg. PSNR (dB)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>LL</td>
<td>LH</td>
</tr>
<tr>
<td>I</td>
<td>1.42</td>
<td>1.39</td>
</tr>
<tr>
<td>II</td>
<td>1.36</td>
<td>1.44</td>
</tr>
<tr>
<td>III</td>
<td>0.87</td>
<td>0.82</td>
</tr>
</tbody>
</table>

**Table 5.13.** Average entropy and average PSNR of models I, II and III.

Figure 5.7 shows the input and output images (only frame number 8 is shown) from the sequence "Newscaster". Figure 5.7 (a) shows the input image. Figures 5.7 (b) and (c) show the output images obtained by using block matching and the pel-recursive motion compensation schemes in model II. Only the output image for model II is shown here because the aim is to compare the effects of block matching and pel-recursive motion compensation schemes on the output images. Figure 5.7 (d) shows the output image from model III. From the output images it was observed that the image in Figure 5.7 (d) (model III) was more 'noisy' than that in Figures 5.7 (b) and (c).
However, a comparison between the block matching motion compensation result in Figure 5.7 (b) and the pel-recursive method result in Figure 5.7 (c) reveals some small artefacts in Figure 5.7 (b) along the edges of the face, left shoulder and the hair, which are attributed to the assigning of common motion estimates to all pixels within each block.

![Figure 5.7](image)

**Figure 5.7.** Comparison of input and output images (frame number 8). - From the sequence "Newscaster". (a) input frame; (b) output frame, using block matching method in model II; (c) output frame, using the pel-recursive method in model II, (d) output frame from model III (using frame differences).

In general, it was noted that model III was computationally simpler to implement than models I and II. Due to its hierarchical structure, model II was found to be computationally more flexible than model I. In particular, it was realised that in model
II subband analysis and motion compensation could be performed by separate programs at the receiver, which was found to be very suitable for the UNIX environment. In model I it was noted that motion compensation and subband analysis had to be performed within the same program, resulting in longer delays for each simulation.

### 5.5 CONCLUSION

This chapter has presented the performance of inter-frame subband models I, II and III. Models I and II have been compared by using block matching and pel-recursive motion compensation. The effectiveness of model III has also been compared to model I and model II.

On the basis of the simulation results and for the test sequence used, model II has a better performance over model I, both with block matching and the pel-recursive motion compensation. This is also reflected in the lower values of the entropy for prediction error signals and higher peak signal-to-noise ratio (PSNR). This result suggest that motion compensation in the subband image is more effective than in the full band image.

The simulation results also suggest that the performance of model I, in terms of the PSNR and the entropy, is about the same under either block matching or pel-recursive motion compensation schemes. However, model II shows some significant differences in performance. The results of employing block matching in model II indicate that for the image sequence used there is no gain in estimating the motion vectors for each subband, suggesting that savings in computation can be attained by sharing the motion information computed from just one subband. The simulations also show that the pel-recursive motion compensation performs best in model II when the motion estimates
are derived from the lowpass subband (LL). This is due to the poor performance of the pel-recursive motion estimation in the subbands with edge details.

In general, model II results show that higher PSNR as well as lower entropy for the subbands can be achieved with block matching motion compensation. However the visual quality of the output images obtained with the pel-recursive motion compensation have slightly less visible artefacts than those obtained with block matching.

Model III has been shown to perform remarkably well with the test sequence "Newscaster". It performs better than model I and is comparable to model II. This result suggests that a simple scheme such as coding of the subband frame differences is sufficient in scenes such as "Newscaster" which seems to fit the "head and shoulder" type of images, containing little motion.
CHAPTER 6
CONCLUSIONS

Three different schemes for coding image sequences using the concepts of subband analysis/synthesis and inter-frame prediction have been simulated and compared.

Models I, and II integrate motion compensation, which is the basis of most current image sequence coding applications, with subband analysis/synthesis. Two most common motion estimation schemes have been employed in the motion compensation, namely, block matching and the pel-recursive methods. Model III is based on direct inter-frame differences of image subbands. In all the models, subband analysis and synthesis was implemented by using 2-D separable near-perfect reconstruction QMF. The performance comparison considers the entropy of the subbands, the peak signal-to-noise ratio (PSNR) and the visual quality of the output images. The simulation results were generated from the 10-frame TV sequence "Newscaster" of size 512 x 512 pixels.

Model I was simulated in two ways, namely, by using block matching and the pel-recursive motion compensation. The results, in terms of the PSNR and entropy of the MC prediction error show that model I performs equally under both the pel-recursive and block matching motion compensation.

Model II was also considered by using block matching and the pel-recursive motion compensation schemes, which resulted in two more image coding strategies. The results for block matching are significantly better than the pel-recursive method. This is mainly due to the slow convergence of the pel-recursive motion estimation algorithm. Test results in chapter 4 have shown that more iterations are necessary in order to allow for the pel-recursive motion estimation algorithm to converge. The reconstructed image
displays slightly more artefacts with block matching than with the pel-recursive motion compensation.

On the basis of the results obtained, and for the test image used, this study has shown that there is no gain, in terms of the PSNR and the entropy of the prediction error signals, in separately estimating motion vectors in the different subbands. Instead, motion vectors derived from one subband can be allocated to all the other subbands. This agrees with the result obtained by Gharavi, cited in Woods, 1991, which was exclusively based on block matching motion compensation. The result obtained here goes further to show that if the pel-recursive motion estimation is employed in model II, the best performance is obtained when the motion vectors are derived from the lowpass subband.

Comparison of models I and II under both block matching and the pel-recursive motion compensation schemes has shown that model II performs better by about 4 dB (PSNR). The superiority of model II was attributed to the better performance of motion compensation in the subband domain than in the full band image. The difference in performance between model I and II is significantly more than that obtained by Gharavi. In their investigation, which was based on block matching motion compensation, model II was better by 0.5 dB.

The performance of model III which encodes direct inter-frame differences of the image subbands was shown to be less than model II but significantly better than model I. Since model III is computationally much simpler than either models I or II, this result demonstrates that in scenes where the motion is small, such as "Newscaster", there is no gain in using motion compensation. Hence, coding the subband inter-frame differences in such scenes is simpler and equally effective.
The following are some suggestions for future improvements to the work that has been discussed.

In this study the effect of the quantiser on the performance was assumed to be irrelevant in the comparison of the models, since the same simple symmetrical quantiser was employed in all the subbands in the models. However, improvement in performance can be achieved by investigating the different ways of quantising the motion compensated prediction error subband signal. One suggestion is to use a separate symmetrical quantiser with a suitable dead zone for each subband. Alternatively, adaptive quantisers could be designed for each subband.

Significant improvement in the PSNR can be obtained for all the models by employing perfect reconstruction (PR) QMF banks. However the PSNR of model I is still expected to be less than that of model II. It has been observed that motion compensation is significantly better in the subband domain than on the full band image.

One of the advantages envisaged for inter-frame subband coding is the flexibility to separately encode each subband by incorporating human visual criteria. This has not been considered in this study. Possible improvement in the perceptual quality of the output image can be achieved if such information is incorporated in the models.

Another possible improvement is in the motion estimation. This thesis has discussed several methods of motion estimation and, in particular, improvements in accuracy and efficiency of both block matching and pel-recursive methods. These techniques and others can be pursued to improve the accuracy of the motion estimation schemes, especially the pel-recursive method. One such area is to optimise the motion detection and image segmentation algorithms.


A.1 SYNTHETIC TEST IMAGE GENERATION

The synthetic image "Rings" was generated according to the following function (Netravali and Robbins, 1978):

\[
f(x, t) = \begin{cases} 
127 & \text{if} \quad \|R\| > 100 \\
127(1 + \exp(-0.05\|R\|))\cos(0.2\pi\|R\|), & \text{otherwise}
\end{cases}
\]  

(A.1)

where \(f(x, t)\) is the intensity of the pixel at the location \((x)\), \(R = x - (x_0 + Dt)\), 127 is the background intensity, where the intensity is ranging between 0 - 255, and \(x_0\) is the centre of the moving pattern at \(t\) (time) = 0. \(D\) is the amount of two-dimensional translation by which the image is shifted from one frame to the next. \(\|\cdot\|\) indicates the magnitude. The image translation is implemented in a utility function called IMG_SHFT. The inputs to this function are; image sequence name, the total number of frames in the sequence, the x-displacement, y-displacement, the size of each frame and the x and y centre coordinates of the first frame.

A.2 TWO DIMENSIONAL GRADIENT ESTIMATION

Gradients are the most commonly used operators in image differentiation. The gradient of an image \(f(x, y)\) at location \((x, y)\) is defined as the two-dimensional vector

\[
\nabla f(x, y) = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix}
\]  

(A.2)
One approach to compute the gradients is by approximating the partial derivatives, \( \frac{df}{dx} \) and \( \frac{df}{dy} \) by first order differences. From Figure (A.1), the gradient at the pixel \((x,y)\) can be obtained by;

\[
\nabla_x = \frac{\partial f(x,y)}{\partial x} = \frac{f_3 - f_1 + f_9 - f_7}{2}
\]

and

\[
\nabla_y = \frac{\partial f(x,y)}{\partial y} = \frac{f_1 - f_7 + f_3 - f_9}{2}
\]

The gradient at \((x,y)\) in Figure A.1 can also be calculated by using the Sobel operator (Gonzalez and Wintz, 1987), to obtain the following expressions;

\[
\nabla_x = (f_3 + 2f_6 + f_9) - (f_1 + 2f_4 + f_7)
\]

\[
\nabla_y = (f_7 + 2f_8 + f_9) - (f_1 + 2f_2 + f_3)
\]

Figure A.1. Pixel configuration used in computing intensity gradients.
A.3 TWO DIMENSIONAL LINEAR INTERPOLATION

Consider an image raster shown in Figure A.2, with four surrounding pixels $a$, $b$, $c$, and $d$. If a non-integral valued two-dimensional displacement, $D$, is estimated, it can be written as the sum of an integral part $D_1$ and a fractional part $D_2$, as shown in Figure A.2. The fractional part, $D_2$, can be resolved into the $x$, $y$ components $\Delta x$ and $\Delta y$, respectively. Let $f$ represent the pixel intensities. Then, at the intervening position (highlighted), the intensity can be interpolated by using the following equation (Netravali and Robbins, 1978):

$$f = (1 - \Delta y)((1 - \Delta x)f_d + \Delta x f_e) + \Delta y((1 - \Delta x)f_b + \Delta x f_a) \quad (A.5)$$

Figure A.2. Two dimensional linear interpolation. The displacement, $D$, is decomposed into an integral part $D_1$ and a fractional part $D_2$. 
APPENDIX B
FIR FILTER COEFFICIENTS

Table B.1 lists the coefficient values for two sets of quadrature mirror filter designs discussed in the example section 4.4. One set is based on the Chebyshev approximation design and the other is based on the Johnston filters type - C (Johnston, 1980).

<table>
<thead>
<tr>
<th>Coeff. #</th>
<th>Chebyshev Filter</th>
<th>Johnston Filter</th>
</tr>
</thead>
<tbody>
<tr>
<td>15</td>
<td>0.433044</td>
<td>0.46640530</td>
</tr>
<tr>
<td>14</td>
<td>0.162219</td>
<td>0.12855790</td>
</tr>
<tr>
<td>13</td>
<td>-0.069041</td>
<td>-0.099802430</td>
</tr>
<tr>
<td>12</td>
<td>-0.070933</td>
<td>-0.039348780</td>
</tr>
<tr>
<td>11</td>
<td>0.026539</td>
<td>0.052947450</td>
</tr>
<tr>
<td>10</td>
<td>0.041960</td>
<td>0.014568440</td>
</tr>
<tr>
<td>9</td>
<td>-0.010174</td>
<td>-0.031238620</td>
</tr>
<tr>
<td>8</td>
<td>-0.025962</td>
<td>-0.0041874830</td>
</tr>
<tr>
<td>7</td>
<td>0.002466</td>
<td>0.017981450</td>
</tr>
<tr>
<td>6</td>
<td>0.015501</td>
<td>-0.00013038590</td>
</tr>
<tr>
<td>5</td>
<td>0.000986</td>
<td>-0.0094583180</td>
</tr>
<tr>
<td>4</td>
<td>-0.008506</td>
<td>0.0014142460</td>
</tr>
<tr>
<td>3</td>
<td>-0.002049</td>
<td>0.0042341950</td>
</tr>
<tr>
<td>2</td>
<td>0.004564</td>
<td>-0.0012683030</td>
</tr>
<tr>
<td>1</td>
<td>0.003594</td>
<td>-0.0014037930</td>
</tr>
<tr>
<td>0</td>
<td>0.000744</td>
<td>0.00069105790</td>
</tr>
</tbody>
</table>

Table B.1. Coefficients for 32-TAP Chebyshev and Johnston (Type -C) filters.