ProQuest LLC.
789 East Eisenhower Parkway
P.O. Box 1346
Ann Arbor, MI 48106 - 1346

Improved Quality Block-Based Low Bit Rate Video Coding

by

Teck Hock KWEH

Thesis submitted to the University of Surrey

for the degree of

Doctor of Philosophy

Centre for Communication Systems Research
University of Surrey
Guildford, Surrey
United Kingdom

June 1998

Summary

The aim of this research is to develop algorithms for enhancing the subjective quality and coding efficiency of standard block-based video coders. In the past few years, numerous video coding standards based on motion-compensated block-transform structure have been established where block-based motion estimation is used for reducing the correlation between consecutive images and block transform is used for coding the resulting motion-compensated residual images. Due to the use of predictive differential coding and variable length coding techniques, the output data rate exhibits extreme fluctuations. A rate control algorithm is devised for achieving a stable output data rate. This rate control algorithm, which is essentially a bit-rate estimation algorithm, is then employed in a bit-allocation algorithm for improving the visual quality of the coded images, based on some prior knowledge of the images.

Block-based hybrid coders achieve high compression ratio mainly due to the employment of a motion estimation and compensation stage in the coding process. The conventional bit-allocation strategy for these coders simply assigns the bits required by the motion vectors and the rest to the residual image. However, at very low bit-rates, this bit-allocation strategy is inadequate as the motion vector bits takes up a considerable portion of the total bit-rate. A rate-constrained selection algorithm is presented where an analysis-by-synthesis approach is used for choosing the best motion vectors in term of resulting bit rate and image quality. This selection algorithm is then implemented for mode selection. A simple algorithm based on the above-mentioned bit-rate estimation algorithm is developed for the latter to reduce the computational complexity.

For very low bit-rate applications, it is well-known that block-based coders suffer from blocking artifacts. A coding mode is presented for reducing these annoying artifacts by coding a down-sampled version of the residual image with a smaller quantisation step size. Its applications for adaptive source/channel coding and for coding fast changing sequences are examined.

Acknowledgements

I would like to take this opportunity to express my sincere thanks to my supervisor Prof. Ahmet Kondoz, whose support, guidance and suggestion throughout my research were most helpful and very much appreciated. I am also indebted to Dr. Faruk Eryurtlu for his assistance and advice, and my friends in the multimedia communication research group, especially Chana Sriratanaban, whose support and friendship have been invaluable.

I would also like to thank my family for their support and encouragement throughout the course of my research. Finally, a special mention must go to Wang Yi-chun as without her support I would not have completed this course of research.

Table of Contents

List of Figures

Glossary and Abbreviations

AbS	Analysis-by-Synthesis
ATM	Asynchronous Transfer Mode
BER	Bit Error Rate
BMME	Block Matching Motion Estimation
CCIR	International Radio Consultative Committee
CIF	Common Intermediate Format
DCT	Discrete Cosine Transform
DPCM	Differential Pulse Code Modulation
DSP	Digital Signal Processor
DWT	Discrete Walsh Transform
FEC	Forward Error Correction
FLOP	Floating point Operation
GOB	Group Of Block
GSM	Global System for Mobile communication
HDTV	High Definition Television
HVS	Human Visual System
IDCT	Inverse Discrete Cosine Transform
ISDN	Integrated Service Digital Network
KLT	Karhunen-Loeve Transform
LAN	Local Area Network
LBG	Linde-Buzo-Gray
LOT	Lapped Orthogonal Transform
MAE	Mean Absolute Error
MB	Macroblock
MPEG	Moving Picture Experts Group
MSE	Mean Square Error
MV	Motion Vector
OBMC	Overlap Block Motion Compensation
PSNR	Peak-to-peak Signal-to-Noise Ratio
PSNR_Y	Luminous PNSR
PSTN	Public Switched Telephone Network
QCIF	Quarter Common Intermediate Format
QP	Quantisation Step size
RCPC	Rate-Compatible Puncture Convolutional Code
ROI	Region of Interest
SAD	Sum of Absolute Difference
SNR	Signal-to-Noise Ratio
UEP	Unequal Error Protect
VLC	Variable Length Code
VLSI	Very Large Scale Integrated circuit

Chapter 1

Introduction

1.1 Background and Objectives

Visual communications have been commonly regarded as a next-generation communication tool, where video coding is a key element to its success. With the advances in digital signal processing, compounded with the rapid developments in the semiconductor technologies, real-time video coding which was once thought to be technically infeasible or economically unreasonable, are now a reality[1].

From the technical point of view where complex coding and decoding systems are concerned, it is not possible to establish communication on a casual basis, and this problem is complicated by the requirement to use the same channel for a multiple of services such as speech, video and data. On the other hand, the economics of market forces dictate that compatibility among video codecs from different manufacturers and applications have to be ensured. Consequently, this leads to the establishment of several digital video standards by standardisation bodies such as ISO and ITU.

For the research community, the setting up of standards is very much of a mixed blessing, since such a move will inevitably lead to a drop in the interest in new algorithm development by industry in the face of the production of hardware to satisfy the agreed specification [1]. In view of this, the main objective of this thesis is to develop efficient techniques that are capable of improving the performance of the

current block-transform video coding standards [32]-[34][60]. The main application is targeted at low bandwidth applications such as visual communication over PSTN, radio links or the increasingly available ISDN network [71]. Thus, the maximum bit-rate adopted in the course of the work will be less than or equal to 64 kb/s to ensure the suitability of the proposed algorithm for a channel with a maximum capacity of 64 kb/s.

1.2 Source Material

To evaluate the efficiency of the developed techniques and algorithms, various standard ITU test sequences of different properties have been utilised. Miss America, Claire, Suzie, Carphone and Foreman are the five sequences used in most of the simulations, although other sequences are also used. Miss America and Claire are typical low motion head-and-shoulder sequences. They can be coded at a very low bit-rate because of their uniform and stationary background. Suzie is another head-and-shoulder sequence with high contrast and moderate noise. It contains a fast head motion when she shakes her hair. Carphone, though not a typical head-and-shoulder sequence, shows a talking head in a moving vehicle with more motion in the speaker and a non-uniform, changing background. Foreman, which is the most complex, shows another talking person but with much more motion and a shaking background due to camera panning. Finally, Silent Voice is the only sequence employed for demonstrating the specific property of a proposed algorithm (Chapter 6). It shows a woman using sign language to communicate with her audience. This sequence has a relatively detailed background compared to the above five sequences. Except for Silent Voice which is a CIF (Common Intermediate Format) sequence, all the test sequences used are of QCIF (Quarter CIF) format containing 176 x 144 pixels. Fig. 1.1 shows some original frames of these sequences with the exception of Silent Voice which has been down-sampled to 1/4 of its original size.

(a) Claire

(b) Miss America

(c) Suzie

(d) Carphone

(e) Foreman

(f) Salesman

(g) Grandma

(h) Silent Voice

Fig. 1.1 : Original picture of some of the test sequences.

1.3 Performance Evaluation and Image Quality Assessment

Since the bit-rates used for encoding the test sequences are very low, quality degradation is inevitable in the output images. Therefore, both subjective and objective methods have been adopted for evaluating the performance of the proposed algorithms. Although the measurement of the subjective quality is quite cumbersome compared to the calculation of numerical values for the objective quality, it is more preferable for very low bit-rate compression because of the inconsistency between the existing numerical quality measures and the Human Visual System (HVS).

There are two broad types of subjective evaluations, rating scale methods and comparison methods [2]. In the first method, an overall quality rating is assigned to the image by using one of several given categories. In the second method, an impairment of a standard type is added to a reference image until the impaired and reference image are of equal quality. However, throughout this thesis, pair comparison where the reference sequence and output sequence from the proposed algorithms are displayed side by side for evaluation.

The quality of the image sequence can also be measured by using some mathematical criteria such as signal-to-noise ratio (SNR), peak-to-peak SNR (PSNR) or mean-square-error (MSE) [1]. These measures are considered to be objective due to the fact that they rely on individual pixel values of the original and reconstructed video frames and do not have any relationship with the human visual system. For image and video, PSNR is preferred for objective measurement and has increasingly been used by the research community, although the other two are still occasionally used [1].

$$PSNR = 10\log_{10}\frac{255^2}{\frac{1}{MxN}\sum_{i=0}^{M-1}\sum_{j=0}^{N-1}[x(i,j)-\hat{x}(i,j)]^2} \tag{1.1}$$

$$MSE = \frac{1}{MxN} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} [x(i,j) - \hat{x}(i,j)]^2 \quad (1.2)$$

where M and N is the size of image,

x and $\hat{x}$ are the original and reconstructed pixels respectively.

In addition, for fair performance evaluation of a video coding algorithm, the bit-rate must also be included. The output bit-rate of a video coder is expressed in bits per second (b/s). Since bit-rate is directly proportional to the number of image pixels and frames per second, both image size and frame-rate have to be indicated in the evaluation process too.

1.4 Outline of Thesis

The work presented in this thesis is primarily based upon the current ITU H.263 standard [3] for low bit-rate video coding. The area of research undertaken was mainly on investigating techniques for improving the subjective quality and coding efficiency of standard block-transform video coder for transmission over channel with bandwidth less than or equal to 64 kb/s.

Chapter 2 gives a very brief survey of some video applications which required low bit-rate video compression. This is accompanied by a short discussion on the transmission networks that these low bit-rate applications are most likely to be used. Finally, we looks into some of the user requirements of real-time video communication. This includes issues such as visual quality, complexity, delay and synchronisation which are important for real-time video communication.

Chapter 3 gives a very brief introduction to the basic features of a video coder. The main characteristics and principles of some of the popular coding schemes, mostly

first-generation pixel-based approaches and a second generation region-based approaches, are then briefly reviewed. The major difference in approach between the two generations is identified. In the second part of the chapter, the main features of the ITU-T H.263 video coding standards are described. Its differences from the other standards are pointed out. Finally, the usefulness of the four negotiable options, available only to H.263, is assessed with the support of simulation results.

Chapter 4 begins with an introduction to the subject on rate control in block-based video coder for fixed-rate transmission. The commonly used methods for rate control are described. Next, a table-based coded-bit estimation techniques for estimating the number of bits required to code a block of prediction error is presented. A feedforward rate control algorithm for achieving stable output bit-rate is described. Simulation results of the above described algorithm is presented. Subjective image quality is addressed in the second part of the chapter. By making use of the prior knowledge of some incoming image sequences, the region-of-interest (ROI) can be located and coded with less coding degradation. For example, for typical teleconference and videophone images, the attention is on the face of a person. A dynamic bit-allocation algorithm is implemented which allocates more bits to the ROI with the constraint that the resultant bit-rate is subjected to the same bit budget.

Chapter 5 looks into the use of motion compensation, which is an essential component of a hybrid video coding system for removing temporal redundancy, for low bit-rate video coding. A rate-constrained selection algorithm based on analysis-by-synthesis approach is devised for motion vector selection. The performance of the algorithm is presented and analysed in detail. A brief discussion on the added computation is also included

The second part of the chapter focus on the issue of mode switching in block-based video coder. First, the different operation modes available for current block-based standards are examined. Then, a mode selection strategy based on the above rate-constrained selection algorithm is presented. The performance of the proposed switching strategy is compared with that of TMN5 [39]. Next, the complexity of the switching strategy is examined and a simplified algorithm is proposed and presented.

Comparison between the simplified algorithm, its complicated counterpart and the TMN5 are then discussed. The chapter concludes with a look into the joint selection of the motion vectors and operating modes.

In Chapter 6, a reduced-resolution coding mode is developed. This mode is expected to be used at very low rate coding when the quantisation step size for coding the prediction error is very large. Its application for adaptive source/channel coding under high channel error rate condition is first examined. Then, its usefulness is studied through simulations carried out on coding sequences containing fast motion

The final chapter is a summary of the results presented in the preceding chapters. In addition, a section is included where possible directions for future research are proposed.

In summary, the work contended in this thesis which is believed to be original or contributory is as follows:

- The development of a table-based coded-bit estimation algorithm for estimating the number of bits required to code a block of prediction error of a block-transform video coder.

- Implementation of a feedforward rate control algorithm based on current frame picture activity for achieving a stable output data rate.

- Implementation of a bit allocation algorithm for improving the subjective quality of a block-based coder through adaptive quantisation.

- The development of a rate-constrained selection algorithm based on the analysis-by-synthesis concept for motion vector selection in block matching motion compensation scheme.

- The development of a rate-distortion optimised operation mode switching strategy for block-based video coder, including a simplified algorithm for reducing the computational complexity.

- The design and implementation of a coding mode for coding of motion-compensated prediction error, in order to improve the subjective quality of the decoded image at very low bit-rate.

Chapter 2

Digital Video : Applications and Transmission Networks

2.1 Introduction

Rapid development of communication networks and computer technology have supported the promotion of the importance of visual information in our communication and information environment. On the other hand, the increasing convergence of the technologies of broadcasting, computing and telecommunications has led to the emergence of a wide variety of communication products and applications. Driving the research and development in the field of digital video are the consumer and commercial applications such as:

- digital TV, including HDTV,
- videoconferencing
- videophone and mobile image communications
- multimedia applications such as interactive multimedia databases, electronic newspapers.

Other applications include surveillance imaging for military or law enforcement, intelligent vehicle highway systems, motion picture production and so on. For the high and medium bit-rate applications, international compression standards such as H.261 [32], MPEG1 [33] and MPEG2 [34] have been available for them for a number

of years. Although the recently established ITU-H.263 standard [3] is capable of achieving a low coding bit-rate, its main targeted application is videotelephony over PSTN and is considered to be too sensitive to bit errors for some applications [35]. As for the increasing demand of a generic standard for addressing the many requirements of a host of multimedia applications, this will be soon answered by the emerging MPEG-4 standard [67].

2.2 Low Bit-Rate Video Applications

Video compression has a wide variety of applications at very low bit-rates. Low bit-rate generally refers applications that require less than 64 kb/s [1]. A primary application is for improving the quality of video communication using ordinary telephone networks, where videophones are the most obvious example [60]. The ongoing boom in the use of Internet and the World Wide Web also increases the needs for very low bit-rate video compression in multimedia applications in this environment. In addition, the introduction of a visual element into wireless communications can only be realised by very low bit-rate video coding as bandwidth is severely limited. There are so many applications of low bit-rate video coding that the following are only some of the more prominent examples:

- Videophone: Videophone service on PSTN and mobile networks. These applications require real-time encoder with easy implementation [72].

- Mobile multimedia communication such as cellular videophones and other personal communication systems [72].

- Remote Monitoring: One way communication of audio-visual information from a remote location, for example, surveillance, security, intelligent vehicle highway [72].

- Multimedia applications such as multimedia electronic mail, multimedia videotex, electronic news service on PSTN and radio channels, etc [72].

2.2.1 Videophone

Videophony is a companion audiovisual service to videoconferencing where low cost and operation over the existing networks is extremely important. It may be distinguished by the following features.

1. It is primarily for person-to-person as with telephony rather than group-to-group audiovisual communication.
2. It is usually an on-demand service provided on customer-switched networks, whereas videoconferencing may need to be scheduled in advance and may use fixed links.

The two persons who mainly communicate expect the motion video to convey emotional information such as expressiveness to support the dialogue, via eye-to-eye contact. The image resolution can be medium or even low, as no details need to be accurately displayed. A low frame rate of 6-12 fps is usually acceptable [38].

The networks most likely to be used for such a service are analogue PSTN and narrowband integrated services digital network (ISDN), but other networks such packet-switched Internet, local area networks (LANs) and the future broadband ISDN are envisaged. Moreover, recognising the increasing demand for low bit-rate applications for mobile communications, intensive research has focused in this area in recent years [64]. Wireless cellular communication such as Global System for Mobile Communication (GSM) pose more difficult challenges for video compression where the available bit-rate may be 13 kb/s and less. The channel noise in this environment is also a major problem for maintaining acceptable video quality.

2.2.2 Remote Monitoring

There are many needs for video surveillance of remote sites and facilities via various combinations of telephone and Internet connections. For example, small business owners can dial up from their homes to visually inspect business premises, without

traveling to the remote location. As very low bit-rate video communications capabilities improve, applications such as these will become more common.

2.2.3 Interactive Video and Multimedia

There are a number of definitions to multimedia. Here it refers to the ability to provide full-motion interactive digital video in the personal computer (PC) or desktop workstation environment. Multimedia also involves text, graphics, sound, and still-images, which have long existed in the PC or workstation environment [72].

Multimedia applications suggest other needs for video compression at low bit-rates. For example, multimedia CDROM programs typically provide interactive choices for users to select video segments and control playback. In the future, individual objects within video segments may be selected and combined with other objects, and a standard video representation is needed to support this. Future video compression systems will also comprehend interactive requirements and support selective video encoding in real time. Users may specify objects to view without the background. Other multimedia needs include selective embedding of video in graphics. The emerging MPEG-4 is envisaged to meet these and many other requirements of the vast variety of multimedia applications by offering a number of functionalities for supporting object manipulation [67]. The following is an example of multimedia application where real-time or near real-time video is desirable.

2.2.3.1 Public Video News-On-Demand

Stored movies are not the only type of motion video that public operators may make available as in the case of movie-on-demand. Any kind of stored information, such as news, may be offered in a similar way.

- General video news-on-demand

Pilot projects have started with a varying degree of success to deliver on-demand stored video news of varying types such as general, sport, traffic, weather forecast, etc. to residential subscribers. These services tend to evolve from the strict availability of motion video sequences, to delivery of multimedia news including textual and still picture information, and offering a higher degree of interaction than that usually available from movie-on-demand applications [72].

- Business news-on-demand

Publishers of business and financial news have shown interest in developing services for video news-on-demand, generally targeting professional subscribers. Like general news, the tendency is to deliver business news in multimedia form rather than motion video only, but also to mix the on-demand mode with spontaneous distribution of hot information such as financial news alerts [72].

2.3 Video Transmission : Network Perspective

A telecommunication system is needed because users are remotely separated from each other or from some server-provided applications. This system must provide for the transport of information signals from the originating source to the required destination. In the following, we shall look at some of the public networks which are most likely to carry the above low bit-rate applications.

2.3.1 PSTN

The traditional analogue telephone networks is a connection-oriented network as connection must first be set up before communication can proceed. The original application for this network is voice-communication only. They have been updated into networks which are expected to form the basis of the emerging world-wide ISDN. However, digitisation of the denser local loop remains elusive. This is partly due to

the high costs involved. Nevertheless, by using modem, digital communication at speeds around 28.8 kb/s is achievable with the current technology. However, many if not most point-to-point telephone connections do not adequately support the full 28.8 kb/s rate, and the maximum effective rates may reduce to 19 kb/s or less which might reduce the acceptance of the video quality. For real-time video communication applications on this network, the major problem for satisfactory performance is the limited available bandwidth and thus the perceptual image quality and end-to-end delay, especially for long-distance connections [72].

2.3.2 ISDN

As mentioned in the above sub-sections, all the major network operators started to convert their networks and switching nodes into digital facilities in the 1980s so that data and telephony can be integrated on one network, which formed the Integrated Digital Network or IDN. ISDN evolved from the IDN when the digital channel of the IDN is taking all the way to the customer so that all services can be integrated on one bearer, i.e. when the local loop is also digitised. With several terminals of different types (e.g. telephone, personal computer, facsimile machine) connected to the same network, it is inappropriate to restrict customers to the use of only one channel at a time. Hence, a basic access offering two 64 kb/s channels is standardised which allows the use of two terminals at a time. These channels are called 'B' channels. A 16 kb/s signaling channel is also offered, which may also be used to provide access to a packet-switched service [71].

For videotelephony services, two cases are identified. One based on using only one B-channel and the other based on using two B channels at the user-network interface [71]. For the second case, it is a requirement that each 64 kb/s connection has similar characteristics particularly in terms of delay, i.e. it is not acceptable for one connection to be routed via a satellite and the other by a terrestrial link. Besides videophony, video communication using an ISDN basic rate interface, operating at a total of 128 kb/s, may be used for such applications as distance learning and access to multimedia information services [72].

2.3.3 Internet and Packet Video

In the multimedia communication services available today, Internet is the most popular environment which is accommodating increasingly growing number of users all around the world. The ease and lower cost of the accessibility to Internet makes it a widely populated recipient for the transport and control of real-time multimedia traffic. Unlike PSTN, Internet is a connectionless-oriented network. The connection setup does not involve the reservation of a well defined path for the routing of the video information. The compressed video bit stream from a video coder as in the case with other data types is packetised before sending it as User Datagram Protocal datagrams on the Internet [72].

Besides the problem of data being corrupted during the transmission, another two types of errors are found in packet-switched Internet. First, unlike connection-oriented communication, the datagrams or data packets of the same communication session might be transmitted along different paths depending on the routing strategies adopted by the network. Due to network latency and jitters, the order of packets might be shuffled and the video depacketiser must retrieve the correct sequence of packets before handing them to the decoder for recovery of the video sequence. Then there is also the possibility of packet loss. Video packet loss as well as the reordered packets can be easily detected by checking the sequence number incorporated in each packet following the expiry of a timeout period at the decoder end. Although the reshuffled video packets can be reordered easily, the delay thus incurred may make real-time application impossible. In the case that the video packet is lost or considered lost after the expiry of the timeout period, the strategy for containing its effect on the reconstructed image quality is important for the successful implementation of real-time video services on the Internet [61].

On the other hand, network congestion is another threat to the feasibility of real-time video on Internet. It exaggerates the time delay problem and hence resulting in higher possibility of packet loss. Therefore, it is necessary for the video encoder to control its output data rate according to the status of the network to prevent the occurrence of a congestion.

2.3.4 Mobile Networks and Video

Mobile networks are characterised by a very special feature which consists of the mobility of their nodes. The bandwidth available for transmission is extremely limited. In general, they suffer from the same types of problems as Internet, namely channel error and packet loss. However, due to their use of radio frequencies for carrying intelligence, mobile networks impose a higher level of corruption on the bit stream due to propagation fading, interference and shadowing in populated areas. Then there is also the possibility of packet loss as the mobile hosts roaming around, moving through and leaving a given mobile radio network. The mobile nodes are more susceptible to technical failure than fixed hosts which contributes to a higher level of lost packets [61].

For compressed video, high error rates and packet loss cannot be tolerated if left unchecked. These problems can lead to a disastrous effect on the quality of video. Therefore, error correction codes and complex modulation techniques are usually employed to make the bit stream more robust to channel errors whereas error concealment is believed to be a better solution for tackling the problem of packet loss [61].

2.4 User Requirements

In most cases, the user has to pay for the service used. Therefore a minimum quality of service has to be maintained for the service to justify its worth. For video communication, many of the user requirements are often conflicting, and a fine balance between them has to be achieved.

2.4.1 Video Quality and Transmission Bandwidth

These are frequently the two most important factors in the choice of a video coding algorithm for any application. It is generally agreed that for a given algorithm, the higher the designed transmission bandwidth or output bit-rate of a coder, the better the output video quality. However, in most applications, bit-rate is limited by scarcity of transmission bandwidth and/or power. Consequently, it is necessary to trade the network bandwidth against quality in order to come up with the optimum performance of a video service and an optimum use of the available network resources. On the other hand, the type of running service demands the minimum required quality of video. For instance, videotelephony application requires the image quality to enable the user to identify his participant on the other end. In surveillance applications, the quality can be satisfactory when it enables the onlooker to identify the shape of a human body appearing in the scene. However, the quality of service for telemedicine applications must enable the remote end user to identify the smallest details of a picture and detect its features with high precision. In addition to the type of application, other factors such as frame rate, number of intensity and colour levels, spatial resolution also affect the quality of video sequence [72].

2.4.2 Complexity

The complexity of a video coding algorithm is related to the number of computations carried out during the encoding and decoding processes. A common indication of complexity is the number of FLoating point OPeration (FLOPs) carried out during these processes. This algorithm complexity is essentially different from the hardware or software complexity of an implementation. The latter depends on the state and availability of technology and the quantities required whilst the former provides a basis benchmark for comparison purposes. Of course, in some applications, such algorithm complexity can be made transparent by the use of sophisticated technology.

For real-time communication applications, low cost real-time implementation of the video coder is desirable in order to attract a mass market. To minimise processing

delay in complex coding algorithms, many fast and costly components have to be used, increasing the cost of the overall system. In order to improve the take up rate of new applications, many of the original complex coding algorithms have been extensively simplified. However, recent advances in Very Large Scale Integrated circuits (VLSI) technology have resulted in faster and cheaper Digital Signal Processors (DSP). The low cost and sophistication of these devices seem to have relegated the complexity problem to the background. Another problem related to complexity is power consumption. For mobile applications, it is vital to minimise the power requirement of the mobile terminals in order to prolong battery life [72].

In general, every video codec on today's market falls under one of the following three categories:

(i) Hardware codecs: With hardware design, the coding is performed by a chip set of three or more VLSI chips tightly connected together. The chip set implements high-speed logic specially dedicated to perform the compression algorithm [72].

(ii) Software codecs: Coding is accomplished in software by using one or more microprocessors embedded on dedicated codec boards or designed directly into a system's motherboard.

(iii)Hybrid solutions: Coding is basically software based, as in software codecs, but relies on specific hardware for performing the most demanding computations, such as motion estimation.

The increasing power of standard computer chips has enabled the implementation of some less complex video codecs in standard personal computer for real-time application. However, it is normally necessary to use some kind of special hardware especially for the encoding for achieving better quality video.

2.4.3 Delay

For real-time applications, the time delay between encoding of an image and its decoding at the receiver must be kept as short as possible. While nothing much can be done for the delay introduced by the communication line, the delay due to the codec processing and its data buffering can vary quite a bit. Time delay tends to change with the amount of motion in the encoded scene, growing longer as movement increases due to the reduction in frame rate. To have the lowest possible delay, it is therefore advantageous to use high frame rate. However, this may be in conflict with the picture quality which may need to be reduced. A compromise therefore often has to be made between picture quality, temporal resolution and coding delay. Time delays greater than 0.5 seconds are usually annoying and cause funny synchronization problems with the other participants in the video communication [1][72].

The end-to-end delay in video communication systems can be divided into encoder delay, the transmission channel delay and the video decoder delay. Each of these components will be discussed in more detail in the following sub-sections.

- **Encoder delay**: which consists of encoder processing delay and encoder-wait-for-data. The former arises from the processing associated with transformation of the data representing the image to the encoded bit-stream presented to the channel for transmission. The processing time depends very much on the different implementation of the codec. Generally, hardware codecs give the smallest time delay while software codecs give the longest processing delay.

  On the other hand, the encoder-wait-for-data arises from a need to wait for the acquisition and processing of a frame that occurs later in the source stream than does the current frame before encoding of the current frame can carry out. This occurs with the use of forward referencing such as B frame of PB-frame option in H.263 [3]. This values is significant if the encoding frame rate is extremely low. However, it ceases to exist if forward referencing is not used.

- **Channel delay**: For video communication, beside the transmission channel throughput delay, another important component exists, namely the buffer delay which is mainly a function of the buffer contents. The bits for the current frame being encoded cannot be sent until the bits in the encoder buffer are transmitted. This is because the bits left in the buffer correspond to the previous frames, which are encoded with more bits than the average bit-rate per frame (i.e. R/F where R and F are the channel and frame rate, respectively). As a result, the time $D_c(i)$ needed for sending the i-th frame through the channel is:

$$D_c(i) = W(i)/R + B(i)/R \tag{2.1}$$

Here $W(i)$ are the bits left in the buffer at the beginning of the i-th frame and $B(i)$ is the number of bits occupied by the i-th frame. $W(i)/R$ and $B(i)/R$ are called the buffer delay and throughput delay, respectively [73]. For low delay, it is desirable to keep $W(i)$ as small as possible. In most land based systems, the buffer delays predominate while in satellite based and long-haul terrestrial systems, the throughput delay is quite substantial.

The above definition of channel delay applies to fixed rate networks. For packet-switch variable rate networks such as Internet, the capability of absorbing a variable output data rate from a source eliminates the buffer delay. However, the throughput delay is no longer a constant value but varies as a function of the status of networks. The arriving order of the transmitted packets will also affects this value.

- **Decoder Delay**: potentially arises from three sources. First is the decoder processing time which is again depending on its implementation but is normally small compared to the encoder processing time. Second, for the case of forward reference where the referred frame does not precede the current frame in the data-stream there will be an additional imposed delay to wait for the frame to become available. This is decoder-wait-for-data. Third, it may be desirable to hold back the display of an available decoded frame to even-out the displayed frame rate. This is decoder-wait-for-display which involves a degree of freedom on the part of the decoder implementation. The total delay is the sum of these three components.

In choosing a low bit-rate coder for a given application, the overall codec delay therefore has to be taken into consideration.

2.4.4 Synchronisation

Most video communication services are accompanied by other source of information such as speech. As a result, synchronisation must be maintained at a certain level in order to ensure satisfactory performance. The best known example is the lip-reading whereby the motion of the lips should coincide with the words that the person is uttering. The simplest and most common technique to achieve synchronisation between two or more traffic streams is to buffer the arriving data at the receiving end and release it as a common playback point [74]. Another possible synchronisation methodology found in the literature consists of multiplexing data together at the source and relying on the sequential reception of packets to maintain temporal consistency [75].

2.4.5 Robustness to Transmission Errors

Since low bit-rate video coding compresses video information into a few parameters, their correct reception is vital for good perception of the received signal. Ideally, we would like the input to the video decoder to be free of errors. However, the transmission channels frequently suffer from degradation which introduce errors into the transmitted parameters. This corruption of the parameters will result in serious degradation in the output image quality if left unchecked. Hence, the use of Forward Error Correction (FEC) techniques is necessary to protect the parameters. This often introduces a high degree of redundancy into the bit stream, especially for channels with high BERs, resulting in inefficient use of the transmission bandwidth. Recently a number of alternative error resilient schemes have been proposed such as two-way code for improving the robustness of the bit-stream [76].

2.5 Conclusion

In this chapter, a brief review of a number of applications which require very low bit-rate video compression have been presented. The types of transmission networks these applications are expected to use and the impact of the associated imperfections on the received video were examined. This was followed by a discussion on some user requirements of video. It emerged that the most challenging of these applications are those of real-time communications over error prone networks such as videotelephony over mobile networks where real-time low power consumption video codecs are required.

Chapter 3

Digital Video Compression Schemes

3.1 Introduction

The digital representation of an image or a sequence of images requires a very large number of bits. The purpose of image compression is to reduce this number as much as possible, and at the same time to reconstruct the original picture as faithfully as possible. The bit-rate reduction is only possible by removing the redundant information from the signal during the coding process and reinserting it during the decoding process. In video signals, there is a vast amount of redundancy between successive frames that can be classified as statistical and psychovisual redundancy [4]. The statistical redundancy results from the fact that pixel values are correlated with their neigbours in spatial and temporal directions. The psychovisual redundancy arise as a consequence of the HVS sensitivity. The human vision has a limited response to fine spatial or temporal detail and bit-rate reduction is possible by allowing distortions that should not be visible to human eyes [1][5].

For many years waveform-based image coding techniques have been the only approaches utilised in image compression [1][5]. These coding techniques share the concept of pixel or block of pixels as the basic entities that are coded. Hence, they are also called pixel-based coding techniques [5]. The lack of consideration for HVS is a major disadvantage of these first generation coding techniques. This leads to the introduction of the second generation image coding where new and more efficient

representations of the image are used. As expected, the HVS becomes a fundamental part of the coding chain.

The major difference between first and second generation image coding can be clearly identified if it is noticed that image and video coding is basically carried out in two steps. First, image data are converted into a sequence of messages and, second, code words are assigned to these messages. Methods of the first generation emphasize on the second step, whereas second generation emphasize on the first step and use the available results from the first generation for the second step [5]. As a result of including the HVS, second generation treats the image as composed of different entities called objects.

3.2 Fundamentals of Image and Video Coding

In this section, the concept of first generation video coding techniques will be reviewed. Fig. 3.1 shows a simplified block diagram of a video encoder and decoder, where the input images undergo four stages of processing as described below :

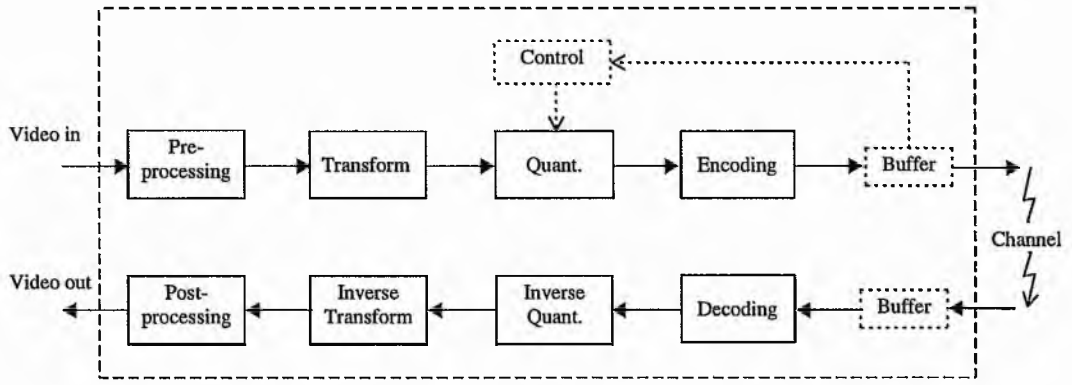


Fig. 3.1 : Block diagram of video coder and decoder

- **Pre-processing**: The efficiency of the encoder can be enhanced if some features of the input images are suppressed or enhanced. For example, for some techniques where motion estimation is employed, noise filtering might improve the result significantly [6].

  At the decoder side, the reconstructed image can be improved at the **post-processing** stage. Edge-enhancement, noise filtering [6] and deblocking-filtering for block-transformed compression are some of the common functions found in this stage [6].

- **Transformation:** This stage is the heart of the compression system in most schemes where the statistical redundancy presented in video sequences is eliminated [1][4][10]. A number of techniques have been used and they are described in the following section.

- **Quantisation:** In this stage, the range of possible values for the transformed signal is reduced, introducing irreversible degradation to the signal, by assigning each value to a member of a finite set of output symbols.

  At the decoder side, the inverse quantisation stage maps the symbols to the corresponding reconstructed values.

- **Encoding:** In this module code words are assigned to the transformed and quantised signal. Lossless coding techniques such as variable word-length code and arithmetic code are often used to take advantage of the different probability of occurrence of each signal [1][5][10].

- **Buffer and control:** Due to the above encoding stage, and also the temporal activity in the incoming video signals, the output bit-rate from the encoder is highly variable. For real-time transmission, the use of a smoothing buffer between the encoder output and channel is necessary. To avoid overflow and underflow of this buffer, a control module is required to regulate the coding process [1][37].

In the following sections, each of the above main module : transform, quantisation, encoding and control will be described in more detail.

3.3 Intraframe Coding

In the literature, there are two main approaches to suppress statistical redundancy, namely the predictive approach and the transform approach [1][5]. Prediction approach is the most straightforward - a number of previous samples are used to estimate the value of the current sample, and the difference between the actual and predicted value is encoded and transmitted or stored. Transform approach, on the other hand, consists of translating the signal to a different space domain, where the information it contains is expected to be separated from the redundancy. Block transform and subband analysis [13] are two obvious examples that belong to the transform approach.

3.3.1 Predictive Coding

In predictive coding, an estimation for a signal element is computed from the previous samples. The difference between the actual and predicted values is then encoded. The prediction can be made from a selection of the previous processed elements or, more complex, a function of several elements. In order to eliminate the accumulation of quantisation error, the prediction is usually based on the previous quantised value samples, as shown in Fig. 3.2. The system operates as follow : at the encoder, the input signal $s(n)$ is subtracted from the predicted value $p(n)$. The error signal $e(n)$ is quantised and encoded. The quantised error $\hat{e}(n)$ is then inverse quantised and added to the predicted value to give the reconstructed signal $r(n)$ for subsequent

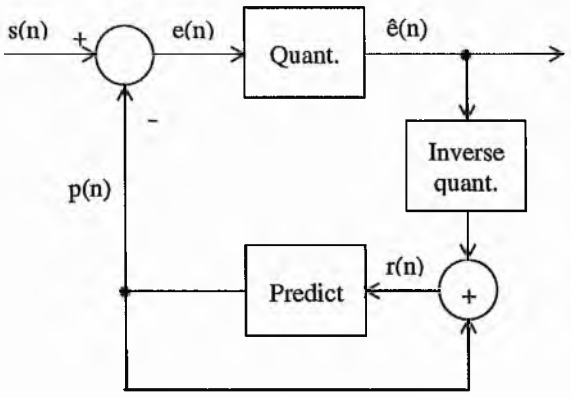


Fig. 3.2 : Predictive coding scheme.

prediction. The last operation is also carried in the decoder for recovering the signal. The above predictive scheme is also termed Differential Pulse Coded Modulation (DPCM) [1][5].

It is possible to improve the performance of the above basic predictive coding scheme by allowing some system parameters to adapt to various influences. Firstly, adaptive quantisation [7][8] which takes into account the HVS factor can be used to adapt the quantisation step according to the visibility of the encoding error in each area of the image. Secondly, adaptive prediction [9] is employed for more accurate prediction of the current sample to enhance the compression result. Nevertheless, predictive coding scheme is outperformed by the transform coding schemes. However, its use in the form of temporal prediction plays an important role in the success of hybrid motion compensated transform video coders [1][5].

3.3.2 Transform Coding

An alternative approach for eliminating the redundancy in the signal is the non-overlapped block transform, in which the input signal is segmented into non-overlapping rectangular blocks of samples and each block is subjected to a linear invertible transform [1][4]. The rationale behind this transform approach is that more

efficient coding can be achieved if the energy of the block of samples can be concentrated in a few transform coefficients. Several orthogonal transforms have been used for this purpose such as Karhunen-Loeve Transform (KLT) [10], Discrete Cosine Transform (DCT), Discrete Walsh Transform (DWT), etc. KLT yields the optimum results but its dependency on the statistics of the input signal and the lack of efficient algorithm for its computation leads to the wide use of the DCT where a fast algorithm is available [1].

Some degree of HVS adaptation in transform coding can be achieved by quantising each transform coefficient differently accounting to its visibility [11]. Moreover, it can be easily integrated with a block based temporal motion compensated prediction for video coding. However, block transform coding suffers from blocking artifacts which consists of the visibility of the boundaries of blocks. Although the use of overlapped orthogonal transform [12] helps to reduce this effect without increasing the total number of transform coefficients for the image, additional complexity is inevitable. Other problems associated with block transform, specially at low bit-rates, include blurred or ringed edges and the checkerboard effect. The latter results from the encoding of a block with very few DCT coefficients [1][5][6].

In spite of these limitations, due to its good performance-complexity trade off for current hardware technology, DCT is adopted in all the current international video coding standards [1][5].

3.3.3 Sub-band Coding

In sub-band coding, the input signal is decomposed into a number of frequency bands by filtering the signal through a filter bank followed by a decimation process, with an overall sample rate equal to that of the original [13]. At the decoder, sub-band signals are up-sampled by Zero Insertion Interpolation(ZII), filtered and added together to recover the original signal. Fig. 3.3 shows a basic two-channel filtering structure for sub-band coding.

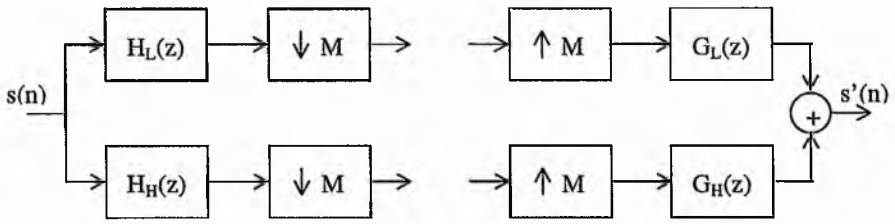


Fig. 3.3 : Basic two-channel filter structure for sub-band coding [1][5].

For image coding, the most obvious extension is to process the signal in two dimension. This can be easily done by sequentially applying the scheme of Fig. 3.3 to rows and columns and so splitting the input image into two bands vertically and horizontally, results in four frequency bands: low-low, low-high, high-low, and high-high. Several variations of this scheme result from iterating the decomposition in all or some of the branches. Without exception, the use of the properties of HVS for improving the coding efficiency can also be incorporated into the coding algorithms by taking into account the non-uniform sensitivity of the eye in the spatial frequency domain [14]. This can also be achieved during the filtering process through the use of special filter structure [15], or during the coding process by allocating more bits for coding the HVS sensitive frequency [16].

DPCM and vector quantisation are popular methods for coding the sub-band signal. However, the lowest sub-band contains significant image structure which can be coded using transform coding [1][5].

3.3.4 Interframe Coding

For video coding, the most direct approach is to extend the above spatial coding schemes to the third dimension, i.e. time domain. For the case of predictive coding, the prediction of a pixel can be obtained either from pixels of the same image or from a previously transmitted image. The latter is referred to as temporal prediction [17].

For the case of transform coding, the use of three dimensional blocks [17][18] has been proposed. However, this scheme introduces a delay and requires additional memory. Three dimensional sub-band [19] has also been proposed. The efficiency of these schemes is limited due to the little correlation that exists in the presence of motion between samples occupying the same position in consecutive images. This leads to the use of motion compensation prediction the scheme [1][5].

Although there are a few motion estimation and compensation techniques suggested for video coding, block matching motion estimation [49] has become the most widespread in use today. It assumes a simple translational motion model for the prediction of a rectangular block of pixels in an image and only one motion vector is used for representing the whole block. Different forms of block matching [49]-[52] are proposed for the search of a block within the reference image that best matches the current block to be encoded, according to some distance criterion such as the MSE or Mean Absolute Error (MAE). The displacement or motion vector is transmitted as side information to the decoder for reconstruction of the prediction image. One reason for its popularity in video coding is its easy integration with the transform coding scheme, which results in the hybrid predictive-transform coding scheme, as shown in Fig. 3.4.

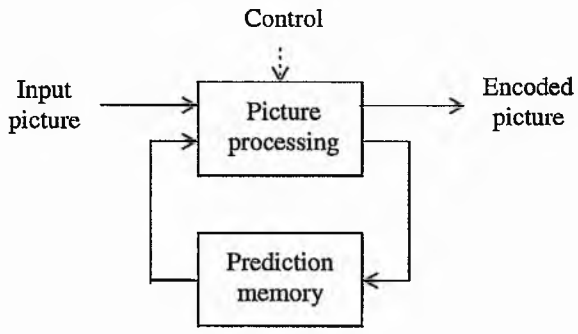


Fig. 3.4 : Hybrid motion compensated video coding.

3.3.5 Segmentation-based Coding

Segmentation-based coding belongs to the second generation video coding technique where the properties of the HVS plays a major role in the coding process of images [20]. A number of algorithms have been proposed which are capable of achieving very low bit-rate with reasonable quality. Basically, it consists of segmenting the image into a number of homogeneous entities called regions or objects with respect to a number of criteria such as contrast, size, etc. [21]. The resulting contours and textures are then coded. Two approaches, namely lossless and lossy coding have been suggested for the coding of the contours. Lossless techniques can be further divided into contour-oriented, in which chain code techniques have been widely used, and shaped-oriented, in which morphological skeleton and quadtree techniques [67] have been proposed. As for texture coding, practically all the texture coding approaches in the first generation can be used, although some modifications are needed to adapt to the arbitrary shape regions [22][23]. The quality and compression ratios obtained depends on the ability of the segmentation technique to provide homogeneous regions and the compression capabilities of both the contour and texture coding techniques used.

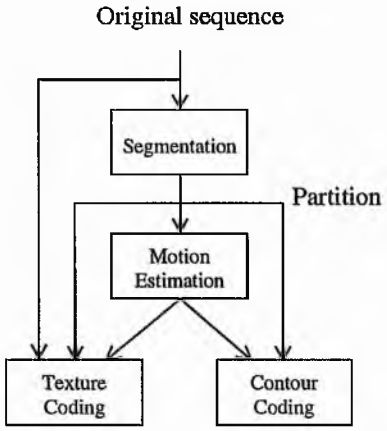


Fig. 3.5 : An example of a segmentation-based coding scheme.

For video coding, in order to achieve higher compression, the use of motion compensation is necessary. Fig. 3.5 shows an example of a segmentation-based coding scheme where motion estimation is incorporated into the coding process. However, the introduction of motion and the use of arbitrarily shaped regions complicates the motion estimation process. Due to motion, the segmented regions in consecutive images are different and this causes difficulty in the prediction of contours and textures. Different approaches have been suggested for solving this problem such as motion segmentation, object tracking, etc. A large improvement in the performance of segmentation-based video coding schemes is expected if this problem of motion estimation for arbitrary shaped regions can be solved.

3.4 Quantisation

The different techniques described above are used for reducing redundancy in video signals for the purpose of higher coding efficiency but the actual operation of video compression is provided by the quantisation stage in an encoder. Hence the most important component of the encoder is the quantisation operation which controls the coding efficiency and the picture quality of the reconstructed video sequences. The coding efficiency and the reconstructed picture quality can be considerably improved if the quantisation operation is based on human visual sensitivity of video signals.

It has been observed experimentally that it is not necessary to convey to the decoder the full numerical precision of the image data to achieve excellent quality reproduction, so the range of possible values which must be accommodated in the encoding stage can be reduced by the process of quantisation. If each sample is quantised independently, then the process is known as scalar quantisation [4][25]. On the other hand, vector quantisation [26] refers to representing a set of vectors, each formed by several continuous-valued samples, with a finite number of vector states. In video coding, there exists several schemes where vector quantisation are used directly to code the image or motion-compensated difference image [27]-[29]. On the other hand, it has also been applied to a transform-domain image representation.

3.4.1 Vector Quantisation

Vector quantisation is also known as block quantisation or pattern matching quantisation. In video coding applications, vector quantisation can be intra-frame of inter-frame. The operation of vector quantisation is based on the selection of a finite set of vectors as representatives of the whole space. This finite set of vector are known as codebook and they determine a segmentation of the input space into cells. The segmentation is carried out according to a minimum distance criterion. For each input vector the quantiser selects the closest vector in the codebook and transmits its label to the decoder as shown in Fig. 3.6. In order to obtain the optimal codebook, the distribution of the blocks is necessary. If it is not known, which is generally the case, the Linde-Buzo-Gray (LBG) algorithms [26] gives a sub-optimal codebook based on a set of training data.

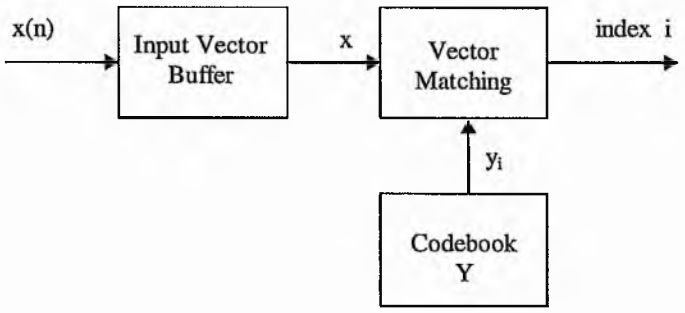


Fig. 3.6 : Block diagram of a simple vector quantiser.

In general, quantisation is a very time consuming task and different ways to structure the codebook have been suggested which simplify the search. An example is the hierarchical vector quantisation [26]. On the other hand, the operation of the decoder is very simple, as it simply accesses a memory which stores the codebook.

3.5 Encoding

The encoding stage usually assumes that the symbols from the quantisation stage are independent of each other, therefore, any technique for the encoding of sources of independent symbols is, in principle, suitable. The simplest approach is the use of Variable Length Codes (VLC) where shorter code words are assigned to symbols that are most likely to occur and longer codewords are assigned to those that are less likely to occur. The Huffman algorithm [68] is used to obtained the optimal VLC codes. However, VLC codes suffer from the restriction of integer number of bits for each code. Arithmetic Coding [30], though more computational intensive, does not suffer from this limitation as fractional number of bits per symbols is allowed. Its use leads to more efficient coding and hence has been included in some of the video coding standards [3][31].

3.6 Rate Control

The rate control module in video compression systems is responsible for regulating the encoder output in according to the available bandwidth of the channel [1][5]. As there is usually a buffer between the encoder output and the channel, the rate control algorithm must prevent the overflow and underflow of this buffer. The fill-level of the buffer is commonly being used for regulating the quantisation operation of the coding process. On the other hand, this buffer also inserts a delay in the video transmission which is proportional to its fill-level. For some services, this delay has to be kept within the limits imposed by the service requirements. Control is usually achieved by adjusting the quantisation step size, although other coding parameters can be used as well, such as the picture rate or the picture resolution [1].

3.7 ITU H.263 Standard

There has been a growing interest in digital video compression technology in recent years. In order to ensure compatibility among video codecs from different manufacturers and applications and to simplify the development of new applications, intensive efforts have been undertaken in recent years to define digital video standards. This has resulted in the ITU standards H.261 [32] and H.263 as well as ISO standards MPEG1 [33] and MPEG2 [34]. These standards were the result of joint development efforts of video and audio compression as well as other system aspects required to support all the applications. Thus they often represent an optimal compromise between performance and complexity. Although all these standards have a generic nature, they address particular applications and also particular ranges of bit-rate. A list of the target applications and bit-rate of each standard is given below.

Standards	Application	Bit-rates
H.261	*Videoconferencing over ISDN*	*p x 64 kb/s*
MPEG1	*Video/Audio storage in CD-ROM*	*1.5 Mb/s*
MPEG2	*Video/Audio broadcast*	*4 ~ 9 Mb/s*
H.263	*Videophony over PSTN*	*< 64 kb/s*

Table 3.1 : List of video coding standards and their applications.

The work leading towards H.263 was initiated in 1993 and targeted at bit-rates below 64 kb/s, in particular, for the application of video transmission over PSTN. At that time, the lowest bit-rate for video coding was achieved using H.261 at 64 kb/s which is beyond the bandwidth available for modems over analog telephone lines. The state of the art for modems at that time was 14.4 kb/s, but rapid advance in technology gave rise to modems with 28.8 kb/s, 32 kb/s and 56 kb/s bit-rates.

The structure of the chosen coding method was very close to the already existing standards. Hence, large improvement in coding efficiency was not expected. Instead,

all possible small improvements were implemented in order to end up with a standard that was significantly better than relevant existing standards (in particular H.261) [35]. As a result, each individual improvement was not necessarily very visible on the decoded pictures.

In the following sections, the structure of the H.263 will be described in detail. The main differences from the previous video standard i.e. H.261 will be pointed out. Lastly, the improvement in performance through the use of the four negotiable options will be investigated.

3.7.1 Picture format

In order to allow a single recommendation to cover use in and between regions using 625- and 525- line television standards, the H.263 source coder operates on pictures based on a Common Intermediate Format (CIF). Pictures are coded as luminance and two colour difference components (Y, C_b and C_r). These components and the codes representing their sampled values are as defined in International Radio Consultative Committee (CCIR) Recommendation 601.

Table 3.2 : Number of pixels per line and number of lines for each of H.263 picture formats.

Picture Format	Number of pixels for luminance (dx)	Number of lines for luminance (dy)	Number of pixels for chrominance (dx/2)	Number of lines for chrominance (dx/2)
sub-QCIF	128	96	64	48
QCIF	176	144	88	72
CIF	352	288	176	144
4CIF	704	576	352	288
16CIF	1408	1152	704	576

There are five standardised picture formats : sub-QCIF, QCIF, CIF, 4CIF and 16CIF. The number of pixels per line and number of lines for each format are given in Table 3.2. As can be seen from the table the number of pixels per line and number of lines

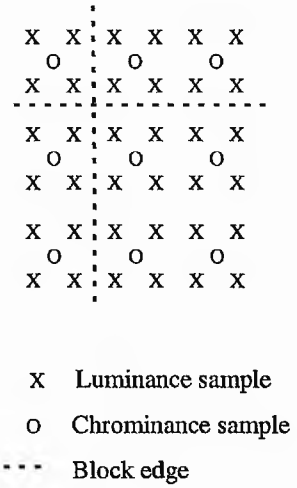


Fig. 3.7 : Position of luminance and chrominance samples

for the colour difference components are half the values of the luminance component due the fact that the human eye is much less sensitive to the details of the colour information than to the details contained in the luminance information. For each of the picture formats, the colour difference samples are sited such that their block boundaries coincide with the luminance block boundaries as shown in Fig. 3.7.

3.7.2 Layering Structure

Each picture is divided into Groups Of Blocks (GOBs). A GOB comprises k*16 lines, depending on the picture format (k=1 for sub-QCIF, 9 for QCIF, and 18 for CIF, 4CIF and 16CIF). The number of GOBs per picture is 6 for sub-QCIF, 9 for QCIF, and 18 for CIF, 4CIF and 16CIF. The GOB numbering is done by use of vertical scan of the GOBs, starting with the upper GOB and ending with the lower GOB. An example of the arrangement of GOBs in a picture is given in Fig. 3.8 for QCIF picture format.

Each GOB is divided into macroblocks. A macroblock relates to 16 pixels by 16 lines of Y and the spatially corresponding 8 pixels by 8 lines of C_b and C_r. Further, a macroblock consists of four luminance blocks and the two spatially corresponding colour difference blocks as shown in Fig. 3.8. Each luminance or chrominance block comprises of 8 pixels by 8 lines of Y, C_b or C_r. The macroblock numbering is done by horizontally scanning of the macroblock rows from left to right, starting with the upper macroblock row and ending with the lower macroblock row. Data for the macroblock is transmitted per macroblock in increasing macroblock number. Data for the blocks is transmitted per block in increasing block number [60].

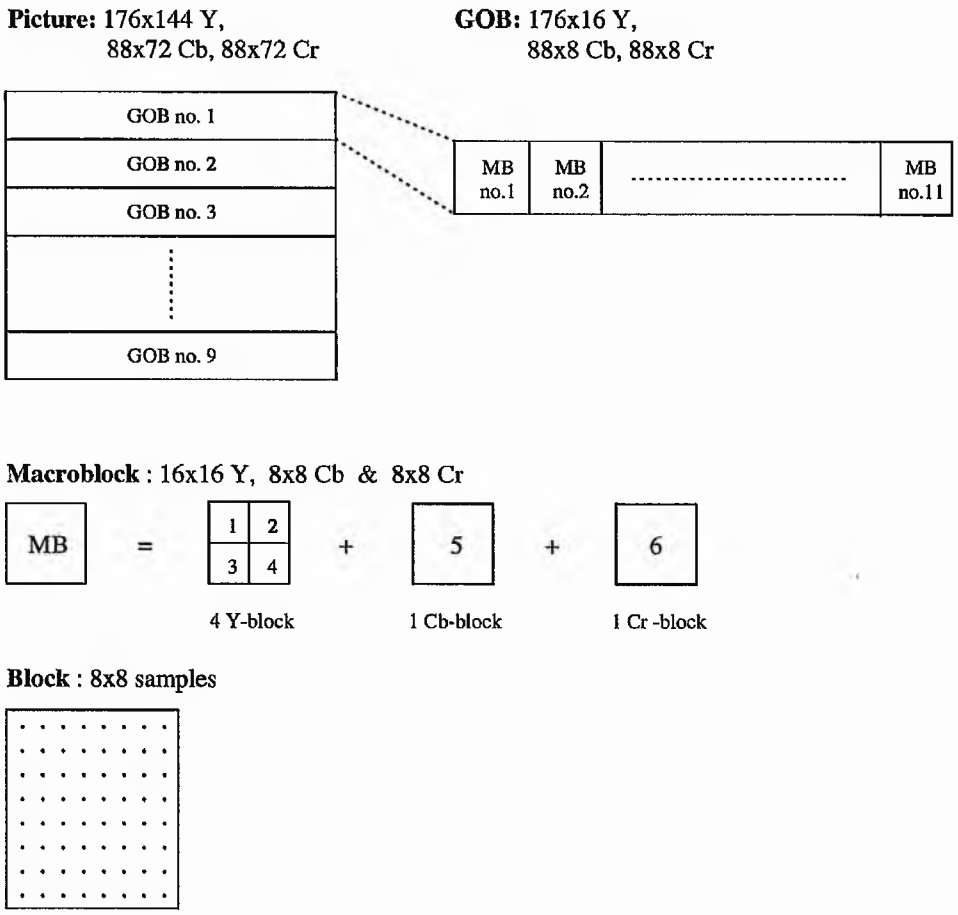


Fig. 3.8 : Hierarchical layering structure of H.263 for QCIF picture format [60].

3.7.3 H.263 Video Coding Algorithm

H.263 consists of a core part and four negotiable options. The core part of H.263 has many similarities with H.261 and MPEG1. In fact, the basic configuration of the coding algorithm is based on H.261. It is block based with motion estimation and motion compensated prediction capability to utilise the temporal redundancy between adjacent pictures. The prediction error signal is then subjected to a two dimensional 8x8 DCT to reduce the spatial redundancy.

The main difference between H.263 and H.261 are :

- H.263 uses less bits for mode information etc. on the block level;
- the coding of motion vectors is achieved more efficiently with better prediction for the motion vectors;
- the use of half pixel precision for motion compensation, as opposed to H.261 where full pixel precision and a loop filter are used;
- the inclusion of four coding options : unrestricted motion vector mode, syntax-based arithmetic coding mode, advanced prediction mode, and PB-frames mode.

A simplified block diagram of video coder of H.263 is shown in Fig. 3.9. The basic operation of the encoder is as follows. The first frame of the incoming video sequence is independently (INTRA) coded: the spatial redundancy of the frame is removed by a de-correlation 8x8 DCT, the resulting DCT coefficients are then quantised and run-length coded. For the following frames, the temporal redundancy between successive frames is removed by subtracting the previous reconstructed frame from the current frame. Motion estimation is usually employed here to further reduce the redundancy. The resulting difference image, commonly referred to as the motion compensated difference image, residual image or prediction error, then undergoes the same processing as the intra mode coding frame i.e. DCT transformed, quantised and

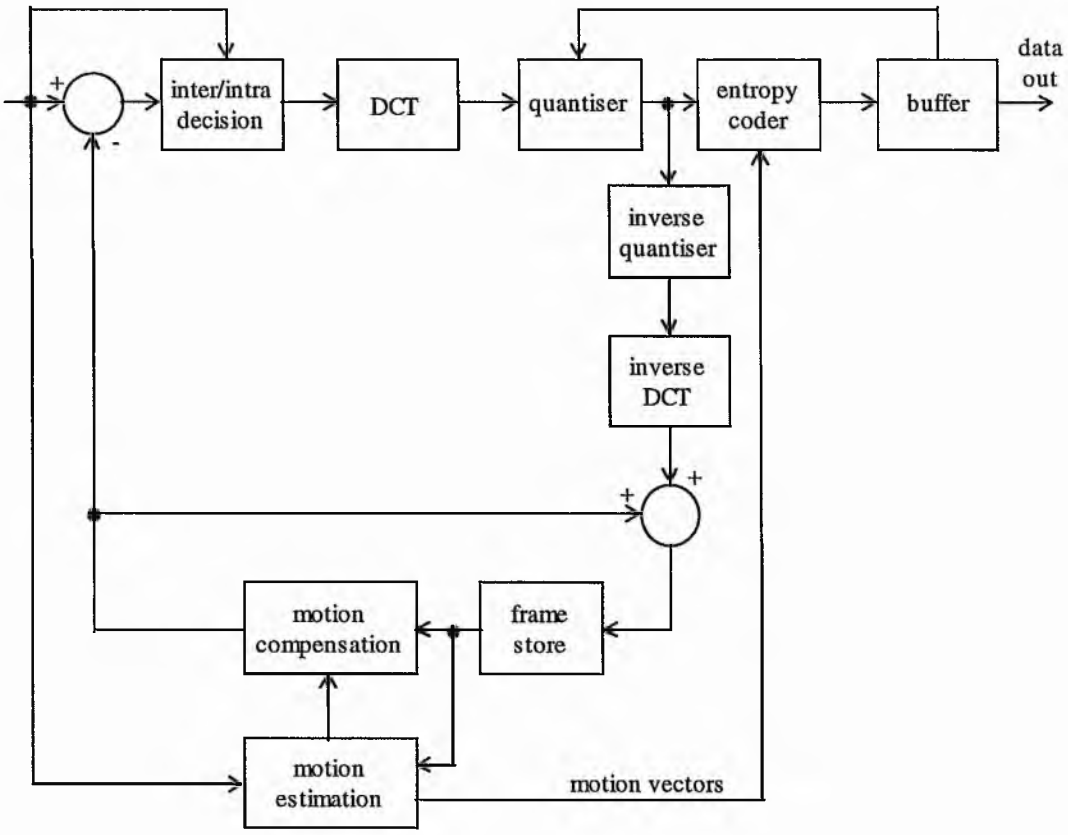


Fig. 3.9 : Simplified block diagram of H.263 encoder.

run-length coded. If motion compensation is used, the associated motion vectors are coded and transmitted to the receiver. This latter mode of operation is called INTER mode in which the previous frame is used to predict the current frame [60].

In a frame where INTER mode is used, it is possible that some of its macroblocks is INTRA coded if they cannot be predicted from the previous reconstructed frame. The decision to use INTRA or INTER mode for each macroblock is made on a macroblock-by- macroblock basis. Usually, the decision is based on the prediction error of the macroblock [39][60].

In the following sections, the main elements of the encoder: motion prediction, block transformation and quantisation will be described in more detail.

3.7.3.1 Motion Estimation

Block Matching Motion Estimation (BMME) is used in the motion estimation process where each macroblock from the current frame is predicted from the previous reconstructed frame. The search is carried out within a window of ± 16 and is restricted such that all pixels referenced are within the reference picture area. The matching criterion may be any error measure such as MSE or Sum of Absolute Difference (SAD) and only luminance component of the image is used in the evaluation. The 16x16 matrix from the previous reconstructed image which give the least error is chosen. The displacement vector between the current macroblock and the best match matrix is called the Motion Vector (MV). The operation of BMME is best illustrated in Fig. 3.10. The technique for finding the best motion vector has not been specified in H.263 but the present day's computer power favours the use of full-search which guarantees that the global minimum distortion will be found.

The motion vector for a particular macroblock will be used for all the pixels in that macroblock for reconstructing the picture at the decoder. As for the chrominance blocks, their motion vectors are derived by dividing the horizontal and vertical component values of the motion vector by two, due to the lower chrominance format. A positive value of the horizontal or vertical component of the motion vector signifies that the prediction is formed from pixel in the referenced picture which are spatially to the right or below the pixels being predicted.

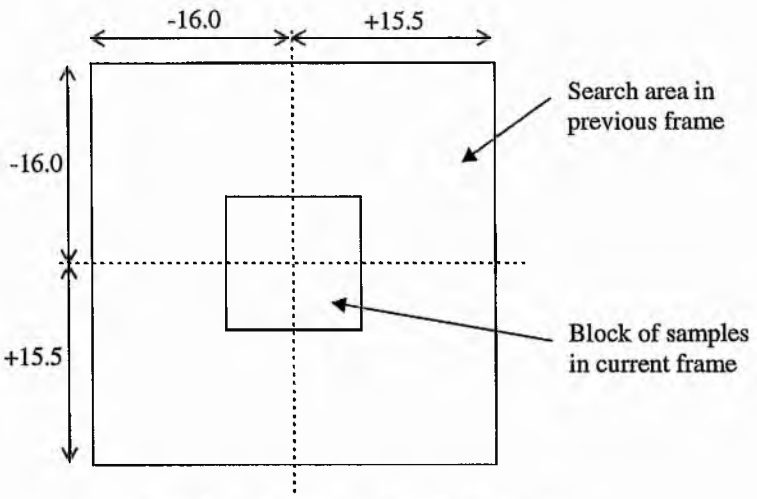


Fig. 3.10 : Principle of Block Matching.

3.7.3.2 Half-pixel Motion Prediction

For more accurate prediction, half-pixel accuracy search is used in H.263 for motion estimation. Instead of comparing blocks with each other that are a multitude of one pixel apart, half-pixel prediction calculates pixel values between two pixels by using interpolation [6][60]. In a way, this implies that half-pixel search actually doubles the size of the search window, hence adding a considerable amount of computational load to the motion prediction if exhaustive search is used. In the Telenor implementation of H.263, a simpler, less computational intensive algorithm is used [39]. First, an exhaustive full-pixel search is performed for blocks in the search window. After that, for the best matching block, a search is done among the eight neighbouring half-pixel interpolated blocks. As a result, half-pixel values of the image samples are required and are found by using linear interpolation between pixel values, as shown in Fig. 3.11.

The use of half-pixel accuracy has enhanced both the objective and subjective quality of H.263 compared to H.261 where full pixel precision and a loop filter is used. This is mainly due to the difference in flexibility between filtering represented by the H.261 loop filter and the half pixel prediction used by H.263. In H.261 loop filtering is performed in both horizontal and vertical directions using taps (1,2,1)/4. In

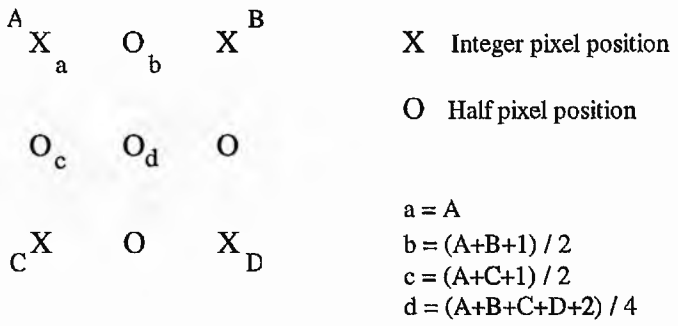


Fig. 3.11 : Half-pixel prediction by linear interpolation.

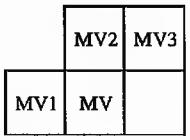
H.263, half pixel values are found using linear interpolations between pixel values. This implies that displacement can be either integer pixel which means that no filtering takes place or half pixel which means using filter taps (1,1)/2 for prediction. As a result, the reduced filtering can be performed in one or two directions. This larger flexibility of filtering is the main reason for the improved subjective quality [35][60].

Both the horizontal and vertical components of the motion vectors have to be sent to the receiver for correct reconstruction of the picture. They are coded separately using differentially coding - only the difference between the actual value and the predicted value is coded as described in the following sub-section. At the decoder, the macroblock motion vector is recovered by adding the predictors to the vector differences [60].

The decision for using INTRA or INTER mode for the current macroblock is normally decided after the motion estimation stage. If INTRA mode is chosen, the current macroblock will be coded directly and no motion vector is sent to the decoder. However if INTER mode is preferred instead, its associated motion vector is differentially coded using the predicted motion vector obtained according to the following sub-section.

3.7.3.3 Motion Vector Prediction

In order to increase the coding efficiency of H.263, motion vectors are differentially encoded to reduce the number of bits required. The candidate predictors for the differential coding are taken from three surrounding macroblocks in the current frame as indicated in Fig. 3.12. The predictors are calculated separately for the horizontal and vertical components. For each component, the predictor is the median value of the three candidate predictors. Only the difference between the actual motion vector and the predictor is coded and sent to the receiver [60].



$MVD_x = MV_x - P_x$
$MVD_y = MV_y - P_y$
P_x = Median ($MV1_x$, $MV2_x$, $MV3_x$)
P_y = Median ($MV1_y$, $MV2_y$, $MV3_y$)

MVD : Differentially coded motion vector
P : Predicted motion vector

Fig. 3.12 : Motion vector prediction [60].

In the special cases where the current macroblock is at the borders of the current GOB or picture, the following rules are applied in increasing order, with reference to Fig. 3.13 [60]:

i. When the corresponding macroblock is coded in INTRA mode or not coded, the candidate predictor is set to zero.

ii. The candidate predictor MV1 is set to zero if the corresponding macroblock is outside the picture (at the left).

iii. The candidate predictors MV2 and MV3 are set to MV1 if the corresponding macroblocks are outside the picture (at the top) or outside the GOB (at the top) if the GOB header of the current GOB is non-empty;

iv. The candidate predictor MV3 is set to zero if the corresponding macroblock is outside the picture (at the right side).

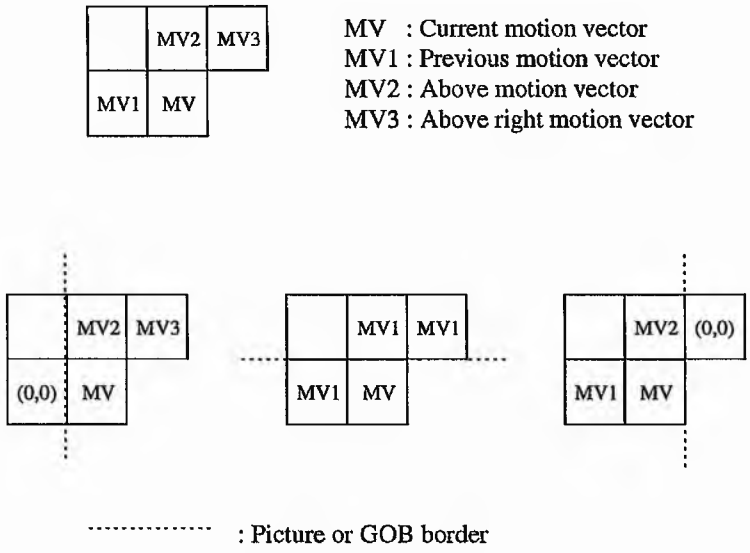


Fig. 3.13 : Motion vector prediction at picture or GOB border [60].

3.7.3.4 DCT Transformation

For macroblocks that are to be INTRA or INTER coded, each of their 4 luminance blocks and 2 chrominance blocks (after motion compensated for INTER macroblock) is subjected to an 8x8 DCT. The two-dimensional 8x8 DCT employed in H.263 is given by equation 3.1 [60].

$$F(u,v) = \frac{1}{4}C(u)C(v)\sum_{x=0}^{7}\sum_{y=0}^{7} f(x,y)\cos[\frac{\pi(2x+1)u}{16}]\cos[\frac{\pi(2y+1)v}{16}] \qquad (3.1)$$

where $F(u,v)$ is the block of transformed coefficients,

$f(x,y)$ is the data block

$C(u) = \frac{1}{\sqrt{2}}$ for $u = 0$; $= 1$ otherwise

$C(v) = \frac{1}{\sqrt{2}}$ for $v = 0$; $= 1$ otherwise

And the inverse DCT transform is defined as [60] :

$$f(x,y) = \frac{1}{4}C(u)C(v)\sum_{u=0}^{7}\sum_{v=0}^{7}F(u,v)\cos[\frac{\pi(2x+1)u}{16}]\cos[\frac{\pi(2y+1)v}{16}] \qquad (3.2)$$

The DCT converts an 8x8 block of pixel values to an 8x8 matrix of horizontal and vertical spatial frequency coefficients. An easy-to-follow numerical example is illustrated in Fig. 3.14. Notice that the distribution of coefficients in the transformed block is far from uniform with a few large coefficients concentrated at the upper left hand corner and the rest hardly significant. Indeed, this is the desired property of DCT which tends to concentrate the energy into the top left-hand coefficients that represent the lower frequencies in the original sample block. The top left-hand coefficient itself represents the dc component of the block (which is actually 8 times the mean). Hence it is called DC coefficient whereas the rest are called AC coefficients [60]. This example has served to highlight the de-correlation property of DCT.

Human eyes are more sensitive to low order DCT coefficients and this has been exploited in H.263 by coding the perceptually important DC coefficient more accurately than the rest as described in the following section [60].

33	36	42	45	51	88	132	140
30	34	38	41	43	60	112	138
47	53	61	69	75	85	103	117
91	92	95	96	97	97	97	94
82	81	82	86	87	86	87	88
71	75	76	76	81	81	81	84
62	64	64	69	71	74	75	79
55	57	55	60	62	64	67	66

8x8 original block

DCT →

597	-109	34.0	-16.6	-1.00	4.64	-4.64	0.58
6.22	-91.5	43.4	-19.3	-4.10	8.70	-2.78	2.39
-75.1	-56.1	30.0	-7.60	-4.92	5.54	-3.21	0.07
8.79	-6.84	7.77	2.14	-4.63	1.78	0.74	0.13
25.3	18.3	-7.42	10.5	-5.25	3.84	0.37	-0.82
29.9	7.19	-11.5	8.17	-4.77	0.26	2.86	-1.74
3.52	-4.58	-8.46	4.82	-3.38	-1.57	1.73	-0.50
-7.19	-10.7	-2.50	1.70	1.37	0.40	1.02	-1.94

8x8 DCT transform coefficients

Fig. 3.14 : An example of DCT transform of a block of pixels.

3.7.3.5 Quantisation

The number of quantiser is 1 for the DC coefficient of an INTRA block and 31 for all other coefficients, including the DC coefficient if it is an INTER block. The same quantiser is used for all the coefficients within a block except the DC coefficient of an INTRA block, which is uniformly quantised with a step size of 8 and no central dead zone to give an 8 bit representation. Each of the other 31 quantisers are basically linear but with a central dead zone around zero and with a step size of an even value in the range 2 to 62 [60].

The encoding process naturally generates a variable data rate, depending upon the picture complexity, and changes from one picture to the next. A target output data rate can be achieved by altering various coding parameters including the picture type (INTRA/INTER), the quantiser step size, and the interval between coded pictures. In general, besides frame skipping, adjusting the quantiser is the most commonly employed method for regulating the output data rate. In H.263 the quantiser is allowed to adjust on the Picture level, GOB level and MB level but the maximum changes are restricted to the range of +/- 2 for the latter.

The following shows how the quantisation and inverse quantisation processes are performed at the encoder and decoder respectively. Here COF is the transformed coefficient to be quantised; LEVEL is the absolute value of the quantised transformed coefficient; COF' is the reconstructed transformed coefficient after inverse quantisation.

Quantisation:

- INTRA DC coefficient:

  $$LEVEL = COF / 8$$

- INTRA AC coefficients:

  $$LEVEL = |COF| / (2 \times QP)$$

- INTER coefficients:

$$LEVEL = (|COF| - QP/2) / (2xQP)$$

Inverse Quantisation:

- INTRA DC coefficient:

$$COF' = LEVEL \times 8$$

- INTRA or INTER coefficients:

$$|COF'| = 0 \qquad \text{if LEVEL} = 0$$

$$|COF'| = 2QP \times LEVEL + QP, \qquad \text{if LEVEL} \neq 0, \text{ QP is odd}$$

$$|COF'| = 2QP \times LEVEL + QP - 1, \qquad \text{if LEVEL} \neq 0, \text{ QP is even}$$

The sign of COF is then added to obtained COF' : COF' = sign(COF) x |COF'|.

Fig. 3.15 continues with the example from the above section whereby the transformed coefficients are quantised, inverse quantised and reconstructed using a quantiser value of 10.

3.7.3.6 Zigzag Scanning and Run-length Coding

The two-dimensional quantised block of coefficients will need to be serialised for transmission along a one-dimensional channel. The concentration of the significant coefficients towards the upper-left hand corner can be exploited by performing a zigzag scan. The order of the zigzag scan adopted by H.263 is depicted in Fig. 3.16. As a result of the scanning, the non-zero low frequencies coefficients will be concentrated at the beginning of the one-dimension stream with a number of runs of successive zeros and a long string of zeros at the end. And this characteristics can be fully exploited by the use of a set of special codewords called run-length code.

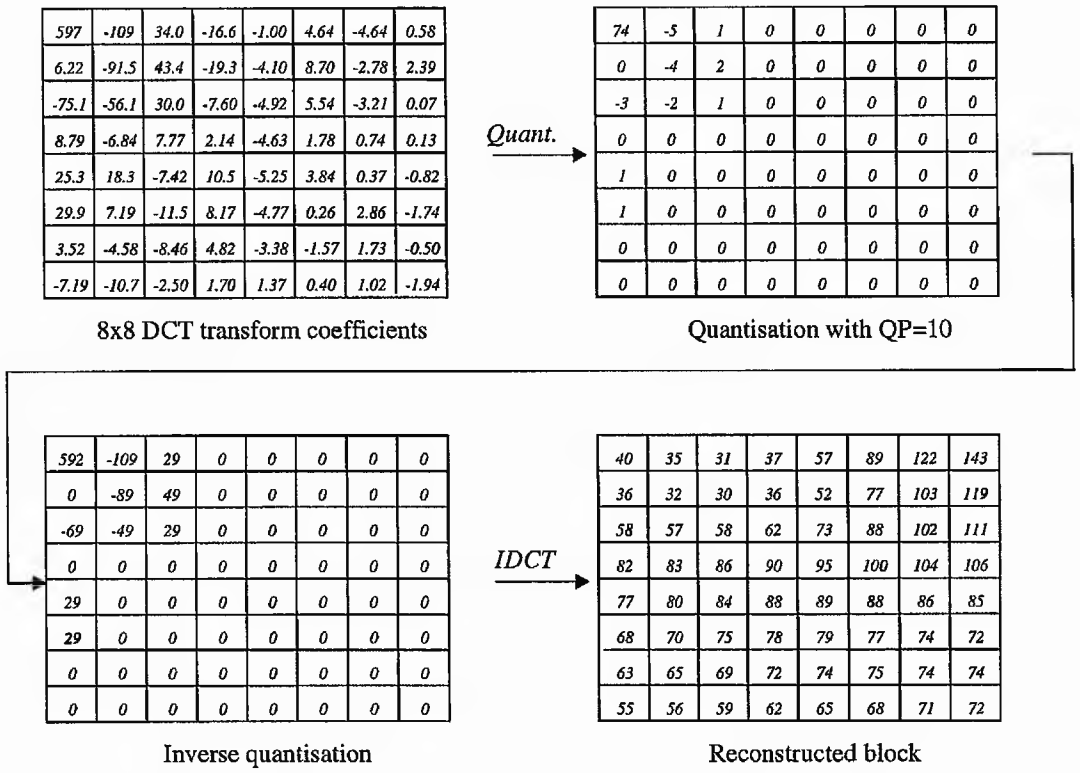


Fig. 3.15 : An example of quantisation, inverse quantisation and reconstruction of a INTRA block of pixels.

In general , as implied by the name, the trick of run-length coding is to represent a run of consecutive identical coefficients with a codeword specifying its value and length of run instead of coding each of them separately. In H.263 only zero coefficients are run-length coded. In addition, it is not difficult to realise from the above example that small coefficient values are more likely to occur than large one, and that short run-lengths of zeros are more likely to occur than long ones. As a result, variable word-length coding is employed to improve the coding efficiency.

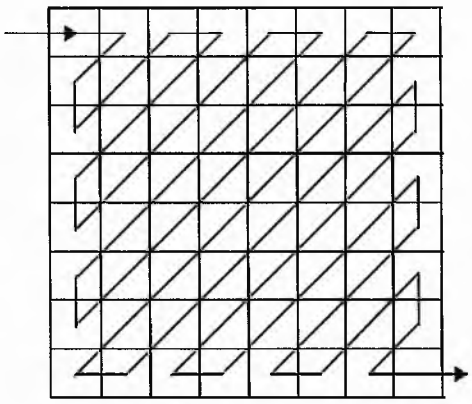


Fig. 3.16 : Zigzag scanning of quantised transformed coefficients.

3.7.3.7 Variable-Length Coding

Variable-length coding has been used in H.263 for efficient coding. It is basically a statistical coding technique that assigns codewords to values to be transmitted. The length of the codeword is chosen depending on the frequency of occurrence of each value. Values with high frequency of occurrence are assigned short codewords and values with sparse frequency of occurrence are assigned long codewords [1][5][60].

3.7.4 Negotiable Options

Another difference between H.263 and the other video coding standards is the inclusion of four negotiable options which the decoder can signals to the encoder which option it has the capability to decode. The encoder proceeds to use them provided it has the capability. The use of negotiable option has made H.263 very flexible. In fact, more new tools in this form of negotiable options have been proposed to ITU for consideration as extension to core H.263 [35]. In the following sub-sections, each of the four options will be described.

3.7.4.1 Unrestricted Motion Vector Mode (Annex D)

This option consists of two parts. In the default video source coding algorithm, motion vectors are restricted such that all pixels referenced are within the coded picture area of the previous reference frame. When Annex D is used, this restriction is no longer valid and motion vectors are allowed to point outside the picture. The benefit of this mode can be easily recognised if the following situation is considered. Assume that the global motion in the picture content is one pixel position in the horizontal direction. In other words, one column of new pixels move into the picture. This means that all pixels except this column of new pixels can theoretically be predicted from the previous picture. However, the restriction on the motion vectors in the default mode imply that those blocks near the edge of the picture will suffer from poor prediction due to the one pixel motion. With the present option, motion vector are allowed to point outside the picture and the referenced pixels outside the picture are replaced by the nearest edge pixel. Therefore, it is possible to do good prediction for all pixels except those coming in as new information. This part of the option is particularly useful when the camera is moving [38][60][61].

The second part of the option deals with an extension to the overall range of the motion vector. In the default prediction mode, the values for the motion vector are restricted to the range (-16.0 to +15.5) pixels. With the present option, the maximum range for the motion vector is extended to (-31.5 to +31.5) pixels. However, it should be noted that not all the vectors may be reached at any time - with the restriction that only values that are within a range of (-16.0 to +15.5) around the predictor can be reached if the prediction is in the range (-15.5 to 16). And if the predictor is outside (-15.5 to 16), all vectors within the range (-31.5 or +31.5) with the same sign as the predictor plus the zero value can be reached. For example, if the prediction of a vector component is +18.5, only vectors in the range (0 to +31.5) can be reached. Obviously, the gain of this part of the option is negligible for a static camera and low activity picture but is particular useful when there is large object or camera motion [38][60][61].

3.7.4.2 Syntax-based Arithmetic Coding Mode (Annex E)

In the core H.263, VLC is used for efficient coding. One of the limitations of VLC is that it is restricted to integer number of bits for each code. This inefficiency can be largely eliminated by the use of arithmetic coding where fractional number of bits per symbol is allowed. Arithmetic coding works in conjunction with a modeller which estimates the probability of a particular symbol in the data stream. In H.263, the used models are switched in accordance to the type of information being coded, hence it is called syntax-based arithmetic coding. The resulting PSNR and reconstructed picture will remain the same with the use of this option but will generally lead to a reduction in the overall bit-rate due to the optimised bit representation of each individual symbol. The amount of bit-rate reduction will depend on individual sequence but an average reduction of 4-10% compared with using VLC can be expected [35].

3.7.4.3 Advanced Prediction Mode (Annex F)

This option includes the possibility of using four motion vectors instead of one per macroblock, and overlapped block motion compensation, which gives smoother prediction image and hence a smoother output image. If this mode is used then the Unrestricted Motion Vector mode (Annex D) must also be in operation to deal with cases where pixels outside the normal coded picture area can be accessed. However, the extended motion vector range feature of Unrestricted Motion Vector mode is not automatically included [60].

In core H.263 16x16 block is used for motion compensation. With the present option, 8x8 block is used instead. Consequently, this may provide more accurate prediction but carries with it an additional overhead for coding the four motion vectors. A trade-off between bit-rate and quality has to be established and this can be decided on a macroblock-by-macroblock basis for which macroblock there is sufficient benefit to use four motion vectors instead of one. As in core H.263, each components of the four

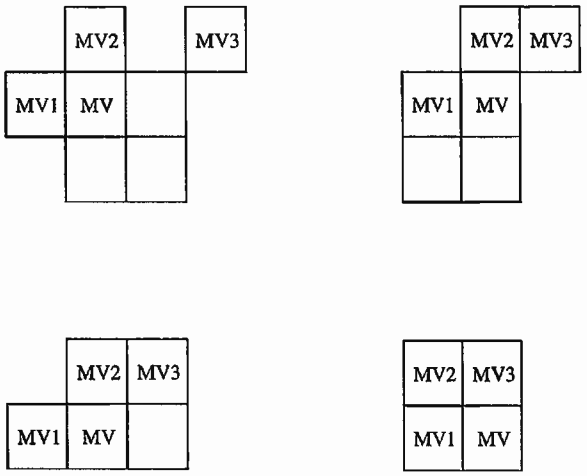


Fig. 3.17 : Candidate predictors MV1, MV2 and MV3 for advanced prediction mode [60].

vectors are differentially coded. Again the predictors are calculated separately for each of the horizontal and vertical components as shown in Fig. 3.17.

The second part of this mode is Overlap Block Motion Compensation (OBMC) [36]. In the core H.263, each pixel is predicted from only one motion vector. This motion vector is the one that are used for the whole macroblock. With the present option, three motion vectors are used to predict each pixel: the motion vector belonging to the current block and the vectors of the two closest blocks. Each pixel is a weighted sum of three prediction values obtained from the three motion vectors. The result is reduced blocking artifacts in the reconstructed picture.

3.7.4.4 PB-Frame Mode (Annex G)

The PB-frames mode introduces a special type of B-frame or bi-directionally predicted frame that is particularly useful for low bit-rate coding. As shown in Fig. 3.18, this B-frame is predicted bi-directionally from the previous reconstructed frame (can be either an I-frame or P-frame) and the P-frame that is currently being coded. This B-frame and the current P-frame are coded as a single unit, thus the name PB-

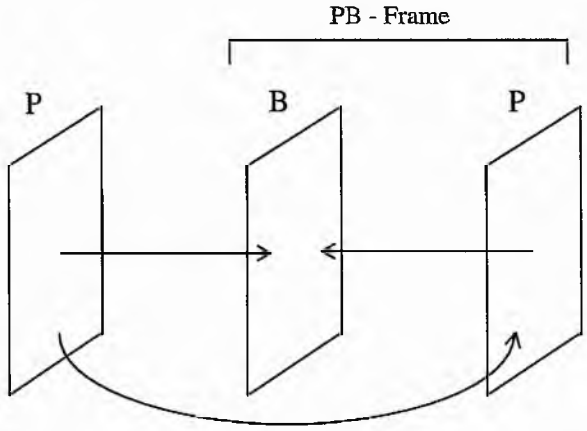


Fig. 3.18 : Prediction in PB-Frame mode

frame. The motion vectors for the B-frame are obtained from scaling down vectors from the relevant P-frame, but additional "delta" vectors may also be transmitted. Therefore, PB-frame mode is a very bit efficient way of coding frames. PB-frames mode is particular useful when the motion between the P-frames is limited. In this case, the frame rate can be easily double without much increase in the overall bit-rate since the number of bits spent on the B-frame is very small. However, this mode becomes inefficient with highly active sequence at low frame rates as interpolation became inaccurate.

3.7.5 Syntax

The order of transmission of the various output parameters of H.263 is followed according to a hierarchical structure starting with a picture, followed by GOB, macroblock and ending with a block. The order of transmission of the GOB is from top to bottom and that of macroblock is from left to right. The exact detail of H.263 syntax will not be covered here but can be found in the recommendation [3].

As with the other existing video coding standards, only the bit-stream syntax is defined for H.263, leaving ample room for improvement to the source coding algorithm. In the next two chapters, we will be investigating techniques for improving the coding efficiency as well as subjective quality of the decoded images.

3.7.6 Performance of H.263 Video Coding Algorithm

The propose of having this section is to assess the usefulness of each of the four negotiable options. Simulation results for the five test sequences namely Claire, Miss America, Carphone, Suzie, and Foreman will be presented and the performance of each option will be discussed.

All original sequences are of QCIF format and are regarded as having a 25 Hz frame rate. Besides, only the first 150 frames of the original sequence will be used. The coded frame rate will be fixed at 12.5 and the quantiser parameter, QP will also be fixed for the duration of the sequence. The output bit-rate and PSNR of each decoded images will therefore vary throughout the sequences, and the tabulated results are the average values for the complete coded sequences.

Table 3.3 shows the quantiser used for coding each sequences and their results obtained using the default mode of H.263. These results will be used as the reference for comparing the performance of the four negotiable options.

Table 3.3 : Quantiser used and results obtained using default coding mode.

Sequence name	Quantiser, QP	PSNR_Y (dB)	Bit-rate (kb/s)
Claire	8	37.27	20.08
Miss America	8	38.05	19.27
Suzie	15	32.68	20.94
Carphone	18	30.51	20.26
Foreman	20	29.03	27.99

3.7.6.1 Annex D - Unrestricted Motion Vectors

It can be seen from Table 3.4 that Annex D gives a small improvement in data compression for most of the sequences. However, the reduction in bit-rate is very significant for Foreman and the perceptual decoded image quality has also improved. The Foreman sequence contains a great deal of motion and a number of very rapid pans. Therefore, Annex D is very useful for coding this kind of sequence since motion vectors are allowed to point outside the coded picture area.

Table 3.4 : Comparison of core H.263 with the use of Annex D.

Sequence name	PSNR_Y (dB)	Bit-rate (kb/s)	% change in bit-rate
Claire	37.30	19.87	-1.05
Miss America	38.11	19.06	-1.09
Suzie	32.76	20.33	-2.91
Carphone	30.57	20.20	-0.30
Foreman	29.18	25.68	-8.25

(a) (b)

Fig 3.19 : 148th frame of Foreman sequence encoded with QP=20 at 12.5 fps (a) Base mode (b) Annex D is turned on.

3.7.6.2 Annex E - Syntax-based Arithmetic Coding

As expected, the results in Table 3.5 indicate that additional compression can be obtained using Annex E, which is more efficient than the variable length coding. The amount of reduction in bit-rate is sequence dependent but generally a reduction of about 4-10% can be expected.

Table 3.5 : Improvement in compression due to the use of Annex E.

Sequence name	PSNR_Y (dB)	Bit-rate (kb/s)	% change in bit-rate
Claire	37.27	19.31	-3.83
Miss America	38.05	18.53	-3.84
Suzie	32.68	19.48	-6.97
Carphone	30.51	18.93	-6.56
Foreman	29.03	26.03	-7.00

3.7.6.3 Annex F - Advanced Prediction Mode

The results below show that there is a reduction in bit-rate for all the sequences when Annex F is used, though theoretically it might lead to an increase in bit-rate due to the coding of 4 motion vectors instead of 1 compared to the default mode. The explanation for this reduction is simple and straightforward. Due to the use of smaller block for motion estimation, more accurate prediction is achieved. Hence, less bits are spent on coding the prediction errors and this saving in bits is more than enough to cover the bits being spent on the four motion vectors. And the results seem to confirm this explanation since the reduction in bit-rate is greater for using a smaller quantiser than a larger one except for Foreman, though one might argue that the sequences Suzie and Carphone used more Annex F option per frame than the sequences Claire and Miss America. The significant reduction in bit-rate for Foreman can again be attributed to the use of Annex D which is automatically turned on when Annex F is used. The most obvious contribution of Annex F is the very significant subjective improvement (principally less blocking artifacts) it brings to all the sequences due to the use of overlapped motion compensation. The results also shown a 0.1-0.3 dB gain in the objective quality.

Table 3.6 : Results of using Annex F for more accurate motion prediction.

Sequence name	PSNR_Y (dB)	Bit-rate (kb/s)	% change in bit-rate
Claire	37.35	19.19	-4.43
Miss America	38.14	18.58	-3.58
Suzie	32.88	20.64	-1.43
Carphone	30.67	19.98	-1.38
Foreman	29.38	26.14	-6.61

(a) (b)

Fig. 3.20 : 148th frame of Foreman sequence encoded with QP=20 at 12.5 fps (a) Base mode (b) Annex F is turned on.

3.7.6.4 Annex G - PB-Frame mode

There are two ways of using the PB-frames mode. For the simpler sequences, it can be used to double the frame rate for a relatively modest increase in bit-rate, giving less jerky images. The alternative way is to use it for achieving higher compression at a given frame rate through the more efficient coding of B pictures. Both experiments were carried out and the two set of results are presented in Table 3.7 and Table 3.8. It can be easily seen from both tables that PB-frames mode becomes ineffective with sequences containing high rates of motion, as the interpolation process becomes inaccurate. The ratio of the number of P frame and B frame coded should give a fairly good idea about the effectiveness of Annex G for the tested sequences.

Table 3.7 : Use of Annex G for attaining higher compression.

Sequence name	PSNR_Y (dB) (No. of P-frame)	PSNR_Y (dB) (No. of B-frame)	Bit-rate (kb/s)	% change in bit-rate
Claire	37.22 (39)	36.89 (34)	16.36	-18.53
Miss America	38.04 (39)	37.65 (34)	14.53	-24.60
Suzie	32.67 (47)	32.37 (26)	19.02	-9.17
Carphone	30.48 (62)	30.15 (11)	19.45	-4.00
Foreman	29.07 (63)	28.54 (10)	26.41	-5.64

(a) (b)

Fig. 3.21 : 148th frame of Foreman sequence encoded with QP=20 at 12.5 fps (a) Base mode (b) Annex G is turned on.

Table 3.8 : Results of doubling the frame rate using Annex G.

Sequence name	PSNR_Y (dB) (No. of P-frame)	PSNR_Y (dB) (No. of B-frame)	Bit-rate (kb/s)	% change in bit-rate
Claire	37.28 (76)	37.20 (73)	21.27	+5.93
Miss America	38.05 (75)	38.00 (74)	20.39	+5.81
Suzie	32.77 (78)	32.53 (71)	24.37	+16.38
Carphone	30.41 (100)	30.20 (49)	26.30	+29.81
Foreman	29.08 (77)	28.88 (72)	32.98	+17.83

(a) (b)

Fig. 3.22 : 148th frame of Foreman sequence encoded with QP=20 at 25 fps (a) Base mode (b) Annex G is turned on.

3.8 Concluding Remarks

In the first part of this chapter, some of the popular image and video coding techniques such as predictive coding, transform coding and subband coding have been briefly reviewed. All of them belong to the first generation pixel-based approach where the influence of HVS has not been seriously considered in the coding process. The difference in approach between first generation coding techniques and second generation region/object-based techniques is identified. A brief description of a segmentation-based video coding scheme has been included to highlight the difference between the two approaches.

A description of the basic features of ITU-T H.263 video coding standard for low bit-rate application was given in the second part of the chapter. Its functionalities were described in detail and its major differences from the other standards were indicated. Finally, the performance of its four optional modes were assessed using some of the standard video sequences.

Chapter 4

Rate Control For Low Delay Transmission

4.1 Introduction

Due to the fact that raw video data contains a large amount of redundant information in both the temporal and spatial domains, the suppression of these redundancies during the coding process generates a highly variable data rate at the output of the encoder. Although packet networks may be capable of handling variable bit-rates, in some applications, constant bit-rate is more desirable either for a simpler network configuration or for channel with fixed bandwidth such as PSTN. To achieve constant bit-rate transmission, a buffer between the coder output and the channel is used to smooth out the bit-rate fluctuations. However, this buffer is only able to absorb short-term variation in the output bit-rate. In videophone and videoconference applications, fluctuations tend to last over several frames. Moreover, one also has to take into account the delay introduced, which might make an interactive, real-time situation impossible. Thus drastic measures such as frame skipping are needed, but the most common method is to use the buffer fill level to control the quantisation process, specifically, the quantisation step size [1]. The outcome is, almost inevitably, a drop in the reconstructed image quality at times of significant detail change in the input images [1].

4.2 Fixed Rate Transmission

Until recently, no form of video transmission was envisaged other than by the use of a fixed rate channel. This is appropriate when no compression process is included in the signal chain from the source to the observer since only fixed length digital codewords generated at a constant rate are involved. However, this situation was altered with the use of image compression techniques such as differential coding, predictive coding and variable length coding. As a result, data is generated at a highly variable rate. The use of a fixed bandwidth transmission system without some kind of matching operation between the source coder and channel is inappropriate - the channel has to accept the peak rate generated by the coder, leaving it under utilised for most of the time. Inevitably, a store or buffer which has the function of smoothing out the fluctuations in the source coder output is required.

The use of a buffer only partially solves the problem. In practical situations, the buffer is only able to absorb short-term variation in the output bit-rate. Hence, drastic measures such as data sub-sampling [37] in the form of frame skipping or down-scaling of picture resolution are needed. However, the most commonly used method is to adjust the quantisation process as a function of the buffer fullness, i.e. by feedback control [1]. On the other hand, the use of current picture activity, i.e. feedforward control, though rarely used, provided another alternative approach. The general situation is as shown below.

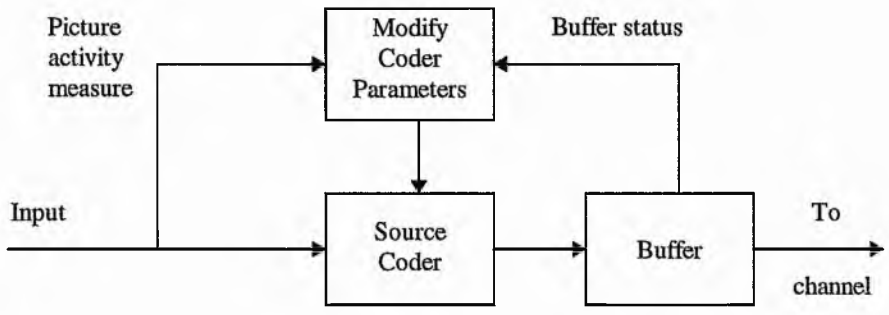


Fig. 4.1 : Control of output bit-rate in a fixed rate system.

4.3 Conventional Rate Control

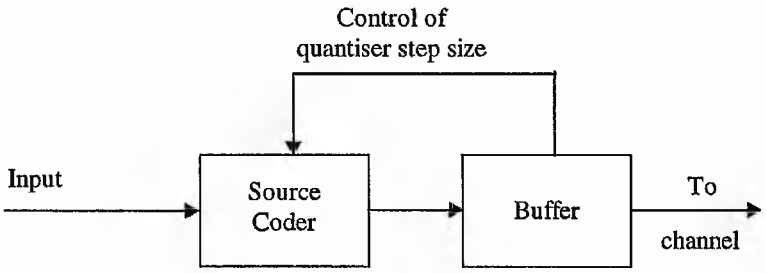


Fig. 4.2 : Control of output bit-rate through buffer fullness.

As mentioned earlier, buffer feedback (see Fig. 4.2) can be used to reduce the rate at which the buffer fills during periods of high activity by reducing the quantisation accuracy of values within the changed regions or by coding fewer samples (sub-sampling) in these regions. The regulation of the quantisation step size value is crucial for video coding because it directly affects the amount of bits generated. This in turn affects the delay and image quality [38]. Fig. 4.3 illustrates the relationship between picture activity, quality and bit-rate. It is obvious from the figure that although the delay for a fixed rate system is kept to a minimum, its image quality will suffer during high temporal activity. In contrast, the variable rate system manages to maintain a constant picture quality at the expense of longer delay which is undesirable for real-time communication applications.

The traditional approach to the selection of quantisation step size for the next frame, GOB or macroblock (MB), is based on the fill level of the buffer. However, the average number of bits generated for each frame is not linearly dependent on the quantisation step size as shown in Fig. 4.4. For example, when QP is less than 5, a unit variation can generate two to five times more coded data and quickly filling up the output buffer. Conversely, the same unit variation may generate only a few dozen more bits when the quantisation step size exceeds 20. Moreover, the content of the current frame also affects the amount of coded data

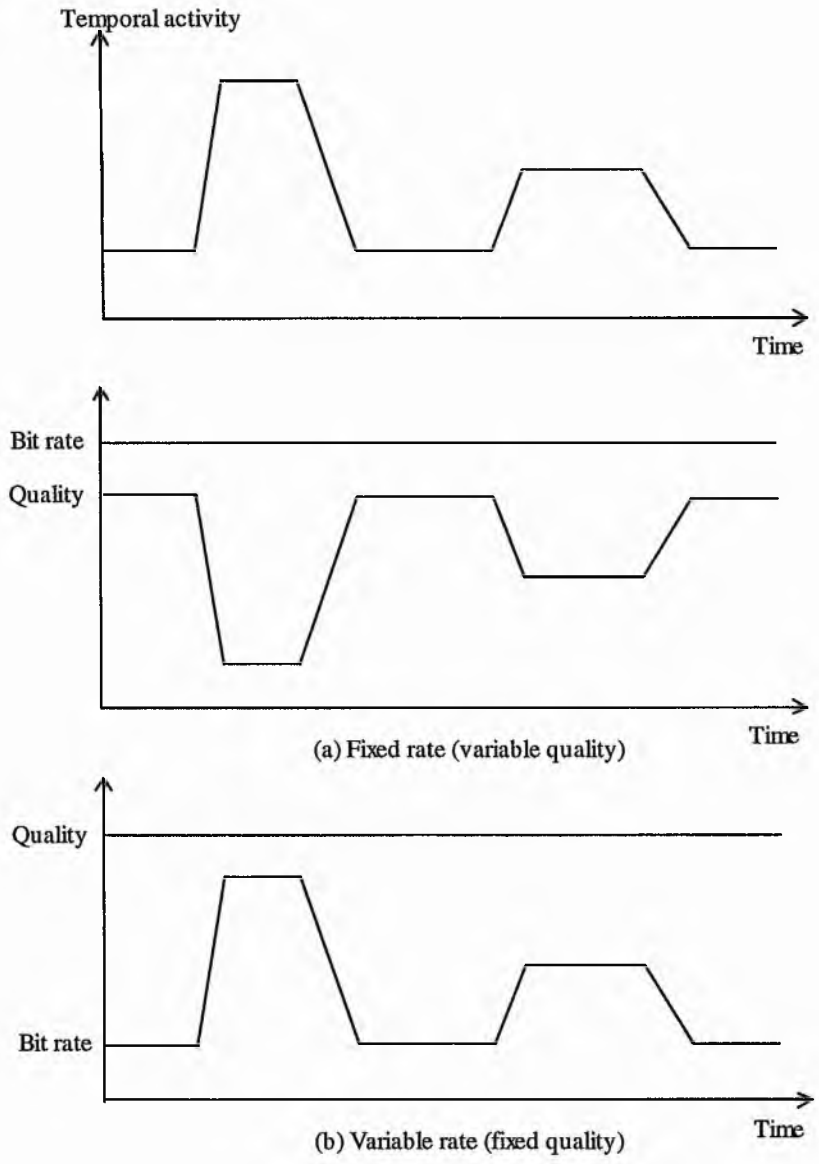


Fig. 4.3 : Schematic portrayal of the pictorial temporal activity/quality relationship in fixed and variable rate systems.

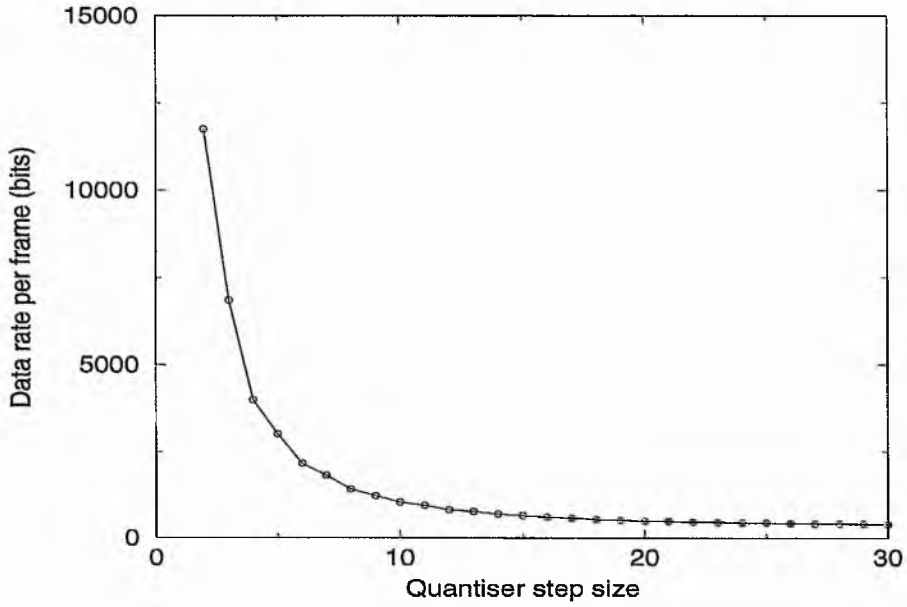


Fig. 4.4 : Average data rate per frame as a function of quantiser.

generated. This make the regulation very critical, especially at low quantisation step size values.

For the conventional rate controller, no information from the current frame, such as degree of activity in the difference image or motion vectors are taken into consideration. This approach gives an unpredictable, highly fluctuating bit-rate and increases the chance of buffer overflow, resulting in a loss of data. In the next section, a rate control algorithm based on Fig. 4.1 will be presented. The activity of the current frame will be used to choose an appropriate quantiser step size such that the resulting bit-rate is close to the target bit-rate. On the other hand, the buffer fullness can be used to prevent overflow and underflow.

4.4 Prediction Error Coded-Bit Estimation

In general, most of the bits generated by typical block-based coding algorithms are spent on the transform coefficients and motion vectors, with the number of bits spent on the transform coefficients being the most unpredictable. On the other hand, experimental results show that there is some correlation between the number of bits spent on the coefficients, prediction error and quantisation step size which can be exploited for estimating the number of bits required to code a certain prediction error value.

First, a large training set including prediction error values and resulting number of bits for different quantisation step size (QP) values is obtained from some of the standard sequences, namely Miss America, Claire, Suzie, Carphone and Foreman. For each QP, the range of the prediction error per block, from 0 to infinite, is divided into L intervals. For each interval, an average bit per error is calculated using the training set. This is equivalent to generating a codebook table with prediction error values as entries. Fig. 4.5 shows a graph of bit per error against the prediction error per block for different quantiser step sizes with L = 100.

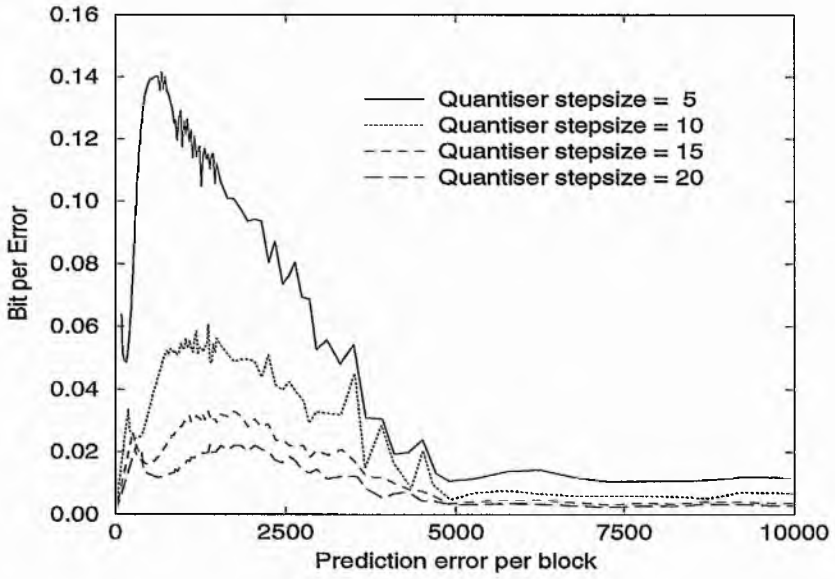


Fig. 4.5 : Bit per error as a function of prediction error per block for different step sizes.

Next, the probability of coding the prediction error has to be considered. Fig. 4.6 shows a graph of probability of coding the error against the prediction error per block. To generate this graph, the number of data being coded as well as not coded are recorded for each error interval. The probability of coding the prediction error of a particular interval is simply the number of data coded divided by the total number of data falling within that interval. As expected, the probability of coding the prediction error increases with the prediction error. These results are used to adjust the bit per error curves with values of those intervals whose probabilities of being coded fall below a threshold value set to zero. Our experiment shows that a threshold value of 50% gives a reasonably good result. The adjustment is done separately for each QP.

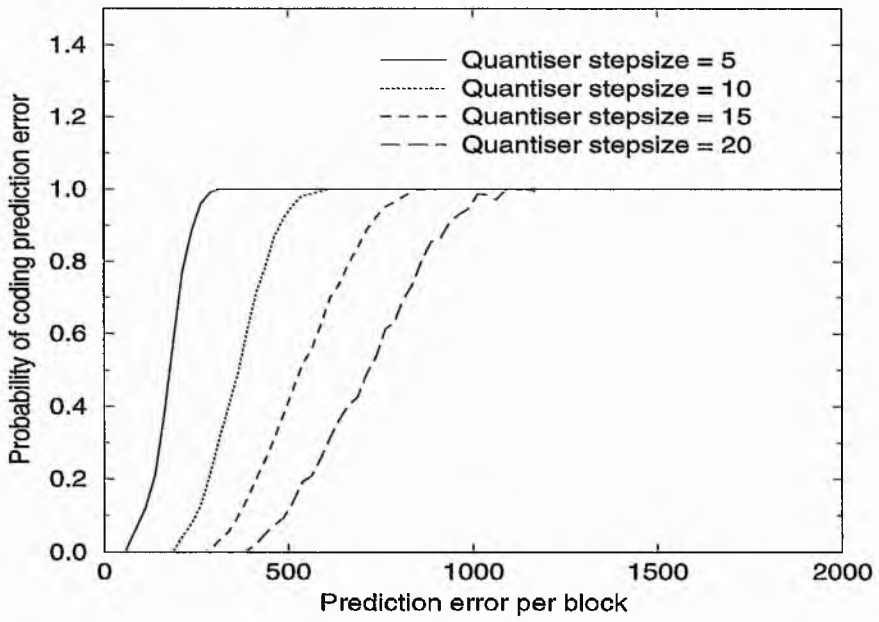


Fig. 4.6 : Probability of coding prediction error as a function of prediction error per block for different step sizes.

4.5 Feedforward Rate Control Algorithm

The prediction error per block for the current frame which is essential for estimating the number of bits required to code the prediction error can be obtained during the motion estimation stage. Next, an initial QP must be selected. Its selection is not very crucial but a value closer to the final selected value will reduce the number of iterations required. Therefore, it is chosen based on the QP and the bit-rate of the last coded frame. With this QP, the number of bits required to code the transform coefficients of the current difference frame is estimated using the prediction error per block and the bit per error curves. These estimated bits are then used for computing the number of bits required to code the motion vectors as well as the Coded Block Pattern for the luminance (CBPY). The computations are straightforward and the numbers of bits to coded them can be readily obtained using the VLC tables [3]. As for the other parameters such as MCBPC, COD, DQUANT and headers, the number of bits spent on them is either constant or negligible. The predicted bit-rate required to code the current frame is the sum of all these values. This bit-rate is then compared with the target bit-rate per frame. QP is increased when the predicted bit-rate is higher or vice versa, and the whole process is repeated. The iteration will stop when there is a cross over 1 or when QP is equal to 31 which is the maximum allowable value. The QP that gives the closest bit-rate is chosen for coding the current frame.

Simulation results give an accuracy of within ± 15% for the bits used to code the transform coefficients and a less fluctuating bit-rate. In order to reduce these fluctuations further, the quantiser step size is adjusted in the MB level. However, the maximum change in step size is limited to ± 2 around the chosen value in order to maintain a more uniform image quality within the same frame. This change in quantiser step size for each MB is based on the following rules:

Let QP_{frame} = selected quantiser step size for the current frame

B_{target} = target bit-rate per frame

B_{total} = total bits spent until the current MB + total predicted bits required to code the remaining MB.

if $(B_{total} / B_{target}) > T_1$ and $QP <= QP_{frame} + 2$, increase QP, where $T_1 > 1$

if $(B_{total} / B_{target}) < T_2$ and $QP >= QP_{frame} - 2$, decrease QP, where $T_2 < 1$

else $QP = QP_{frame}$

One may think that using values closer to 1 for both thresholds might give a better result but this proved to be not true. This is mainly due to the fact that the prediction algorithm is not very accurate in the MB level, though it gives a rather impressive overall result.

4.5.1 Simulation Results and Discussion

The above rate control algorithm had been implemented in TMN5 [39], a H.263 test model developed by Telenor R&D, and simulations were performed using only the base mode. In order to give a fair comparison, the original rate controller in TMN5 was modified so that the regulation of the quantiser step size was made possible in the MB level. Unexpectedly, this actually led to a worse result in some cases. The values of T1 and T2 used in the experiments are 0.95 and 1.05 respectively.

Table 4.1 : Results of sequences encoded at 20 kb/s, 7.5 frame/s.

	Original TMN5		Proposed Controller	
	Actual bit-rate (kb/s)	PSNR (dB)	Actual bit-rate (kb/s)	PSNR (dB)
Foreman (240)	20.33	28.27	20.01	28.14
Carphone (200)	23.06	31.29	20.16	30.86
Suzie (149)	19.91	32.65	20.13	32.81
Salesman (200)	20.86	31.64	19.99	31.57

Table 4.1 tabulates the results obtained which show that the bit-rates of the 4 sequences encoded using the proposed algorithm are very close to the target bit-rate. As for the image quality, the small decrease in the overall PSNR did not really give any significant degradation in the subjective quality except for the sequence Suzie. Fig. 4.7 gives an example of the variations in the output bit-rate. As can be seen, the proposed rate controller gives a less fluctuating output bit-rate. Fig. 4.8 is a plot of the corresponding objective quality of the coded sequence. As expected, the PSNR values become less stable.

Subjective evaluations using pair comparison method were carried out and no significant changes in the output decoded image quality is found except for the sequence Suzie. Although there is a sudden drastic drop in the image quality when she started to shake her hair, the image quality recovers rapidly, unlike that of TMN5 where the distortion lasted for a longer period.

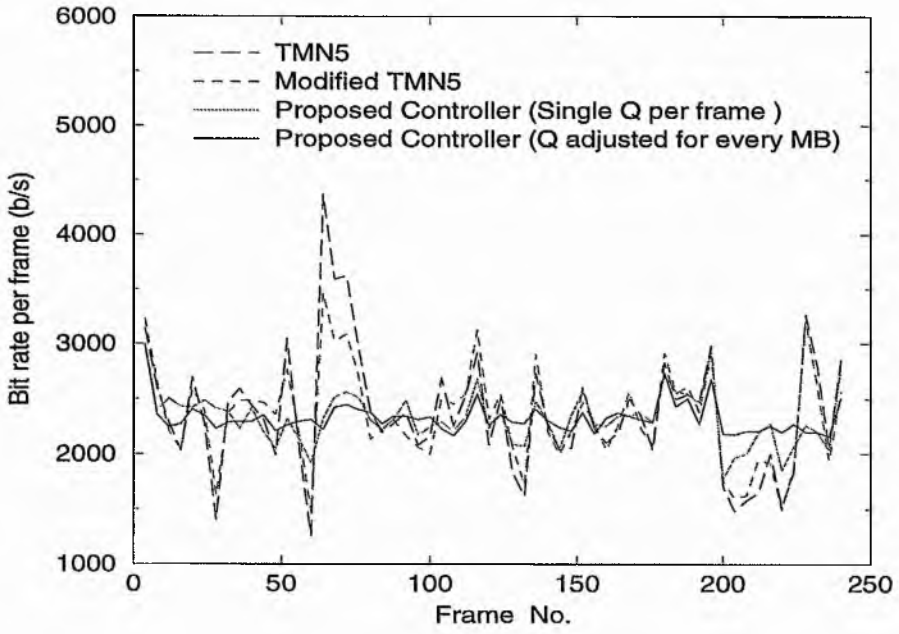


Fig. 4.7 : Bit-rate per frame of Foreman encoded at 20 kb/s, 7.5 f/s.

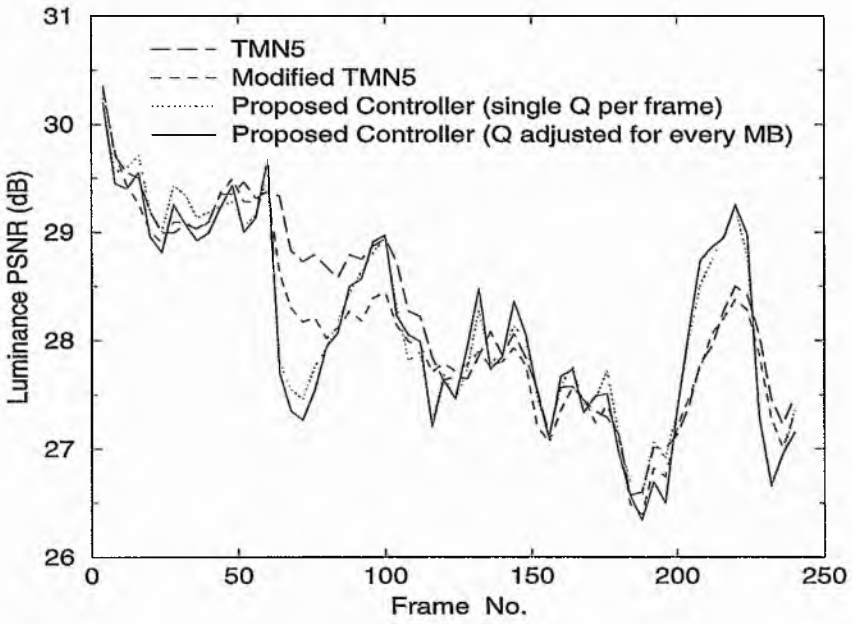


Fig. 4.8 : PSNR of Foreman encoded at 20 kb/s, 7.5 f/s.

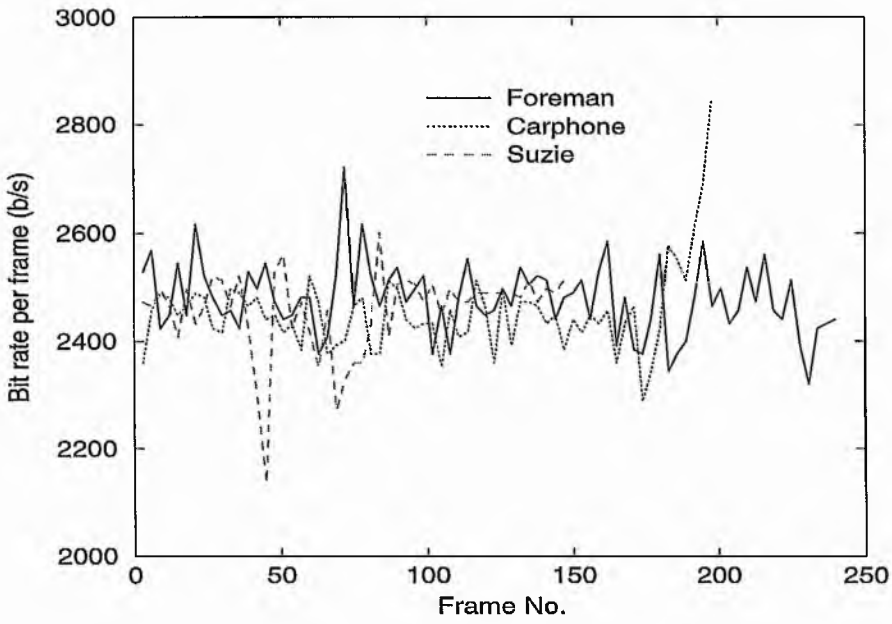


Fig. 4.9 : Bit-rate per frame of sequences Foreman, Carphone and Suzie encoded at 2.5 kb/frame.

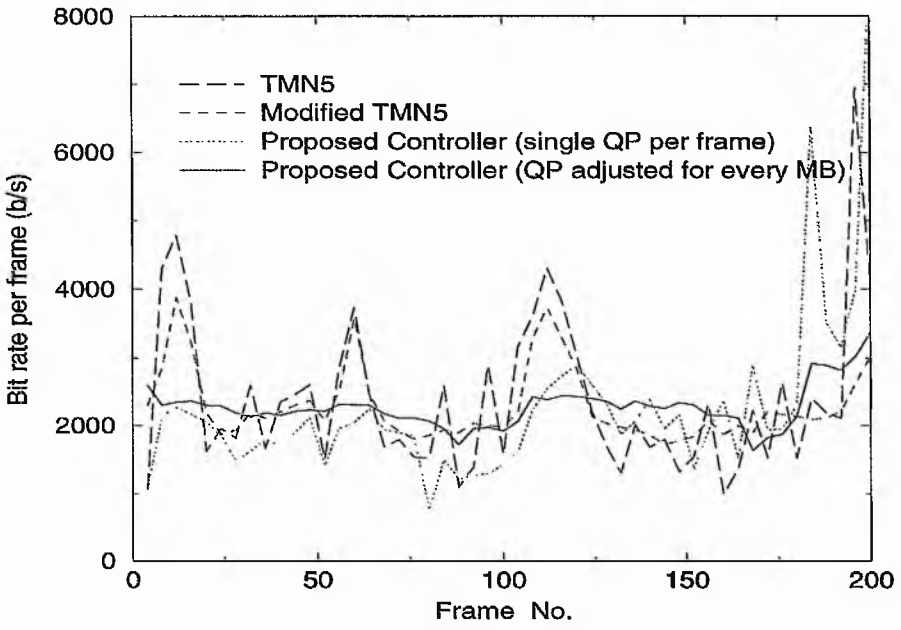


Fig. 4.10 : Bit-rate per frame of Salesman encoded at 20 kb/s, 7.5 f/s.

Fig. 4.9 plots the bit-rate per frame for the sequences Foreman, Carphone and Suzie encoded at a fixed bit-rate of 2.5 kb/frame. As can be seen, the deviation of the actual bit-rate from the targeted bit-rate is reasonably small. Fig. 4.10 compares the 4 controllers using the sequence Salesman which is not in the training sequences. Although the result is not as good as those used in the training sequences, the feedforward rate control algorithm still gives a more constant bit-rate. On the other hand, by adapting the coded-bit estimation tables to the statistics of the current image sequence, it is possible that a more stable output bit-rate can be achieved. A simple method for making the coded-bit estimation tables adaptive is implemented and the same experiment is performed for Salesman sequence. The bit-rate per frame is plotted in Fig. 4.11 which shows a much stabilised output.

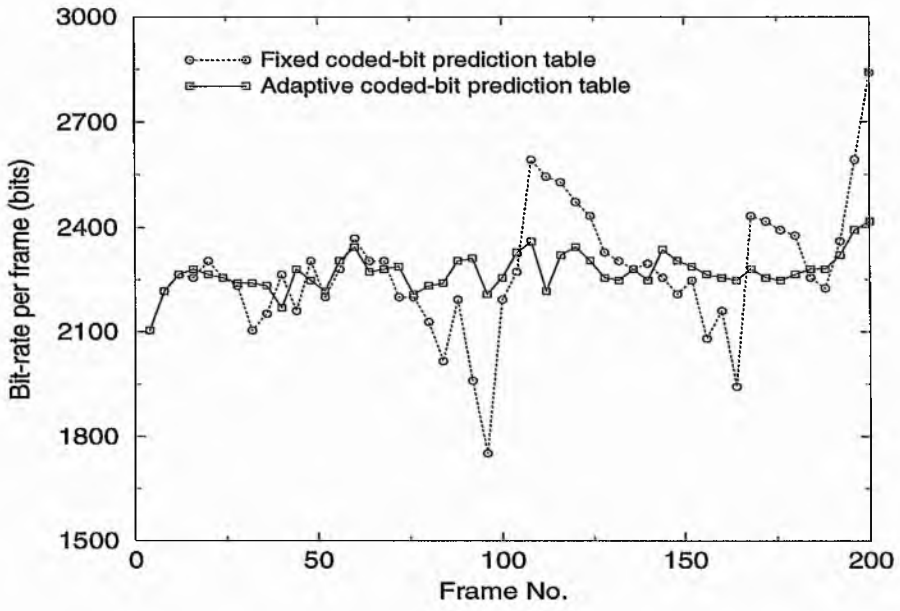


Fig. 4.11 : Output bit-rate per frame of Salesman sequence encoded with rate control algorithm using fixed coded-bit estimation tables and one with adaptive tables.

4.6 Subjective Quality Improvement

The types of images in certain applications are known a priori and this knowledge can be exploited to increase the coding efficiency by coding those Region-Of-Interest (ROI) more accurately than the rest of the image. For example, in videoconference and videophone images, one tends to concentrate on the face with special emphasis on the eyes and the mouth. Hence it is reasonable to spend more bits coding these regions of an image more accurately at the expense of coarser coding of the remaining image. To achieve this, a recognition algorithm is required to locate the position of the ROI in the incoming image sequence.

In the following section, a dynamic bit-allocation algorithm for improving the subjective quality of typical head and shoulder images will be described. The advantage of this algorithm is that the above stated objective is achieved with the output bit-rate remaining almost at a constant.

4.6.1 Face and Background Segmentation

In recent years, a number of papers had suggested methods of adaptive coding using a priori knowledge about the image [40][41] in which important features are identified and accounted far more favourably. In videoconference and videophone images, the face is an area of high movement and is the part most intensively observed. Therefore, it seems quite reasonable to spend more bits coding this part of image more accurately at the expense of coarser coding of the remaining parts. To achieve this, a face location algorithm is required to identify the position of the face in a head and shoulder image.

Methods of extracting the human body location and facial position in a head and shoulders image had been described in the literature [42][43]. For block-based video coders such as H.263, accurate location of the face is not required. For the sequence Miss America, we found that the method of M. Soryani and R J Clarke [42] is accurate enough to suit our purpose. The method first converts the input image into a binary image and the face is identified as a large white region. A binary map, for example of 9 x 11 for the QCIF picture format, is then created which identified the face from the background.

4.6.2 Bit Allocation Based On Buffer Feedback Control

This is a simple algorithm based on the TMN5 rate control module [39] in which a smaller quantisation step size is used to code the face region and a coarser quantisation step size is used to code the background. The selection of the 2 step sizes depends on the rate control algorithm which calculates the quantisation step size (QP) for the next frame to be coded in order to meet the target bit-rate. Basically, we set the minimum size of the gap between these 2 step sizes, for example QP_f for the face region is set at QP-2 and that of the background, QP_{nf} at QP+6 at the beginning of the coding process. During the coding process, if the generated bits are less than the target bits, QP will decrease, so that more bits are generated to meet the target bit-rate. When QP is reduced to a threshold value say QP_{lower}, it will stop to decrease.

However, QP_f for the face will continue to decrease until the target bit-rate is met. As a result, the gap between QP_{nf} and QP_f increases. The idea is to hold QP_{nf} at the value of $QP_{lower}+6$. On the other hand, if the generated bits is higher than the target bit-rate, QP will be increased. When QP is increased above a threshold value say QP_{upper}, the gap between QP_{nf} and QP_f will start to reduce by holding QP_{nf} at QP_{upper}. This simple control technique gives satisfactory results with Miss America sequence.

4.6.2.1 Simulation Results and Discussion

Simulation were carried out using Miss America sequence. Table 4.2 is the results obtained for various bit-rates. All the simulation were performed using only the base mode.

Table 4.2 : Results for Miss America

	Face Enhanced			TMN5		
Target bit-rate/frame rate	Face PSNR (dB)	Overall PSNR (dB)	Actual bit-rate (kb/s)	Face PSNR (dB)	Overall PSNR (dB)	Actual bit-rate (kb/s)
20000/10	34.69	36.82	20.29	32.36	37.89	20.23
17000/10	33.37	36.53	17.29	31.51	37.22	17.18
14400/10	31.83	36.02	14.57	30.52	36.43	14.53
9600/06	30.96	35.71	9.73	29.77	35.91	9.73

Table 4.2 shows an improvement of 1.2 - 2.3 dB in PSNR using the above described algorithm. In Fig. 4.13 and Fig. 4.14 the number of bit per frame and PSNR around the face area are plotted for Miss America test sequence coded at 20 kb/s. In subjective tests, pair comparison method has been used. The results confirmed the improvement in subjective picture quality, favouring the above described algorithm which gives smoother (less noise) and sharper perception around the face area. For subjective assessment, a frame of Miss America sequence encoded at 14.4 kb/s is displayed in Fig. 4.12. In fact, the sequence coded at a rate of 17 kb/s using the described algorithm has the same subjective quality as that coded at 20 kb/s using the original algorithm, which gives a reduction of 15% in terms of bit-rate. Objectively, the PSNR around the face is also higher as shown in Table 4.2.

Fig. 4.12 : A frame of Miss America sequence encoded at 14.4 kb/s. The left image is encoded using TMN5 and the right image is encoded using the described algorithm.

Though the subjective quality of the reconstructed image has improved and the average bit-rate is close to the target bit-rate, the goal of this experiment is only partially fulfilled as the bit-rate per frame is highly fluctuating as shown in Fig. 4.13. As stated above, the aim of this experiment is to design a bit allocation algorithm that allocates more bits to the region-of-interest and at the same time maintains the resultant bit-rate close to its allocated value. In the next section, we shall see that this can be achieved using the feedforward rate control algorithm in section 4.4, which is essentially a coded-bit estimation algorithm.

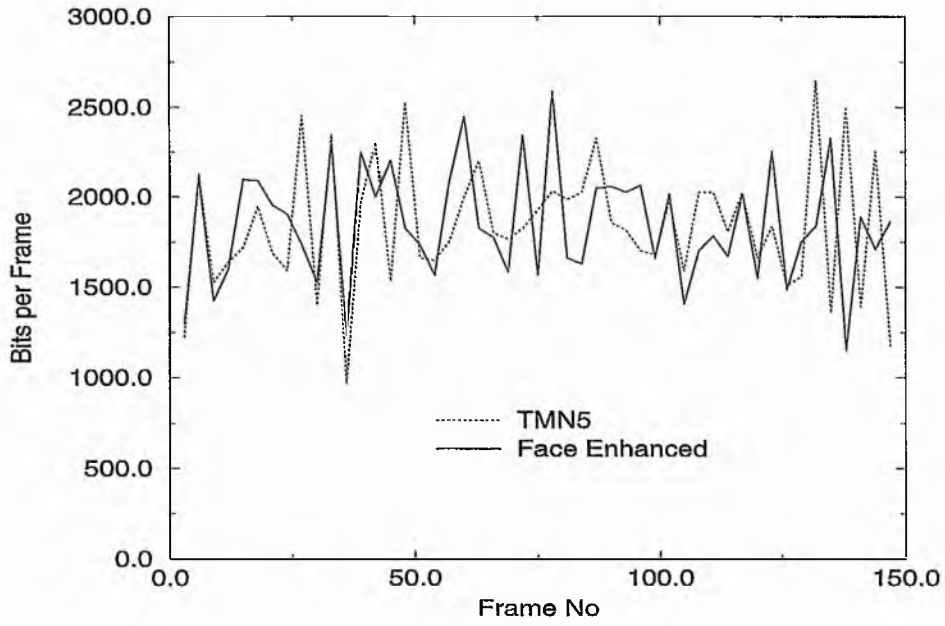


Fig. 4.13 : Bits generated per frame of Miss America encoded at 20 kb/s.

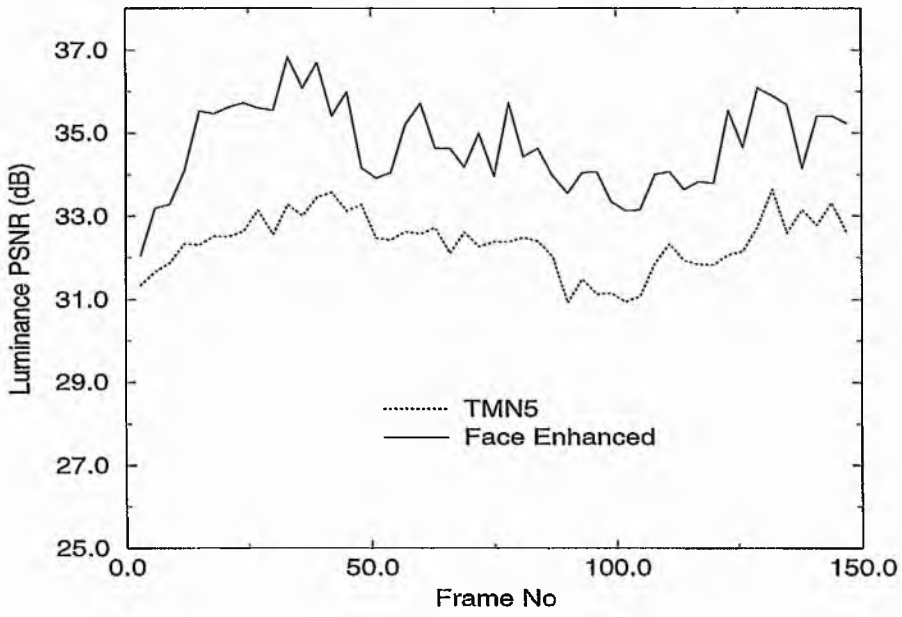


Fig. 4.14 : PSNR of Miss America encoded at 20 kb/s.

4.6.3 Bit Allocation Based On Picture Activity

In order to obtain a stable output bit-rate, the feedforword rate control algorithm in section 3.4 is used to select two quantisation step sizes for coding the current picture such that the resulting bit-rate per frame is close to the target value. As in section 4.6.1, the region-of-interest is coded using a smaller quantisation step size (QP_f) and the rest of the picture is coded with a coarse step size (QP_{nf}).

Initially, the minimum size of the gap between these 2 step sizes is set to g, i.e.

$$QP_{nf} = QP_f + g, \qquad \text{where } g = 0,1,2,..12. \ (= 8 \text{ initially})$$

Next, the above-mentioned rate control algorithm is used to estimate the number of bits required to code the current image. When the predicted bit-rate is greater than the target bit-rate, QP_f will be increased or vice versa until a pair of quantisers capable of giving a predicted bit-rate close to the target value is found. When QP_f is greater than a threshold, for example 15, g is increased for every increment of QP_f until it reaches 12 which is the maximum value allowed. On the other hand, if QP_f decreases below a threshold, for example 5, QP_f is held at 5 and g starts to decrease. When g is 0, i.e. $QP_{nf} = QP_f$ and the predicted bit-rate is still less than the target bit-rate, then QP_f will be decreased below 5. However, for very low bit-rate application, the chance of QP_f decreases below 5 is very small.

A fairly stable bit-rate per frame is achievable using the above method. However, as expected, the associated image quality becomes more fluctuating when there is significant movement in the incoming images. In order to maintain a better image quality, more bits are allocated to those frames that contain significant movement. However, a frame which has large movement does not necessarily mean that it requires more bits if it can be well predicted from the previous reconstructed image. Looking at the variance or SAD of the difference image might give a better hint. Here, the quantisation step size is used to determine the extra amount of bits the current image is allowed to have with the following rules.

if ($QP_f > 25$), target bit-rate = 1.25 x target bit-rate;
else if ($QP_f > 20$), target bit-rate = 1.20 x target bit-rate;
else if ($QP_f > 15$), target bit-rate = 1.15 x target bit-rate;
else target bit-rate remain unchanged.

4.6.3.1 Simulation Results and Discussion

The sequences Miss America, Foreman, Carphone and Suzie were used in the simulation. The face location algorithm of M. Soryani and R. J. Clarke was only used for locating the face of Miss America. For the other 3 sequences, the binary maps for identifying the face from the background were manually created. Table 4.3 are the results obtained using the above bit-allocation algorithm.

Table 4.3 : Results for Foreman, Carphone, Suzie and Miss America.

	Face Enhanced			TMN5		
Target bit-rate/frame-rate	Face PSNR (dB)	Overall PSNR (dB)	Actual bit-rate (kb/s)	Face PSNR (dB)	Overall PSNR (dB)	Actual bit-rate (kb/s)
Foreman (0-240) 48 kb/s, 10 fps	33.81	30.35	48.00	31.47	30.91	48.14
Carphone(0-200) 32 kb/s, 6.25 fps	35.03	32.35	32.04	32.52	33.73	35.05
Suzie (0-149) 28 kb/s, 6.25 fps	36.15	34.81	28.15	33.12	34.56	28.01
Miss Am (0-149) 14.4kb/s, 6.25fps	40.03	37.41	14.50	38.16	37.89	14.35

A gain of about 2-3 dB in luminance PSNR around the face region of each test sequences is achievable with the face enhanced algorithm when compared with TMN5. In Fig. 4.15 the bit-rate per frame is plotted for the sequence Foreman. A less fluctuating bit-rate, comparable to that obtained with a stabilised bit-rate algorithm, is achievable using the proposed algorithm. Fig. 4.16 shows that the PSNR around the face of Foreman coded with the face enhanced algorithm is always higher than that of a stabilised bit-rate algorithm. On the other hand, the overall PSNR of the stabilised bit-rate algorithm is always higher than the proposed algorithm as shown in Fig. 4.17.

Compared to TMN5, the PSNR around the face of Foreman for the face enhanced algorithm is almost always higher except for a few frame where TMN5 gives a much higher values. Fig. 4.18 is a plot of the PSNR around the face of Suzie which again confirmed the advantage of the face enhanced algorithm compared to the stabilised bit-rate algorithm. As for TMN5, the distortion in the image quality lasted for a long period after Suzie started to shake her hair. For the face enhanced algorithm, the image quality recovers rapidly, although there is a drastic drop in the image quality when Suzie shakes her hair.

Subjective tests using pair comparison method also confirm the improvement in the decoded picture quality. The reconstructed images from the described algorithm are very significantly better, with smoother and sharper perception around the face area.

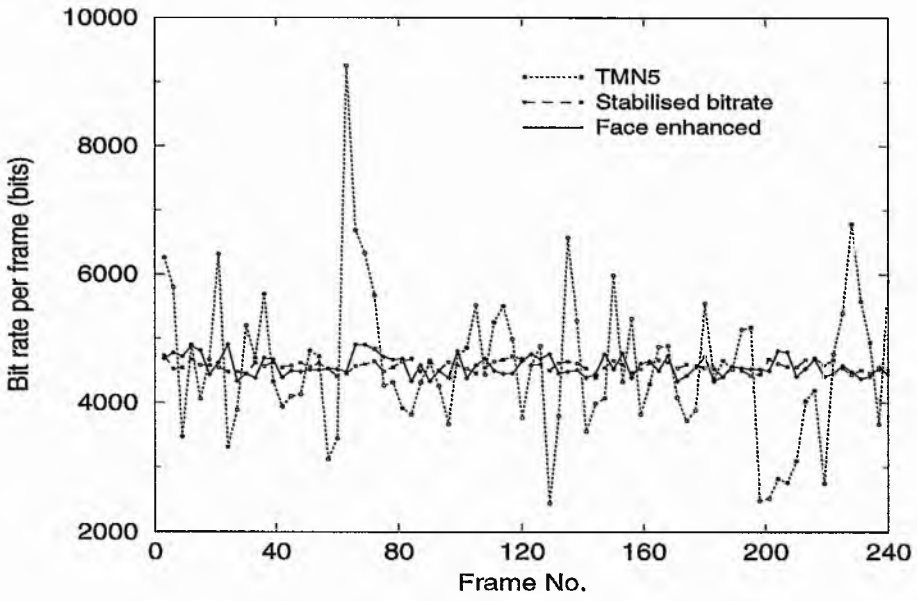


Fig. 4.15 : Bit-rate per frame of Foreman sequence.

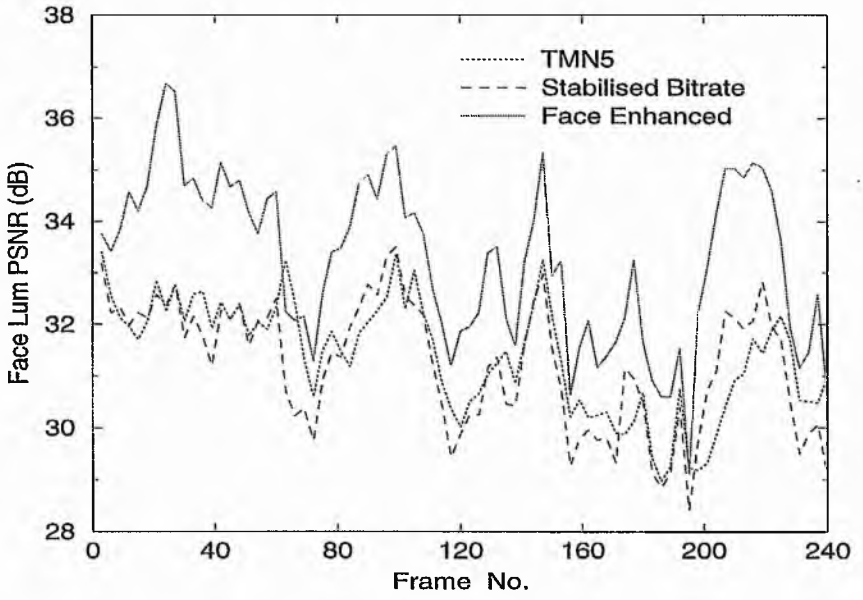


Fig. 4.16 : Luminance PSNR around the face of Foreman.

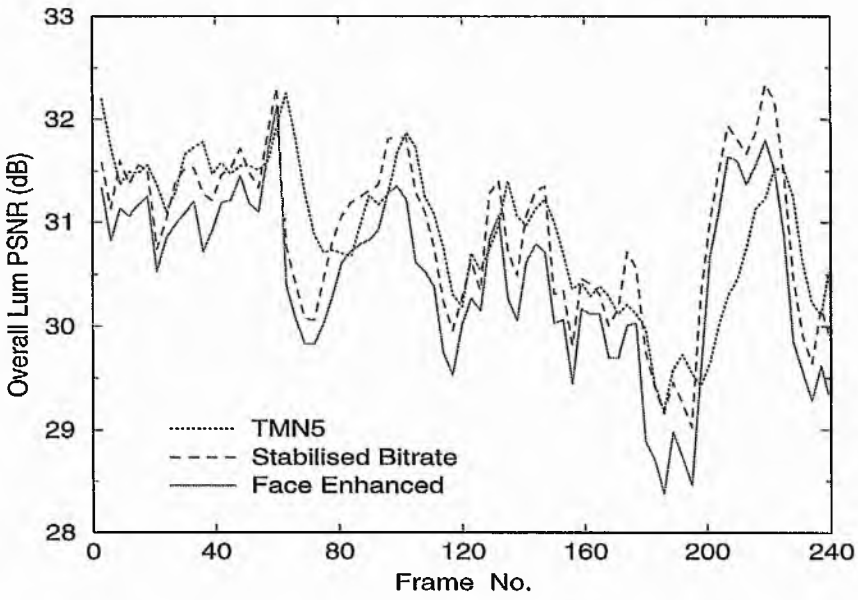


Fig. 4.17 : Overall PSNR of Foreman sequence.

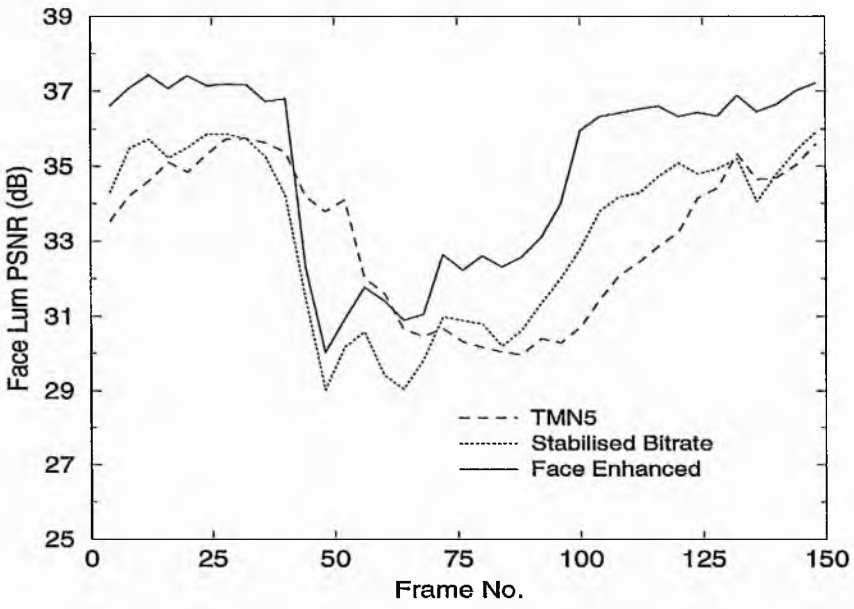


Fig. 4.18 : Luminance PSNR around the face of Suzie.

4.6.4 Constraints

The employed face location algorithm is a very simple one. It fails to locate the face when applied to images with white background or the person wearing brighter clothing, for example in Claire sequence. Hence in practice a more robust technique is required, for example one that is capable of detecting the eyes and mouth of a person and thus the face can be determined [43].

Due to the required segmentation between face region and background, a full image is needed to be captured before locating the position of the face can begin. This imposes an additional delay of one frame. However, we think that the size of the QCIF is relatively small, thus the delay is tolerable. One way to avoid this problem is to perform the segmentation on the previous original or reconstructed image. Simulation carried out on Miss America sequence gave a slight decrease of about 0.1 dB in the PSNR around the face. However, this decrease in PSNR depends on the translational movement of face between each encoded frame.

The maximum quantisation step size that is allowed to change between each transmitted MB in the same slice of macroblock (GOB) is limited to ± 2. Consequently, the background in the same GOB as the face is coded more accurately than the rest. Ironically, it is also partly due to this reason that when the locating of the face is carried out on the previous frame, there is only a small decrease in PSNR of the face.

4.7 Concluding Remarks

A feedforward rate control algorithm which is capable of tracking closely the target bit-rate per frame is presented. In contrast to the traditional approach of selecting the quantiser step size using the fill level of the buffer, the proposed algorithm makes use of the prediction error for selecting a suitable step size in order to achieve a constant output bit-rate. A coded-bit table with prediction error per block as the entry is created for estimating the required bits for coding the residual image. Simulation results show that stable output bit-rate is achievable with this rate control algorithm. A simple method for making the coded-bit estimation table adaptive to the image statistics is also implemented. Further work is needed for improving the algorithm. For instance, subjective quality can be taken into consideration so that image degradation due to the changes in quantisation step size between each coded frame can be kept to a minimum and at the same time without introducing too much delay.

Using the above rate control algorithm, which is essentially a coded-bit estimation algorithm, a bit allocation algorithm for improving the subjective quality of typical head and shoulder image is implemented. It is based on segmenting the scene into face region and background and coding them with different quantisation step sizes. Simulation results showed a significant improvement in the objective as well as subjective quality of the reconstructed images. The other advantage is that the output bit-rate is less fluctuating.

Chapter 5

Rate - Constrained Motion Compensation And Mode Selection

5.1 Introduction

In general, all the recently recommended video coding standards employ a hybrid coding configuration, in which a motion-compensated difference frame is subjected to a two-dimensional intraframe transform coding operation [1][32]-[34][60]. The intraframe transform coding is used for removing spatial redundancy in a single frame whereas the interframe motion-compensation coding, taking advantage of the strong correlation between successive frames of a video sequence, is used for reducing the temporal redundancy [1][18].

Several successful motion estimation techniques have been reported in the literature for reducing the temporal redundancies between successive coded frames. Element recursive techniques [44]-[46], gradient techniques [47][48] and Block Matching Motion Estimation techniques (BMME) [49] are some of the well-known techniques. Due to its easy implementation, very simple motion model assumed and cost associated with transmitting or storing the motion vectors, BMME enjoys wide popularity in a number of applications, especially in hybrid block-transform video compression system where it is adopted by all the current video coding standards [32]-[34][60].

Partly because inter-frame information such as motion vectors and additional side information usually contributes to a very small portion of the overall bit-rate, classical BMME allocates the bit-rate necessary for choosing the motion vectors which result in minimum distortion between the estimated and the actual image blocks. This minimum distortion selection criterion does not take into account the quantisation operation of the residual, assuming that this will lead to fewer bits for coding the resultant residual images. While this is true for high bit-rate coding in which small quantisation is used for coding the prediction error, unfortunately, this bit allocation strategy becomes increasing inefficient as temporal information begins taking up a considerable proportion of the available bit budget in low bit-rate coding [69][70]. In other words, the problem of optimal bit allocation between motion vector rate and prediction error rate subjected to a distortion constraint has not been formulated in conventional hybrid coding schemes. Obviously, the ideal solution is to select the motion vectors that minimise the overall distortion and bit-rate after the residual image is coded.

In this chapter, an efficient motion compensation algorithm where motion vectors are selected based on a rate-distortion measure will be presented. The algorithm minimises the rate subjected to a constraint on the overall distortion by optimising jointly the motion vector coder and residual coder. After that, the rate-distortion constrained selection algorithm is extended to the selection of coding mode. A simplified algorithm for reducing the computational complexity is implemented for the latter operation. Finally, the two operations are combined to form a rate-distortion optimised coder.

5.2 Block Matching Motion Estimation

BMME was first used by Jain and Jain [49] who simply matched the element luminance value in a given block of size M x N in the current frame with those of a similar M x N region within a search window of (M+2p) x (N+2p) in the previous

frame. Here p is the allowable maximum displacement. The matching criterion may be an error measure such as MSE or MAE [1][20].

$$\text{MAE} = \frac{1}{M \text{ x } N} \sum_{i=0,j=0}^{M-1,N-1} |x(i,j) - \hat{x}(i,j)| \tag{5.1}$$

If full search motion estimation is used, a total of $(2p+1)^2$ search points are required which is very computationally intensive. Although the number of search point can be significantly reduced by limiting the size of the search window, which is 2p by 2p in this case, it may lead to poor results, or complete failure for sequences containing fast motions. As a result, numerous schemes such as 'logarithmic' search [49] and 'three-step' search [50] have been proposed to reduce the number of search points whilst not sacrificing the reliability of the estimation too much.

On the other hand, various error measures have also been proposed for reducing the complexity of error evaluation. An example is the above-mentioned MAE which replaces the square operation required by MSE with an absolute value operation (which requires less computational power). The trade off is a small decrease in efficiency - a small increase in the variance of the prediction error when compared to the result of MSE which is considered as the optimal error measure. Another example is the 'matched' and 'mismatched' criterion [51] used by Gharavi and Mills which required (MxN) comparison and (MxN) addition operation per search and can achieve a PSNR almost as high as that obtained when using MSE, and significantly better than that produced by the use of MAE.

In spite of these efforts, the power of present day's computers allows the use of full search procedure, which guarantees to find the correct global minimum distortion. As for the error measure, SAD which is an equivalence of MAE is often preferred over the MSE for fast implementation purpose in special hardware architectures [1][20].

BMME is a simple and cost effective technique and, as far as reducing the variance of an interframe prediction is concerned, there is no doubt that it can be made to function efficiently. One drawback associated with it is that it operates primarily on the

assumption of translational motion. Therefore, other varieties of motion such as rotation, zooming are not well predicted by this technique. However, this problem can be partially solved by using a smaller block size [52][53].

5.3 Analysis by Synthesis Technique

Analysis-by-Synthesis (AbS) was proposed initially for speech processing [54] and has been successfully applied to low bit-rate speech coding [55]. The general AbS technique is a trial and error approach towards choosing the best candidate from a number of candidates. The effect of a certain input candidate to a process is analysed by examining the output and then comparing it to some preset reference. The input candidate for which its output is closest to the reference is chosen. The benefit of such an approach is that the best candidate is guaranteed to be determined. The cost is that the procedure has to be repeated as many times as the number of candidate inputs even if only one is chosen in the end. However, the number of candidates can be reduced by setting some conditions for them to be chosen as candidates in the first place. Some loss in performance is inevitable but the reduction in the computational load may be significant.

Analysis-by-synthesis algorithm corresponds to a closed-loop system, as depicted in Fig. 5.1 [55]. There is a feed-back from the output of the system to the processes which makes the decisions. In the case of motion vector selection, the feedback information can be the resulting bit-rate and image quality. Inevitably, this will lead to a substantial increase in the complexity of the system, but better performance is expected.

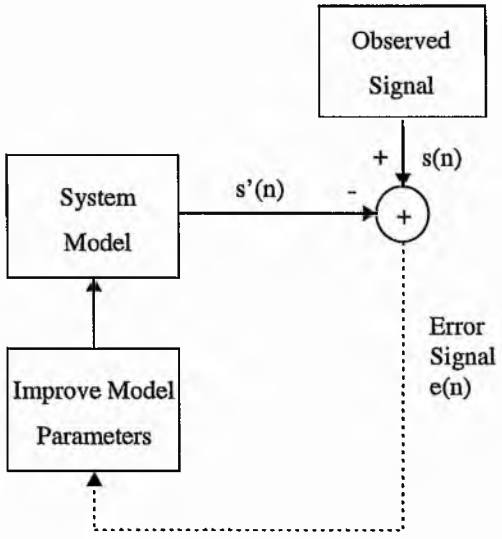


Fig. 5.1 : General block diagram of analysis-by-synthesis closed-loop analysis.

Fig. 5.2-5.4 illustrated the plots of sum of absolute differences (SAD), resulting bit-rate and reconstruction MSE values for a particular macroblock of the Foreman sequence. It can be seen from these 3 plots that using SAD as the selection criterion for the motion vectors does not always lead to the minimum reconstruction MSE or bit-rate. If analysis by synthesis approach is employed, the optimum motion vector that gives an improved result in terms of bit-rate and image quality might have a better chance of being selected.

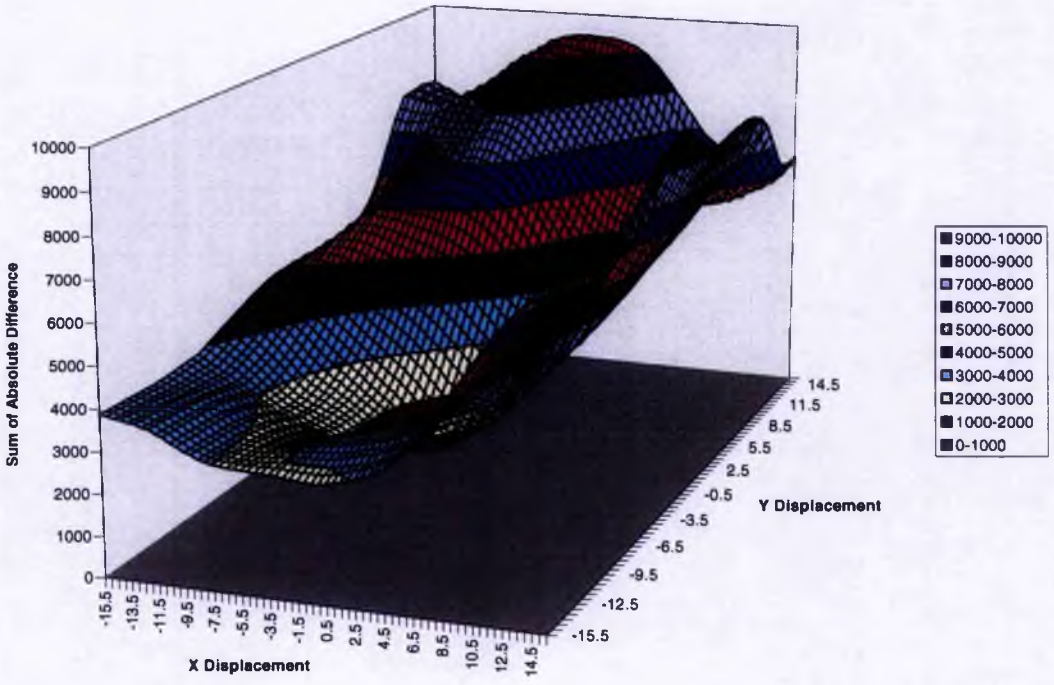


Fig. 5.2 : Sum of absolute difference of a particular macroblock of the Foreman sequence in all allowable motion vector displacements.

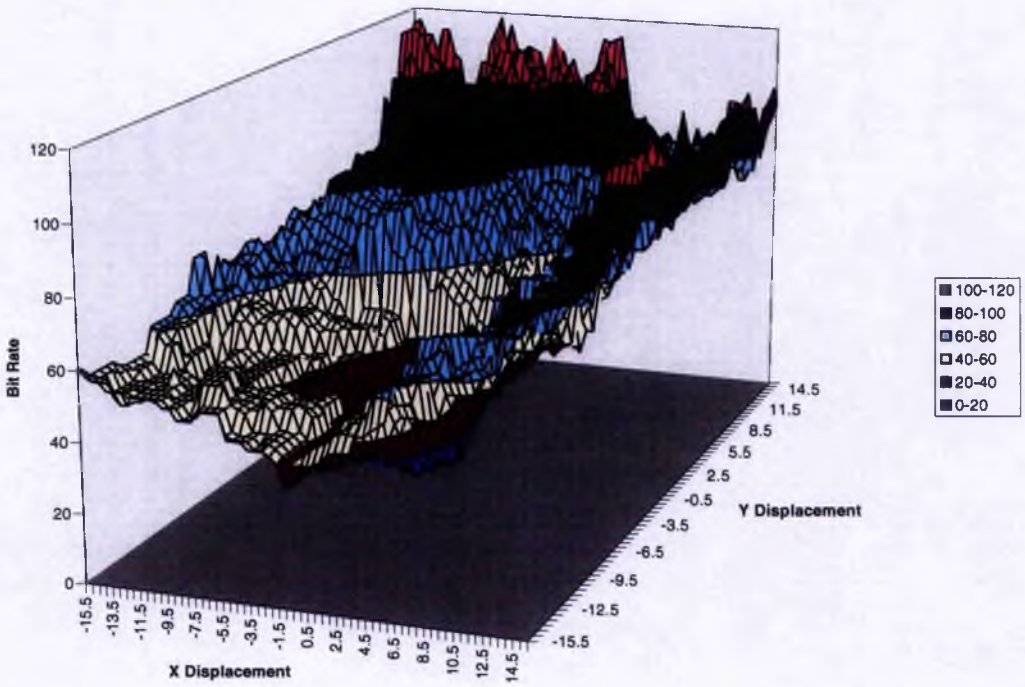


Fig. 5.3 : Resulting bit-rate of the corresponding macroblock in all allowable motion vector displacements.

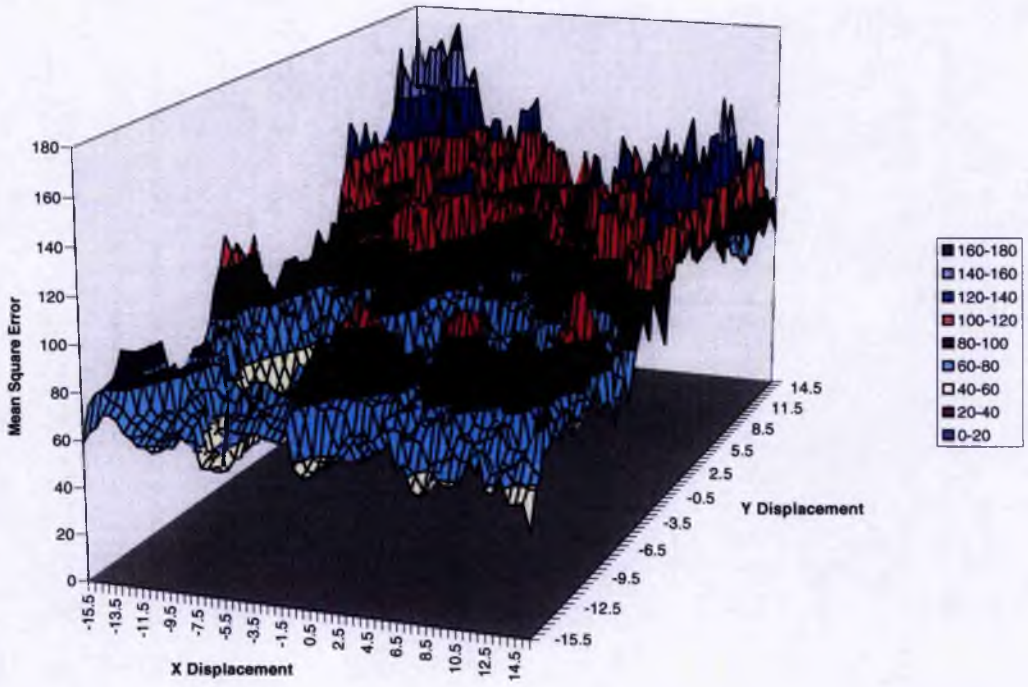


Fig. 5.4 : Reconstruction mean square error of the corresponding macroblock in all allowable motion vector displacements.

5.4 Motion Estimation Based on Analysis-by-Synthesis Approach

In order to accurately determine the number of bits required to encode a given residual image and its associated motion vectors, and the reconstruction error, dummy encoding, decoding and reconstruction of the image are required. For an exhaustive search with a (m+2p) x (n+2p) window, the number of motion vector candidates is $(2p+1)^2$ for a given m x n block for integer pel accuracy. This number will increase by a few times if half-pel accuracy is required [3]. If all these candidates are to be dummy encoded, the increase in computational load will be tremendous and seems unjustified. Hence, only the *N* best motion vector candidates in the sum of absolute difference sense will be submitted to the analysis-by-synthesis process.

During the motion estimation search, the N best motion vectors which give the smallest SAD are chosen. These motion vectors are refined to half-pel accuracy using a search window of +/- half-pel around the chosen vectors. This will generate $9N$ half-pel candidates from which the new best N candidates are chosen. The algorithm has to avoid choosing duplicated half-pel candidates to improve its efficiency. For instance, if motion vector MV_1=(1,1) and motion vector MV_2=(1,2) are chosen during the integer-pel search, they both end up having the same half-pel motion vectors MV=(0.5,1.5), (1,1.5) and (1.5,1.5) after the half-pel search.

Following the choice of candidates, the number of bits required to encode the motion vectors and its associated residual image, and the resulting reconstruction error will be determined through dummy encoding for each vector. The candidate which requires the least number of bits is chosen initially. A very significant reduction in bit-rate can be achieved using this criterion for motion vector selection [56]. Nevertheless, this reduction in bit-rate is achieved at the expense of reconstruction error. Compared to the case where only the prediction error is used as the selection criterion, this is the other extreme case where only the bit-rate is considered. A compromise between the bit-rate and reconstruction error has to be made. Further investigation has shown that there are situations whereby using another candidate will lead to a significant decrease in the reconstruction error, but requires only a few more bits to encode.

5.4.1 Iterative Rate-Constrained Motion Vector Selection

After selecting the candidate which gives the minimum bit-rate, the following parameters are calculated for the rest of the N-1 candidates according to the following formula :

Let g_i = gain per extra bit used,

b_s = number of bits required to code the block with the selected motion vector,

SAD_s = reconstruction SAD associated with the selected motion vector. (Note that the SAD referred here is the sum of absolute difference between original block and the reconstructed block.)

For the rest of the N-1 candidates,

$$\begin{aligned} \Delta SAD_i &= SAD_s - SAD_i \\ \Delta b_i &= b_i - b_s \\ g_i &= \Delta SAD_i / \Delta b_i \end{aligned} \tag{5.2}$$

Next, the candidate with the maximum g_i is chosen and if it is greater than a threshold T, the initially selected motion vector candidate is replaced by this candidate. Thus, the amount of bit-rate reduction and the associated quality degradation are determined by the threshold T. Obviously, using a large threshold value will result in greater bit-rate reduction but higher quality degradation and vice versa.

The operation of the selection process is clearly illustrated in Fig. 5.5. As can be seen, the algorithm is essentially calculating the slopes between the initially selected point (candidate), A and the other N-1 best points. The calculated slopes correspond to the gain, g_i of the above algorithm. The one with the maximum gain is selected, which is point B in the example. However, there is a possibility that this might not be the optimum point. Point C could be the optimum point if slope b2 is greater than the preset threshold T. Therefore, the selection of the optimum point is repeated using this newly selected point B as the new reference point and this iteration is repeated until there is no change in the selected point.

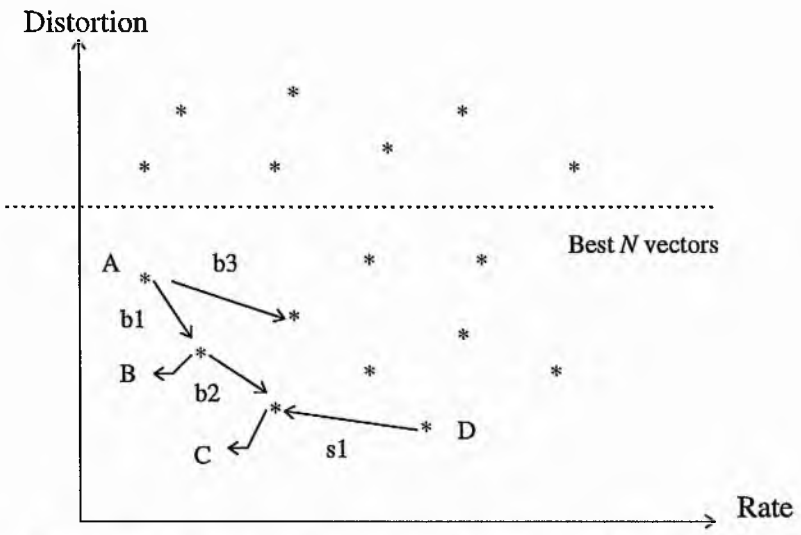


Figure 5.5 : Illustration of the selection algorithm with $N = 10$.

From the above figure, it also seems possible to obtain the optimum point by starting from the point which gives the minimum SAD (point D). The same algorithm can be used but this time only point which gives the minimum positive gain and of value less than a preset threshold is selected. However, the final selected point might not be the same for both cases, especially when N gets larger. This emphasizes the importance of the threshold selection in finding the optimum motion vector.

5.5 Simulation Results and Discussion

The above rate-constrained selection algorithm was incorporated into TMN5 for motion compensation. Fixed quantisation step size was used in the experiments in order to show the reduction in bit-rates. The results obtained using TMN5 were used as the reference for the calculation of improvement obtained using the proposed algorithm.

Table 5.1 : Results obtained using TMN5, with all negotiable options off.

Sequence	Frame No.	Frame Rate	Quantisation step size	Lum PSNR (dB)	Bit rate (kb/s)
Suzie	0 - 124	12.5	12	33.48	28.10
Carphone	150 - 274	6.25	15	30.40	27.02
Foreman	200 - 324	6.25	23	28.56	28.83

The sequences Suzie, Carphone and Foreman were used in the simulations and encoded with the conditions stated in Table 5.1, using the following 6 criteria for motion vector selection :

C1 : minimum prediction error, SAD;
C2 : minimum reconstruction error;
C3 : minimum bit-rate;
C4 : maximum SAD gain per bit starting from the minimum bit-rate point(1 iteration);
C5 : Same as C4 but iterated while the new gain is greater than the threshold;
C6 : Same as C5 except starting from the minimum SAD point.

Experimental results showed that using threshold $T = QP$, quantisation step size gives a reasonably good result in terms of bit-rate reduction without much quality degradation. As shown in Fig. 5.6, the threshold value determines the trade-off between bit-rate and quality.

Fig. 5.7 shows a graph of percentage change in bit-rate as a function of the number of candidates, using the above 6 criteria for the sequence Foreman. Fig. 5.8 is the corresponding change in objective quality. As expected, the minimum reconstruction error criterion and the minimum bit-rate criterion form the upper and lower bound for the percentage change in bit-rate respectively, with the other 4 criteria fall in between them. Foreman sequence is able to achieve a slight drop in bit-rate using the minimum prediction error criterion but this is not always true for all sequences. For example, Carphone has a slight increase in bit-rate using this criterion. This is mainly due to the quantisation in the encoding process.

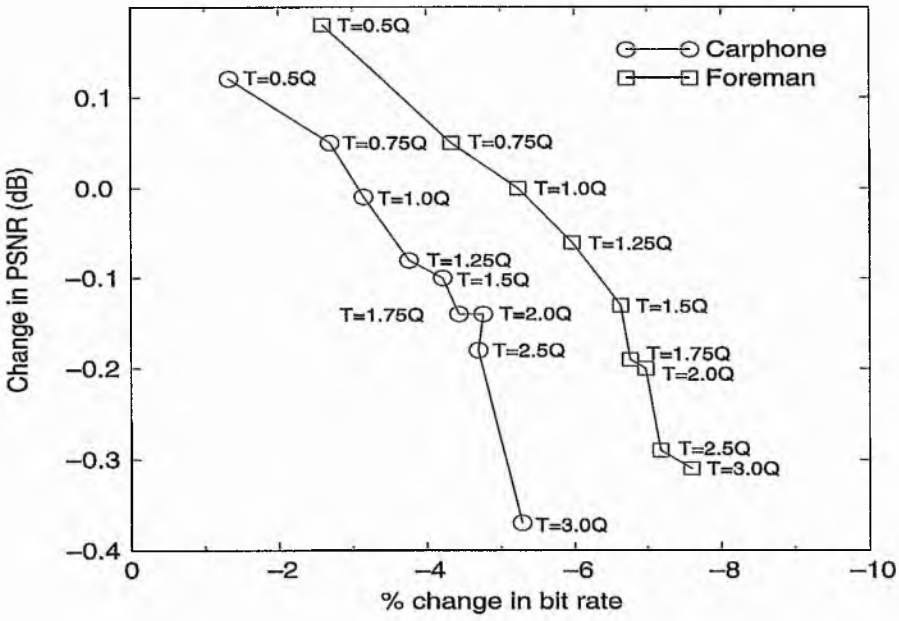


Fig. 5.6 : Results of the sequence Foreman and Carphone obtained using different threshold for criterion 5, with $N = 10$.

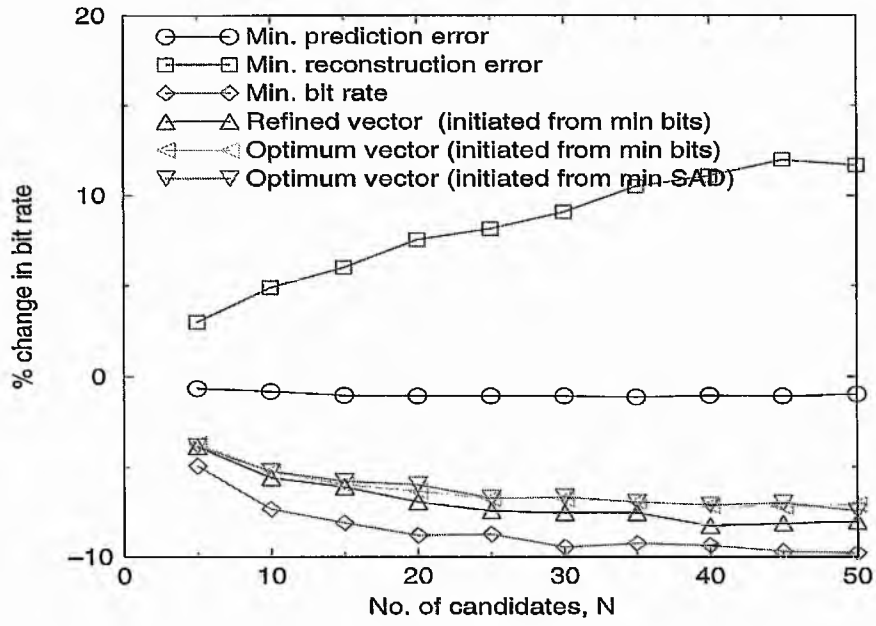


Fig. 5.7 : Percentage change in bit-rate for the sequence Foreman encoded using 6 different selection criteria.

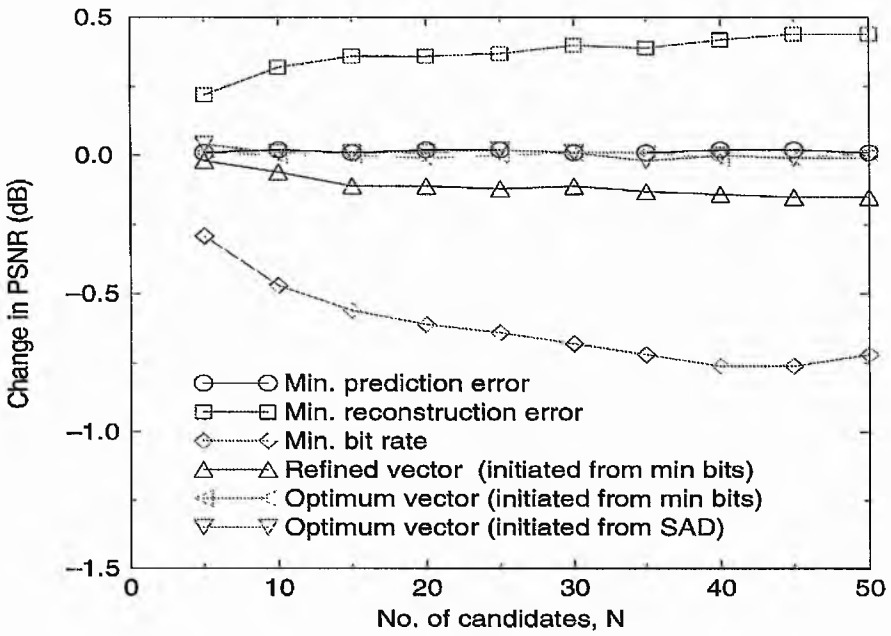


Fig. 5.8 : Change in PSNR for the sequence Foreman encoded using 6 different selection criteria.

As can be seen, the minimum bit-rate criterion gives the maximum bit-rate reduction but this is accomplished with about 0.7 dB decrease in PSNR. On the other hand, about 7% reduction in bit-rate can be achieved with virtually no change in the objective quality using criterion 5 or 6. Another 0.5% - 1% of bit-rate reduction is achievable, with changes in the PSNR remains within ± 0.04 dB, if MSE is used as the distortion measure for criterion 5. Criterion 4 is shown to give a bit-rate much closer to the minimum bit-rate criterion but with a lower PSNR than criteria 5 and 6. Even though the degradation in objective quality is not obvious, its advantage is lost when a less active sequence is used, such as Carphone in Fig. 5.10. As for the subjective quality using pair comparison, criteria 5 and 6 did not give any degradation whereas there is a noticeable degradation for criterion 3, for the case when N=50.

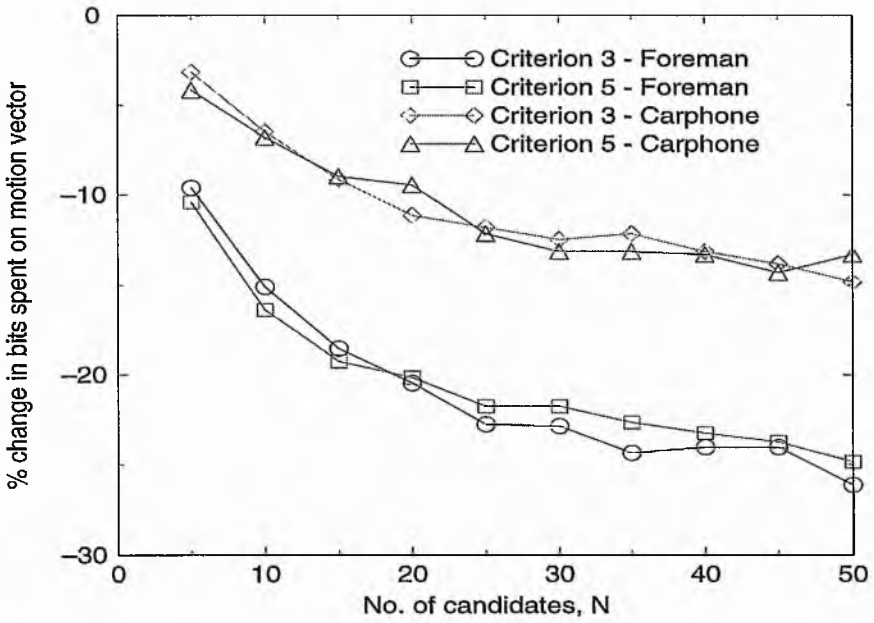


Fig. 5.9 : Percentage change in bits spent on motion vector for the Foreman and Carphone sequence using criteria 3 and 5.

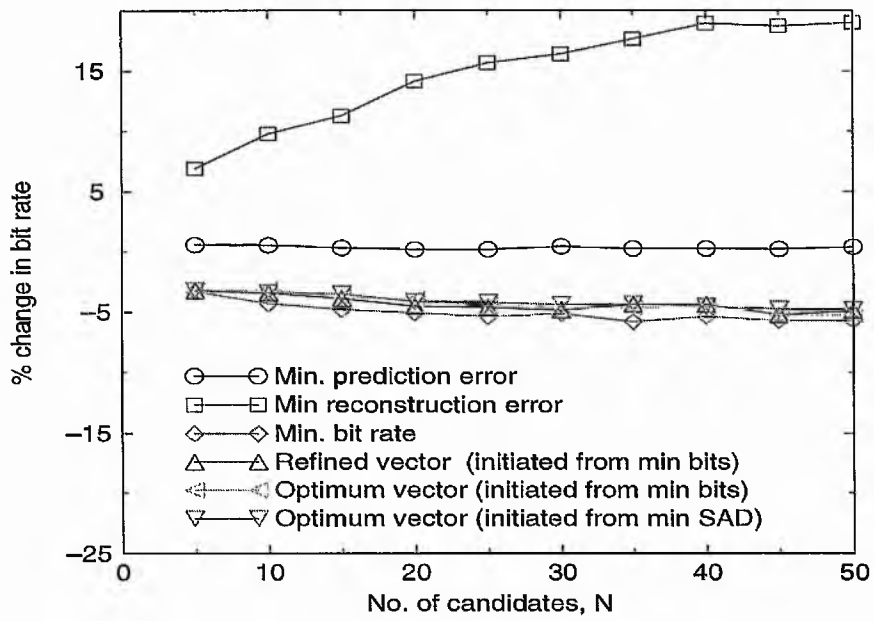


Fig. 5.10 : Percentage change in bit-rate for the sequence Carphone encoded using 6 different election criteria.

Experimental results showed that the extra saving in bit-rate using the minimum bit-rate criterion compared to criterion 5 or 6 comes from the saving in coding the transformed coefficients, thus leads to a degradation in the objective image quality. As for the bit saving for coding the motion vectors, criteria 3, 5 and 6 achieved roughly the same percentage as shown in Fig. 5.9. This observation is confirmed by Fig. 5.10 which shows that the difference in bit-rate reduction between these 3 criteria is less prominent. Since the Carphone sequence is not as temporally active as Foreman which also contains a screen change when the camera swings away from the foreman, we expect the saving in bits for coding the transformed coefficients to decrease for criterion 3 and increase for criteria 5 and 6. Fig. 5.11 shows that the bit saving for coding the coefficients decreases about 3% for criterion 3 but increases slightly for criterion 5.

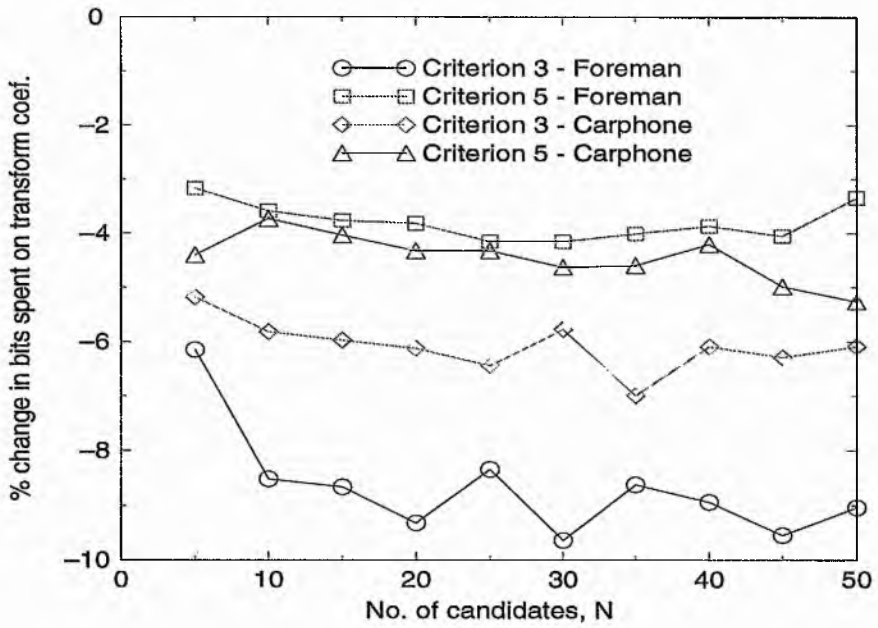


Fig. 5.11 : Percentage change in bits spent on luminance transform coefficients for the sequence Foreman and Carphone using criteria 3 and 5.

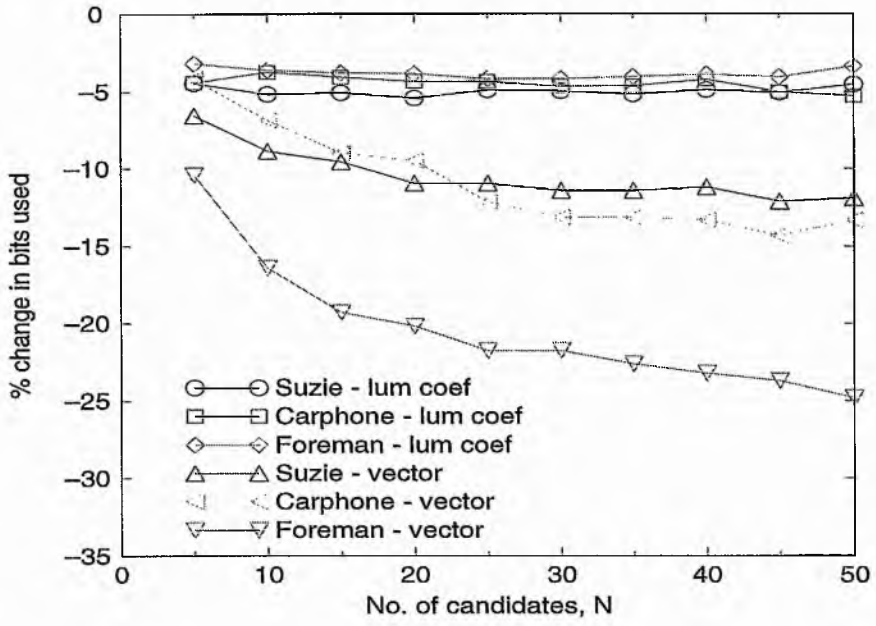


Fig. 5.12 : Percentage change in bits used to encode the transform coefficients and motion vectors for the sequences Suzie, Carphone and Foreman, using criterion 5.

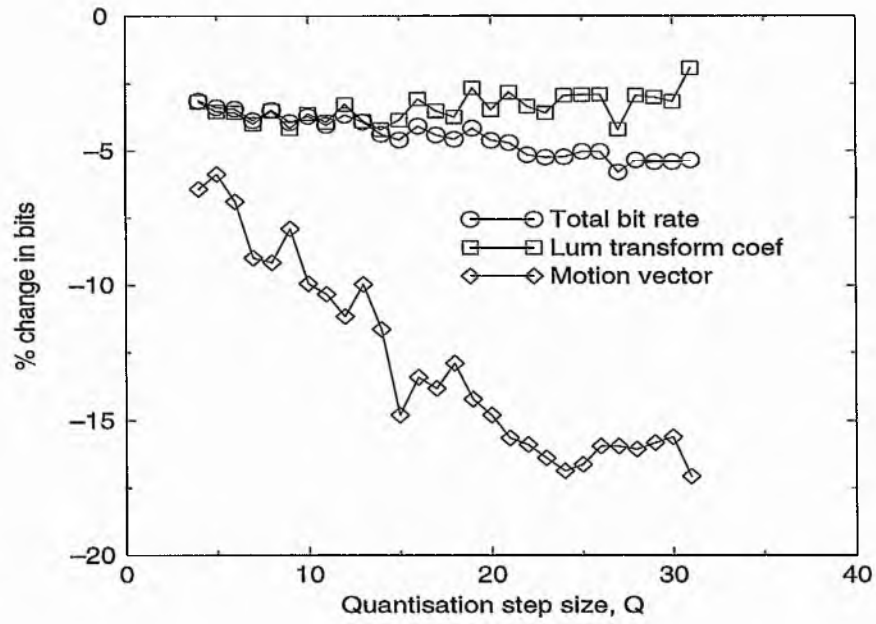


Fig. 5.13 : Percentage change in bits spent on transform coefficients and vectors as a function of quantisation step size for the sequence Foreman, using criterion 5 with $N = 10$.

Fig. 5.12 shows the breakdown of the reduction in bit-rate into its components, i.e. bits spent on the transform coefficients and the motion vectors. As can be seen from this graph, increasing the number of candidates does not have much effect on further reducing the number of bits spent on the transform coefficients. However, its effect on reducing the number of motion vector bits is significant. The motion fields estimated from minimum SAD criterion and criterion 5 are plotted in Fig. 5.14. It is clear that the motion field is much smoother using the rate-constrained criterion than the minimum SAD criterion where large motion vectors are predicted for the background. Fig. 5.12 also shows that the amount of reduction depends greatly on the temporal activity of the input sequence. For instance, Foreman sequence which contains a lot of motion in all directions and camera panning, has a very significant reduction in the number of motion vector bits whereas less active sequences such as Suzie achieved only half of this value. In contrast, the reduction in bits spent on the transform coefficients is slightly higher for less active sequences.

Fig. 5.13 plots the percentage change in bits spent on transform coefficients and motion vectors as a function of quantisation step size. The corresponding change in PSNR is within $\pm$ 0.05 dB. The percentage change in bits spent on each parameter for a particular step size is calculated using the results obtained with TMN5 as the reference and the proposed algorithm both encoded with that step size. It can be seen that the number of bits saved from the motion vectors at high bit-rates, i.e. using a small quantisation step size, contributed very little to the overall bit-rate reduction since most of the total bit-rate is spent on the transform coefficients. On the other hand, at low bit-rate, the saving in bits from motion vectors becomes important; about 3 times higher than the number of bits saved from the transform coefficients, as the number of bits spent on the motion vectors constitutes a more significant proportion of the total bit-rate. Fig. 5.15 is a rate-distortion curve of the sequence Foreman which shows that better performance can be obtained using the proposed algorithm.

(a) Bit spent on motion vector = 773.

(b) Bit spent on motion vector = 612.

Fig. 5.14 : Motion field estimated by (a) minimum prediction error, SAD, criterion and (b) rate-constrained criterion on Foreman sequence.

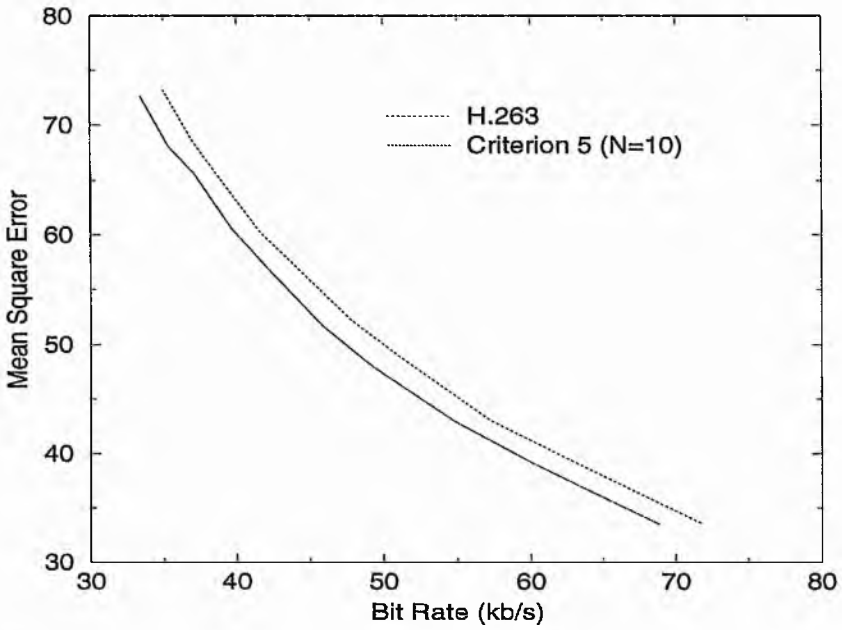


Fig. 5.15 : Rate distortion curve of Foreman sequence using criterion 5 with $N = 10$.

5.6 Computational Complexity

As for the increase in computational complexity, profile analysis on TMN5 shows that about 90% of the encoding time is spent in motion estimation related routines. For the proposed algorithm, there is a negligible increase in computation for the integer-pel motion vector search, but an increase of (N-1) times for the half-pel motion vector search. However, the increase in computation for the latter case can be reduced by eliminating the re-processing of duplicated half pel motion vectors which constituted a significant portion (about 40%) of the total $9N$ half pel motion vectors. Moreover, another N times of computing the DCT, coding, IDCT and reconstruction for determining the required bits and reconstruction error are also required. In view of the large proportion of time being spent on motion estimation related routines, these two major increases in complexity will not increase the overall computational load significantly. Besides, the extra computation required for DCT, IDCT and

reconstruction can be accomplished through parallel processing if speed is vital, as for the case of real time application.

5.7 Mode Selection Strategy

A key problem in high compression video coding is the operational control of the encoder. As described in section 3.7.2, the established block-based standards subdivided the current frame into unit regions called macroblocks, each consists of four 8x8 luminance blocks and two 8x8 chrominance blocks. Several modes of operation such as uncoded mode, intra mode and inter mode are provided by these standards which can be selected on a macroblock by macroblock basis. Since typical image sequences contain a widely varying contents and motion, these available modes allow the encoder to efficiently code different regions using a suitable mode. For example, block-based motion compensation followed by transform-quantisation of the prediction error (interframe coding) may be used for macroblocks that can be predicted from the previous decoded frame. For relatively unchanged macroblock, simply copying it from the previous region (uncoded) is preferred since no extra bits are required. On the other hand, coding a particular MB independently (intraframe coding) may be more efficient when the block-based translational motion model breaks down. As a result of this multi-mode operation, an improved rate-distortion performance is expected if the modes are chosen properly for each type of MB. Consequently, the strategy for selecting the proper mode for each MB of the current frame becomes an important issue for the efficiency of an encoder.

In all the established standards, the mode selection criteria is not subjected to recommendation and is left open for individual implementation as part of the coding control strategy. Past papers on video coding have applied rate-distortion theory to improve the performance of MPEG encoders by optimising the frame type and/or the quantiser selection [57][58]. One drawback with these approaches is that the problem of selection the best encoding strategy for a frame is not considered at the MB level. Instead, the optimisation is accomplished by assuming a fixed number of quantisation

choices for each frame. Recently, a more efficient, macroblock-by-macroblock mode selection strategy was proposed in [59] to overcome this drawback. In general, both the above proposed methods optimise the encoder operation within a rate-constrained product code framework using a Lagrangian formulation. The associated Lagrangian formulation leads to an unconstrained cost function and a trellis whose associated paths correspond to all possible operational rate-distortion points. A dynamic programming solution based on the Viterbi algorithm is used in [59] to locate the optimal path in the trellis.

In the following sections we will be looking into the strategy of efficient mode switching. The rate-constrained selection algorithm used in section 5.4.1 for motion vector selection will be applied to block-based multi-mode coder for operation mode selection. This is a much simpler and less complex solution compared to the above proposed methods since it does not involve any kind of trellis. The operating mode of each macroblock is decided immediately after its associated motion vectors are found, unlike the strategy used in [59] where the mode of each macroblock is optimised for the whole slice of macroblocks in which it belonged, i.e. the mode for each macroblock can only be decided after motion estimation for the whole GOB is finished. In view of the above mentioned difference, the results obtained using this proposed strategy may be sub-optimal compared to that of [59].

5.7.1 Operation Mode of Standard Block-Based Coder

In standard block-based coders, the two picture coding types commonly found in them are INTRA and INTER. The INTRA picture type is a very bit consuming coding operation because it allows only INTRA coding to be used throughout the whole frame. Therefore, it is only used for special purposes such as coding the first frame of a Group Of Picture (GOP) as in MPEG or whenever there is a screen change. It is also employed for error control by transmitting INTRA coded picture at regular interval or at the request of the decoder for flushing out a corrupted picture. On the other hand, the INTER picture type is more complicated but flexible in that it allows individual macroblocks to be coded using one of the several available coding modes. The coder

is allowed to choose a proper mode for each macroblock according to their statistics, thus leading to a significant improvement in coding efficiency compared to the INTRA picture type. The operation of each of the available modes will be briefly described below. A detailed description can be found in [3].

- The UNCODED mode indicates to the decoder through a single bit that the current macroblock is to be represented by simply copying the contents of the corresponding macroblock in the previous decoded picture. This mode is especially useful for stationary background.

- For the INTRA mode, the macroblock is coded independently and no information from the previous decoded frame is involved in its coding processing. The four luminance blocks and 2 chrominance blocks are transformed by a 8x8 DCT. The resulting coefficients are quantised, zigzag scanned and run-length coded. This mode is preferred for coding regions that cannot be predicted from the previous picture such as uncover background.

- In the INTER1MV mode, the current macroblock is predicted from the previous decoded frame using a single motion vector. After motion compensation, the prediction error is transformed and quantised in the same manner as the INTRA mode. The use of this mode is more bit efficient as less bits are usually needed compared to the INTRA mode.

- INTER4MV mode, which is available only to H.263, is an advanced prediction mode that allows the use of four motion vectors per macroblock. It functions basically similar to the INTER1MV mode except that smaller block size is used for achieving more accurate prediction. Moreover, an overlapped block motion compensation technique is employed in the reconstruction stage. As a result, blocking artifacts has been significantly reduced.

5.7.2 Rate - Distortion Optimised Mode Selection Strategy

After motion-compensation, the encoder needs to choose an appropriate mode (among UNCODED, INTRA, INTER1MV and INTER4MV) for the current macroblock. Using dummy encoding, the number of bits associated with each mode and the corresponding reconstruction distortion are calculated. In order to reduce the complexity, the correlation between the current macroblock, its preceding and succeeding macroblocks in terms of bit-rate and distortion have not been taken into account during the calculation. Hence, the use of overlapped block motion compensation has been eliminated when calculating the resulting reconstruction distortion.

It is not difficult to realise that these four coding modes are equivalent to four motion vectors. Hence, using the rate-constrained selection algorithm described in section 5.4.1, the mode which gives the best result in terms of bit-rate and distortion is selected for the current macroblock.

5.7.3 Simulation Results and Discussion

The proposed Rate-Distorion (R-D) optimised mode switching strategy is applied to the H.263 video coding standard and simulation results are shown in Fig. 5.16 to Fig. 5.19. Results obtained by TMN5 is used for comparison. For fairness, both sets of the experiments were carried out under the same conditions, i.e. only advanced prediction option is used and the frame rate is held constant at 8.33 fps so that the same frames from each video sequence are encoded by both algorithms. Due to the lack of a suitable rate controller, which can affect the compression operation significantly, fixed quantisation step size was used for these simulations.

Experimental results have shown that the proposed mode selection strategy outperformed TMN5 for all the test sequences. Fig. 5.16 and 5.17 are the rate-distortion plots for the Carphone and Foreman sequences respectively. Both plots indicate a substantial improvement in objective quality or bit-rate when compared

with TMN5. On the other hand, by using the same rate controller as in TMN5, similar objective improvement are observed for most of the bit-rate. Subjective assessments through side-by-side comparison confirmed these objective improvement with much better viewing quality for sequences encoded with the proposed algorithm.

Fig. 5.18 and Fig. 5.19 illustrated the relative frequency, hence the relative efficiency, of each mode being selected at various bit-rates. Although these plots of relative frequency of each mode differ significantly for different video sequences depending on their contents, there are some noticeable common characteristics in them. The UNCODED mode, as expected, is used more often at low bit-rate than at high bit-rate. In contrast, the more accurate and expensive (in terms of bit-rate) INTER4MV mode is almost not used at very low bit-rate but is increasingly selected as bit-rate increases. As for the INTER1MV mode, its frequency of being selected increases as the bit-rate increases, but remains unchanged or begins to fall after 30 kb/s.

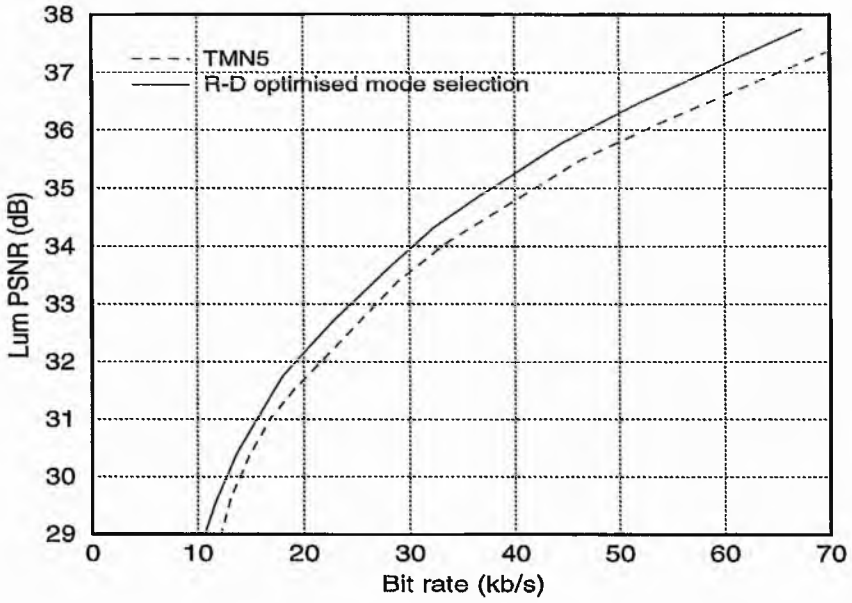


Fig. 5.16 : Comparison in coding performance between the proposed mode selection strategy and TMN5 for the first 150 frames of Carphone sequence.

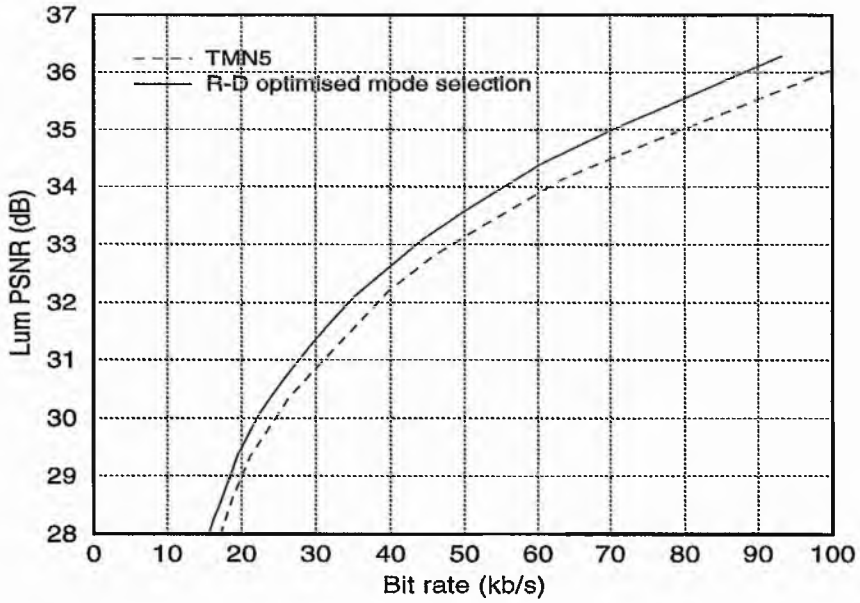
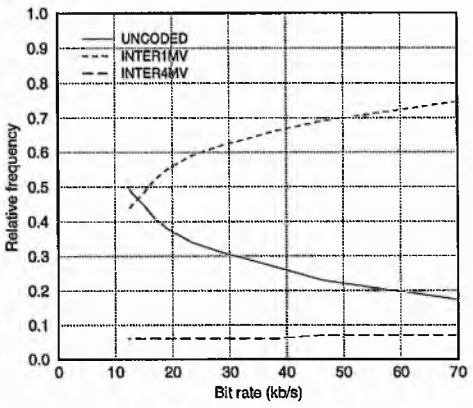
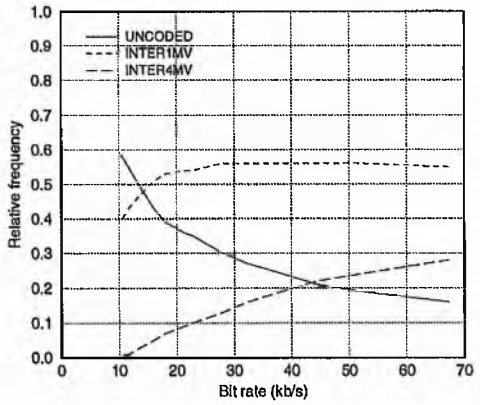


Fig. 5.17 : Comparison in coding performance between the proposed mode selection strategy and TMN5 for the first 150 frames of Foreman sequence.

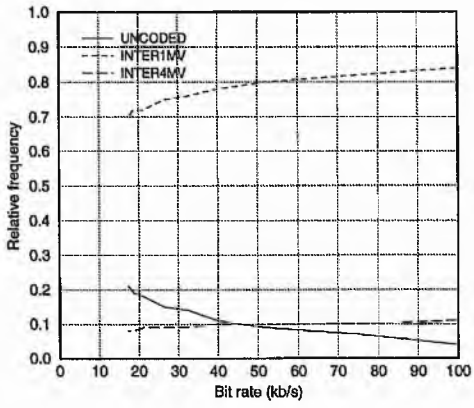


(a)

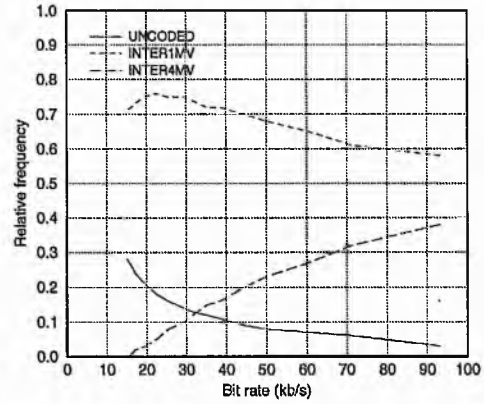


(b)

Fig. 5.18 : Relative frequency of mode versus bit-rate for the first 150 frames of carphone sequence (a) TMN5 (b) Proposed mode selection strategy.



(a)



(b)

Fig. 5.19 : Relative frequency of mode versus bit-rate for the first 150 frames of foreman seq. (a) TMN5 (b) Proposed mode selection strategy.

5.8 Computational Complexity

For the above mode selection algorithm, 4 extra set of encoding and decoding operation, i.e. DCT, IDCT, reconstruction of image for calculating the reconstruction distortion, are required for the INTRA, UNCODED, INTER1MV and INTER4MV modes. Although this increase in computational load is relatively insignificant, it might not be desire for real time application.

Experimental results have shown that using the same criteria as TMN5 [39] for selecting between INTRA mode and INTER modes give almost the same results. The criteria used by TMN5 is as followed :

$$\text{MB\_mean} = \left(\sum_{i=1,j=1}^{16,16} original \right) / (16 \times 16) \tag{5.3}$$

$$\text{Absolute distortion, } A_{MB} = \sum_{i=1,j=1}^{16,16} |original - MB\_mean| \tag{5.4}$$

$$\text{INTRA mode is chosen if : } A_{MB} < (SAD_{INTER} - 500) \tag{5.5}$$

The decision on whether to use INTRA or INTER prediction in the coding is made immediately after the integer-pixel motion estimation. If it is adopted in the proposed selection strategy, the set of dummy encoding and decoding operations will be reduced to 3 if INTRA mode is not selected. And if INTRA mode is selected, no further operation is necessary, either for half-pixel motion search or mode selection among the other three modes i.e. UNCODED, INTER1MV and INTER4MV mode. However, the reduction in computational load will not be significant since the average percentage of INTRA mode being chosen in the coding process of the test sequences is always accounted to less than 1%.

5.8.1 Simplified Algorithm Using Coded-bit Estimation

Most of the additional computational load due to dummy encoding and decoding is attributed to the computation of the prediction error rate and reconstruction MSE : DCT, quantisation, encoding of transform coefficients, dequantisation, IDCT, reconstruction, and computation of the MSE. The computation of the motion vectors rate contributes only to a small proportion of the overall added computational load as it is straightforward and the numbers of bits can be readily obtained using the VLC tables. For these reasons, a substantial reduction in the added computational load can be obtained if the coded-bits for the transform coefficients and the reconstruction MSE are estimated. Using the method described in section 4.4, two tables are generated from a large training data set for estimating the number of bits required to code the AC coefficients and its associated reconstruction distortion. For both the tables, instead of using SAD, the absolute distortion, A_{BLK} of each 8x8 block (equation 5.7) is used as the entries to the tables. The bits for coding the DC coefficient is calculated separately as follows.

Computation of DC bits:

$$\text{BLK\_mean} = \left(\sum_{i=1,j=1}^{8,8} residual \right) / (8x8) \tag{5.6}$$

$$\text{Absolute distortion, } A_{BLK} = \sum_{i=1,j=1}^{8,8} |residual - BLK\_mean| \tag{5.7}$$

$$\text{LEVEL} = [abs(\text{BLK\_mean}) - (\text{QP}/2)]/(2*\text{QP}) \tag{5.8}$$

where *abs(x)* means taking the absolute value of x.

The DC bit for INTER modes can be obtained from the following 2 tables [60]:

Table 5.6a : if AC bit greater than 0.

LEVEL	1	2	3	4	5	6	7	8	9	10	11	12	>12
BIT	3	5	7	8	9	10	10	11	11	12	12	12	22

Table 5.6b : if AC bit is zero.

LEVEL	1	2	3	>3
BIT	5	10	12	22

If INTRA mode is chosen, 8 bits is used for coding the DC coefficient of each block.

Unlike section 4.4, the estimated AC bits and DC bits are computed separately because this leads to a more accurate bit-rate prediction. A few examples of the tables are depicted in Fig. 5.20. The solid curves are the average number of bits spent on AC coefficients without taking into consideration the probability of the prediction error being coded whereas those dotted curves are adjusted according to the probability by multiplying with their respective probability curves in Fig. 5.21. As discussed in section 4.4, the probability factor is important especially for the larger QP when the relative frequency of each distortion is considered. Fig. 5.22 shows that most of the distortion for the larger QP falls within the portion of the dotted AC coded-bit curves where they will not be coded.

Using straight-lines approximation, these AC coded-bit curves and reconstruction distortion curves are approximated with 2-3 straight lines. Some examples are shown in Fig. 5.23 and Fig. 5.24 for the AC coded-bit curves and reconstruction MSE curves respectively. The advantage of using these straight-line approximated functions is that they require only a relatively small additional memory compared to the tables which consist of 100 entries for each of the 31 quantisers.

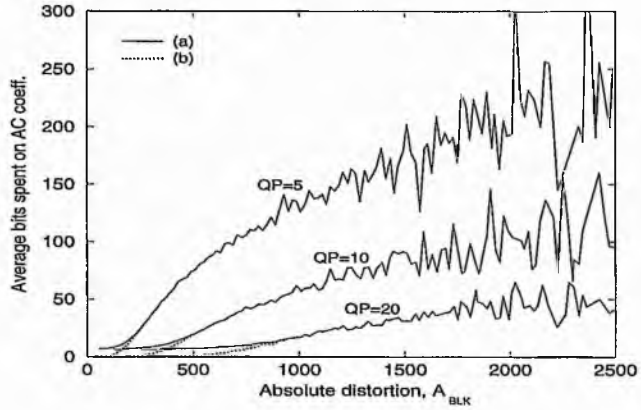


Fig. 5.20 : Average number of bits spent on AC coefficients as a function of prediction error per 8x8 block.

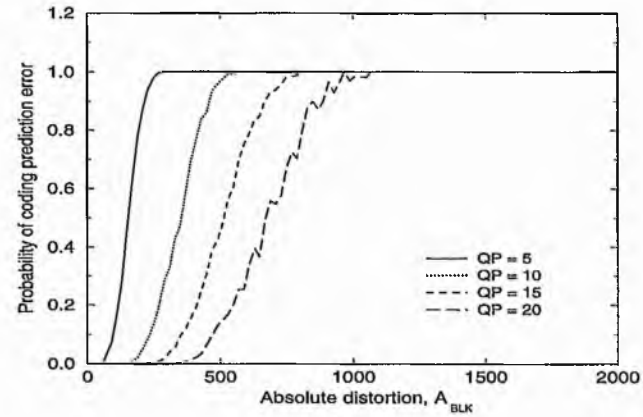


Fig. 5.21 : Probability of coding the prediction error of an 8x8 block.

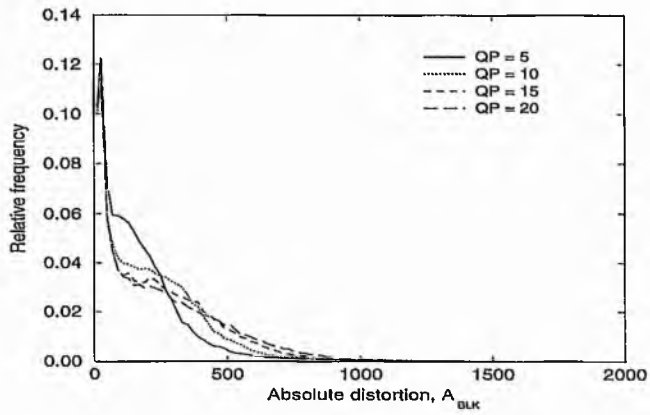


Fig. 5.22 : Relative frequency of occurrence of absolute prediction error.

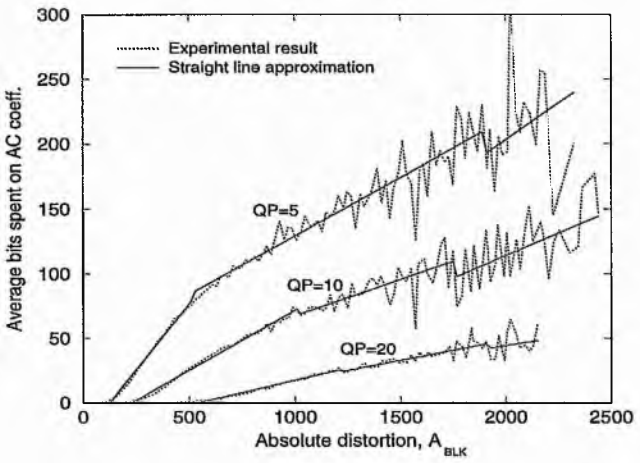


Fig. 5.23 : Modeling AC coded-bit curves using straight-line functions.

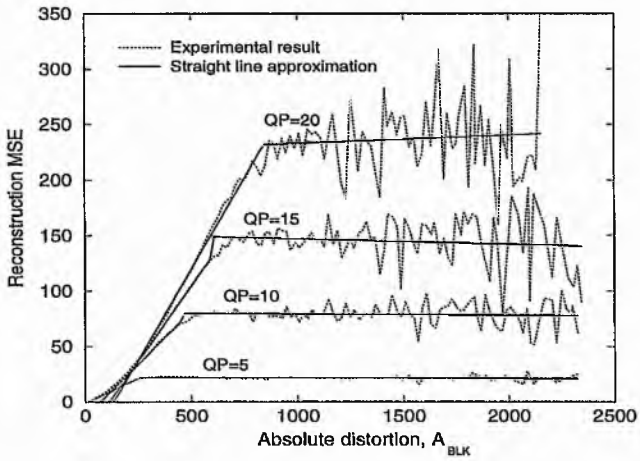


Fig. 5.24 : Modeling reconstruction MSE curves using straight-line functions.

5.8.2 Simulation Results and Discussion

The dummy encoding and decoding stages in the R-D optimised mode selection algorithm for estimating the prediction error rate and reconstruction MSE of each operating modes is replaced by the above simplified method. The experiments in section 5.5.2 were re-ran. Similar results but with some loss in rate-distortion performance were observed as shown in Fig. 5.25 and Fig. 5.26. The relative frequency of selecting each mode is also very similar to Fig. 5.18 and Fig. 5.19.

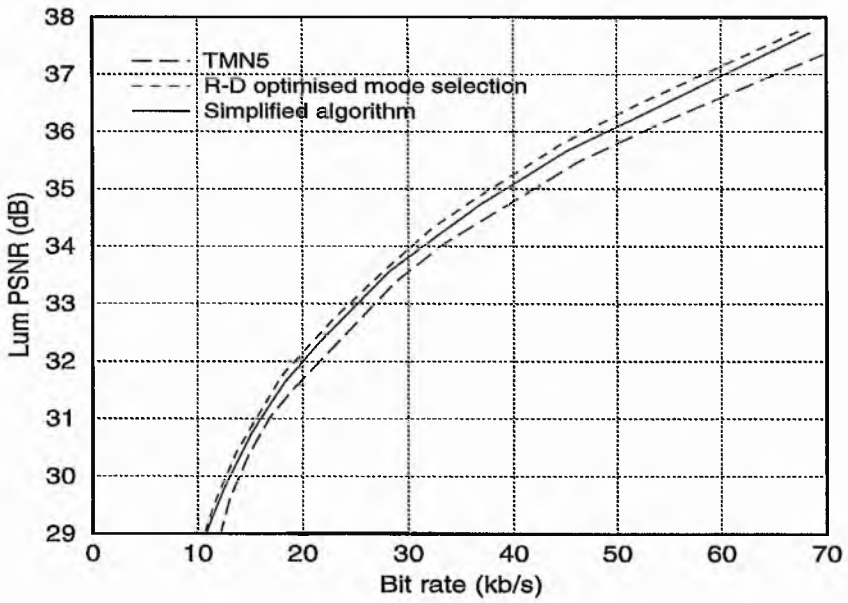


Fig. 5.25 : Comparison in coding performance between the proposed mode selection strategy and TMN5 for the first 150 frames of Carphone sequence.

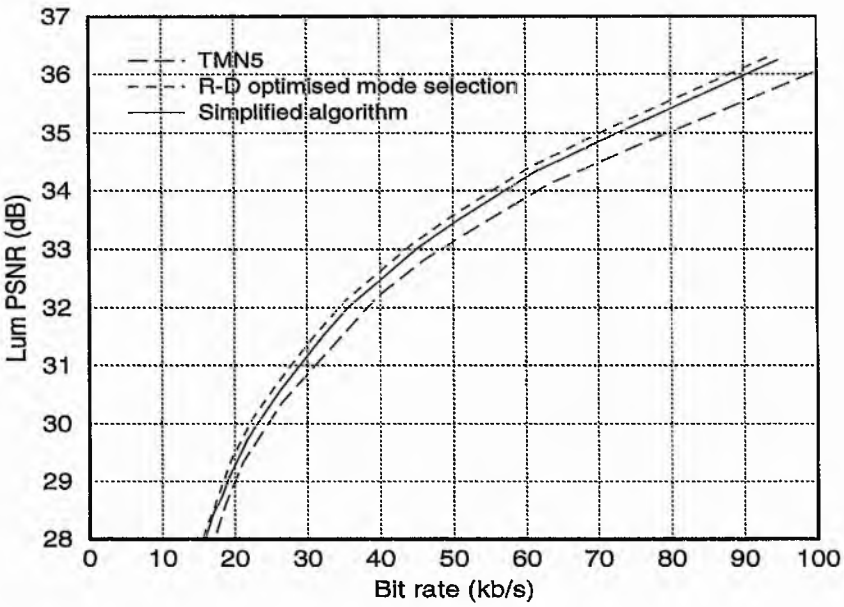


Fig. 5.26 : Comparison in coding performance between the proposed mode selection strategy and TMN5 for the first 150 frames of Forman sequence.

5.9 R-D Optimised Coder

Since both the motion vector and coding mode are selected under the same framework, it is natural that they are jointly selected. In TMN5, the advanced search for the 4 8x8 motion vectors is carried out after the 16x16 motion vector search with a window centered around the integer 16x16 vector. In order to reduce the amount of computational load, the advanced 4 motion vectors search is performed only for the best 16x16 motion vector with a search window of 5 half pixel around it. For both cases, only the best N candidates in turn of SAD are selected, resulting a total of $2N$ candidates. Together with the UNCODED mode, a total of $2N+1$ candidates are submitted to the rate-constrained selection algorithm.

Simulations were performed using the same coding conditions as in section 5.5.2 with N=10. The rate-distortion performance of the R-D optimised coder and its simplified version using the coded-bits estimation method of section 5.8 are plotted in Fig. 5.27 and Fig. 5.28 for the sequences Carphone and Foreman respectively. As can be seen

from both figures, only a small improvement of about 0.1 dB is observed when the rate-constrained motion vector selection algorithm is incorporated into the R-D optimised mode switching algorithm. The reason is that the performance of the coder has been significantly improved by the appropriate selection of the coding mode for each macroblock and the more accurate motion prediction offered by the INTER4MV mode. These two factors have reduced the usefulness of the rate-constrained motion vector selection algorithm. This seems to suggests that improved performance of a multimode block-based video coder can be achieved with an efficient operating mode switching strategy, without resorting to the motion estimation stage. However, this is only true because H.263 adopted a two level hierarchical motion estimation scheme. For other hybrid standards such as MPEG1 and H.261 where only one level of motion estimation is employed and a large block size is used, the rate-constrained motion compensation algorithm may still prove to be useful in enhancing the coding efficiency.

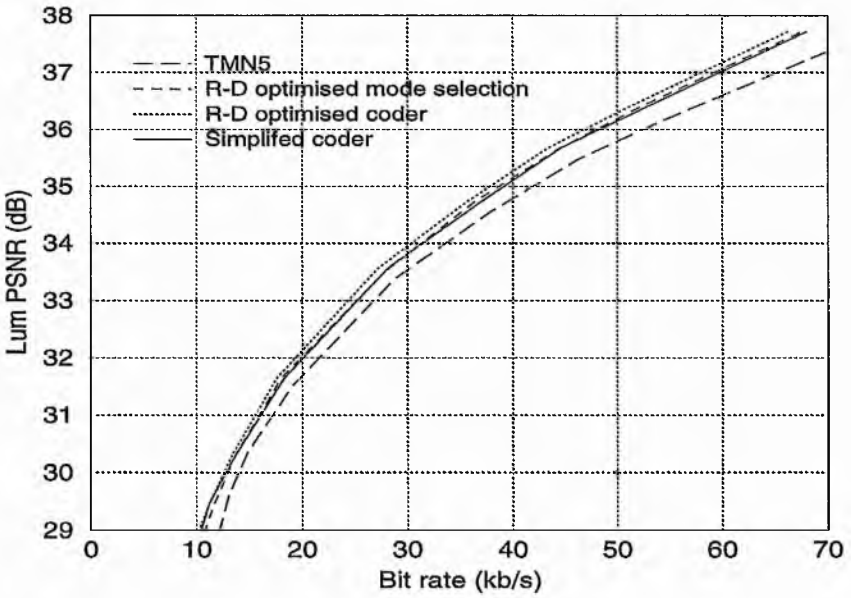


Fig. 5.27 : Rate-distortion performance of the R-D optimised coder and its simplified version for the first 150 frames of Carphone sequence, coded at 8.33 fps.

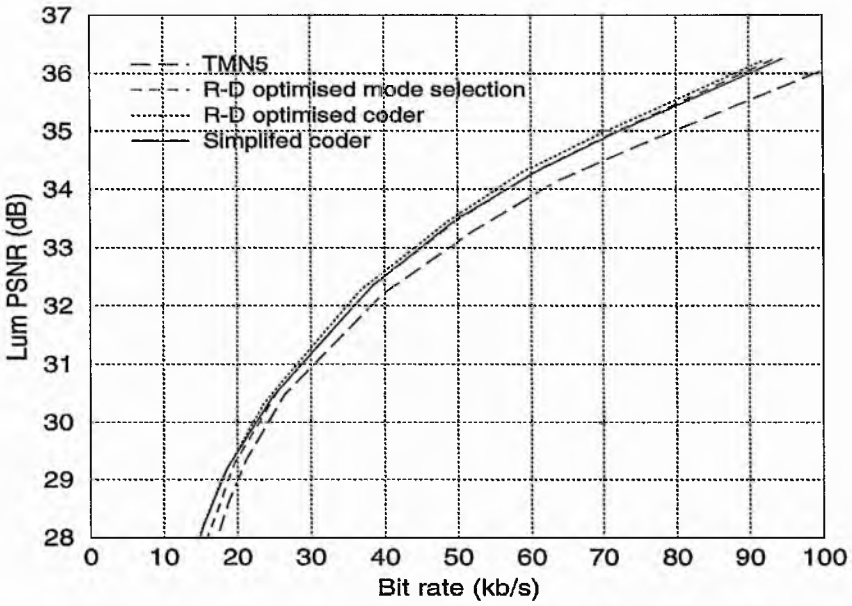


Fig. 5.28 : Rate-distortion performance of the R-D optimised coder and its simplified version for the first 150 frames of Foreman sequence, coded at 8.33 fps.

5.10 Concluding Remarks

A rate-constrained motion estimation algorithm that gives an improvement over the conventional minimum distortion scheme has been developed. In contrast to the classical approach of selecting the motion vector that gives the minimum distortion, the proposed algorithm takes into account both the bit-rate and the resulting distortion during the motion vector selection process. The algorithm which employed an analysis-by-synthesis technique was incorporated into a hybrid motion-compensated transform coding scheme and proved to be capable of improving the coding efficiency.

The reduction in bit-rate depends very much on individual sequence. In general, a temporally active sequence containing a great deal of movement is able to achieve more significant reduction than a quiet sequence. Moreover, the number of candidates used in the analysis-by-synthesis stage and the preset threshold will also affect the

results. A reduction in bit-rate in the range of 3% to 10% is achievable with negligible degradation in objective and subjective image quality when the proposed algorithm is used. On the other hand, the complexity of the system has also been increased, as a function of the number of candidates, due to the use of the analysis-by-synthesis technique for choosing the best motion vectors. But the increase in complexity was found to be acceptable if the number of candidates was kept at a small value.

In the second part of the chapter, a rate-distortion optimised mode switching strategy for a multimode block-based coder is presented. In general, at least 3 basic coding modes are available in the current video coding standards, namely intra mode, uncoded mode and inter mode. For H.263, an additional mode called advanced 4 motion vectors mode is also available. In the proposed selection strategy, these modes are treated as motion vectors and the above rate-constrained selection algorithm is employed for their selection. Simulation results showed an improvement in the rate-distortion performance. A simplified algorithm based on coded-bit and distortion estimation was also implemented. As expected, a reduction in performance is traded off for a substantial reduction in the computational complexity. Finally, the rate-constrained motion estimation algorithm is incorporated into the mode switching algorithm to form a R-D optimised coder. However, here only a small improvement was observed. The reason is that the performance of the coder has been significantly improved by the appropriate selection of the coding mode for each macroblock and the more accurate motion prediction offered by the INTER4MV mode. These two factors have reduced the usefulness of the rate-constrained motion vector selection algorithm.

Chapter 6

Subjective Quality Improvement Using Reduced-Resolution Coding

6.1 Introduction

The regulation of output data rate is crucial for the video encoding process because it directly affects the amount of bits being generated. This in turn affects video image delay, frame rate stability and image quality. As discussed in Chapter 4, frame skipping and reducing the accuracy of the quantisation process are the two most commonly used methods for reducing the data rate of a block-transform video coder, although other alternatives are also available. They are adequate under normal circumstances for typical teleconference and videophone images. However, sudden and unpredictable large motion often appears in real-time video communication. Under such situation, if too many bits are assigned to a few frames, extreme frame dropping or extreme degradation of image quality for the following frames will result if the channel bandwidth is limited. For other situations such as transmission of video over error-prone mobile link where bandwidth is severely restricted and more bits are required for error control, increasing the quantisation step size and skipping more frames might not be the best solutions.

Firstly, it is well known that hybrid predictive-transform coding technique on which H.263 standard is based suffers from blocking artifacts caused by the truncation of

high frequency DCT coefficients and subsequent coarse quantisation of the remaining coefficients when large quantisation step size is used for achieving the target bit-rates. As a result, discontinuities become visible along block edges [1][38].

Secondly, besides giving perceptually unpleasant jerking images, skipping several frames to allocate more bits for each coded frame might reduce the correlation between successive coded frames if scenes are rapidly changing. As a consequence, this may lead to the failure of the motion compensation stage and subsequently more bits will be spent on intra-coding the failed regions. And for certain applications such as communication through the use of sign language, skipping of several frames might lead to a break down in communication, as every movement and sign of the hands contributes to the understanding of the language.

For an already low coding frame-rate sequence, instead of using a large quantiser for coding the residual image, down-sampling the latter to allow the use of a smaller quantiser might give a subjectively better decoded image. The down-sampled residual image will have a smaller number of coefficients. Hence, finer quantisation can be applied to them under the same bit-rate budget constraint.

In this chapter, we will be investigating the advantages of using this approach for coding residual image under very tight bit-rate budget. First, its application to adaptive source/channel coding under very poor channel condition will be examined. Then, its use for coding highly active scene will be studied. In both cases, we are more interested in the subjective quality instead of objective quality of the decoded pictures. Similar work has been carried out in [60] and has been proposed to ITU for consideration as an extension to H.263.

6.2 Transmission Error

For efficient compression, most coding schemes always include an entropy coding operation as the final stage of processing. The entropy coding can be either variable VLC or arithmetic coding which takes into account the different probabilities of occurrence of various data output levels, lengths of symbols, etc. The output bit-stream thus consists of codewords which can only be correctly decoded at the decoder if they correspond with members of the predetermined code table.

In such a situation, transmission errors will have a potentially catastrophic effect upon the reconstructed image. Much work has been expended over the past years in the development of algorithms which attempt to minimise this problem. The situation is complicated by the fact that not all data may have the same degree of significance. For example, coding/decoding control information is most vital and will need to be heavily protected. Motion vectors are the next category, especially when they are differentially coded, where a single error can propagate rapidly to the rest of the image. Then, in transform schemes, some spectral coefficients are more important than other - generally the loss or corruption of higher order terms matters less than that of DC/low frequency information [61].

The use of conventional forward error control for error protection can lead to a very significant increase in the total number of bits transmitted. Moreover, the design of such error protection scheme needs to consider the worst case Bit Error Rate (BER) and also maximum burst error length. It is possible to allocate bits dynamically between source and channel coding according to the channel error condition [61]. For fixed-rate transmission, it is often the case that good error protection of a basically low quality coded image (few bits available for source coding) will give better results than those obtained by using more bits for a higher quality image and thus reducing the number of bits available for error protection. Such source/channel coding schemes have achieved some popularity though other error resilient schemes are also available and have been intensively investigated in some of the core experiments for MPEG4 [64].

In general, source/channel coding is an effective way of maintaining the performance of a predictive scheme in the presence of high channel error rates by preferentially allocating the spare bandwidth obtained from reducing quantisation accuracy to the provision of error protection.

6.2.1 Channel Protection Techniques

To mitigate the effect of quality degradation encountered in error prone environments, Forward Error Correction (FEC) techniques are widely used for detecting and correcting errors at the expense of more overhead [62]. The main problem is that coded video is a variable rate traffic so output video parameters cannot be presented to the channel encoder as fixed length symbols. On the other hand, rate-compatible punctures convolutional (RCPC) codes have been used in many video services to provide a multi-rate channel error control [63]. The principle behind these codes is that the same convolutional code can be used to give different levels of error protection by simply removing certain bits.

Obviously, for optimum use of the available bandwidth, the power of the channel coder has to be made adaptive to the channel condition. With convolutional code, this can be easily achieved by adjusting the redundancy bits according to the status of the channel. As a result, a very fast back channel signalling scheme which keeps the encoder updated of the latest variations in the network conditions must be employed in order for this adaptive channel coding strategy to work.

For efficient use of the available bandwidth, another FEC technique namely Unequal Error Protection (UEP), is sometimes applied for robust video communications. UEP consists of assigning a variable level of protection to video parameters based on their sensitivity [61] as mentioned in the previous section. In general, FEC techniques can either be used alone or combined with other error resilience techniques [61]. These techniques, when applied, introduce redundant information that acts against the compression efficiency of the video signal. They are more effective for channels with predictable BER and limited burst length unless multi-rate functionality such as that

offered by RCPC is incorporated. However, they fail with high BER channels or long bursts while giving a disastrous effect on the occupied bandwidth due to the large amount of overhead they imposed on the encoded signal.

6.2.2 Rate-Compatible Punctures Convolutional Code

The RCPC code is capable of providing adaptive error protection for channel with time varying condition. The channel encoder starts off by sending the mother code only, i.e. with no error protection bits. If the transmitted data is corrupted by errors in a way that it cannot be interpreted by the channel decoder, the encoder is informed through a back channel signalling technique. Therefore, the channel encoder increases its rate and sends the first level of protection bits. Take a 4-register convolutional code for example, 4 rates or 3 levels of error protection are available for the channel encoder. The channel encoder starts with the rate set to 1 (in this case no error correcting code is transmitted) and steps up its rate when requested by the channel decoder which is usually a viterbi decoder. For degraded channel conditions, the channel coder should provide more powerful protection to the output symbols. Hence more bits are allocated to the output symbols in order to enable the channel decoder to correct a higher number of errors. Depending on the back channel reports, the channel coder increments its rate of transmission. The rate keeps on increasing from one level to another until the decoder is able to reconstruct all the contained data without any detected error.

Therefore, the rate of the convolutional coder is made variable depending on the decoder's ability to correct the corrupted bits. The higher the requested rate, the more redundancy is added into the transmitted bit-stream in order to increase the error correcting power of the channel decoder. This multi-rate error control code is derived from a single code by removing particular bits from its output. At the decoder, these removed bits are inserted back before error detection begins. This adaptive multi-rate error protection scheme using RCPC has been used in conjunction with backward channel signaling in some MPEG4 video resilience experiments [64].

6.2.3 Layered Coding

In digital video compression, scalability usually denotes the feature that there is a scale or hierarchy of quality or spatio-temporal resolution. The methods for scalable coding employed in MPEG2 are divided in two main categories : a first category for coding a hierarchy of coding noise and a second one for coding a hierarchy of spatio-temporal resolution [65]. One of the applications is for graceful degradation in channel with a transmission quality that fluctuates in time. As illustrated in Fig. 6.1, scalability provides a way of gradual or graceful deterioration of the received video quality. The corruption by transmission errors of a non-scalable bit-stream is more abrupt than in the case of layered bit-streams. As errors in enhancement layer bit-stream are less visible, and with an unequal error protection, reception of the base layer is guaranteed up to a more severe level of transmission distortion.

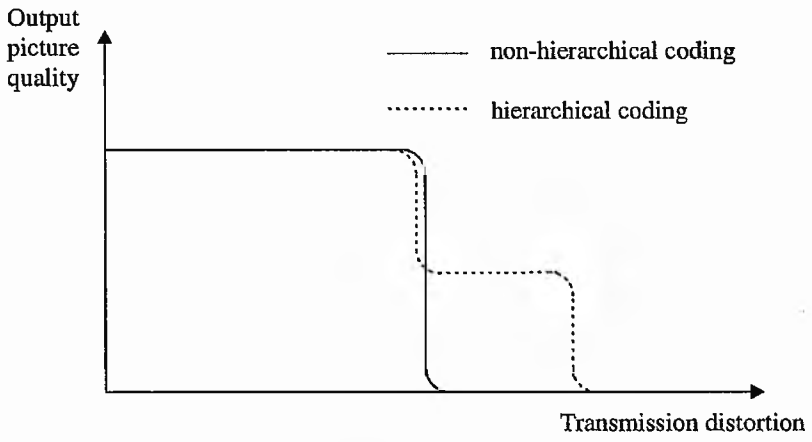


Fig. 6.1 : Picture quality as a function of transmission conditions.

On the other hand, an adaptive two-layer coding technique has been proposed and applied to H.261 for transmission over ATM networks [66]. The base layer consists of a low quality image generated at a very low bit-rate while the second layer is the difference between the input picture and the output of the base layer. In order to ensure graceful degradation, the data from the base layer is transmitted with high priority using the guaranteed bandwidth of an ATM network which makes them

immune to packet loss. Packets from the second layer are transmitted through the channel that is not guaranteed and may be lost if congestion arises. If these packets are received, the decoded picture will be improved [66].

In low bit-rate video coding over fixed-rate channel, the bandwidth is usually very limited. Hence layered coding such as the above two examples is usually not employed as it increases the number of overhead bits. However, the above two examples highlight the advantage of having a well-protected or a guaranteed low quality image.

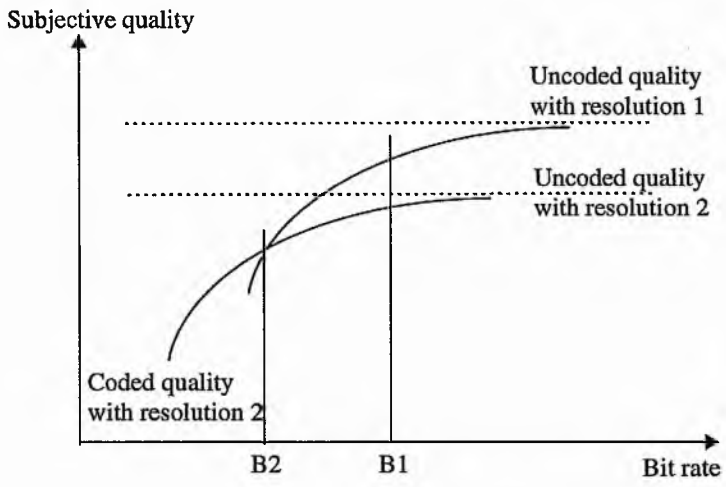


Fig. 6.2 : Relation between subjective picture quality and bit-rate for different picture formats.

6.3 Subjective Image Quality

Fig. 6.2 depicted the relation between the subjective picture quality and bit-rate for different picture formats [35]. The horizontal dotted lines define the subjective picture quality of the original picture for different format. The corresponding curves show the subjective quality of the decoded picture as a function of bit-rate after coding. As can be seen from the figure, if bit-rate B1 is available, coding using resolution 1 gives better subjective quality than using resolution 2, although much smaller coding

degradation is achieved with resolution 2. However, we were more interested in the region where the two curves intercepted. It can be easily seen that at bit-rate lower than B2, the subjective quality of using resolution 1 for coding is poorer than that of resolution 2 due to the large quantisation needed for achieving the target bit-rate. This indicates that for optimum subjective picture quality, choice of coding resolution should be part of the compression method.

6.3.1 Jerkiness

Jerking is easily noticeable at frame rate of 7 Hz or lower. Under this low coding frame-rate, if an object is moved at a constant speed across the camera, the decoded image will be detected as "jumped" and does not copy the movement smoothly. Jerking is usually acceptable when it is stable and constant but becomes annoying when provoked by an unstable degradation of the frame rate as illustrated in Fig 6.3. For the latter case, the same smooth movement is reconstructed with some frames at high frame-rate, followed by some frames with a larger delay between each other, then again with a series of fast frames, and so on. The movement becomes unnatural and unpleasant to view [38].

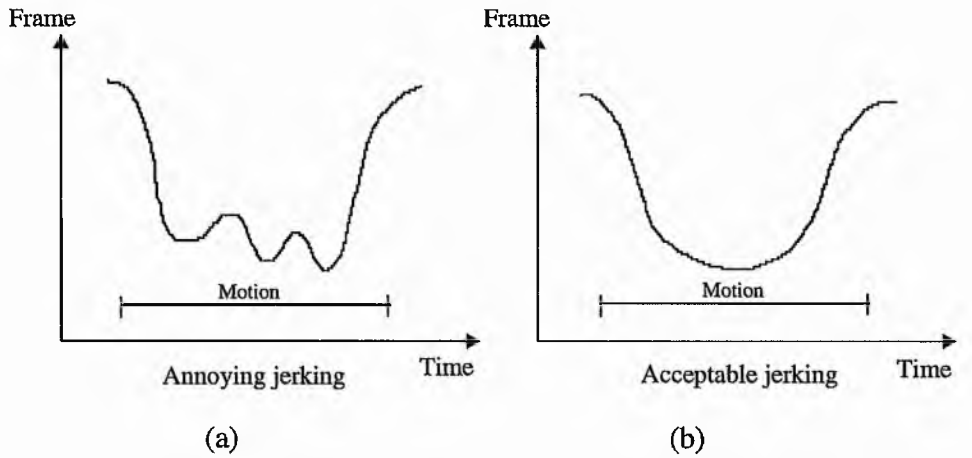


Fig. 6.3 : (a) Annoying image jerking results from repeated abrupt variations of frame-rate, usually caused by scene motion; (b) a smooth and uniform decay frame-rate is always preferable.

6.4 Reduced-Resolution Coding of Prediction Error (RRC mode)

In Section 5.7, the use of multi-mode methodology in current video coding standards to cater for the different types of scene statistics has been examined. Results shown that more efficient coding can be achieved if the modes are applied appropriately to different regions of an image. Inter-coding mode is chosen at the assumption that the resulting prediction error will be encoded and transmitted to the decoder. This coding method is effective when relatively small quantiser is used. However, when the quantisation of the prediction error is very coarse, most of the prediction error will not be coded and quantisation noise will become visible in the decoded images.

In the following sections, the coding efficiency and advantages of the reduced-resolution coding method (RRC mode) will be investigated for some of its target applications. In general, RRC mode is expected to be used when the bit-rate budget is extremely low resulting in very coarse quantisation being carried out on the transformed coefficients. Due to the down-sampling process before it is encoded and subsequent up-sampling process at the decoder, the reconstructed images are expected to be slightly blurred. However, this does not imply that the subjective quality will be degraded. In contrast, the opposite is anticipated due to the smaller quantisation made possible by the reduced number of transform coefficients needed to be coded.

6.4.1 Coding Structure

The coding structure of the RRC mode is almost the same as the original H.263 coder except that whenever the RRC mode is used, the prediction error is down-sampled before it is DCT transformed. As in the original H.263, motion-compensation is carried out at full resolution of the image. After motion compensation, the resolution of each macroblock of prediction error is reduced by half for each dimension, producing an 8x8 luminance block and two 4x4 chrominance blocks. The rest of the coding operation is as usual as H.263 except that the chrominance blocks are

transformed by a 4x4 DCT and its resultant coefficients are scanned in a similar zigzag order. The result will not be significantly affected if the chrominance blocks are not down-sampled when the added complexity due to the use of a 4x4 DCT is not desirable.

At the decoder, the reconstructed 8x8 luminance block is up-sampled to give the 16x16 MB and added to the motion compensated prediction MB to give the final 16x16 reconstructed MB, as illustrated in Fig. 6.4. Similar processing is performed for the chrominance blocks.

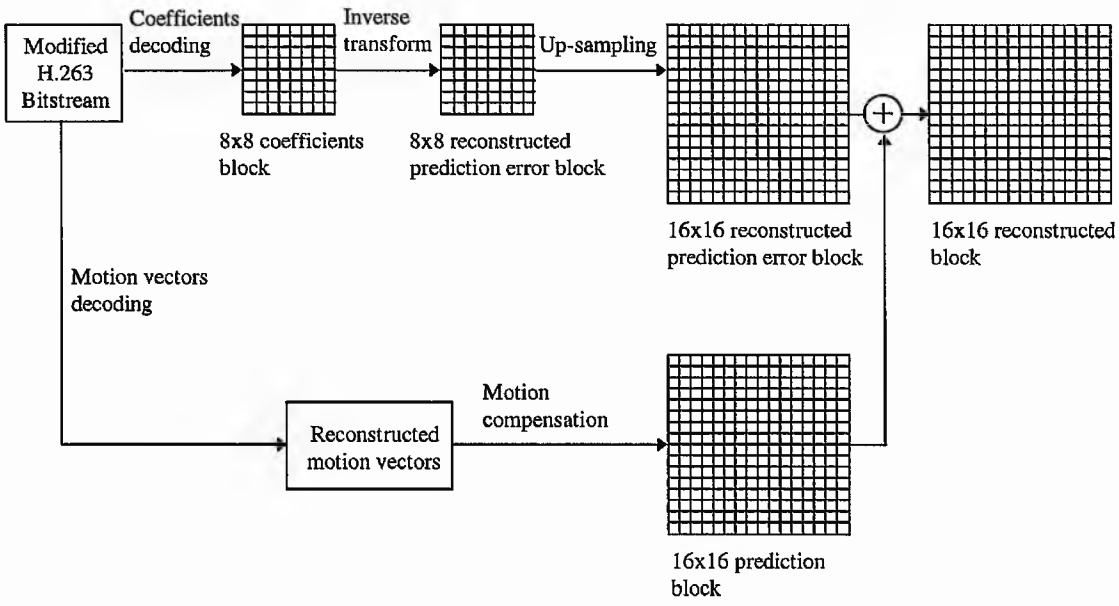


Fig. 6.4 : Decoding process of a macroblock.

6.4.2 Down-Sampling of Prediction Error

After motion estimation, the prediction error image is reduced by half in each dimension. In order to make it compatible to H.263 and realise a simple implementation, the down-sampling and corresponding up-sampling process are carried out on macroblock basis. Therefore, the four luminance blocks are reduced to a single 8x8 block while the 8x8 chrominance blocks are down-sampled to 4x4 blocks.

There are a number of methods for reducing the resolution of images. Linear interpolation filter is used in the experiment due to its simplicity. Fig. 6.5 shows the down-sampling filter which is similar to the one used in H.263 for interpolation of the reconstructed image for half-pixel motion vector search. The down-sampled luminance block is transformed by a 8x8 DCT while the 4x4 chrominance blocks are transformed by a 4x4 DCT. Their transformed coefficients are quantised, zigzag scanned and run-length coded as in the original H.263 except that the order of the zigzag scan is slightly modified to adapt to a 4x4 block of coefficients for the chrominance blocks.

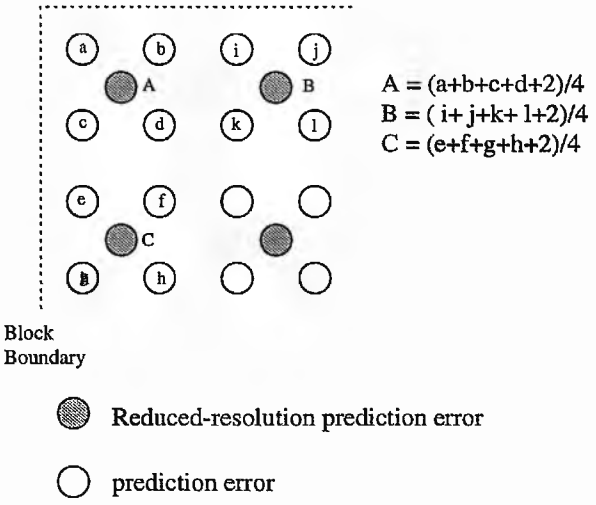


Fig. 6.5: Down-sampling of prediction error.

6.4.3 Up-Sampling of Prediction Error

At the decoder, the transformed coefficients are recovered and inverse-transformed to reconstruct the prediction error. The bilinear filter for up-sampling the 8x8 prediction error block to the 16x16 macroblock is shown in Fig. 6.6 and Fig. 6.7 [60]. The up-sampled macroblock is added to the motion-compensated prediction blocks to obtain the final reconstruction macroblock. As usual, similar processing is carried for the chrominance blocks.

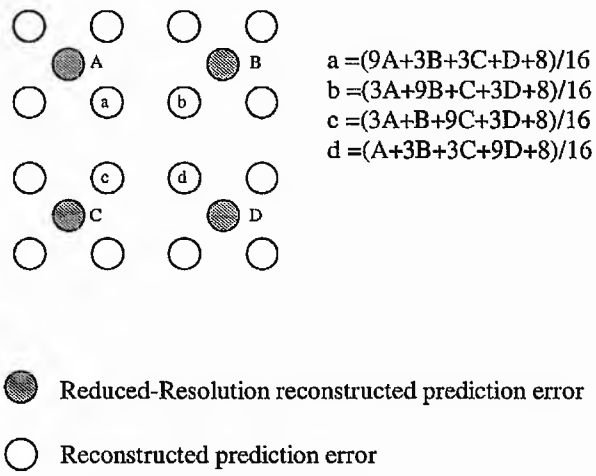


Fig. 6.6 : Up-sampling of reconstructed prediction error inside a block.

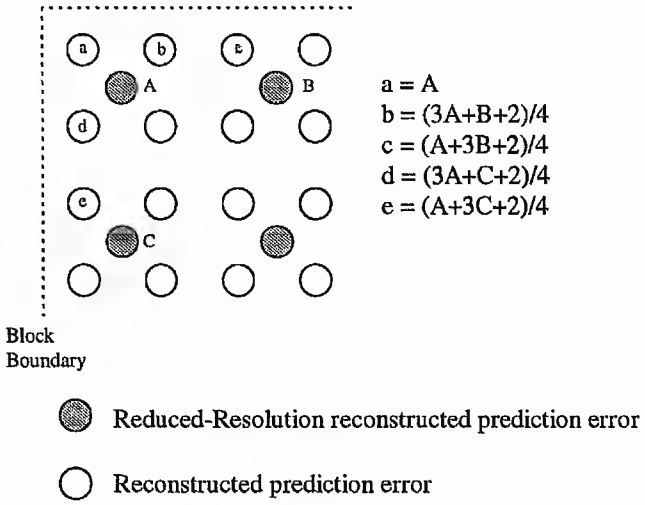


Fig. 6.7 : Up-sampling of reconstructed prediction error at block boundary.

6.4.4 Bit-Stream Syntax

A small modification to the encoder output bit-stream syntax is required so that the compressed bit-stream can be decoded at the receiver. As the RRC mode is designated to be used under very tight bit-rate budget circumstances, unnecessary increase in overhead bits is undesirable. As mentioned in the above section, the RRC mode is designed to operate on a macroblock structure. However, if the decision to use it is based on a macroblock-by-macroblock basis, then an extra bit is required for every coded macroblock to indicate its use. This gives a maximum of 99 bits if QCIF picture format is used which is a relatively large value at very low bit rate (approximately 5% ~ 10% of the total bit-rate per frame in some cases). Taking this into consideration, the down-sampling mode is decided on a frame by frame basis and only 1 extra bit is required to be included in the picture header to signal its use.

6.5 Adaptive Source/Channel Coding for Fixed-Rate Transmission

In this section, the use of the RRC mode for adaptive source/channel coding will be investigated. For fixed-rate transmission, it is widely accepted that good error protection at the expense of the image quality is a better scheme than its opposite counterpart. If this coding strategy is adopted, when the channel condition is very poor, more bits will be allocated to error control by reducing the quantisation accuracy of the prediction error. Under extreme circumstances, the quantisation step-size used will be very large that most of the prediction error will not be transmitted. As a result, quantisation noise becomes visible in the decoded images. By using RRC mode less bits will be required for coding the prediction error, thus a smaller quantisation step-size can be used for the same bit-rate budget.

In order to realise adaptive source/channel coding, a multi-rate puncture convolution channel coder is implemented for error protection. The channel coder has a variable rate of 1, ¾ , $^2/_3$, and ½.

6.5.1 Simulation Result And Discussion

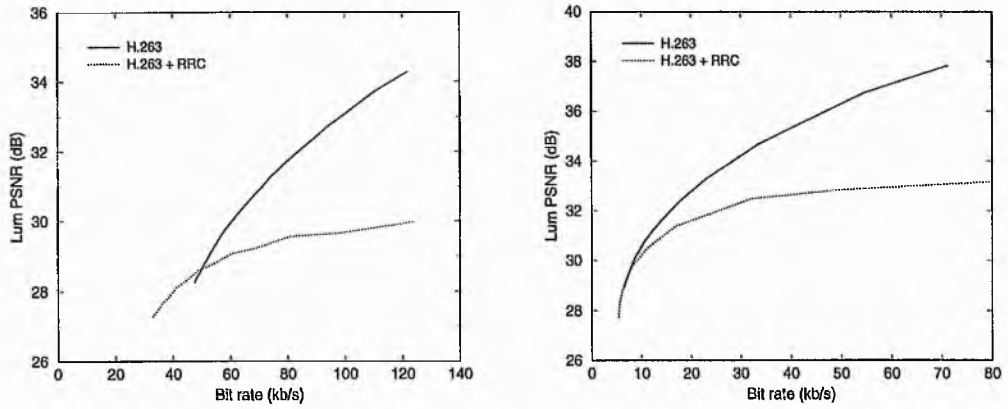
In order to highlight the efficiency of RRC mode and for easy evaluation, the RRC mode is always turned on on the assumption that the channel error condition is very hostile and ½ rate error protection is employed in the RCPC channel coder.

Simulations were carried out on several standard video sequences and the results were presented in Table 6.1. Fixed quantiser was used throughout the encoding process and it had been chosen for each sequence such that the average resultant bit-rate was close to the target rate. The resultant bit-rate in Table 6.1 includes the bits spent on channel coding and in this case it is half of the stated value.

Table 6.1 : Results of using RRC mode for coding the prediction error under very bad channel condition when the quantisation step size is very large.

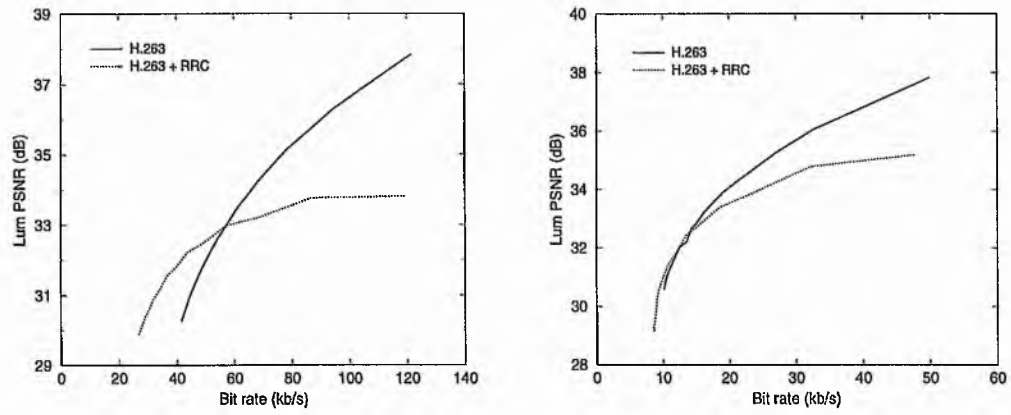
Sequence	RRC mode		H.263	
	PSNR	Bit Rate (QP)	PSNR	Bit Rate(QP)
Miss America (0-149, 10 fps)	33.36	20.34 (10)	32.90	22.33 (30)
Claire (0-149, 10 fps)	30.10	24.07 (12)	30.80	24.13 (26)
Grandmother (0-149, 10 fps)	29.17	20.53 (10)	29.12	20.64 (26)
Suzie (0-149, 6.25 fps)	30.98	26.97 (10)	29.99	27.54 (28)
Carphone (0-149, 10 fps)	28.22	40.03 (10)	28.93	39.51 (25)
Foreman (0-149, 6.25 fps)	26.40	39.90 (13)	26.81	41.34 (30)

The results in Table 6.1 show that using the RRC mode does not always give better PSNR than H.263. Fig. 6.8 and 6.9 are the rate-distortion plots of the sequences Grandma and Suzie respectively. The plots show that at very low bit-rate i.e. when the quantiser is very large, using the RRC mode tends to give a better PSNR result than the original H.263. As expected, this advantage of RRC mode is very obvious for intra-frame coding than inter-frame coding. On the other hand, subjective quality assessment shows that all the decoded sequences are significantly better for the RRC mode than H.263. The decoded pictures from the RRC mode are smoother, containing less quantisation noise although they are not as sharp as that of H.263, whose decoded pictures contain a lot of blocking artifacts which is very unpleasant to view.



(a) (b)

Fig. 6.8 : Rate-distortion curve for the first 150 frame of the sequence Grandma coded at 10 fps; (a) all frame are intra coded; (b) except the first fame, all the other frame are inter coded.



(a) (b)

Fig. 6.9 : Rate-distortion curve for the first 150 frame of the sequence Suzie coded at 10 fps; (a) all frame are intra coded; (b) except the first fame, all the other frame are inter coded.

6.6 Adaptive RRC Mode Selection

The Reduced-Resolution Coding mode is expected to be used when encoding a highly active scene. It allows the encoder to send update information for a residual picture that is encoded at a reduced resolution, while preserving the detail in a higher resolution reference image to create a final image at the higher resolution. This avoids overspending of bits on a few frames, resulting in an increase in the time delay. It also provides the opportunity for maintaining the coding frame-rate while maintaining sufficient subjective quality. Besides, this capability might be useful in certain circumstances such as communication through sign language for disabled people.

6.6.1 Rate Control

The rate controller in section 4.5 will be used in the experiments. The controller tries to assign the same amounts of bits to each frame through the use of a suitable quantiser so that a target bit-rate can be achieved. The quantiser is selected with the aid of a bit-rate prediction algorithm. Since the results in section 4.5.1 show that an almost constant bit-rate can be achieved using this controller, a constant coding frame-rate is expected. Nevertheless, a variable frame-rate controller is also implemented in both the modified and original H.263 to enable comparisons to be carried out.

6.6.2 Frame-Layer Rate Control

At the beginning of the encoding process, the number of bits in the buffer between the encoder output and channel, W is set to zero, $W=0$. The first frame of the video sequence is intra coded using a fixed value of QP (by default QP=15). The other parameters are defined as follows:

Let B' = Number of bits spent on the previous encoded frame,
R = Target bit-rate per second (e.g. 24 kb/s, 48 kb/s),
G = Frame-rate of the original video sequence in frame per second (e.g. 25 fps, 30 fps),
F = Target frame-rate in frames per second (e.g. 7.5 fps, 10 fps),
M = Threshold for frame skipping, $M = R/G$.

After coding each frame, the number of bits in the encoder buffer is $W = W + B'$. The number of frames need to be skipped is calculated as:

```
frame_skip = 0;
While ( W > M )
{
        W = W - R/G;
        frame_skip++;
}
```

The next "*frame_skip*" frames of the original video sequence will be skipped.

The target bit-rate for the next frame (to be coded) is :

$B = R/F + \Delta$ where $\Delta = W/F$

The rate controller will be used to find an appropriate QP for coding the next frame.

6.6.3 Switching Strategy

The use of RRC mode is decided frame by frame in order to reduce the complexity. In general, the subjective impression of decoded video sequence very much depends on the frame-rate and individual image quality. And the image quality is mainly determined by the quantisation step size (QP). Taking these factors into consideration, the degradation of the subjective quality from the coding frame-rate and QP can be estimated. If degradation is detected, the lower resolution coding mode will be switched on as below:

Using the rate controller, the QP for the current frame is estimated (QP_{est}) so that the target bit-rate per frame can be achieved. If QP_{est} is greater than a threshold, QP_{thres}, the RRC mode will be used and a new QP_{est} is chosen for this mode.

```
if ( QP_est > QP_thres )
{
        switch to RRC mode;
        new QP is chosen for RRC mode;
}
```

6.6.4 Simulation Results and Discussion

In the following simulation, the optional advanced mode is turned on. In order to show the effectiveness of this mode, a highly active MPEG4 test sequence Silent Voice is used for the simulation. It contains a woman using sign language to communicate with her audience. The background of the pictures is very detailed and there is very rapid hand movement throughout the whole sequence. Therefore, it will be interesting to see how the RRC method performs compared with the original H.263.

For fair comparisons, the rate-controller in section 4.5 will be used for both the modified and original H.263. This rate controller tries to assign the same amount of bits to each frame by selecting a QP that can achieve the target rate. As a result, fixed coding frame-rate is expected if each frame can be coded with the assigned bit-rate. Nevertheless, for the first part of the simulation, fixed frame-rate will be used. And for the second part, the above variable frame-rate controller will be used.

(a) Fixed Frame-Rate

Fig. 6.10 is a plot of the bit-rate per frame for fixed frame-rate coding. As can be seen, the average bit-rate per frame is almost constant for the modified H.263 compared with that of original H.263. Fig. 6.10 also shows that the RRC mode is almost always used throughout the whole sequence. Fig. 6.11 is the corresponding luminance PSNR values. The objective quality obtained using the original H.263 is almost always higher than that of the modified H.263 when RRC mode is used except at the second half of the sequence when the original H.263 starts to run out of bits due to overspending of the available bit-rate at the beginning of the sequence. Subjective assessment through side-by-side viewing shows better perceptual quality from that of modified H.263. Although the decoded images are slightly blurred due to the up-sampling process, they are smoother and contain less blocking artifacts.

b) Variable Frame-Rate

Fig. 6.12 is the bit-rate per frame plot. Due to the use of the above variable frame-rate controller, both the encoders skipped about 40 frames after intra-coding the first frame before they started to encode the first inter-frame. As a result, a large number of macroblocks could not be predicted from the first reconstructed frame and have to be intra-coded. The use of RRC mode helps to reduce the amount of bit-rate for the modified H.263 whilst the original TMN5 cannot do much to reduce the amount of bit-rate required as its QP is already 31. Moreover, whenever there are overspending of bit-rate, more frames are skipped to empty the output buffer before the next frame is coded. As a result, the decoded sequence becomes more jerky whereas the same frame-rate of the same sequence coded using the modified H.263 can be maintained through the use of RRC mode. The corresponding PSNR plot in Fig. 6.13 shows that whenever RRC mode is switched off, the PSNR immediately returns to the same level, or even higher in some frames, as the original H.263. Subjective tests show the hand movement of the woman is smoother and the picture quality is also better for the modified H.263, but that of original H.263 is jerky and certain hand signs are skipped.

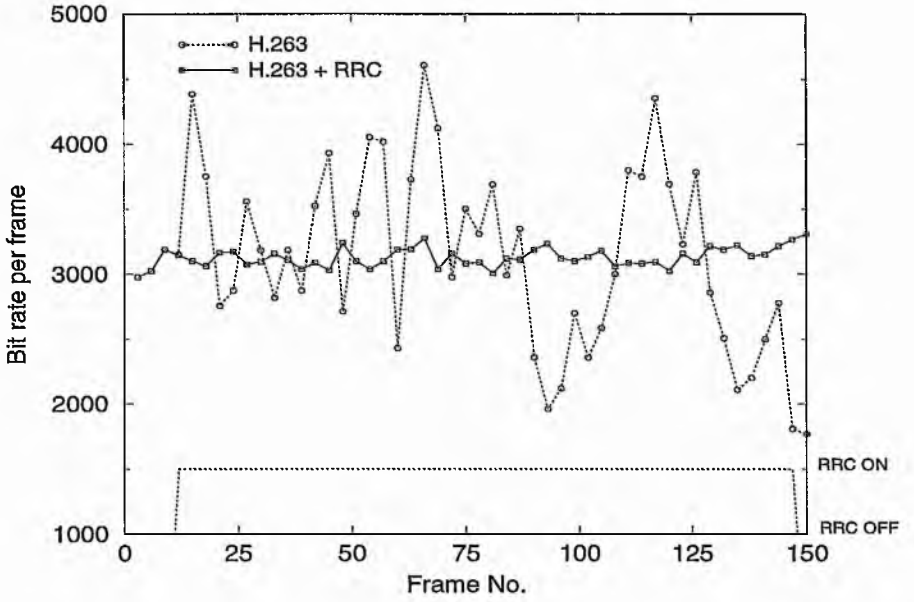


Fig. 6.10 : Bit rate per frame of Silent Voice coded at 40 kb/s using fixed frame rate of 10 fps.

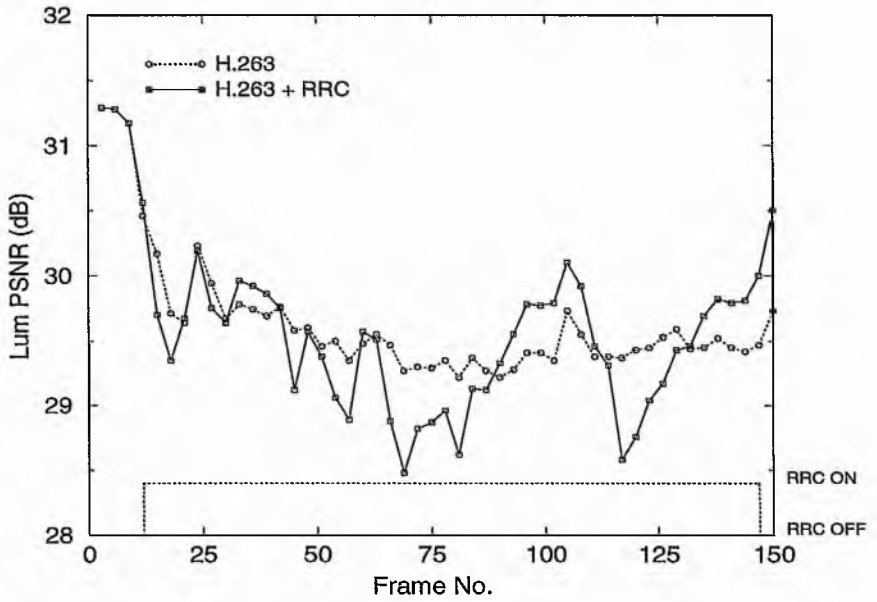


Fig. 6.11 : Luminance PSNR of Silent Voice coded at 40 kb/s using fixed frame rate of 10 fps.

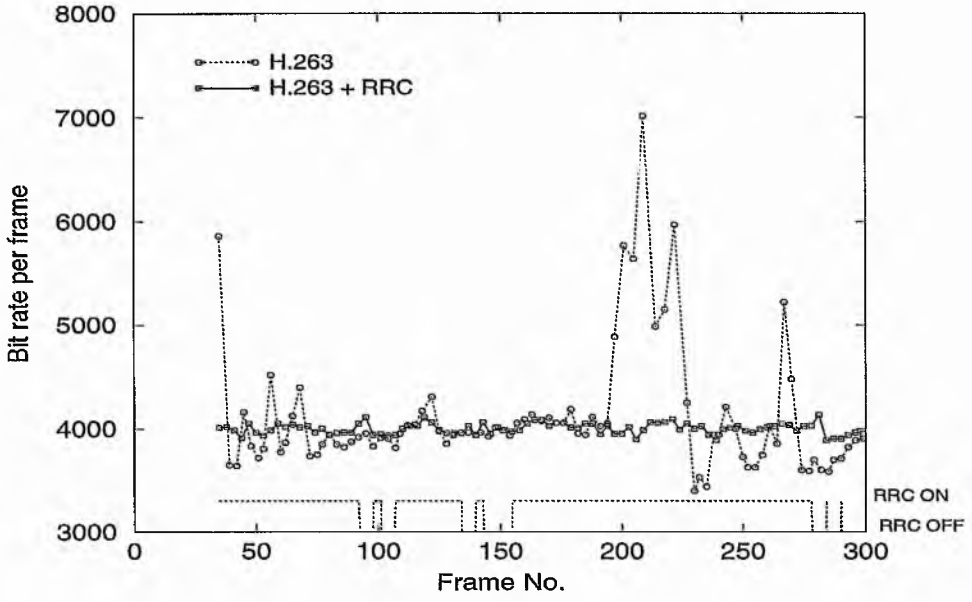


Fig. 6.12 : Bit rate per frame of Silent Voice coded at 40 kb/s using variable frame rate.

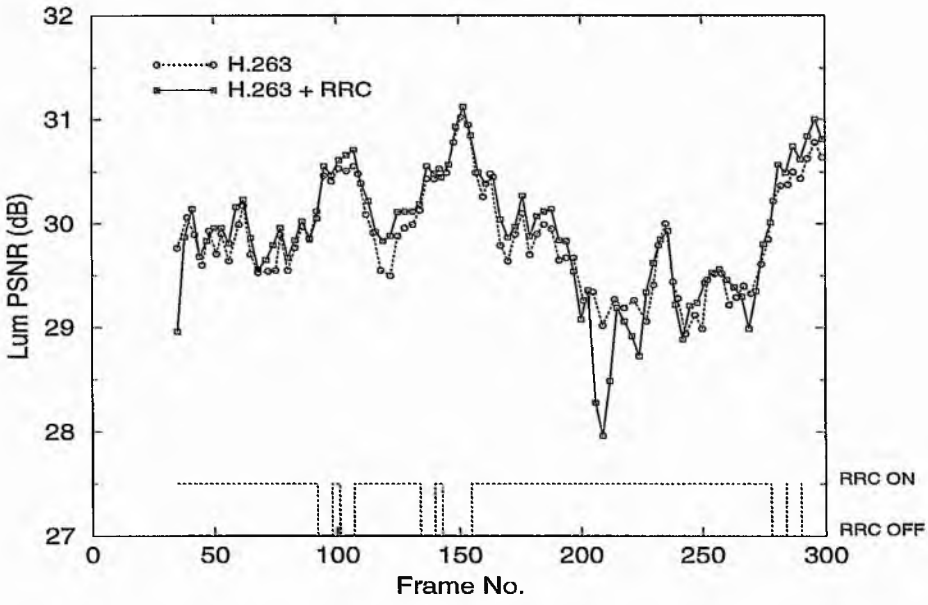


Fig. 6.13 : Luminance PSNR of Silent Voice coded at 40 kb/s using variable frame rate.

6.7 Concluding Remarks

In this chapter, a new coding mode is presented for coding residual image under very low bit-rate budget when the quantisation step size is extremely large. The use of large quantisation step size leads to the appearance of quantisation error and blocking artifacts on the decoded pictures. On the other hand, when the reduced-resolution coding mode is used, the subjective quality of the decoded pictures is significantly improved because the blocking artifacts are suppressed. Basically, the RRC mode down-samples the motion-compensated prediction error before it is being transformed and coded. As a result, finer quantisation of the transformed coefficients is possible while the resultant bit-rate is maintained at the allocated value. Consequently, the coding frame-rate can be maintained and the decoded sequence looked smooth even when sudden and unexpected motion appeared.

Chapter 7

Conclusion

The work in this thesis has primarily been based on the ITU H.263 video coding standard for low bit-rate applications. It was originally developed for video telephony over modems and analog telephone lines. However, with its moderate complexity and the increasing computational power of the standard computer chips, it is now becoming possible to implement both the encoding and decoding algorithms in standard PC. Moreover, with the increasingly available ISDN, its use in the years to come is assumed to be connected to the PC environment.

The aim of this research programme was to investigate techniques for improving the performance of standard block-transform video coders in terms of coding efficiency and subjective quality of the output images. Techniques for achieving both objectives have been developed and implemented in a H.263 coder. In the following section, a brief conclusion of the research achievements reported in this thesis is presented. This is followed by a short discussion on possible directions of future research activity.

7.1 Concluding Overview

In Chapter 3, a brief overview of several video compression techniques was given. Next the hybrid motion-compensated block-transform standard H.263 was described in detail. Its differences from the other established video coding standards such as MPEG and H.261 were identified which lead to the significant improvement in coding efficiency and perceptual quality. The usefulness of its four negotiable options were then assessed with the support of simulation results. All of them proved to be useful in one way or another in enhancing the performance of the coder. However, the improvement very much depends on the contents of individual video sequence with the exception of the syntax-based arithmetic option which helps to increase the compression ratio when the default VLC coder is replaced by it.

In Chapter 4, the rate control module whose function is to regulate the output data rate of a video coder was investigated. For block-transform video coders, the buffer fill-level is usually used for controlling the output data rate and the two parameters most commonly used for this purpose are the coding frame-rate and quantisation step size. For low-delay, fixed-rate transmission, this buffer feedback control technique may not be adequate. A feedforward rate control technique based on the statistics of the prediction error was devised and showed to be capable of achieving a stable output bit-rate. Basically, the technique makes use of a coded-bit prediction algorithm for finding a quantisation step size that can achieve the allocated bit-rate. A coded-bit estimation table with sum of absolute difference (SAD) as the entry was created for estimating the residual image coding bits.

Next, the above coded-bit prediction algorithm was incorporated into a bit allocation algorithm for adaptive quantisation of the prediction error in order to improve the subjective quality of the decoded picture. In brief, the region/object of interest (ROI) is coded more accurately at the expense of coarser quantisation of the rest of the picture for the same bit-rate budget. In order to achieve this coding strategy, an object locating algorithm is required for finding the location of the ROI. Simulations showed

that the objective and subjective quality of the decoded images of typical head and shoulder images are significantly improved by the above bit allocation algorithm.

Chapter 5 presented a rate-constrained motion compensation algorithm for improving the coding efficiency of standard hybrid video coder where block matching motion estimation technique (BMME) is employed for reducing the temporal redundancy. Analysis-by-synthesis technique was used for selecting the best motion vectors that result in the reduction of the overall bit-rate subjected to a distortion constraint. The heart of the algorithm is a rate-distortion constrained selection algorithm which searches for the best motion vector from the available candidates. The drawback of this algorithm is the increase in computational complexity associated with AbS technique. The advantage is that a significant reduction in the overall bit-rate was achieved without affecting the objective and subjective quality of the decoded images.

The switching strategy of standard block-based coder was next examined. Due to the multi-methodology approach adopted in these standard coders, efficient coding is expected if appropriate operating mode is chosen for coding each region of a picture. The above AbS technique was used for selecting the appropriate mode. More efficient coding results were obtained due to the more sensible use of the available modes. A simplified algorithm based on coded-bit and reconstruction distortion estimation was implemented. This resulted in a substantial reduction in the computational complexity while sacrificing a loss in performance. The rate-constrained motion compensation algorithm was incorporated into this operating mode selection algorithm. Only a small improvement in performance was observed. This led us to the conclusion that a better rate-distortion optimised result can be achieved with an efficient operating mode switching strategy without resorting to the motion estimation stage. However, this conclusion is only true when a two-level hierarchical BMME scheme is employed in the coder. For a video coder where only a single level BMME is adopted, the rate-constrained motion compensation algorithm is still useful in enhancing the coding efficiency.

In Chapter 6, an alternative method for coding the residual image under very tight bit-rate budget was investigated. Mainly due to coarse quantisation of the residual image,

blocking artifacts become visible at very low rate coding. However, if the residual image is down-sampled, less coding bits will be needed due to a smaller number of transform coefficients. As a result, smaller quantisation step size will be allowed to use given the same bit-rate budget. This idea was implemented for the coding of prediction error. Its application for adaptive source/channel coding was first examined. Then, its use for coding sequences containing fast motion was investigated. Both experiments showed significantly better results in terms of the subjective quality of the decoded images than using a large quantisation step size.

7.2 Thoughts of Future Work

There is a growing interest in using compressed video in a wide variety of areas. Besides its use for communication over mobile networks, the main area where very low bit-rate video will be used in the years to come is assumed to be connected to the PC environment. For both applications, computational complexity will be a vital factor for their practical implementation. Hence, future research for these applications should be aimed for low complexity algorithms. Moreover, for mobile communication where bit errors in the transmission is a major problem, error resilient coding, which is currently being intensively investigated, is worth investigating.

At the moment, much compression activity is taking place in MPEG4, a new video coding standard based on object-oriented concept. The coding structure is a merge between the segmentation-based scheme and the current block transform coding scheme. As a result of being object oriented, coding the contour and texture of arbitrary-shaped regions is necessary. Contour coding is a relatively mature area, several efficient coding methods exist. However, most of them belong to the lossless approach which is unnecessary in lossy video coding for communication applications. Therefore, new techniques which result in lower bit-rate by allowing some distortion in the contours will be preferred. As for texture coding, a number of techniques have been proposed but it is coded in very much the same way as in H.263 using DCT coding. A fairly computationally intensive padding technique [67] is used for padding

those blocks that contain the contour into a 8x8 block. Therefore, more effort can be spent on developing new techniques which are much simpler and efficient. Moreover, the motion estimation approach is only a small modification of the block matching technique. Hence, future work should also include the design of new techniques which allow interframe prediction of the contour and texture information of arbitrarily shaped regions in order to reduce the temporal redundancy.

On the other hand, the ITU group that developed H.263 have been collaborating closely with MPEG4 for the last two years. This led to the results of additional functionalities being added to the core H.263. These functionalities, like the 4 optional modes described in section 3.7.4, are optional and their performance is sequence dependent. Further work in the direction of mode selection strategy for achieving optimum performance is worthwhile.

List of Publications

1. T.H. Kweh, F. Eryurtlu, A.M. Kondoz, "Rate control algorithm for block-based variable rate video encoders", IEE Electronic Letters, Vol. 32, No. 14, pp. 1277-1278, July 1996.

2. T.H. Kweh, F. Eryurtlu, A.M. Kondoz, "Closed-loop motion compensation for video coding standards", IEE Proceedings on VISP, Vol. 144, No. 4, pp. 227-232, August 1997.

References

[1] R. J. Clarke, "Digital Compression of Still Images and Video", Academic press, 1995.

[2] A.N. Netravali and J.O. Limb, "Picture coding: A review", Proc. IEEE, vol. 68, pp. 366-406, Mar. 1980.

[3] ITU-T Recommendation H.263, "Video coding for low bitrate communication", Nov. 1995.

[4] P.A. Wintz, "Transform Picture Coding", Proc. IEEE, vol. 60, no. 7, pp. 809-820, Jul. 1972.

[5] "Video Coding: The Second Generation Approach", Edited by L. Torres, M. Kunt, Kluwer Academic, 1996.

[6] A. Murat Tekalp, "Noise Filtering", Digital Video Processing, Prentice Hall PTR, chapter 14, pp. 262-282, 1995.

[7] H.G. Musmann, "Predictive Image Coding", Image Transmission Techniques, Academic Press, New York, pp. 73-112, 1979.

[8] H.G. Musmann, P. Pirsch, H.J. Grallert, "Advances in picture coding", IEEE Proc. vol. 73, pp. 523-548, 1985.

[9] K.A. Prabhu, "A predictor switching scheme for DPCM coding of video signal", IEEE Trans. Commum. Vol. COM-33, pp. 373-379, 1985.

[10] D. Pearson, "Image Processing", McGraw-Hill, 1991.

[11] J. Oest, F.J. Guirao, N. García, "Digital transmission of component coded HDTV signals using the DCT : design of a visibility threshold matrix", Proc. EUSIPCO 90, pp. 881-884, Barcelona, Sept. 1990.

[12] H.S. Malvar, "Signal Processing with Lapped Block Transform", Artech House, London, UK, 1992.

[13] A.N. Akansu, "Multiresolution signal decomposition: transform, subbands, and wavelets", Academics Press, 1992.

[14] R.J. Clark, "Transform coding of Images", Academic Press, San Diego, 1985.

[15] H. Li, Z. He, "Directional subband coding of images", ICASSP Proc., pp. 1823-1826, 1989.

[16] T.D. Lookabaugh, M.G. Perkins, "Application of the Princen-Bradley filter bank to speech and image compression", IEEE Trans. Acoust. Speech, Signal Process, ASSP-38, pp. 1914-1926, 1990.

[17] A.N. Netravali, J.O. Limb, "Picture coding: a review", Proc. of the IEEE, vol. 88, no. 3, pp. 366-406, Mar. 1980.

[18] J. Rose, W. Pratt, G. Robinson, "Interframe cosine transform image coding", IEEE Trans. On Commun., vol. 25, no. 11, pp. 1329-1339, Nov. 1977.

[19] G. Karlsson, M. Vetterli, "Three dimensional subband coding of video", Proc. of ICASSP 88, pp. 1100-1103, New York, Apr. 1988.

[20] Kunt, A. Ikonomopoulos, M. Kocher, "Second generation Image Coding techniques", Proc. of the IEEE, Vol. 73, no. 4 , pp. 547-575, Apr. 1985.

[21] P. Salembier, "Multi-Criterion segmentation for image coding", Int. Workshop on Mathematical Morphology and Its Applications to Signal Processing, Barcelona, May 1993.

[22] M. Gilge, T. Engelhardt and R. Mehlan, 'Coding of Aritrarily Shaped Image Segments Based on A Generalized Orthogonal Transform', Signal Processing : Image Communication 1, pp. 153-180, 1989.

[23] A. Kaup and T. Aach, 'A New Approach Towards Description of Arbitrarily Shaped Image Segments', IEEE International Workshop on Intelligent Signal Processing and Communication Systems, Taipei, Taiwan, March 1992.

[24] S.P. Lloyd, "Least squares quantisation in PCM", IEEE Trans. on Information Theory, vol. 28, no. 2, pp. 129-137, Mar. 1982.

[25] J. Max, "Quantising for minimum distortion", IEEE Trans. on Information Theory, vol. 6, no. 1, pp. 7-12, Mar. 1960.

[26] M. Gray, "Vector quantisation", IEEE ASSP Magazine, vol. 1, no. 2, pp. 4-29, Apr. 1984.

[27] F. Lavagetto, S. Zappatore, "Adaptive vector quantization for fixed bit-rate video coding", Signal Processing, vol. 34, no. 1, pp. 19-31, Oct 1993.

[28] M. Goldberg, P.R. Boucher, S. Shlien, "Image Compression Using Adaptive Vector Quantization", IEEE Trans. On Commun, vol. COM-34, no. 2, pp. 180-187, Feb 1986.

[29] H.M. Hang, J.W. Woods, "Predictive vector Quantisation of Images", IEEE Trans. On Commun", vol. COM-33, no. 11, pp. 1208-1219, Nov. 1985.

[30] N.S. Jayant, P. Noll, "Digital coding of waveforms", Prentice-Hall, Englewood Cliffs, USA, 1984.

[31] ISO/IEC IS 10918 (JPEG).
"Information Technology -Digital Compression and Coding of Continuous-Tone Still Images", 1994.

[32] ITU-T Recommendation H.261.
"Video codec for audiovisual services at p x 64 kbit/s", Rev. 2, 1993.

[33] ISO/IEC IS 11172 (MPEG-1).
"Information Technology - Coding of Moving Pictures and Associated Audio for Digital Storage Media Up to About 1.5 Mbit/s", 1993.

[34] ISO/IEC DIS 13818 (MPEG-2).
'Information Technology - Generic Coding of Moving Pictures and Associated Audio. ITU-T Recommendation H.262', Mar. 1994.

[35] G. Bjontegaard, "Very Low Bitrate Video Coding Using H.263 And Forseen Extensions", ECMAST 96, Proc Part II, Belgium, pp. 825-838, May 1996.

[36] M.T. Orchard, G.J. Sullivan, "Overlapped Block Motion Compensation: An Estimation-Theoretic Approach", IEEE Trans. On Image Process., vol. 3, No. 5, Sept. 1994.

[37] J.O. Limb, "Buffering of data generated by the coding of moving images", Bell Syst. Tech. J. 51, pp. 239-259, 1972.

[38] M. Denicolai, "Evaluating and Improving Video-Codec Image Quality", Electronic Design, pp 81-92, 10 Jul. 1995.

[39] Telenor R & D, "H.263 video codec test model", Nov 1995.
Http://www.nta.no/brukere/DVC/tmn5/tmn5.html.

[40] J.A. Saghri, A.G. Tescher, "Knowledge-based image bandwidth compression and enhancement", Proc. SPIE, vol. 804, pp. 201-216, 1987.

[41] R.H.J.M. Plompen, J.G.P. Groenveld, F. Booman, D.E. Boekee, "A image knowledge based video codec for low bitrates", Proc. SPIE, vol. 804, pp. 379-384, 1987.

[42] M. Soryani, R.J. Clarke, "Image segmentation and motion-adaptive frame interpolation for coding moving sequences", Proc. ICASSP, pp. 1882-1885, 1989.

[43] A. Henry, A. Rowley, B. Shumeet, T. Kanade, "Neural Network-Based Face Detection", IEEE Trans. On Pattern Analysis and Machine Intelligence, vol. 20, no. 1, Jan. 1998.

[44] A. N. Netravali and J. D. Robbins, "Motion-compensation television coding : Part 1", Bell Syst. Tech. J. 58, pp. 631-670, Apr. 1979.

[45] C. Cafforio, F. Rocca, "The differential method for motion estimation", Image Sequence Processing and Dynamic Scene Analysis, T.S. Huang, Ed. New York : Springer-Verlag, pp. 104-124.

[46] D. R. Walker, K. R. Rao, "Improved pel-recursive motion compensation", IEEE Trans. On Commun., vol. COM-32, pp. 1128-1134, Oct. 1984.

[47] B. K. P. Horn, B. G. Schunck, "Determining optical flow", Artif. Intell., vol. 17, pp. 185-203, 1981.

[48] H. H. Nagel, "Displacement vectors derived from second-order intensity variations in image sequences", Computer Graphics and Image Process., vol. 21, pp. 85-117, 1983.

[49] J.R. Jain, A.K. Jain, "Displacement measurement and its application interframe image coding", IEEE Trans. On Commun., vol. COM-29, pp. 1799-188, 1981.

[50] T. Koga, K. linuma, A. Hirano, Y. lijima, T. lshigun, "Motion-compensated interframe coding for video conferencing", NTC 81, Proc., New Orleans, pp. G5.3.1-G5.3.5, LA, Dec. 1981.

[51] H. Gharavi, M. Mills, "Block matching motion estimation algorithms - new results", IEEE Trans. Circuit Syst. CAS-37, pp. 649-651, 1990.

[52] M. Bierling, "Displacement estimation by hierarchical block matching", Visual Communication and Image Processing, Proc. SPIE, vol. 1001, pp. 942-951, Nov. 1988.

[53] M. Accame, D.D. Giusto, "Adaptive-size hierarchical block matching for efficient motion compensation of video sequences", Proc. SPIE, Vol. 2451, pp. 112-120, 1995.

[54] L. Rabiner, R. Schafer, "Digital Processing of Speech Signals", Prentice-Hall, Signal Processing Series, 1978.

[55] A.M. Kondoz, "Analysis-by-Synthesis Coding of Speech", in Digital Speech, John Wiley & Sons, 1994.

[56] Andre Vogt, "Applications of block matching motion estimation utilising an analysis by synthesis technique in present video coding standards", Internal report, CSER, Uni. of Surrey, Mar. 1996.

[57] K. Ramchandran, A. Ortega, M. Vetterli, "Bit allocation for dependent quantisation with applications to multiresolution and MPEG video coders", IEEE Trans. Image Processing, vol. 3, no. 5, pp. 553-545, Sept. 1994.

[58] J. Lee and B.W. Dickinson, "Joint optimization of frame type selection and bit allocation for MPEG video coders", in Proc. ICIP, vol. 2, pp. 962-966, 1994.

[59] T. Wiegand, M. Lightstone, D. Mukherjee, T.G. Campbell, S.K. Mitra, "Rate-Distortion Optimized Mode Selection for Very Low Bit Rate Video Coding and the Emerging H.263 Standard", IEEE Trans. On Cct. And Syst. For Video Tech, vol. 6, No. 2, Apr. 1996.

[60] ITU Telecommunication Standardisation Sector, Draft ITU-T Recommendation H.263, "Video Coding for Low Bitrate Communication", Feb. 1997.

[61] A.H. Sadka, "Error Control Strategies in Block-Transform Video Coders for Multimedia Communication", Ph.D Thesis, University of Surrey, 1997.

[62] Peter Sweeney, "Error Control Coding: An Introduction", Prentice Hall, 1991.

[63] J. Hagenauer, "Rate-Compatible Punctured Convolutional Codes and their Applications", IEEE Trans. on Commun, vol. 36, no. 4, pp. 389-400, Apr. 1988.

[64] International Organisation for Standardisation, ISO/IEC/JTC1/SC29/WG11, "E4: Core Experiment on Error Resilient Methods Based on Back Channel Signalling and FEC", July 1996.

[65] J.D. Lameillieure, "Scalable Video Coding For Fast Sequence Previewing in multimedia browsing", European Conference on multimedia applications, services and techniques, Proc. Part II, pp. 635-654, May 1996.

[66] M. Ghanbari, "An Adapted H.261 Two-Layer Video Codec for ATM Networks", IEEE Trans. On Comm., vol. 40, no. 9, pp. 1481-1490, Sept. 92.

[67] International Organisation for Standardisation, ISO/IEC JTC1/SC29?WG11. "MPEG-4 Video Verification Model Version 8.0 : Coding of Moving Pictures And Associated Audio Information, Document No. 1796", July 1997.

[68] D.A. Huffman, 'A method for the construction of minimum redundancy codes', Proc. IRE, 40, pp. 1098-1101, 1952.

[69] M.C. Chen, A.N. Willson, "Rate-distortion optimal motion estimation algorithm for video coding", pp. 2096-2099, 1996.

[70] W.C. Chung, F. Kossentini, J.T. Smith, "An Efficient Motion Estimation Technique Based on A Rate-Distortion Criterion", pp. 1926-1929, 1996.

[71] J.M. Griffiths, 'ISDN Explained', 2nd Ed., John Wiley & Sons, 1992.

[72] F. Fluckiger, 'Understanding Networked Multimedia: Application and Technology', Prentice Hall, 1996.

[73] Richard Fryer, "Delay Computation for H.263 Novel Algorithm Proposals", University of Strathclyde, Apr. 97.

[74] D. Deutsch, J. Escobar, C. Partridge, "A Multi-Service Flow Synchronisation Protocal", BBN STC Tech Report, Cambridge, Mass, Mar. 1991.

[75] Wu-Hon and al., "A software architecture for workstations supporting multimedia conferencing in packet switching networks", IEEE Journal on Selected Areas in communication, vol. 8, no. 3, pp. 380-390, Apr. 1990.

[76] Toshiake Watanabe, "Designing process for reversible variable-length code (RVLC)", Technical Report from Toshiba, Oct. 1996.