University College London
Department Of Electrical and Electronic Engineering
MSc in Telecommunications

MSc Dissertation

Reliable Video Transmission Over Packet Networks

Isaac Kokkinidis[†]

August 1998

[†] sponsored by the **Greek State Scholarships Foundation**

Abstract

A survey of current research on reliable video transmission over lossy packet networks is presented in this dissertation. Reliability can be provided by encoding the data in such a way that transmission errors would have minimal effect, and by using error concealment techniques at the receiver. Some proposed solutions for enhancing the error robustness and the quality of service for video transmission schemes that utilise interframe prediction at the encoding stage are presented, using a simulation model of a H.263-compliant codec.

Preface

This thesis is submitted as part of the requirements for the MSc in Telecommunications, University College London.

During my work for the project, for which this document is reporting the results, I had to deal with cutting-edge issues in the area of digital video compression and transmission. In this research area, especially in its Internet-related part, proposals are made and implemented so quickly, that it is very difficult for an individual to keep pace with the proceedings.

In a masters final project the time period is usually limited to *implement* any 'novel' ideas that could probably be presented in a research event. However, I have the ambition that the careful reader would agree with me that this thesis cannot be described as the result of purely bibliographic and theoretical work.

Acknowledgments

The author is sponsored during this postgraduate course by the **Greek State Scholarships Foundation**, Lysikratous 14, GR-105 58, Athens, Greece.

Many thanks to Dr. Vicky Hardman for her helpful comments and interesting discussions. Also to Dr. Lionel Sacks for helping me find such an interesting project, and for his comments on early versions of this report.
Ditto to my friends and colleagues Notis Stamos and Panos Gevros, who, apart from their support, made my year in London very enjoyable.
I am also grateful to all my new friends, fellow MSc students and not, we had some unforgettable moments this year.

Isaac Kokkinidis
August 1998

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Digital Video Communications

Until recently, digital video communication was only used in research and academic experiments. It is the unexpected growth of the Internet that stimulated the proliferation of networked multimedia, including video sequences transmission. The video component requires usually the largest portion of computing power and bandwidth of a multimedia coding and transmission system, and imposes strict latency constraints. Therefore, it is the milestone application for future multimedia systems.

Currently, there is a large scale deployment of digital terrestrial and satellite television networks, which are bound to gradually replace their analogue counterparts. There is also a growing demand for streaming video services over the Internet, as well as for video conferencing services over data networks. The trend is towards *convergence* of the telecommunications, broadcasting and information technology (IT) industries, in order to provide seamlessly interactive multimedia services to the end users.

This trend is supported by international standards for multimedia coding. Particularly for video sequences, there is a plethora of standards for different qualities and applications, each applying efficient techniques to remove the source redundancy inherent in digitised video, in other words, compress the information. This is a very demanding task in terms of computing performance, but advances in processing capabilities enable its execution within the real-time requirements imposed by the industry. With compression, it has become practical to store and manipulate digital video with currently available storage systems.

Whether it has become practical to *deliver* video to the users, depends on the underlying network technology, as well. While broadcast networks, be them cable, terrestrial, or satellite, have been designed with relatively large bandwidth capacity, they are not yet optimised for transmitting digital information. On the other hand, Internet Protocol (IP) data networks do not currently fulfil neither the bandwidth, nor the latency and Quality of Service (QoS) requirements for the average user. It is well understood that future broadband systems using Asynchronous Transfer Mode (ATM) technology, will accommodate digital video transmission in their wide range of services.

1.2 Applications

The entire range of applications for digital video cannot be defined exhaustively, as the versatility of digital information enables the multiplexing of several media types and the creation of unforeseeable combinations between them. Currently, digital video applications include:

- *Face-to-face communications,* as desktop and mobile video telephony and video conferencing. Video could be delivered over IP-based networks, Integrated Services Digital Networks (ISDN), or mobile networks, serving home or corporate users. The connections could be point-to-point, or point-to-multipoint, the latter enabling applications such as teleworking, computer supported cooperative work (CSCW), distance learning.
- *Broadcast entertainment video,* which could be offered in a variety of qualities, as low as the quality of streaming video currently available over the Internet, and as high as the quality of cinema films, and even higher. Digital TV applications, for which the MPEG2 standard caters, are in this category.
- *Video databases,* including video on demand, which is the holy grail of the media and telecommunications industries. It comprises the smart archiving of digital video, enabling search by criteria, and its delivery with one of the aforementioned network technologies.
- *Telemedicine applications,* which are more special than face-to-face communications, as they should enable, apart from collaborative video conferencing between medical practitioners, the high-quality, preferably lossless, encoding and transmission of medical images.

Potential future applications include object-based schemes, suitable for interactive transaction with objects on the video scene and viewpoint selection, for example, video games or virtual reality applications. Three-dimensional and stereoscopic digital video coding and transmission is also a promising research area.

1.3 Goals

In this thesis, the target scheme is a reliable, or, more precisely, robust video transmission system, which operates over the Internet. The system should be able to provide acceptable video output even when operating under unfavourable conditions, where not all encoded information is received at the decoder.

The inherent redundancy of video information is effectively removed by video coding algorithms. Apart from making the resulting bitstream more sensitive to errors, this removes a lot of information that could help the receiver to recover potential losses of information. The common remedy for this problem is to trade error-resilience for video coding compression efficiency by intentionally adding redundancy or using retransmission. Moreover, picture enhancing techniques are applied at the receiver to make errors less noticeable.

There is a plethora of available techniques which can be combined in many different ways in order to design the above robust schemes. This work aims to point out which are more appropriate for the target scheme, by identifying technology matches and mismatches, categorising applications, determining requirements, spotting possible design flaws and pitfalls, and proposing end-to-end solutions.

In order to support the arguments expressed in the above design stage, and as a case study, a subset of the above proposals is applied to a H.263 encoder and experimental results are obtained. Should the latter confirm the validity of the most of these suggestions, it will be considered that this work has achieved its main goals.

1.4 Outline

The logical outline of this thesis is organised in two parts. Part I, comprising Chapters 2, 3, and 4, contains the essential background and the technologies applied to digital video encoding, transmission over packet networks, and reliable video communications, respectively. Part II, with Chapters 5, 6, and 7, is the decision and implementation stage, where specific technologies are selected according to which are the most appropriate for the target scheme, the results of experiments on some of them are presented, and conclusions are drawn.

Chapter 2

Video Coding: Principles and Standards

2.1 Introduction

Video transmission systems are characterised by relatively high bandwidth and processing power requirements, due to the large amounts of information involved. For example, if digitised TV-quality video was transmitted 'as is', it would require a raw throughput of hundreds of Mbps.[1] Although such scales are currently feasible in modern communication, storage and processing systems, they are completely uneconomical. Therefore, the need to compact the video information and keep the required bit rates at a reasonable level arises.

The main principle applied to video coding is that the limitations of the human visual system should be exploited, by throwing effectively away the information that cannot be perceived by the average viewer. This principle, which exploits the inherent *subjective* redundancy of video sequences, is applied universally, from the video capturing to the displaying systems, and has been the basis for considerable reductions to the bandwidth requirements of analogue broadcast TV systems, exploiting the limited human sensitivity to the spatial, temporal and colour resolution of the transmitted video sequences.

Furthermore, video information exhibits high correlation in the temporal and spatial direction. For the former, that is because when a moving scene is segmented into a number of still frames, they tend to be very similar to each other, over relatively long time intervals. The latter stands for regions of

[1] Mega-bits per second.

each of these still images that have similar brightness and colour intensities. Digital video can be further compressed by using numerical redundancy elimination techniques such as entropy and arithmetic coding.

Following the introductory paragraph on representation of video information, some techniques of video coding and compression are briefly described. In the literature, these are usually categorised to still and moving image coding techniques. However, an integrated presentation according to the underlying technology used, was considered more appropriate for this document. The applications of some of these techniques in standards are concluding the chapter.

2.2 Digital Representation of Video Information

From the concept of the first systems storing and transmitting digital visual information, some constraints and assumptions were identified. First, the scene is restricted to a finite two-dimensional image size, sampled at a specified frame rate. This introduces the concepts of temporal and spatial resolution. Obviously, the former is chosen to be sufficient to preserve smoothness in the reproduction of scene movement with respect to the human visual system, usually in the range of 24–30 frames per second (fps). The spatial resolution determines the physical size and total number of the picture elements, or *pels*, that compose a still frame.

Second and last, there is a finite colour depth, allocating a number of bits for each pel, according to the desired display quality. Technically speaking, this determines the sampling rate used to capture the light frequencies of a visual scene. These are represented in various ways, using sets of component signals, which can be combined to synthesise any colour. One of the most frequently used, inspired by the way the human visual system senses light frequencies, is the red, green and blue primary colours representation. It has been shown that 8 bit unsigned numbers suffice for the representation of each of the primary colours, as the average human viewer cannot discriminate more distinct colours.

Other colour coordinate systems exist, usually a result of a linear transform from the RGB system. One of the most practical separates the image brightness information, or luminance, from the colour information, or chrominance. This is accomplished by taking a weighted sum of the RGB signals as the luminance signal, Y, and using colour differences for the chrominance components, C_b and C_r, as shown with the transform pair in (2.1). This colour space is specified in the CCIR 601 recommendation for digital video, yielding an effective range for Y from 16–235, and for the chroma signals C_b, C_r from 16–240.

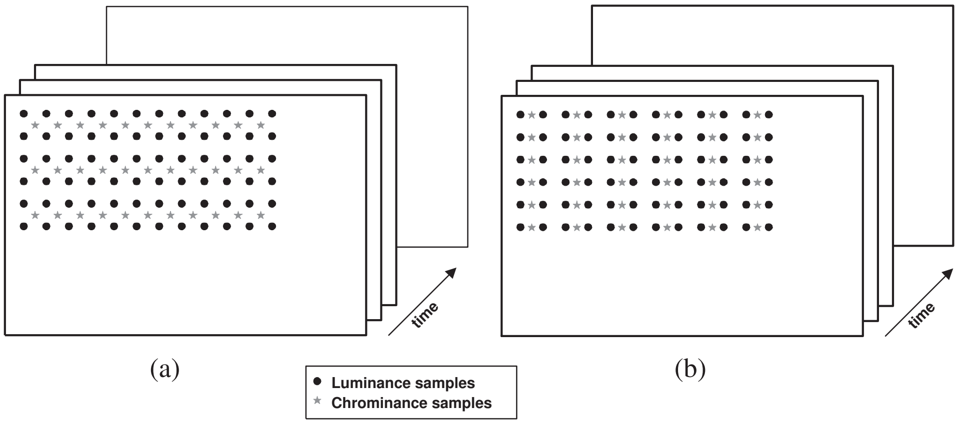


Figure 2.1: Sampling patterns for (a) 4:2:0 and (b) 4:2:2 formats

$$\begin{bmatrix} Y \\ C_b \\ C_r \end{bmatrix} = \begin{bmatrix} 0.257 & 0.504 & 0.098 \\ -0.148 & -0.291 & 0.439 \\ 0.439 & -0.368 & -0.071 \end{bmatrix} \cdot \begin{bmatrix} R \\ G \\ B \end{bmatrix} + \begin{bmatrix} 16 \\ 128 \\ 128 \end{bmatrix} \tag{2.1}$$
$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 1.164 & 0 & 1.596 \\ 1.164 & -0.392 & -0.813 \\ 1.164 & 2.017 & 0 \end{bmatrix} \cdot \begin{bmatrix} Y - 16 \\ C_b - 128 \\ C_r - 128 \end{bmatrix}$$

The partitioning to luminance and chrominance has the advantage that the human eye is more sensitive to the former, and can tolerate lower spatial resolutions of the latter without noticeable effect. Therefore, for digital video coding, each of the chrominance signals is usually subsampled with a ratio of 1:2 or 1:4 to the luminance sampling rate, as shown in Figure 2.1.

2.3 Removing Source Redundancy

As described above, compression techniques are used to eliminate the spatial, temporal and statistical redundancy of the video signal. Generally, they can be categorised in *lossless* and *lossy* video coding schemes. The former are preserving every bit of the original information, producing identical reconstructed pictures at the decoder, while the latter sacrifice some of the least significant visual information to improve compression efficiency. Currently, lossless schemes are only used in medical imaging applications, where, even for psychological reasons, it is justifiable to maintain maximal decoding quality at the expense of performance, while lossy schemes have been adopted for nearly all other applications.

The algorithms are usually applied in two stages; first waveform encoding and then entropy encoding. Some of the most popular generic algorithms for coding of video sequences, *i.e.* content-independent schemes, are described in the following paragraphs.

2.3.1 Predictive Coding

General signal processing techniques that have been successfully applied to encode and compact audio information, such as Differential Pulse Code Modulation (DPCM), can be applied to the two-dimensional raster signal of each video frame. These usually use an average of the neighbouring pels as prediction and send only the prediction error. This mode, called intraframe prediction, exploits the spatial redundancy within each frame. If the predictor uses pels from different frames, usually the temporally previous to the current, it is called interframe prediction, and it removes temporal correlation as well.

Block Matching

In the interests of simplicity and transmission efficiency prediction is usually done in terms of blocks of pels. According to the block matching technique [NH95], each frame is segmented into small orthogonal blocks which are labelled by the coordinates of their upper-left corner. For each of these blocks, a search is performed to find the best matching block of the previous in time frame (backward prediction). This is done in order to maintain the current frame's grid, as shown in Figure 2.2.

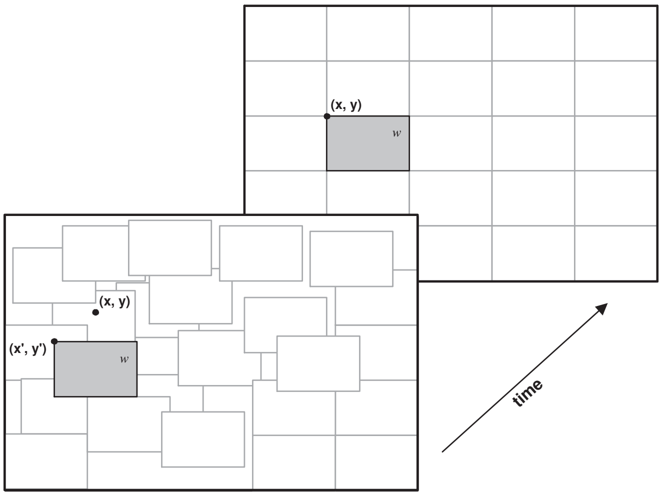


Figure 2.2: Backward prediction for block matching.

Considering the block of the current frame with its upper-left corner at the position (x, y), the search is done in a restricted area around the pel (x, y) of the previous frame, as it is assumed that the block translations are small from frame to frame. Exhaustive search, checking every possible displacement in the search region, is the simplest approach, while more sophisticated approaches, as logarithmic or three-step search are also applied.

The matching criterion is usually the minimisation of a norm applied to the luminance values of the two blocks. It does not need to be applied to the chrominance values as well, as these do not provide additional information about motion. The whole process, depicted in Figure 2.3, is summarised:

$$\begin{gathered}\forall(dx, dy) \in W, \quad \text{find the vector } [dx, dy]^T \\ \text{that minimises the norm:} \\ \mathcal{L}_{(x,y)\in w}[cur(x, y), prev(x + dx, y + dy)]\end{gathered}$$

where *cur* stands for the current frame's block, *prev* for the previous frame's block, w is the block area, W is the search area, and dx, dy are the horizontal and vertical displacements of the reference block. The most commonly used norms are the sum of absolute differences or $\mathcal{L}_1$, and the sum of squared differences or $\mathcal{L}_2$.

If the best-matching block has its upper-left corner at the position (x', y') on the previous frame, the *motion vector* $[dx, dy]^T = [x - x', y - y']^T$ represents the estimated displacement of this block in order to overlap the block (x, y) of the current frame.

The collection of all motion vectors of a specific frame, called the vector

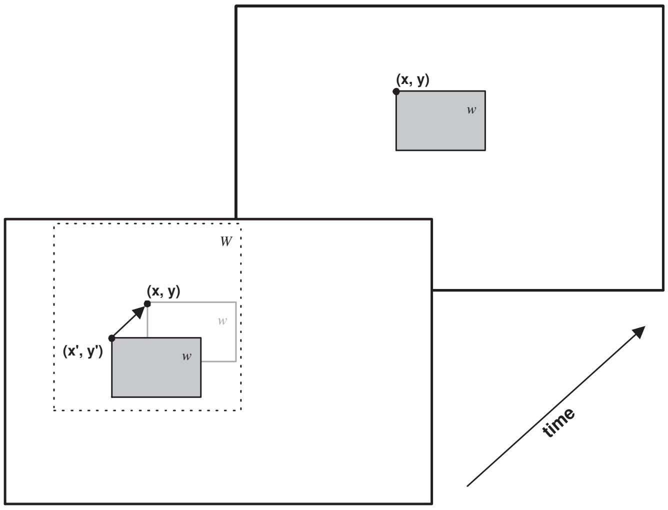


Figure 2.3: The block matching algorithm.

field, is a motion descriptor of the video sequence. Block matching is a special case of motion estimation, as it only deals with translational motion, and does not cope well with rotational motion. However, the simplicity of the algorithm, and the compliance of most video sequences with its smooth motion assumption, has made the industry to adopt it in most video compression standards.

Block matching is a moderately processor-intensive procedure, and it is done only at the encoding stage. At the decoder, motion compensation takes place, which comprises displacing the previous frame's blocks according to the received vector field. This task is much less processor demanding and it is easily implemented in hardware.

2.3.2 Block Transform Coding

The use of transform coding for blocks of pels has proven to be an effective way of compacting the block energy, namely the information, in a few transform coefficients. Not all of these need to be transmitted to maintain good image quality, and they need not be represented with full accuracy [NH95]. The most effective with respect to compression performance for images is the Karhunen-Loeve Transform (KLT). However, its high computational complexity has led to the adoption of the Discrete Cosine Transform (DCT) for the majority of applications.

Discrete Cosine Transform

The two-dimensional DCT of a $N \times N$ block of pels is described by the following pair of equations.

$$F(u,v) = \frac{2}{N} C(u) C(v) \sum_{i=0}^{N-1} \sum_{i=0}^{N-1} f(i,j) \cos\left(\frac{(2i+1)u\pi}{2N}\right) \cos\left(\frac{(2j+1)v\pi}{2N}\right) \tag{2.2}$$

$$f(i,j) = \frac{2}{N} \sum_{i=0}^{N-1} \sum_{i=0}^{N-1} C(u) C(v) F(u,v) \cos\left(\frac{(2i+1)u\pi}{2N}\right) \cos\left(\frac{(2j+1)v\pi}{2N}\right) \tag{2.3}$$

where

$$C(x) = \begin{cases} \frac{1}{\sqrt{2}}, & x = 0 \\ 1, & \text{otherwise} \end{cases}$$

and $f(i,j)$ are the pel values, which are integers usually in the range 0–255, with $F(u,v)$ their correspondent DCT coefficients, which are real numbers.

The DCT converts a block of pels to a same-sized block of coefficients, representing the image data in the spatial frequency domain. Upper-left corner coefficients represent low spatial frequencies, and therefore contain most of the information for the majority of blocks within a picture, while lower-right corner coefficients represent high spatial frequencies, most of which are usually near zero. The first coefficient, at the top-left position of the block is called the DC value and represents zero spatial frequency, while all the remaining are the AC coefficients.

Quantisation

Real numbers, and consequently DCT coefficients, are very expensive in terms of transmission efficiency. Therefore, quantisation is applied to reduce the representation accuracy of the transform, and delete, or set to zero, the frequently appearing near-zero coefficients. This is the point where the information leakage that makes the scheme lossy resides, as, in the inverse process at the decoder, it is not possible to restore precisely the DCT coefficients, but only an approximation of them, depending on the selected quantisation parameters.

DCT coefficients quantisation does not have the effect of, for example, DPCM samples quantisation, where a particular quantised sample affects relatively few pels in the reconstructed image, as in the former all pels of the block are affected. This leads to a *spreading* of the quantisation error, and much better subjective quality of the reconstructed images [NH95]. However, the unavoidable loss of information produces distortions at the block boundaries, or 'block artifacts,' which are emphasised by the limited precision of block-based motion estimation, usually used in conjunction with DCT.

Encoding

Further compression can be achieved by exploiting the statistical redundancy in the resulting quantised DCT coefficients. This can be achieved with run-length or entropy encoding, such as Huffman encoding. The general idea behind these encoders is that the most frequently occurring values are assigned shorter codes, with longer codes representing the less frequent values.

Particularly, for DCT coefficients, it is better to select the input to the run-length encoding by applying a special zig-zag block scanning pattern, as shown in Figure 2.4. This scanning ensures that the correlation of adjacent samples will be maximal, increasing the overall coding efficiency.

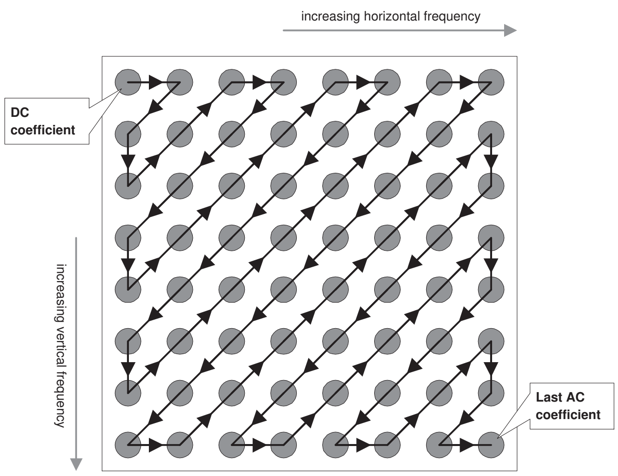


Figure 2.4: The zig-zag scanning pattern for the DCT coefficients.

2.3.3 Other Techniques and Current Research

Apart from transform coding, other video coding techniques include block-based and non-block-based algorithms. An example of the former is fractal encoding, which segments the image in small enough blocks to apply self-similarity theory, and finally describe the whole image with a formula. This leads to very high compression efficiency, but it is also a very computationally intensive task at the encoder. Subband coding is an example of the latter, which applies several bandpass filters to the source image, producing bands of spatial frequencies, which are further compressed in the same manner as the DCT coefficients.

All the techniques described above are content-independent, that is, they deal with any type of source images, as they apply generic waveform encoding. For special applications as personal communications, model-based coding schemes are applied, which achieve very high compression efficiency, leading to very low bitrate coding schemes. For example a human speaker is usually modelled as 'head and shoulders before a stationary background' video sequences, and geometrical transforms are applied to handle temporal changes, as lips and eyes movement.

2.4 Standards and Intended Applications

The most popular video coding standards, described in this section, were developed by two organisations. The one is the International Telecommunications Union-Telecommunicatinons sector (ITU-T), which has specified the H.26x standards, targeted to videoconferencing applications. The second is the International Standards Organisation (ISO), particularly its Moving Pictures Expert Group (MPEG), which developed the MPEG standards, targeted to entertainment video applications.

2.4.1 The ITU H.26x Family

Historically, the H.261 recommendation [ITU93] was developed in the mid-80s, and was developed for video transmission over ISDN channels. The final CCITT H.261 recommendations, released in late 1989, were made for a $p \times 64$ kbps video codec, where p is between 1 and 30, whence came the alternative name p*64 [NH95]. The H.263 standard [ITU95] was developed in the mid-90s, when speeds near 30 kbps over the Public Switched Telephone Network (PSTN) were available to the majority of users, and was targeted to such low bitrates. Currently, an improved version, H.263+, is proposed by the industry to support error resilient very low bitrate video delivery over wireless networks.

The ITU-T has standardised only the *decoder* for H.261 and H.263, and leaves the design details of a compatible encoder deliberately unspecified. Both standards are parts of umbrella standards, the H.32x standards, which describe the multiplexing of all data streams that compose a videoconferencing system. These include video, audio, data, control and signalling information, and packet frame structures. In the following paragraphs only the video part is described.

H.261

For H.261 only two input video resolutions are supported: the Common Intermediate Format (CIF) at 352×288 pels for the luminance, and the Quarter CIF (QCIF) at 176×144 pels. The chrominance values are sampled according to the 4:2:0 sampling, see Figure 2.1. The maximum frame rate is 29.97 fps, however in most applications frames are dropped before encoding and transmission to fit the available bandwidth.

The images are segmented in macroblocks, which comprise six 8×8 pel blocks, four for Y values and one for each of their respective C_r, C_b values. There are two types of frames in H.261. The first frame and specific frames in the sequence inserted for stability and resynchronisation, are encoded independent of their adjacent frames, *i.e.* intra-coded. All the other frames are inter-coded, that is, they are predicted using motion compensation from the previous frame.

Intra coding is DCT transformation of the 8×8 blocks of luminance and chrominance values, quantisation, and run-length encoding of the coefficients. An inverse quantisation and decoding procedure is also performed, in order to provide a local copy of the non-identical reconstructed image. The latter is used as a reference for prediction, rather than the original input. Therefore, for error-free transmission, the same prediction will be formed to the decoder and the encoder.

Inter coding applies block matching to the 16×16 luminance values of each macroblock, and determines the respective motion vectors, which are differentially encoded and transmitted. The motion compensation procedure is performed locally in order to calculate the prediction error, which, if above a threshold, is DCT encoded and transmitted. If the prediction error is very high, the motion vectors are ignored, and the corresponding macroblock is intra-coded. As in intra-frames, the new reference for prediction is the locally reconstructed motion compensated frame, which is stored in a buffer for this task. Inter coding achieves much higher compression efficiency than intra coding, in other words, the quality of inter-coded pictures is much better than that of intra-coded, for the same number of bits.

The resulting bitstream has a hierarchical structure. Blocks of variable length coded coefficients are collected together to form macroblocks, which are collected to form Group of Blocks (GOB), and a complete frame is made up of several GOB, see also Figure 2.5.

It is worth mentioning that the output bitrate of H.261 is variable, depending highly on the amount of motion in the encoded scene. As H.261 is using ISDN connections, which are constant bit rate, there is a rate control mechanism used, which changes the quantisation parameters when buffer overflows due to increased scene activity are imminent [RR96].

H.263

The bandwidth requirements of H.261 are considered high for the majority of domestic and mobile users, so a new standard was developed for bitrates

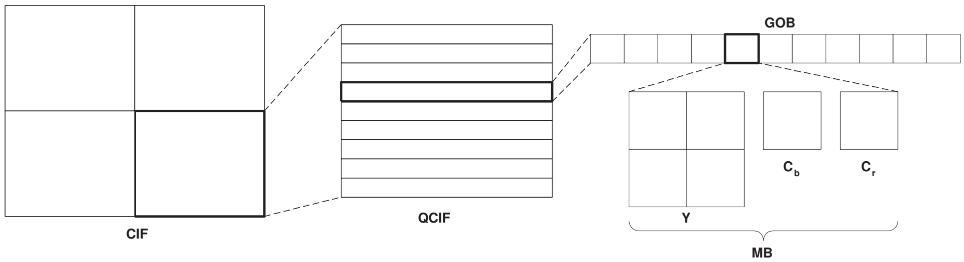


Figure 2.5: The hierarchy of structures for a H.263 CIF frame.

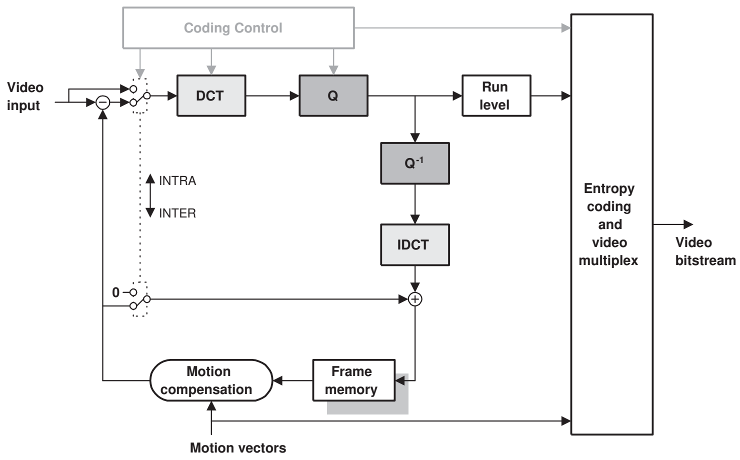


Figure 2.6: A simplified block diagram of an H.263 encoder

near 30 kbps. The H.263 encoder is fundamentally the same with the H.261 encoder, as shown in Figure 2.6. However, some modifications were incorporated in the recommendation to improve the video quality at low bit rates, including:

- The use of new input video formats, as sub-QCIF (128×96), 4CIF (704×576) and 16CIF (1408×1152).[2]
- The use of half-pel prediction in block matching, unlike H.261, where candidate blocks are displaced only to integer pel positions, and motion estimation is coarser.
- Unrestricted search range for motion vectors, allowing better prediction near the frame boundaries.
- Advanced prediction mode, which assigns different motion vectors to each of the four 8×8 blocks comprising a macroblock.
- Arithmetic coding rather than Huffman encoding.
- PB-frames, which are combinations of a forward and a bidirectionally predicted frame, see the next paragraph for definition of P and B frames.

These features make the codec much more complex than the respective H.261 implementations, but, with the continually improving hardware available at continually decreasing prices, they can be justifiable, especially for mobile users and terminals.

[2]Frame dimensions are with respect to the luminance samples, unless otherwise stated.

2.4.2 The ISO MPEG Family

The time line for the ISO video coding standards starts with the creation of the MPEG group, commissioned to standardise video and associated audio coding, in 1984. The MPEG1 standard was finalised in the early 90s, and was intended to storage and offline playback digital video applications at bitrates up to 1.5 Mbps. In the mid-90s the MPEG2 standard for digital broadcasting of TV and HDTV quality video sequences at higher bitrates was released. Currently, the MPEG4 standard, providing a framework for integrated visual communications and interactive multimedia is being finalised.

The group has thoroughly specified the *syntax* of the MPEG bitstreams, leaving the encoder design to the implementors. The MPEG standards have three key parts: Systems, Video and Audio. The systems part concerns the synchronisation and multiplexing of video, audio and auxiliary data. Only the video part will be described in the following paragraphs.

MPEG1

The input pictures for MPEG1 [ISO91] are progressive, *i.e.* non-interlaced, Standard Input Format (SIF) images. There are two versions of SIF, one corresponding to the American NTSC with $352 \times 240 \times 29.97$fps, and one to the European PAL with $352 \times 288 \times 25$fps. These provide quality that is broadly similar to that of VHS video [RR96].

There are three types of coded pictures in MPEG1: I, P, and B frames. I frames, or intra-coded frames, are partitioned to macroblocks and encoded with virtually the same way as in H.261 intra coding. P frames, or forward predicted frames, are motion compensated and encoded similarly to the H.261 inter mode. B frames, or bidirectionally predicted frames, are encoded using interpolated motion estimation between the previous in time I or P picture (forward prediction), and the next I or P frame in the sequence (backward prediction). B pictures are never used as a reference for predicting other frames in the sequence.

The temporal positioning, and consequently the display sequence of these frames is shown in Figure 2.7. However, the sequence of pictures that is transmitted is different, in order to limit the decoder's buffer requirements to two frames only. For example, a sequence of pictures:

$$I_0 B_1 B_2 B_3 B_4 P_5 B_6 B_7 B_8 B_9 P_{10} B_{11} B_{12} B_{13} B_{14} I_{15} \ldots$$

should be transmitted or stored as:

$$I_0 P_5 B_1 B_2 B_3 B_4 P_{10} B_6 B_7 B_8 B_9 I_{15} B_{11} B_{12} B_{13} B_{14} \ldots$$

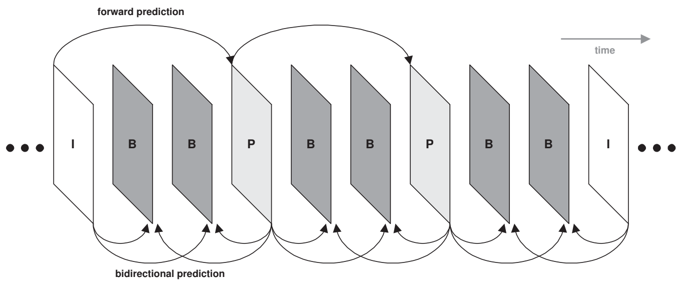


Figure 2.7: The display sequence of MPEG-encoded pictures.

This out-of-sequence transmission or storage pattern, has made the delay requirements for MPEG streams higher than H.261's, which reflects that real-time communications was not one of the design considerations, as in H.261. The bitstream hierarchy is similar to the H.261, with the slice layer being the equivalent of the GOB layer, and the Group of Pictures (GOP) layer added, which consists of all the frames between two I frames, the first of the two included. MPEG1 supports rate control to avoid buffer overflows, as described in previous paragraphs.

MPEG2

MPEG2 [ISO93] was developed to efficiently encode television sequences, or CCIR601 video. It supports various combinations of input formats and bitrates, with the three main formats as in Table 2.1. As a superset of MPEG1, it provides backwards compatibility, but it is targeted at applications well above 2 Mbps.

While MPEG2 is largely based on the MPEG1 encoding techniques, it specifies a number of enhancements, including:

- Support of interlaced video in addition to progressively scanned video.
- More chrominance sampling modes, including 4:2:2 and 4:4:4.
- Support of scalable video coding, implementing spatial, temporal, SNR and data partitioning scalability, which are described in later chapters.

Application	*Input video*	*Coded bitrate*
Home entertainment video	352×240, 30fps	1.5 Mbps
Digital television	720×486, 30fps	5–10 Mbps
Extended definition TV	1920×1080, 30fps	30–40 Mbps

Table 2.1: MPEG2 main bit rates and applications.

MPEG2 does not prescribe these capabilities to the encoder, but allows the selection of some or all of them, through combinations of profiles and levels. This provides the very desirable flexibility that MPEG1 lacked, and broadens the range of potential applications for MPEG2.

MPEG4

Unlike MPEG1 and MPEG2 which had one clear application in mind when they were developed, MPEG4 is a much broader umbrella-type standard. Initially it was aimed at very low bitrate communications, however its scope was expanded to be much more of a multimedia coding standard. The additional functionalities of MPEG4 include [Tal98]:

- The ability to efficiently encode mixed media data, such as video, graphics, text, images, audio, and speech, with the introduction of the audiovisual objects concept.
- The ability to create a compelling multimedia presentation by composing these mixed media via scripts.
- Error resilience through content-based scalability, resynchronisation, data partitioning, reversible variable length codes, and header extension codes, discussed in Chapter 4.
- The ability to encode arbitrarily shaped video objects.
- Multiplexing and synchronisation of the data associated with these objects, so that they can be transported according to Quality of Service (QoS) constraints over networks.
- The ability to interact with the audiovisual scene generated at the receiver end.

These functionalities, are specified in an open toolkit approach, allowing for the incorporation of emergent coding algorithms. Applications will be able to select and download additional tools as required.

2.5 Summary

In this chapter, an overview of the most popular video coding techniques and their applications to standards was presented. Techniques that achieve compression by exploiting the spatial, temporal, statistical and subjective redundancy inherent in video sequences have been developed in the recent years.

Of these, one of the most effective, widely tested and mature is motion compensated DCT video coding, which is applied in most of the described standards, tabulated in Table 2.2. These include the communications-oriented ITU standards, H.261 for ISDN transmission, and H.263 for video delivery over PSTN or mobile networks. ISO MPEG standards are more influenced by the broadcasting, signal processing and computing industries, but with the rapid expansion of the Internet, more communications aspects are inserted to them, as MPEG4 shows.

Organisation-Standard	*Bit rate*	*Intended network platform*
ITU-H.261	64k–2 Mbps	ISDN
ITU-H.263	14.4k–56 kbbs	PSTN, IP-based, mobile
ISO-MPEG1	Up to 1.5 Mbps	Offline storage
ISO-MPEG2	2M–15M Mbps	Digital Broadcast TV
ISO-MPEG4	8k-35 kbps	Interactive multimedia, mobile

Table 2.2: Video coding standards at a glance.

Chapter 3

Video Transmission Over Packet Networks

3.1 Introduction

Digital video, and generally multimedia, are presenting most of today's network technologies with new challenges, unforeseen when they were designed. With the exception of Broadband ISDN (B-ISDN) networks based on ATM, which are a relatively new development, the other popular network technologies were either dedicated to a specific type of analogue or digital information, for example speech or TV signal, or were designed for the delivery of computer generated data, as email messages and text or binary files. Of course, there have been work-arounds for the delivery of different information types, for example, modems enabled the analogue speech PSTN to deliver digital data, but these are only half-measures not applicable for resource and QoS-demanding applications as multimedia.

Currently, the most promising digital video delivery network technologies are the Internet and ATM-based[1] networks. In the recent years, the Internet has emerged as one of the most important candidate platforms for the delivery of multimedia information. Its global presence and exponentially increasing user base, has driven the proliferation of both applications and content provision in it. Furthermore, it has attracted attention from worldwide research and evolves rapidly to a multi-capable network, accommodating information types that were not in its initial scope. ATM, on the other hand, as a later design, provides flexibility for the efficient delivery of information with differing requirements. Internet and ATM technologies are

[1]ATM is better described by the terms cell switching or cell relay, but, for the sake of simplicity, it will be included in the networks referred to as packet switched.

not mutually exclusive, as it is possible to implement the former over the latter.

In this chapter, the network services will be discussed and assessed from the user's point of view, *i.e.* the QoS will be the main criterion. The focus will be on the Internet technologies, however, the ATM capabilities will also be discussed for comparison. Although not strictly required, a familiarity of the reader with the principles of computer networks, such as the OSI seven-layer stack, would make the chapter more apprehensible.

The outline of the chapter follows. After a taxonomy of digital video communications applications, their requirements as far as it concerns the network service are presented. Last, the current and emerging network technologies are described and assessed with respect to the offered QoS.

Classification of Video Communications Applications

From the networking aspect, digital video applications can be categorised to point-to-point and point-to-multipoint applications, the latter including multicast and broadcast applications. Another classification distinguishes between real-time applications and non-real-time applications. Real-time implies that the pre-recorded or live information is transmitted from the source to the sink(s) and is displayed at approximately the same time. In other words, the receiver does not need to download and buffer the whole video information before displaying it.

Real-time transmission is a prerequisite of interactive applications such as videoconferencing and personal communications. Obviously, only real-time video communications as they were defined above are in the scope of this document, and these will be examined in the following sections. In the Internet context, such applications are also called *streaming* applications, especially when they do not involve two-way video communication.

3.2 Requirements of Digital Video Transmission

The basic points that should be fulfilled by the network service in order to provide QoS for video communications are examined in the following paragraphs. These include the throughput, delay and error characteristics of the network service. Extending the scope of QoS to the advanced functionality of the service, multicasting capability is also examined.

It has to be stressed here that the concept of QoS is applying to end-to-end connections, in the particular case multimedia connections. Video is just one of their components, however, it is the most demanding in terms of network service and it has the greatest impact on the overall QoS.

3.2.1 Throughput

The available data transmission rate offered by the network service is critical for compressed video quality. In the lossy compression techniques described in Chapter 2, the output bitrate of the encoder, and consequently the compression ratio, is highly dependent on the information 'thrown away' at the quantisation stage. Larger quantiser steps lead to better compression ratio, and poorer quality of the reconstructed images. The latter can be thus used as a measure for the throughput requirements of the network service.

Constant Bitrate Transmission

Compressed video has bursty characteristics, which makes it uneconomical for transmission over traditional videoconferencing or broadcast network platforms, such as ISDN or digital TV. The main reason for this is the disparity between the required bitrate of intra-coded and of motion predicted images. In other words, the former have poor compression efficiency compared to the latter, which require much less bits to be encoded with the same quality.

In order to overcome this problem, data is buffered before transmission, smoothing out the short-term variations in bitrate. However, longer-term variations, due to changes in the spatial and temporal content of the sequence, cannot be smoothed out in this way, unless arbitrarily large buffers are used [RR96]. What is usually applied in these cases is rate control, which has the drawback that for high-motion sequences the quality is degrading significantly, as described in the previous chapter.

Variable Bitrate Transmission

Variable bitrate connections are offered in packet-switched networks, with ATM being the one better supporting it with QoS guarantees, such as peak bitrate and average bitrate constraints. Even with ATM, however, the long-term variations of these constraints, do not alone guarantee the effective transmission of unsupervised, or *open-loop encoded*, digital video. For live video transmission these variations cannot be reliably estimated, and could violate the traffic contract.

3.2.2 Delay

Generally, delay is introduced in all three stages of a video transmission scheme. The first is video encoding. The waveform encoding stage requires a good deal of integer and floating point calculations. Other factors as B pictures introduce delay as well, requiring the buffering of 'future' frames for backward prediction. Encoding delay has been reduced to non-critical levels by the wide availability of powerful hardware.

The second stage is the transmission itself. This delay can be predictable for circuit-switched networks, such as ISDN, or unpredictable in packet-switched networks. For example, in IP networks, packets are delivered in a store-and-forward way using buffering, and whenever congestion occurs, routers can discard packets. It is thus not a rare occasion when packets corresponding to the same video bitstream are following different routes, experiencing different delays, and arriving out-of-sequence at the receiving end.

Last, delay is introduced during decoding, as there is usually buffering of the reference frames for motion compensation. This delay has been kept to non-critical levels, through the reordering of frames before transmission, and the availability of cheap computing power.

In two-way videoconferencing systems, the total, or *transit* delay is critical. A delay of more than a few hundred milliseconds makes conversations and interactivity frustrating for the communicating parts. In one-way communication systems, however, the delay *variation* is more critical, as it has an increasing effect to the buffering requirements at the decoder, and consequently its cost.

3.2.3 Error Rates

Practical communications systems, are not completely reliable in delivering information intact. There are errors and data losses, which affect more or less the quality at the receiver. Uncompressed video information is considered one of the least sensitive targets of transmission errors, as occasional leakages are usually not noticeable in the bulk of information received. Compressed video, on the other hand, is extremely vulnerable to errors or packet loss, as described in the following paragraphs.

The entropy encoding stage of popular video compression schemes uses Variable Length Codes (VLC) to take advantage of long runs of bit patterns. However, this makes the stream asynchronous, and more prone to packet loss or bit errors, if resynchronising markers are not used. Even with their use the error is spatially propagated until the reception of the next marker.

The motion compensation used to reconstruct images at the decoder can provide low quality results if erroneous reference images are used, or if motion vector information is not correctly received. This makes the errors to propagate temporally as well as spatially until the next intra-coded frame appears in the sequence, and, if this is not done in regular intervals, the results could be very frustrating for the user.

3.2.4 Multicasting

Though not a traditional QoS measure, multicast capability is one of the basic requirements for particular video transmission schemes, such as videoconferencing systems. Multicasting is the intermediate case between point-to-point or unicast communications, and broadcast communications. Quoting Fluckiger [Flu95]:

> *"Multicasting is the capability of the network to replicate, at certain internal points, the data emitted by a source. Replicated data should be forwarded to the recipient end-systems which are part of the multicast group so as to avoid segments of the network to be traversed by multiple copies of the same data."*

The resource savings by using multicast transmission are considerable, as shown in Figure 3.1. New versions of IP provide the functionality for multicast connections, but it is not currently implemented in all the Internet routers. Currently there are some interconnected via tunnels multicast-capable islands, forming an overlay network on top of the Internet, namely the Multicast Backbone (Mbone).

The Mbone, as a testbed for multimedia applications, is a glimpse to the future of the Internet. It has been successfully used to broadcast several Internet Engineering Task Force (IETF) video and audio meetings to whoever joining the multicast session. Current research is targeting in improving the QoS for such connections. It is well understood that this task is much more complex than the respective for unicast connections.

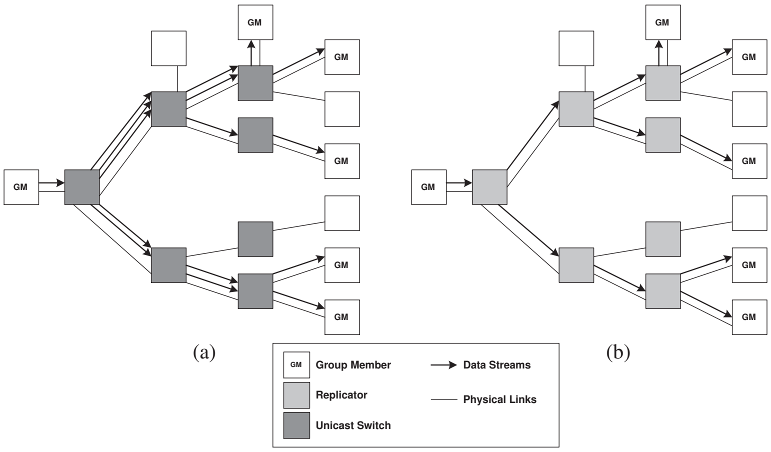


Figure 3.1: (a) Unicast and (b) multicast model for multiparty conferences.

3.3 Network Service Models for Video Delivery

Looking at the transmission problem from the networking service point of view, the individual operations can be classified according to the OSI seven-layer model. For example, video encoding/decoding reside in the Application layer of the OSI stack. The following paragraphs deal mainly with the Transport and Network layers functions,[2] and how they can be customised to satisfy the aforementioned QoS requirements.

3.3.1 Traditional Internet Network Service Models

The Network layer service of the Internet, provided by IP, is often described as a best-effort service. Every packet requesting IP service, is given a 'fair' chance to reach the destination by the routers,[3] on the basis that there is a known route only, irrelevant of traffic conditions along that route [CWH+96]. The probability of congestion is thus very high when there is high network service demand, forcing packet dropping at overloaded routers. IP is connectionless, as it treats each packet individually, and not as a part of a stream of information forming an end-to-end connection.

Over IP, two main Transport layer protocols are used: the User Datagram Protocol (UDP) and the Transport Control Protocol (TCP). UDP resembles little of a traditional Transport layer functionality, as it just extends the best-effort, or datagram, service of IP, providing a lightweight unreliable way of transmitting non-critical or bulk information. No transport layer action is taken for lost UDP packets.

On the other hand, TCP enhances the IP datagram service, providing an end-to-end connection-oriented service to deliver reliably streams of information to the destination. Lost TCP packets are all acknowledged and retransmitted, providing a lossless service to the Application layer functions. Furthermore, TCP provides flow control to make sure a fast sender cannot swamp a slow receiver with more messages than it can handle [Tan96].

The TCP network service model is working finely for traditional information exchange between computers, such as file transfer. However, in real-time video transmission, the receiver is a human and not a computer. Humans cannot tolerate long pauses during multimedia playback for error recovering through retransmission, or excessive delays due to congestion. They do tolerate temporary quality degradation as a result of packet loss,

[2] The OSI Session and Presentation Layers can be safely ignored for the following discussion, as they do not have a direct effect to the QoS, and they do not have an Internet equivalent.

[3] Routers are essentially the switches of IP networks.

as long as the overall quality is kept in acceptable levels.

This has led to the use of UDP in early streaming multimedia applications, but its functionality was inadequate to provide any QoS guarantees, and its packets are very prone to loss. Therefore, the IETF has standardised supplementary architectures for the delivery of multimedia over the Internet, which are discussed in the following paragraph.

3.3.2 Evolution to Multimedia Network Service Models

The Internet is currently being upgraded to provide QoS for real-time applications, a process that involves the development of new protocols for supporting new classes of service. The process is greatly influenced from the ATM design principles and network service models, so it is essential to describe the main concepts of ATM before moving on to the Internet multimedia architecture.

The ATM Paradigm

ATM is using short fixed-sized packets, or cells, which are transported at regular time intervals, so as to minimise the processing time associated with their processing, buffering and routing. It is a connection-oriented network, usually operating at very high speeds.

ATM provides a mechanism for logical connections, called Virtual Circuit Connections (VCC). A VCC is setup between two end users through the network, and a variable rate, full-duplex flow of cells is exchanged over the connection [Sta98]. A Virtual Path Connection (VPC) is a bundle of VCCs that have the same end points, and it helps to restrain the signalling and control cost of multiple connections by grouping them into a single unit, as shown in Figure 3.2.

The real-time services of ATM are classified to constant and variable bitrate services, or CBR and VBR respectively. ATM defines a range of QoS guarantees for its connections. These can be generally described by the term traffic control, and refer to the actions taken to avoid congestion

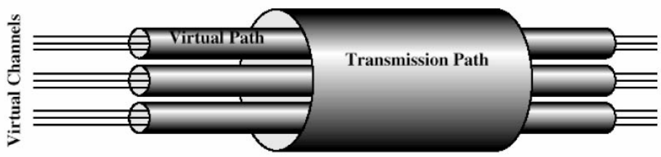


Figure 3.2: ATM connection relationships, source: [Sta98].

conditions or to minimise congestion effects, and include [Sta98]:

- *Resource management using virtual paths,* which is allocating the available network resources in such a way as to separate traffic flows according to their required service characteristics.
- *Connection admission control,* which ensures that a new connection can get the QoS agreed in the traffic contract, otherwise it rejects the request.
- *Usage parameter control,* which monitors established connections to determine whether the traffic conforms to the traffic contract.
- *Selective cell discard,* which discards low-priority cells tagged by the data source or the usage parameter control.
- *Traffic shaping,* which regulates and smooths out the traffic flow so it does not exceed the agreed QoS parameters.

Multimedia over the Internet

The proposed changes to the Internet are made in a supplementary way, without affecting its current basic best-effort service. At the network layer, IP has been upgraded to IPv6, which supports QoS by a priority value in its header, and by flow labels, identifying individual data streams [Bra97a]. Over IPv6, the Real-time Transfer Protocol (RTP) has been introduced supplementing UDP, as depicted in Figure 3.3. Newer developments include the Real-Time Streaming Protocol (RTSP), which supports 'remote control' operations for real-time audio/video sessions.

RTP was developed to provide real-time service among a number of participants in a session. The latter is defined by the RTP port number,

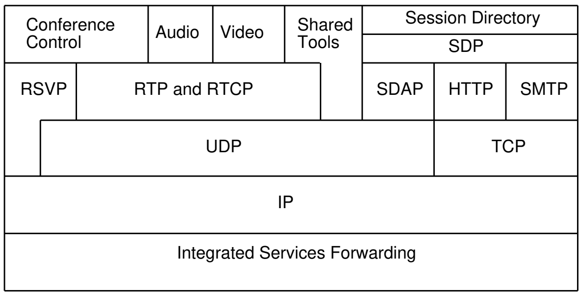


Figure 3.3: The Internet multimedia protocol stack, source: [CWH+96].

the RTCP[4] port number, and the participant IP addresses. One of the main features of RTP is that it supports multicast connections by including a source identifier pointing to the member of the multicast groups that generated the data. It also includes a timestamp for synchronisation at the receiver, and a payload format identifier.

Identifying a series of packets as a data flow, is the first step in order to provide QoS functionality, similar to the ATM. The second step is to manage the network resources and guarantee delivery within QoS constraints, such as throughput and delay. This is accomplished by the Resource ReserVation Protocol (RSVP), which permits applications to reserve resources for an IP packet flow without establishing an explicit connection [Bra97b]. RSVP supports multicast efficiently by making the receivers responsible for requesting and reserving resources according to their QoS requirements.

With the development of the above mechanisms, it has become possible to the IETF to define classes of service for the Internet, which are roughly in line with those developed for ATM and presented in the previous paragraph. The most important of these are controlled-load service and guaranteed service. The former does not guarantee bandwidth or delay, as the latter does, but relies on admission control functions to avoid network overload and congestion [Bra97b]. It targets adaptive real-time video applications, while guaranteed service is more suitable for applications with hard delay requirements.

3.4 Summary

This chapter dealt with efficient ways of delivering video over packet networks, particularly over the Internet. The requirements of real-time multimedia do not map well to the current IP service, thus, new protocols and services, inspired from ATM service models, were developed, and are currently deployed. This evolution is taking place so as to minimally affect the existing popular Internet services.

[4]The Real-Time Control Protocol is transferring control information, such as QoS status reports, which are related to the RTP data stream.

Chapter 4

Reliable Video Communications

4.1 Introduction

All practical communications systems introduce errors during the transmission of information. Errors alter or delete parts of the information and usually have a deep impact on the quality of the received information. Their effect depends on the extent and the significance of the error-struck information segment, and in many cases they deem the transmitted message unusable.

For packet networks particularly, errors are usually modelled as packet loss. This could refer to the actual loss of the packet in networks that discard packets when congested, and to the reception of a packet altered by the network infrastructure above a certain degree. In real-time video communication, the packet loss concept is further extended to packets arriving too late to be displayed at the decoder. For example, in streaming video applications over the Internet, delay is the biggest source of packet loss.

The outline of the chapter follows. The following two introductory paragraphs are dealing with the effects of packet loss on coded video and the model applied for error control, respectively. Next, some of the techniques which have been developed in order to protect and recover from packet loss, are presented. They are categorised according to the place where error control or concealment is performed.

4.1.1 Effect of Packet Loss on Compressed Video

Compressed video is extremely vulnerable to transmission errors, as discussed in previous chapters. First, in intra-coded pictures, a packet usually holds a good deal of information, spanning over a region of the picture. Loss of the packet causes the whole region to be unavailable to the decoder. The results aggravate if appropriate measures, for example resynchronisation headers and location identifiers, are not used to prevent loss of synchronisation at the decoder leading to spatial propagation of the error.

Second and last, interframe prediction is causing spatiotemporal propagation of potential errors, as depicted in Figure 4.1 for the case where a H.263 GOB is erroneously received. The error decays over time due to the leakage in the interframe prediction loop, see also Figure 2.6, page 15. Intra-coded frames terminate the effect of such errors, but, in practical low-bandwidth transmission schemes such as H.263, they are inserted in the sequence only after a large number of inter-coded frames. In such cases techniques as conditional replenishment are used, as described in the following sections.

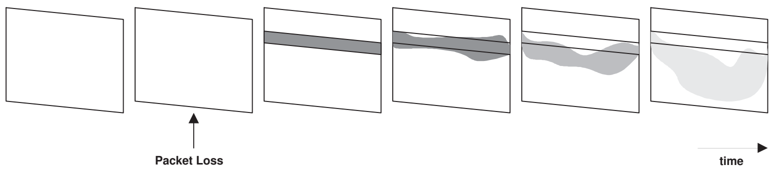


Figure 4.1: Temporal error propagation in H.263, source: [FGV98].

4.1.2 Formulating Error Resilience

The functional diagram depicted in Figure 4.2 is introduced in order to describe the techniques for enhancing the robustness of real-time video communications systems. The familiar from the previous chapters waveform and entropy encoding functions are appearing in the source encoding stage. Before the actual transmission, transport coder functions, such as packetisation, channel coding, and transport-layer control, are performed to convert the bistream to data units suitable for transmission [WZ98]. In Figure 4.2, double arrows are used to indicate that it is possible to have feedback information from the receiver to the sender.

Generally, redundancy could be added at each of the functional stages to help the receiver recover lost packets or conceal errors. The problem of error control and concealment can be formulated as designing a pair of source encoder/decoder and transport encoder/decoder so that the error

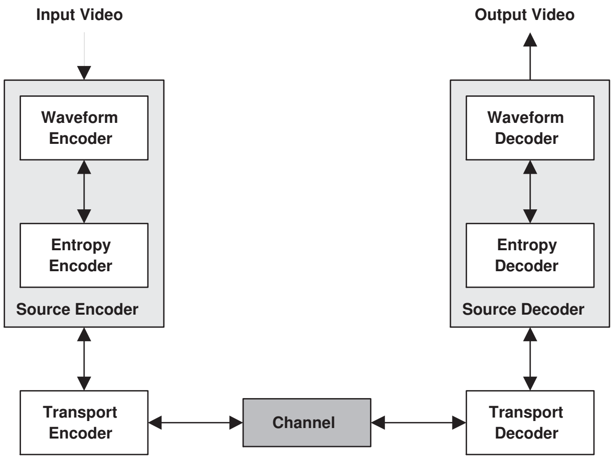


Figure 4.2: A functional block diagram for a video communications system, source: [WZ98].

distortion at the decoder is minimised with a given video source model, channel bandwidth, and channel error characteristics [WZ98]. Obviously, the solution to this problem is not straightforward.

4.2 Sender Based Error Control

These techniques are also referred in the literature as forward error control or concealment, and they could be generally described as preventive. In other words, should a transmission error occur, the decoder is left alone to deal with the problem, by exploiting the concealment redundancy added at the coding stage. This can be done either by exploiting niches that standards intentionally left to add such information, or by slightly modifying standards and providing an error-resilient version of them. A brief overview of the techniques follows.

4.2.1 Robust Waveform Coding

The traditional waveform encoders were designed to maximise compression efficiency. However, eliminating all the statistical and subjective redundancy of the video sequences makes error concealment at the decoder very difficult and the bitstream more sensitive to information loss. Error-resilient waveform encoders apply single-channel and multiple-channel techniques, which are presented in the following paragraphs.

Single Channel Techniques

The first approach in this category is to intentionally add auxiliary information to the bitstream. The most illustrating example of this technique is the concealment motion vectors mode of MPEG2, an optional feature designed to provide robustness in error-prone channels. In this mode, motion estimation is performed for I frames, as well as for P and B frames. The respective motion vectors are sent and used by the decoder to motion compensate I-frames if errors are detected during their normal decoding.

A second approach is to restrict the prediction domain. This is applied in the independent segment decoding mode of H.263, where spatial and temporal prediction is confined within each GOB. With this way, the effect of error propagation, is reduced significantly compared to the one illustrated in Figure 4.1, as shown in Figure 4.3. The drawback is that the compression efficiency is reduced as well, as restricted prediction does not perform well compared to unrestricted.

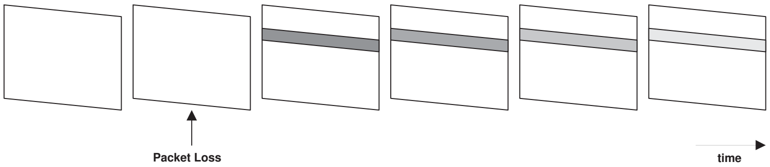


Figure 4.3: Temporal error propagation in H.263 Independent Segment Decoding mode, source: [FGV98].

Multiple Channel Techniques

Layered video coding, or, according to the MPEG terminology, scalable video coding, partitions the bitstream to more than one layers. It is assumed that there is a channel for the guaranteed transmission of the base layer, while the the other layers, which are transmitted via non-guaranteed delivery channels, are progressively enhancing the decoded video quality. If errors occur, the base layer is always providing a low-quality version of the video sequence.

The partitioning usually involves temporal and spatial subsampling or coarser quantisation parameters at the base layer, for temporal, spatial and SNR[1] salability, respectively. This assigns a low-quality version of the sequence to the base layer, leaving the fine details to the other layers. A similar partitioning is including the lower spatial frequency transform coefficients in the base layer, leaving the higher frequency components for the

[1]Signal to Noise Ratio.

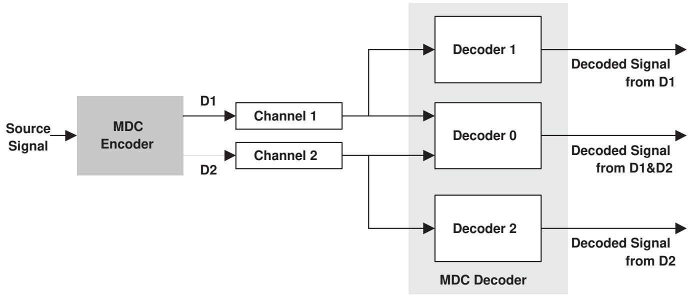


Figure 4.4: A two-description MDC codec, source: [WZ98].

enhancement layers. Last, data partitioning, is the inclusion of the most important and sensitive information, such as motion vectors, in the base layer, leaving the less sensitive data for the enhancement layers.

The guaranteed delivery base layer channel assumption on which layered coding is based, is not valid for many packet networks, such as the Internet. For these networks it has been shown that Multiple-Description Coding (MDC) works better. This coding scheme assumes that there are several parallel channels between the source and the destination and that each channel may be temporarily down, or suffering from long burst errors. It transmits different versions, or descriptions, of the same source signal over these channels. At the destination, depending on which descriptions are received correctly, different decoders are invoked, as illustrated in Figure 4.4.

The MDC encoders and decoders are designed in such a way, as to provide acceptable quality of the reconstructed images, with one only of the descriptions, and that incremental improvement is achievable when receiving more descriptions. Therefore, every description should carry sufficient information for the original signal, implying that there will be overlap, or correlation, in the information contained in different descriptions [WZ98]. This reduces the compression efficiency of the scheme, but eliminates the need for use of other error-control bits or latency, which an error-resilient single description codec might require.

4.2.2 Robust Entropy Coding

The most simple of the robust entropy coding techniques is self-synchronising entropy coding, which uses one codeword as a synchronisation marker and inserts it at regular intervals in the bitstream. An error-resilient entropy coding example are the Reversible Variable Length Codes (RVLC) which are used in MPEG4's error-resilient mode. These enable the backward decoding of VLCs as well as the forward, enabling the decoder to continue until the last backwards decodable VLC on the event of an error while it was decoding in the forward direction.

Forward Error Coding (FEC) could be also classified here, as it adds redundancy after the entropy encoding stage. FEC has been long used in data communications for error detection and correction. It can be used in video communications for recovery of lost packets, but its use should be judicious, as it occupies a considerable portion of the usable bandwidth. Techniques that change the encoding design parameters so as to provide a bitstream which performs optimally for given channel error characteristics, called joint source and channel coding techniques, could be also applied at this stage.

4.2.3 Robust Transport Coding

Techniques that are used at the transport coder, are mainly concerned with the packetisation of the video stream. This should ideally decorrelate information stored in different packets, so that errors are effectively isolated. For RTP, payload formats have been released by the IETF, for all the major video coding standards described in Chapter 2.

For example, in RFC 2190 [Zhu97], the RTP payload format for H.263 is defined. The general format of a video packet is illustrated in Figure 4.5. The H.263 payload header contains essential information such as predictor, entropy encoder, and quantiser parameters, which make the decoding of each packet nearly independent of whether its predecessors or successors have been correctly received and decoded. The task of decorrelating information is not always straightforward; particularly, in the H.263 example the RFC suggests intervention in source coder parameters, such as picture type and prediction range decisions, in order to accomplish this task.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                 RTP header                                    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                 H.263 payload header                          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                 H.263 bitstream                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 4.5: The layout of an RTP H.263 video packet, source: [Zhu97].

The above RTP packets contain data from several blocks, usually whole GOBs. Thus, a loss of one packet affects a contiguous region of the picture, with frustrating results. Other schemes apply interleaved packetisation, by which successive image blocks are stored into nonadjacent packets. However, this increases the complexity not only of the transport codec, but also of the source encoder, as it has to confine block-level prediction to only the blocks within a packet.

Other techniques applied at the transport coder stage, use the idea of layered coding, replacing channels with transport protocols. According to them, the base layer, containing the most important information, could be transmitted using the reliable TCP, while all the remaining information could be sent with UDP.

4.3 Receiver Based Error Concealment

In order to take appropriate action against errors at the receiver, they have to be detected first. This is usually done by examining packet headers. For example, RTP provides a minimal service,[2] as its packets have a sequence number, and timestamp information, which make the detection of lost packets automatic. Sequence numbers are used in the RTP payload data for specific applications as well, for example, there is a GOB sequence number in the aforementioned H.263 payload header.

Once an error is detected, the decoder should resynchronise itself. This is straightforward in packet loss, as the next valid packet is identifying the next valid data, in contrast to bit errors, where special techniques such as FEC have to be applied. The next step is to recover the lost packet, possibly using auxiliary information from the encoder. If this is not possible, an attempt to make the error less noticeable to the human viewer, or *conceal* the error, takes place.

The main principle of error concealment at the receiver is the exploitation of the limitations of the human visual system. For instance, people tolerate more distortion to the high spatial frequency components, corresponding to detail and edges in an image, than to their low frequency counterparts. For typical video sequences, most image data is varying smoothly in the spatial and temporal direction, thus low-frequency components are predominant, which can be exploited as well. The results of error concealment are highly dependent on the forward error control techniques used, namely the type of the concealment redundancy inserted in the encoded bitstream.

[2] As already described, RTP was intentionally designed with minimal transport functionality, according to the Application Level Framing (ALF) guidelines.

Two main ways exist for the techniques applied for postprocessing at the decoder: prediction and interpolation. These are described in the following paragraphs.

4.3.1 Prediction-Based Techniques

A good estimation of the lost blocks, can be provided by using the spatially corresponding blocks of the previous frame. This technique performs well at low-motion scenes. If the motion vectors for the lost blocks are received correctly, for example in MPEG2 concealment vectors mode, the results are much more improved by using the motion compensated blocks of the previous frame. This technique has been shown to work very well in conjunction with data partitioned layered coding, where the motion information is transmitted with the base layer.

Other techniques that exploit the spatial correlation, as well as the temporal in order to estimate the lost blocks, are the maximally smooth recovery techniques. Briefly, they apply spatiotemporal measures in the neighbourhood of the lost blocks, to obtain an estimation that minimises them, so that the resulting video signal is as smooth as possible.

4.3.2 Interpolation-Based Techniques

Lost DCT coefficients can be determined by interpolation among the corresponding frequency coefficients of spatially adjacent blocks. Two things have to be noted on interpolation. First, it can only be applied within intra-coded blocks, as the inter-coded blocks contain prediction errors, which do not exhibit high levels of correlation. Second, it is preferably applied to the DC and the low-frequency AC coefficients, as the smoothness assumption is not valid for the high frequency coefficients.

Motion vectors can be determined through interpolation. Various combinations of reference motion vectors for the interpolation can be used. One is motion vectors from the corresponding block of the previous frame, assuming rigid motion for the video sequence. Another is the motion vectors corresponding to the spatially adjacent blocks, assuming a monotonic distribution for the vector field. Combinations of the above with averaging and medians are also used.

4.4 Interactive Error Control and Concealment

The postprocessing error concealment techniques presented in the previous section are adequate for providing an acceptable video quality at the decoder only when the packet loss rate is very low. These techniques perform much better when they are used in conjunction with forward error control techniques. Obviously, the interaction between the encoder and the decoder should take place over a two-way communications channel. This prerequisite cannot be satisfied for real-time video applications in specific network architectures, such as broadcast or multicast channels, as the backward requests from the decoders would place a great burden on the network, further increasing packet loss.[3]

For real-time video communications, plain buffering and retransmission are not applicable due to the latencies incurred. One of the simplest approaches is that the decoder acknowledges the loss of specific packets to the encoder and requests the transmission of an intra-coded frame to stop error propagation. This corrects the error for the future frames, after a time period which can be considerable if the round-trip delay is large. However, it is not economical, as the low-compression-efficiency intra-coding might be invoked for every — small or large — quality degradation due to packet losses. Furthermore, low-bandwidth video coding standards such as H.263 are sending whole intra-coded frames on very rare occasions during the video sequence.

More sophisticated schemes are sending back information on the specific region of the picture that has not been correctly received, and the encoder intra-codes and sends only the corresponding macroblocks. These selective encoding schemes have been further elaborated by using techniques such as error tracking. The idea behind error tracking is to simulate at the encoder the loss that the decoder experiences, according to the feedback information it receives. Then, motion estimation is applied to the locally reconstructed error-struck images, and the updated motion vectors and prediction errors are sent as in the normal, compression efficient, interframe mode. On their reception, the receiver can effectively reconstruct the new correct video frames.

Other solutions include H.263's reference picture selection mode, in which both the encoder and the decoder have multiple frame buffers, which are selectively used depending on the nature of the messages exchanged between them. It comprises sets of coding parameters which trade compression efficiency for error resilience when the loss rate is high, and vice-versa.

Modern Internet video streaming applications use techniques such as

[3]However, this argument is currently challenged, as discussed in page 45.

prioritised multicopy retransmission [WZ98]. As their name suggests, these schemes combine prioritised layered coding, with multiple channel retransmission for the base layer. Since the transit delay is not crucial for one-way video communications, these techniques provide flexible tradeoff between delay and reconstruction quality.

4.5 Summary

In this chapter, the techniques used in order to control or conceal transmission errors for video applications were presented. These can be applied at the sender, the receiver, or both of them. Some of them are suitable for bit errors, while others handle more efficiently packet loss. Not all of them are giving the desired results over IP-based networks, which are in the scope of this report. The selection of the most appropriate will be done in the next chapter, along with the decisions on the other parts of the video communications scheme.

Chapter 5

Decisions

5.1 Introduction and Problem Formulation

The previous chapters reviewed the technology applied in video coding, communication, and robust transmission. This chapter aims to select appropriate technologies for a reliable video communications scheme over packet networks, particularly IP-based networks.

Therefore, this chapter deals with decisions. One way of taking decisions is to adopt a centralised pyramidal model, following top-down or bottom-up approaches. Another option is that of a serialised procedural approach. And yet another is a fully parallel model, in which concurrent decisions are taken for all the parts of a system.[1]

For this work, a combination of the procedural and the parallel model was considered simpler and easier to formulate. Particularly, the design problem is formulated in three phases. The first is the selection of the video sequences for the scheme, in other words, this is the system input decision stage. The second is the internal system decisions stage, which comprises the selection of a video codec, the network model for the transmission, and the techniques to provide robustness. The third deals with the selection of appropriate methods of assessing the quality of the system output. The decision-taking model is illustrated in Figure 5.1.

In the following sections, the criteria that have to be set and the requirements that have to be met for each of the parts depicted in Figure 5.1 are discussed, along with an evaluation of the available technologies. In the end of each section, the most appropriate options for the target reliable video communications scheme are suggested.

[1] Of course, this list of options is not exhaustive.

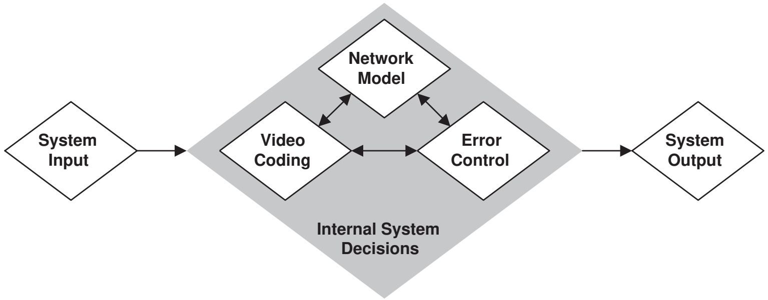


Figure 5.1: The decision-taking model adopted for the reliable video communications scheme.

5.2 Choosing Input Video Sequences

The choice of input video sequences determines the range of the potential applications for the scheme. The attributes that are important here, are the picture size and frame rate of the sequence.

Picture sizes that are acceptable for entertainment video, are at least the size of current PAL and NTSC formats. These are approximately the size of 4CIF, which has both horizontal and vertical dimensions double the size of CIF, see page 13 for details. For personal communications applications, picture sizes of CIF and even QCIF provide adequate resolution for head and shoulders images, and fit better to small screen mobile terminals.

Frame rates in films and television are in the range of 24-30 fps. If lower values are used, the eye can distinguish the succession of images, which causes a frustrating flickering effect. However, for smaller resolutions, and for sequences with low motion and not major scene changes due to camera pan and zoom, smaller frame rates can be used without notice from the user.

The selection of the above attributes has to be judicious for two main reasons. First, the bandwidth requirements of high quality video could become unbearable for the network infrastructure. Second, the real-time decoding and display of TV-quality pictures in current desktop or mobile terminals is not a trivial task yet, and has a direct impact on the decoder's cost.

Taking into account these facts, and as this work will evaluate mainly schemes for reliable personal communications, the input sequences that have been selected are all head and shoulders QCIF images, sampled at 15 fps. Three sequences with different motion and spatial detail characteristics were selected in the conducted experiments, described in the next chapter.

5.3 Choosing Video Codec

The above selection of input video sequences dictates the selection of the video coding standard. The main candidates are MPEG4 and H.263, as MPEG1 and MPEG2 are aimed at the coding of at least TV-quality images for entertainment video, and H.261 considered more suitable for circuit switched networks, such as ISDN.

MPEG4 was initially designed for very low bitrate communications, but on the way, it supported much more input formats. Now it is envisaged more as a complete integrated multimedia platform, which supports interactions between all the media streams. Examples include overlay text and graphics information which is changing according to the displayed video context, and could point to further information via the Web.[2] MPEG4 is still in the standardisation process — in the final stages, though — and its main architecture has been changed from a simple scheme implementing cutting-edge technology to a versatile and highly modular scheme which accommodates a wide range of interactive multimedia applications.

On the other hand, H.263 is much more compact as a concept and has a restricted range of applications. Very low bitrate applications are destined to mobile multimedia applications, and there is a global research effort on adapting H.263 for mobile networks, which has resulted to an updated specification called H.263+. However, it emerges as a more efficient and economic candidate than MPEG-like schemes for personal communications and videoconferencing over wide area computer networks, as well. In fact, the standard caters for different network requirements, as the implementations are free to include or omit the optional parts of it.

From the above discussion, it has become clear that the H.263 standard was selected for video coding. This selection, has practical advantages as well, as there are public domain implementations for the specific standard, available at the source code level. One of them, the *Telenor TMN5*, was used for the implementation stage of this work.

5.4 Choosing Network Model

It has been stated from the early chapters that IP networks will be the target platform for this work. Therefore, this decision deals with the modelling of the network service rather than the selection of the network technology itself. Maintaining consistency with the previous chapters, QoS will be the main

[2]The Web, or WWW for World Wide Web, is the set of interlinked documents throughout the Internet, accessible through point-and-click applications known as browsers.

criterion for this decision.

As far as it concerns throughput, the current network service is considered generally adequate, provided that H.263 is used. The latter, is designed to provide video services over PSTN-level bandwidths, so the access network is less than an issue. However, from the QoS point of view, the service is definitely inadequate, without any minimum or average throughput guarantees. A solution could be the adoption of Internet multimedia architectures, however, it requires a considerable effort to implement them in all the Internet routers.

Similar remarks can be made for the delay requirements. Moreover, depending on the error control techniques that will be selected in the next paragraph, the applications will experience different latencies. For example, if interactive error control is used, increased delay is one of the main consequences, and the design should target its minimisation. Note that at this stage, decisions are taken concurrently, according to the model depicted in Figure 5.1.

The simplest approach to model error rates in the Internet is apply random packet loss. This can be arguably simulated as a random dropout of packets just before the decoding. However, the use of techniques such as RSVP to allocate resources and provide a level of QoS makes the simulation of packet loss much more complicated, as routing decisions are taken to minimise error rates. Modelling packet loss is alone a field of ongoing research, with divergent results. Therefore, and for the sake of simplicity the initial random model is usually applied, and has been selected for this work.

Multicasting is currently implemented in 'island' subnetworks but is well on the process of extending everywhere on the Internet. For open-loop encoding,[3] multicast connections can be effectively simulated by unicast equivalents for this work's purposes. However, for interactive sender-receiver encoding and error control schemes, congestion control measures have to be taken to handle the feedback traffic effectively. The requirements of such schemes were considered well beyond the timeline and scope of this project, and were not implemented.

5.5 Choosing Error Control Techniques

Of the several error control schemes examined in the previous chapter, the ones that apply interactive error control and concealment between the encoder and the decoder are those that provide the most robust solution. The

[3]See also page 23.

aforementioned latency drawback of these schemes can be eliminated by adopting the error tracking technique or its variants, which do not interrupt the playback to correct the errors. Instead, they apply error concealment at the decoder, sending feedback details about the error to the encoder, which heals the problem in the subsequent frames.

As discussed in paragraph 4.4, the techniques that apply selective retransmission are considered inappropriate for multicast schemes. This claim is challenged today, with multipoint control units, which could group error control feedback information between the encoder and a multicast group of decoders. Furthermore, sophisticated schemes retransmit only selected regions of the picture, keeping the encoder bandwidth and the retransmitted data below a total data rate, decreasing the packet loss probability.

In addition to the above, packetisation can be applied to isolate the packet loss effect, and reduce error propagation significantly. The packetisation scheme of RFC 2190 [Zhu97] is effectively decorrelating the information stored in adjacent packets, so that the loss of one of them does not affect the correct decoding of the rest. It suggests that whenever possible, segmentation of the H.263 bitstream should be done in GOB boundaries and that the optional resynchronisation GOB headers should be always used.

All the techniques discussed above, are fairly complex to implement, especially the interactive error control scheme. In the next chapter, the main focus will be on the effect of packetisation and receiver-based error concealment to the output video quality. These were much more straightforward to implement, however they remain a subset of the decisions taken in this paragraph.

5.6 Choosing Output Quality Evaluation Measures

The ultimate judgement on the output video quality is made by the human viewer. However, subjective evaluation of images has a number of technical and practical drawbacks. The former include the inability of human observers to maintain a consistent evaluation for longer than a few seconds sequences, as they tend to regard the error and coding-related artifacts as natural after some exposure to them. Practical drawbacks are that the process of evaluation is time-consuming, non-automated, and has to be done with a large and different in every test sample of observers in order to give reliable results.

Therefore, analytical methods are usually preferred to conduct comparative analysis of video coding and transmission schemes. There are drawbacks here as well, as it is difficult to quantify how better is one scheme from another, by relying only on these measures, and there are certain images that mislead such statistical evaluations. Thus, subjective testing could be used in conjunction to the analytical evaluation, to provide the most reliable results.

The most common measure of the distortion of an image relative to a reference image, is the Peak SNR (PSNR), a mean square error (MSE) measure defined as:

$$\mathrm{PSNR} = 10\log_{10}\left(\frac{255^2}{\mathrm{MSE}}\right) \quad (\mathrm{dB}) \tag{5.1}$$

$$\mathrm{MSE} = \frac{1}{MN}\sum_{j=1}^{N}\sum_{i=1}^{M}[f(i,j) - \hat{f}(i,j)]^2 \tag{5.2}$$

where M and N are the horizontal and vertical dimensions of the image, and f and $\hat{f}$ are the pel values matrices for the original and the reconstructed frames, respectively.

Generally speaking, values of PSNR over 40 dB correspond to reconstructed images that are near perfect quality, that is, they seem identical to the original. Practical video coding schemes are producing images with lower average PSNR values, usually in the range of 32–40 dB. It has to be noted, however, that for the same average PSNR, distortions due to transmission errors are much more frustrating than distortions due to video coding. Therefore, some selected reconstructed frames will be illustrated in the next chapter for comparison, in addition to the PSNR statistics.

5.7 Summary

This chapter presented an elaboration to the contents of the previous chapters. Following a comparative analysis of the candidate technologies, criteria were set, and the most appropriate techniques for the target video transmission scheme were selected. A subset of these proposals will be tested in the experiments of the following chapter.

Chapter 6

Experiments

6.1 Introduction

Scientific claims cannot be considered as valid if they are not validated by experimental results. In the previous chapters, there have been expressed several such suggestions, some of which will be proven by the experiments conducted as described in this chapter.

In this area of research, the most popular and time-efficient way of obtaining results on new technologies is software simulation. The reliability of these results relies heavily on the simulation model chosen. Even with careful modelling, it is essential to test the designed schemes before deployment in practical conditions, as the involved systems are very complex, and occasionally exhibit unexpected behaviour. Of course, these final tests are not in the scope of this document.

In the following paragraphs the simulation parameters are described, then the results are presented in tables, charts and sample images, as required. These results are usually comparative and are accompanied by comments where it was considered appropriate. Last, a discussion section is concluding the chapter.

6.2 Configuration

6.2.1 Assumptions

As in most simulations, a lot of compromises and assumptions have been made to model the real-world system. The most important assumptions for the following simulation include:

- *Random packet loss* is assumed for simplicity, as discussed in the previous chapter. A specified percentage of the packets is 'dropped,' that is, they are considered unavailable, before decoding.

- *Sufficient decoder parameters* are also assumed. This implies that the decoder does satisfy real-time requirements, be them buffer size, network access connection, decoding and display speed.

- *Zero packetisation cost,* with respect to bitrate and latency, is assumed. Since RTP packetisation was not implemented in this work, all the presented bitrates are not including the additional load caused by packet headers.

6.2.2 Input Sequences

The video sequences used in the simulation were three. They are all QCIF size, 4:2:0 colour sampling, which means 176×144 pels for the luminance component Y, and 88×72 pels for each of the chrominance signals, C_b and C_r. All images are sampled at 15 Hz, that is, the frame rate is 15 fps.

Therefore, the sequences differ only at the content. The first, named 'Akiyo,' is a typical single person head and shoulders image with medium spatial detail and low foreground motion with stationary background. The second, named 'Mother & Daughter' or 'M&D,' differs only at the point that motion is not restricted in a small region, as it features two subjects. The third, which will be the toughest test for the experiments, is the 'Foreman' sequence, featuring medium foreground and low background motion, as the camera is unstable and finally pans away from the subject's face.

(a) (b)

Figure 6.1: Original frames, (a) Akiyo frame 111, (b) M&D frame 22.

Sample frames of the first two sequences are illustrated in Figure 6.1, and for the Foreman sequence in Figure 6.4a, page 51. Note that these sequences are widely used by researchers for testing video coding algorithms, particularly, they are all MPEG4 test sequences.

6.2.3 Software Development

The video encoding and decoding software used is based on the public-domain H.263 implementation from Telenor. This *C* implementation has the versatility of using simple raw binary files for the input and output sequences. All the software development was done on a BSD-Unix platform, with use of the X11R6 library for video display.

Additions and modifications to the encoder and the decoder implementations were considered as the fastest and simplest way to provide the error control functionality. Despite the neatness and flexibility of a modular design which leaves the existing algorithms intact, it was also considered as time-consuming and very complex. Main enhancements include modification of the picture and macroblock type decision, enabling forced intra-mode for them, and temporal prediction error concealment at the decoder.

Apart from the software developed for the simulation itself, some non-trivial tasks were implemented, including programs for the display of still and moving image sequences, file conversion utilities, and various scripts for the calculation, collection, and graphical representation of the results.

6.3 Tests and Results

6.3.1 Encoding Parameters

The video encoder operates in several modes. Some of them correspond to parameters which fine-tune the resulting bitstream, such as quantisation, motion vector search range. Others are toggling the use of H.263's advanced prediction options, including arithmetic coding, unrestricted block matching search and PB frames.

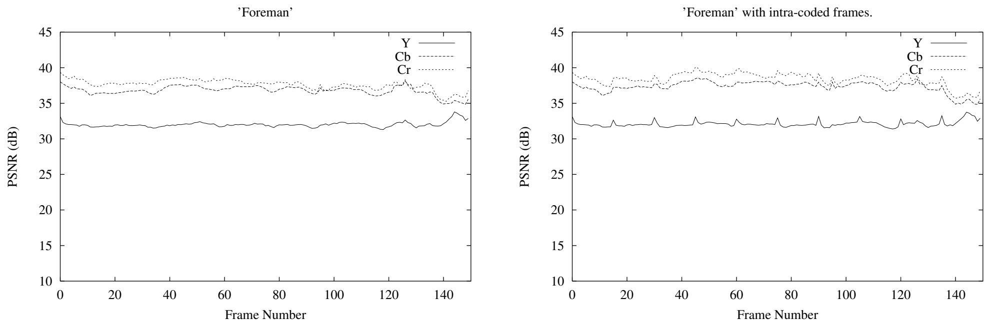
Advanced prediction mode provides considerable savings of bitrate for the same quality, as shown on Table 6.1. However, RTP packetisation is much more complex for this mode, and it was not used in the remaining experiments.

Some remarks have to be noted regarding Table 6.1. The first is that the Foreman sequence is the most demanding in terms of bitrate. This is

Video sequence	*Bitrate (kbps)*	*AP-Bitrate (kbps)*	*Savings*	*Luminance* $\overline{\text{PSNR}}$ *(dB)*	*AP-Luminance* $\overline{\text{PSNR}}$ *(dB)*
Foreman	84.96	65.84	22.50%	32.01	32.22
Foreman-I	95.48	76.55	19.82%	32.11	32.28
M&D	32.24	22.13	31.36%	32.87	32.81
M&D-I	44.59	34.98	21.55%	33.07	33.02
Akiyo	19.19	12.82	33.19%	34.26	34.30
Akiyo-I	33.65	26.41	21.51%	34.55	34.55

Note: the suffix '-I' means that 5% of the sequence is forced intra-coded.

Table 6.1: H.263 normal versus Advanced Prediction (AP) mode.



(a) (b)

Figure 6.2: Colour components PSNR for (a) Foreman and (b) 5% forced intra-coded Foreman.

expected, as in such scenes that feature relatively high levels of motion and the camera is moving,[1] motion estimation errors are much higher. Non-zero motion estimation errors are encoded and transmitted for motion compensation, so they create the observed disparity in bitrate requirements.

The second point is that only the luminance PSNR is used for comparison. This is the common practice in video-related simulations, as chrominance information follows the luminance information in predictions and errors, therefore it exhibits nearly the same statistical behaviour, as shown in Figure 6.2.

Third and last is the suffix '-I'. Normal H.263 bitstreams are using intra-coding only in the first frame of the sequence. These '-I' bitstreams are formed by forcing the encoder to insert at least 5% intra-coded information,

[1]Camera movement can refer to physical movement (panning), but also to lens movement (zooming).

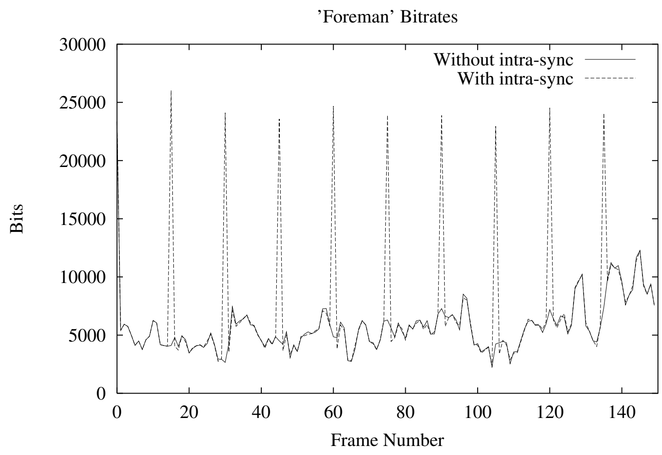


Figure 6.3: Number of bits per encoded frame for Foreman.

in order to make the stream more error-resilient. This has the effect of increasing the required bitrate, as shown in Figure 6.3, without significant gain in the decoded picture quality for lossless transmission, as Table 6.1 suggests.

Another issue in the encoding is the use of GOB markers for resynchronisation. If they are used, packet loss affects only a GOB, that is, a row of macroblocks in the QCIF frame. The results if GOB resynchronisation is not used are illustrated in Figure 6.4. In the following experiments, GOB markers are always used.

(a) (b)

Figure 6.4: Foreman frame 111, (a) original, (b) effect of packet loss without GOB sync.

6.3.2 Error Concealment Results

Two methods of error control and concealment were tested. The one is adding intentionally redundancy by forcing a percentage of the encoded sequence to intra-coding. This makes the bitstream more robust, as all error propagation stops at the next intra-coded block. However, it is not the sender-receiver interactive error control scheme that was described above, as the insertion of the intra-coded information is done at random positions, and is not guided by the decoder.

Obviously, the tested scheme has the advantage that it can be applied when a decoder feedback channel is not available. This is a feature extremely desirable in multicasting applications. The drawback is its randomness, which means that the scheme does not detect errors and correct them, but acts preventively and, under certain conditions, might not provide the desired results. Two versions of the scheme were tested under 5% random packet loss, one forcing 5% and the other 15% intra-coding.

The second error concealment method is temporal prediction, that is, in the event of a packet loss, the error-struck current frame region is repaired using the corresponding region from previous frames. Unfortunately, there is no point in using this technique in conjunction with forced-intra coding, due to the randomness of the latter.

The sequences were all tested under 5% packet loss conditions. Packetisation is done according to RFC 2190 Mode A, that is, packets are aligned at GOB level. Therefore a lost packet corresponds to a lost row of macroblocks for the current frame.

The variation of luminance PSNR for every decoded frame for the Foreman sequence, is depicted in Figure 6.5. A comparison of the average quality results of the tested techniques on all the sequences, is shown in Table 6.2. Comparative subjective results are also provided for all the sequences in Figures 6.6, 6.7, and 6.8, in the following pages.

Scheme	*Foreman* $\overline{\text{PSNR}}$ (dB)	*M&D* $\overline{\text{PSNR}}$ (dB)	*Akiyo* $\overline{\text{PSNR}}$ (dB)
No Loss	32.11	32.87	34.55
No Forced Intra-coding	20.72	22.37	21.62
Temporal prediction	23.58	24.21	27.67
5% Intra-coded	24.33	26.81	29.41
15% Intra-coded	26.57	30.48	32.94

Table 6.2: Comparison of error concealment results for 5% packet loss.

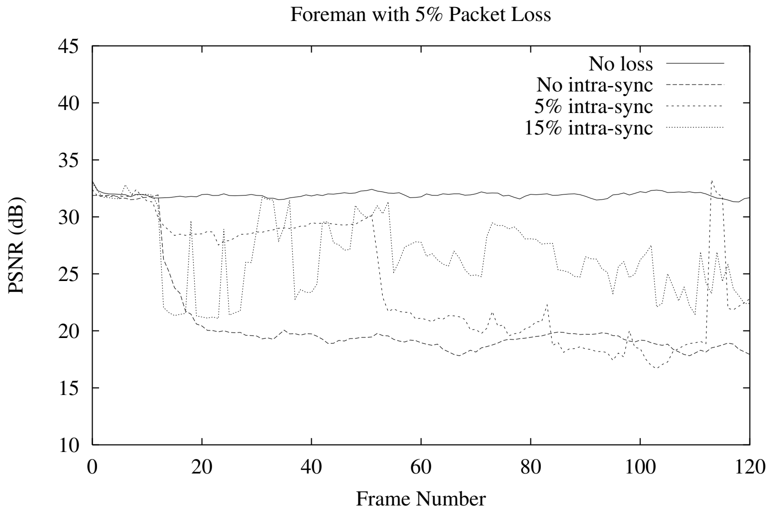


Figure 6.5: Foreman PSNR per decoded frame for 5% packet loss.

6.3.3 Discussion

From the statistical results of Table 6.2 and the following sample frames, it is obvious that all error control techniques perform well for the given packet loss conditions. The effects of packet loss are shown in Figures 6.6b, 6.7b, and 6.8b. These images feature severe distortions including block artifacts and other spatial deformations, caused by the absence or incorrect decoding of either macroblock information, or motion vectors information.

As discussed in previous chapters, the temporal prediction error concealment performs well only on low-motion sequences, and only for individual errors (not error bursts). The artifacts that are caused because of the use of the previous frame's information, are frustrating for moving parts of the image. This is clearly seen in Figure 6.6c for the helmet and part of the face, in Figure 6.7 for the mother's hand, and in Figure 6.8 for the subject's eyes. However, these sequences are played at 15 fps, and these errors are hardy notifiable. The problem with error repair, is that it is not actually repair, as the error will be preserved for next frames if the encoder is not notified, or intra-coded blocks do not appear. This is reflected on its relatively low average performance statistics of Table 6.2.

On the other hand, forced intra-coding provides a much more robust bitstream. The more intra-resyncs inserted, the better picture quality is obtained, as shown in Figure 6.5. However, this comes at the price of significantly increased bandwidth requirements. It should be stressed that the frames illustrated in Figures 6.6d, 6.7d, and 6.8d, are not intra-coded.

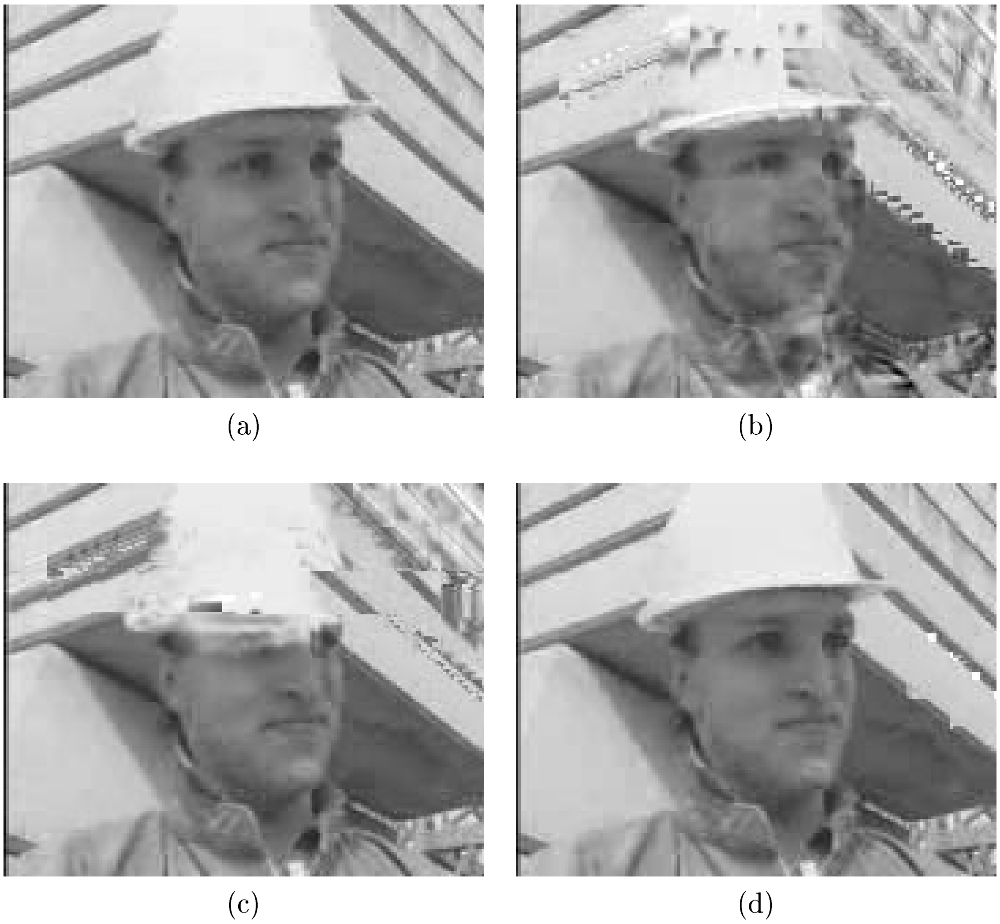


Figure 6.6: Foreman with 5% packet loss inter-coded frame 111, (a) no packet loss, (b) no forced intra, (c) temporal prediction concealment, and (d) 5% forced intra.

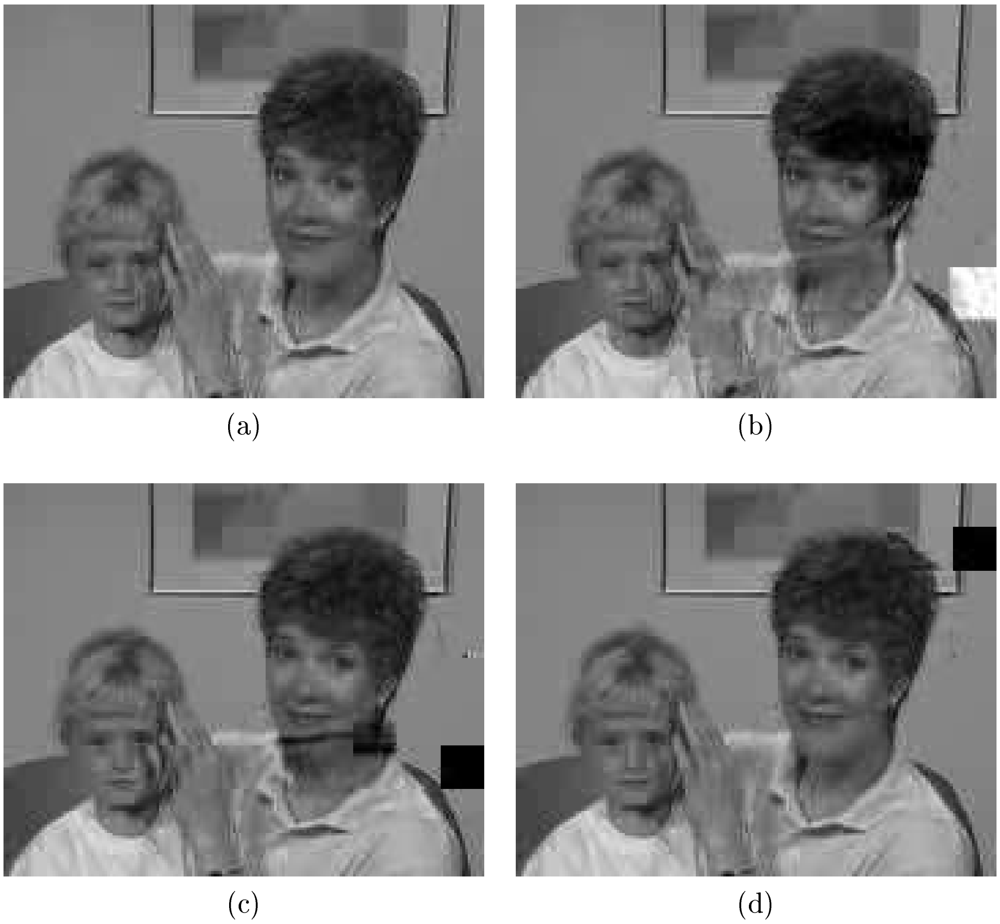
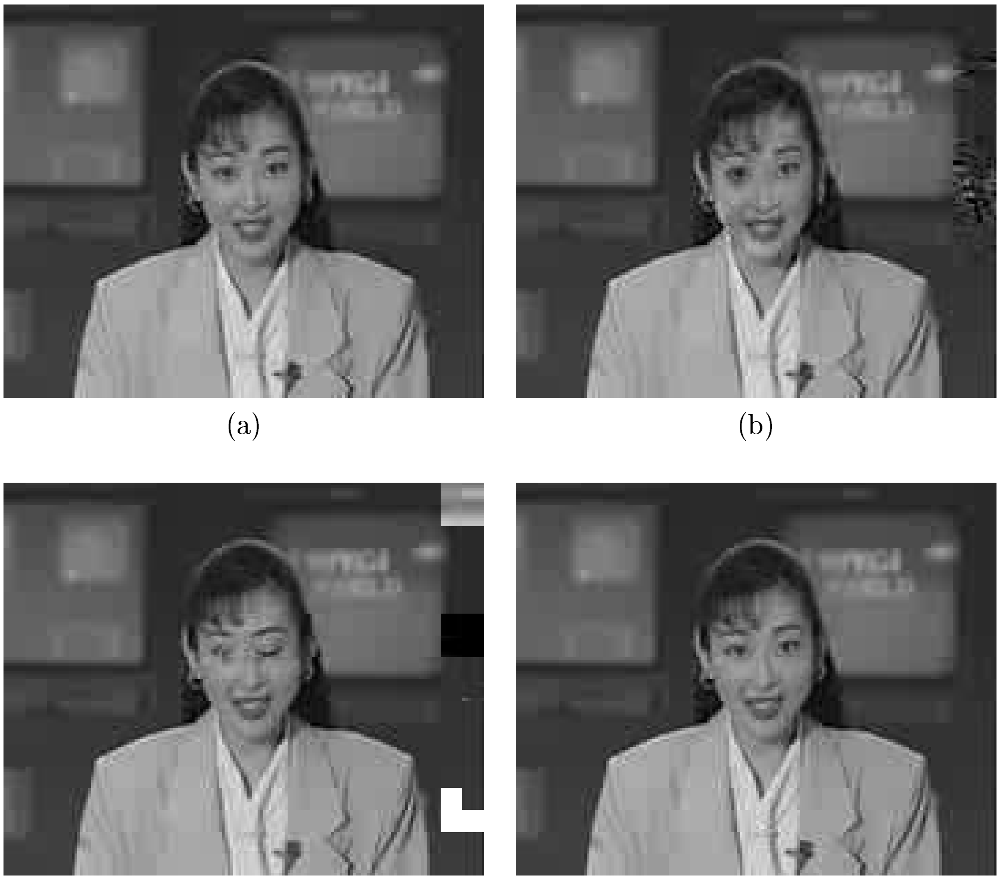


Figure 6.7: M&D with 5% packet loss inter-coded frame 22, (a) no packet loss, (b) no forced intra, (c) temporal prediction concealment, and (d) 5% forced intra.



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

Figure 6.8: Akiyo with 5% packet loss inter-coded frame 111, (a) no packet loss, (b) no forced intra, (c) temporal prediction concealment, and (d) 5% forced intra.

6.4 Summary

This chapter presented the details of the implementation part of this work. The results obtained were presented analytically and graphically, and some brief remarks on the output video sequences were made. These were found to be fully in line with the corresponding theoretical evaluations in the previous chapters.

Chapter 7

Conclusions

The target in this work is a reliable digital video transmission scheme over IP-based networks. The latter, as non-reliable best-effort delivery systems, are prone to packet loss, as described in Chapter 3. Moreover, compressed data exhibit very high sensitivity to partial loss, due to the concentration of significant information to the resulting bitstream. This is clearly visible in video sequences, where, due to the predictive compression techniques used, errors are propagated in the spatial and temporal direction, leading to very low quality, nearly imperceptible, images.

Therefore, the plain standardised compression algorithms used in H.261, H.263, and MPEG codecs, are shown not to perform well in such conditions. Some solutions sacrifice much of the compression efficiency of these schemes, by disabling interframe prediction, in order to provide a more robust, self-synchronising bitstream. This is arguably acceptable, as tradeoffs between compression efficiency and error-resilience are common practice.

However, the price of sacrificing all interframe prediction is very high, as bandwidth requirements are significantly and wastefully increased. Recent advances in research on coding and transmission schemes, target on retaining the compression efficiency of interframe prediction and using judiciously intraframe synchronisation to increase robustness in packet loss conditions. The most sophisticated techniques involve cooperation of the sender and the receiver in order to recover from packet loss.

Apart from enhancing the reliability and robustness of the transmitted information, it is desirable that quality degradation does not reach levels frustrating for the users, at any time. Therefore, besides the encoding, error concealment techniques are applied at the decoder, to increase the perceptibility of error-struck parts of the video sequence. It is well understood that these techniques are only enhancement measures, and do not remedy the information loss effects.

As in most engineering design problems, there is not a panacea providing solutions for any application. Robust video transmission schemes are limited scope systems, optimised and orientated to particular application requirements. For this work, these requirements were set for personal communications applications over the current unreliable IP network service model.

Taking into account all the above considerations, the resulting proposed solution was to insert a limited amount of redundancy in the waveform encoding stage, by slightly modifying the intra-inter coding decision rule to be more biased than usual towards intra-coding. This provides the desired robustness to the bitstream and a good reference to the decoder to fully recover from errors in only a few fractions of a second. The performance of the scheme is further improved if a simple temporary error repair techniques, such as temporal prediction is used, as shown by the experimental results obtained through simulation.

The proposed scheme copes well with prolonged error bursts due to congestion, as intra-coding provides effective isolation of groups of frames, making their decoding independent of the correct reception of other groups. It does not require two-way communication between the sender and the receiver, which could make it a good candidate for multicasting.

Its main drawback is the limited compression efficiency associated with intra-coding. In addition, it does not perform well under unfavourable conditions such as packet loss on contiguous intra-coded packets, which have a very low, but non-zero occurrence probability.

Future research should focus on interactive error control and transmission, which undoubtedly provides the best solution for robustness. The very simplistic network model used, with random packet loss and no QoS guarantees, could also be replaced by a more near to real world networks simulation model. Furthermore, efficient ways of handling errors of a multicast group during a session, could be investigated as well.

Bibliography

[Bra97a] Torsten Braun. Internet Protocols for Multimedia Communications. Part I: IPng—The Foundation of Internet Protocols. *IEEE Multimedia*, 4(3):85–90, July-September 1997.

[Bra97b] Torsten Braun. Internet Protocols for Multimedia Communications. Part II: Resource Reservation, Transport and Application Protocols. *IEEE Multimedia*, 4(4):74–82, October-December 1997.

[CWH+96] Jon Crowcroft, Ian Wakeman, Mark Handley, Stuart Clayman, and Paul White. *Internetworking Multimedia*. UCL Press, 1996.

[FGV98] Niko Färber, Bernd Girod, and John Villasenor. Extensions of ITU-T Recommendation H.324 for Error-Resilient Video Transmission. *IEEE Communications Magazine*, 36(6):120–128, June 1998.

[Flu95] Francois Fluckiger. *Understanding Networked Multimedia: Applications and Technology.* Prentice Hall International, 1995.

[ISO91] ISO/IEC 11172. Information Technology—Coding of moving pictures and associated audio for digital storage media up to about 1.5 Mbit/s. Technical report, ISO, 1991.

[ISO93] ISO/IEC JTC1/SC2/WG11. MPEG Phase 2. Test Model 5, ISO, 1993.

[ITU93] ITU-T Recommendation H.261. Video Codec for Audiovisual Services at px64 kbit/s. White book, ITU-T, 1993.

[ITU95] ITU-T Recommendation H.263. Video Coding for Low Bitrate Communication. Draft, ITU-T, 1995.

[NH95] Arun N. Netravali and Barry G. Haskell. *Digital Pictures: Representation, Compression, and Standards.* Plenum Press, 2nd edition, 1995.

[RR96] Martyn J. Riley and Iain E. G. Richardson. *Digital Video Communications.* Artech House, 1996.

[Sta98] William Stallings. *High Speed Networks: TCP/IP and ATM Design Principles.* Prentice Hall, 1998.

[Tal98] Raj Talluri. Error-Resilient Video Coding in the ISO MPEG4 Standard. *IEEE Communications Magazine*, 36(6):112–119, June 1998.

[Tan96] Andrew S. Tanenbaum. *Computer Networks.* Prentice Hall International, 3rd edition, 1996.

[WZ98] Yao Wang and Qin-Fan Zhu. Error Control and Concealment for Video Communication: A Review. *Proceedings of the IEEE*, 86(5):974–997, May 1998.

[Zhu97] C. Zhu. RTP payload format for H.263 video streams. RFC 2190, Internet Engineering Task Force, September 1997.