University of Cyprus
Department of Computer Science
Networks Research Laboratory

Adaptive Methods for the Transmission of Video Streams in Wireless Networks

Deliverable 1.2

Stream Concealment Techniques and Layering Techniques

Project Title	: Adaptive Methods for the Transmission of Video Streams in Wireless Networks
Deliverable Type	: Final WP report (M12)

Deliverable Number	: D1.2
Title of Deliverable	: Stream Concealment Techniques and Layering Techniques
Work Package Title	: Content Adaptation Techniques
Work Package Number	: WP1
Internal Document Number	:
Contractual Delivery Date	:
Actual Delivery Date	:
Author(s)	: Pavlos Antoniou Vasos Vassiliou Andreas Pitsillides Andreas Panayides Aris Vlotomas Demetris Piperis Petros Christoforou
Email(s)	: paul.antoniou@cs.ucy.ac.cy vasosv@cs.ucy.ac.cy Andreas.Pitsillides@ucy.ac.cy

Abstract

Delivering video data of satisfactory quality over unreliable networks – such as the Internet or wireless networks – is a demanding area which has received significant attention of the research community over the past few years. Streaming video guaranteeing the necessary quality is a very well debated and challenging problem. Given the fact that packet loss is inevitable and therefore the presence of errors granted, the effort is directed towards limiting the effect of these errors. A number of techniques have been developed to address this issue, each one incorporating a set of characteristics. This deliverable aims to introduce the most considerable approaches of error resilience, error concealment and joint encoder-decoder error control techniques and to provide a thorough discussion of the benefits and drawbacks of these error control methods. Moreover we investigate the Content Adaptation Techniques that aim to the adaptation of content to the desirable rate without the need for re-encoding or regenerating. In particular we introduce encoding of video information into multi-layered video emphasizing in the three video coding techniques SNR (Signal-to-Noise Ratio), temporal and spatial.

Keywords: error concealment techniques, error resilience techniques, content adaptation techniques.

Table of Contents

List of Figures

List of Tables

1. Introduction

A typical video communications system consists of five steps as illustrated in *Figure 1*.

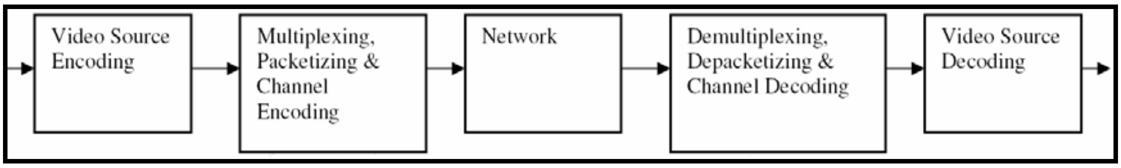
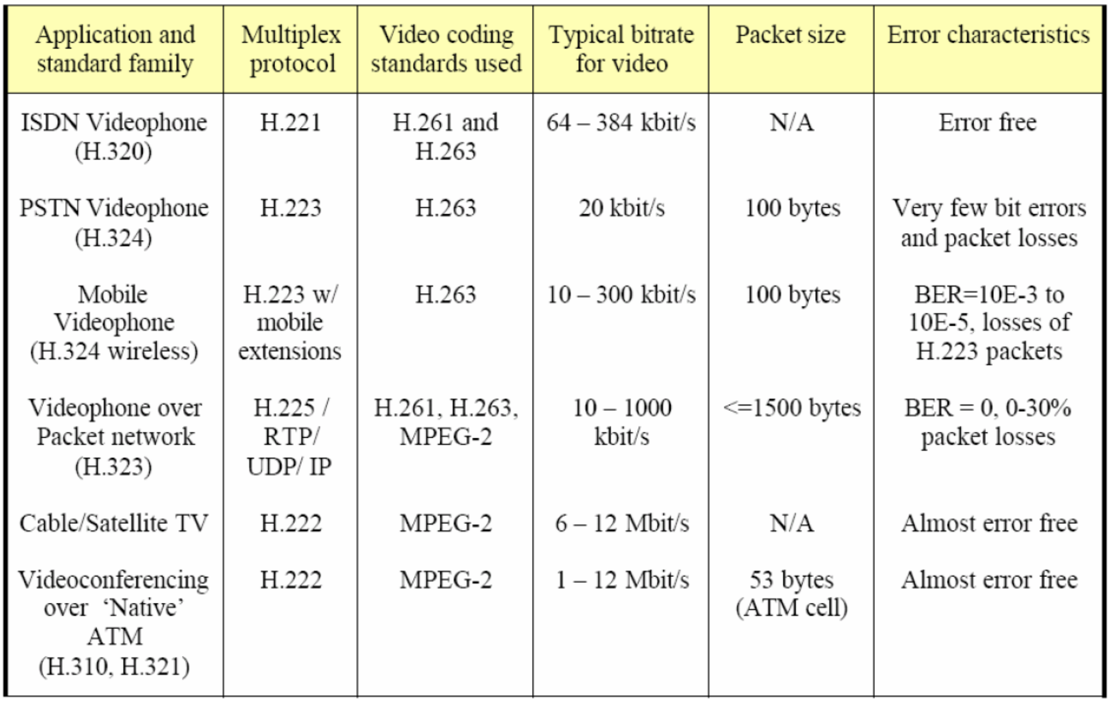

Video Source Encoding
Multiplexing, Packetizing & Channel Encoding
Network
Demultiplexing, Depacketizing & Channel Decoding
Video Source Decoding

Figure 1: A typical video communication system [3].

Unless a dedicated link is employed between source and destination, Quality of Service (QoS) for real time applications cannot be provided. This is due to the fact that packet loss or corruption caused by a number of reasons such as physical impairment, traffic congestion or bit errors is unavoidable. Such losses are even more intense in video streaming because of the use of predictive coding and variable length coding (VLC) performed by the source coder. Today's Internet, a packet based network, is described as a best effort delivery service. This states clearly that current Internet technology and protocols involved may not supply guarantees for an application's quality. Transmission control protocol (TCP) uses retransmission and traffic monitoring to secure packet delivery to destination. However, this approach is not applicable for video streaming and real time applications in general since retransmission time is simply unacceptable in most cases. In addition, retransmission may result in alternations of temporal relations between audio and video. User Datagram Protocol (UDP) is used instead which on the other hand, does not provide any error handling or congestion control. Consequently, designing the compression algorithm and the compressed bit stream in such a way that is resilient to errors is the prudent way to address this issue. Studying the background relative literature on error resilient techniques one can identify three major approaches which have been developed to tackle error control and are categorized according to where error control actually takes place: Constructing the compressed bit *error resilient* takes place at the encoder using a number of source and channel coding techniques. As mentioned above though, the presence of errors is inevitable. For that reason, *error concealment* mechanisms have been developed and are triggered by the decoder to limit the effect of errors once they have occurred. Lastly, schemes based on *encoder-decoder* interaction have been introduced, where the decoder provides all the necessary information for the encoder to adapt to network conditions and detain packet loss.
Below we investigate error resilient techniques as well as error concealment methods and joint encoder-decoder error control techniques.



Application and standard family	Multiplex protocol	Video coding standards used	Typical bitrate for video	Packet size	Error characteristics
ISDN Videophone (H.320)	H.221	H.261 and H.263	64 – 384 kbit/s	N/A	Error free
PSTN Videophone (H.324)	H.223	H.263	20 kbit/s	100 bytes	Very few bit errors and packet losses
Mobile Videophone (H.324 wireless)	H.223 w/ mobile extensions	H.263	10 – 300 kbit/s	100 bytes	BER=10E-3 to 10E-5, losses of H.223 packets
Videophone over Packet network (H.323)	H.225 / RTP/ UDP/ IP	H.261, H.263, MPEG-2	10 – 1000 kbit/s	<=1500 bytes	BER = 0, 0-30% packet losses
Cable/Satellite TV	H.222	MPEG-2	6 – 12 Mbit/s	N/A	Almost error free
Videoconferencing over 'Native' ATM (H.310, H.321)	H.222	MPEG-2	1 – 12 Mbit/s	53 bytes (ATM cell)	Almost error free

Table 1: Standard families for video transmission [3].

2. Error Resilient Techniques

Error resilient techniques [30] aim to encode the compressed bit stream in such a way that the transmission errors impact upon decoding and reconstruction of the video data will be minimal. To achieve this, the encoder must add redundancy to the compressed bit steam. Redundancy bits are additional to data bits and are the ones responsible for improved quality in the presence of transmission errors. The involved mechanism employed of course does not come at any cost, as encoders adding redundancy become less efficient than normal encoders. However, in the long run, the benefit is obvious. The problem is then focused on maximizing error resilience with the smallest possible amount of redundancy bits.

Error resilient video coding techniques can be subdivided in the following approaches: Robust Entropy Coding, Incorrect State and Error Propagation, Unequal Error Protection (UEP)-Layered Coding and Multiple Description video coding. These approaches are presented in the rest of this Section.

2.1 Robust Entropy Coding

Compressed video stream's vulnerability to transmission errors emerges from the fact that a video coder employs variable length coding (VLC) to represent various symbols. Consequently, once a bit error occurs or a bit is lost, immediately the involved and subsequent code word(s) are constituted non-decodable, since the decoder is not able to match the appropriate bits to appropriate parameters.

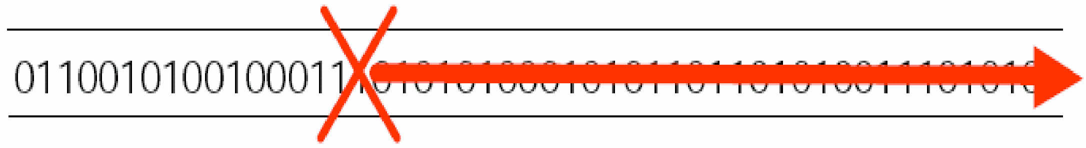


Figure 2: Code words following an error are constituted non-decodable.

One should note that fixed length codes (FLC) are not susceptible to this problem since the knowledge of the beginning and the end of each codeword limits the loss to that single codeword. However, FLC's provide poor compression efficiency and therefore are not considered.

2.1.1 Re-Synchronization Markers

Re-synchronization markers constitute a simple, yet efficient way to address the bit stream synchronization problem. Such markers may be placed strategically in the bit stream (MPEG-1/2, H.261/3) or periodically (MPEG-4). Placing re-synchronization markers strategically (every fixed number of blocks, variable number of bits) suffers from the increased probability that active bit stream areas may be corrupted. Conversely, periodically placing the markers (variable number of blocks, fixed number of bits) reduces this probability while it simplifies the search of resynchronization markers and supports network packetization. Therefore re-synchronization markers are placed periodically. A marker should be designed in such a way that it will notably differ from other code words, concatenations of code words and minor permutations of code words.

The end of a marker should follow some kind of header information incorporating spatial and temporal locations or other in-picture predictive information concerning the subsequent bits. The decoder can then resume decoding properly after the interruption which occurred by the presence of the resynchronization marker.

The frequency and length of re-synchronization markers can be thought of redundancy-wise. That is, the more frequent and longer the markers are, the more redundancy bits are employed and reversely. In addition, the presence of a marker typically interrupts in-picture prediction mechanisms, such as MV or DC coefficient prediction, contributing in this way in increasing the added redundancy. However, longer and frequent re-synchronization markers enable the decoder to recover faster from transmission errors and therefore restrain their effect by gaining synchronization quicker. Consequently, in practical video coding systems, long and frequent markers are employed.

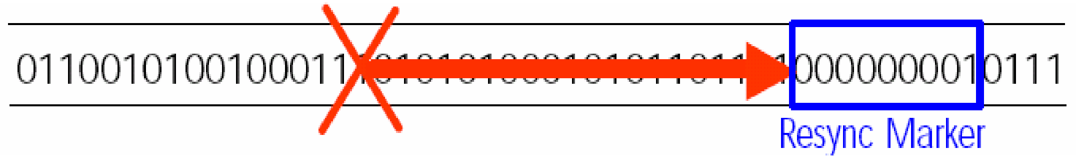


Figure 3: Re-synchronization markers enable the decoder to recover from transmission errors and continue decoding properly.

2.1.2 Reverse Variable Length Coding (RVLC)

While conventional VLC's only enable forward unique decoding, reverse variable length coding RVLC's are enhanced to also provide backward unique decoding. The underlying idea is why waste the information carried by the bits that intervene between the corrupted bits and the next resynchronization marker when you can make use of them; So, instead of jumping to the next resynchronization marker and continue decoding onwards, jump to the next synchronization marker and start decoding backwards to make worthy the non corrupted bits.

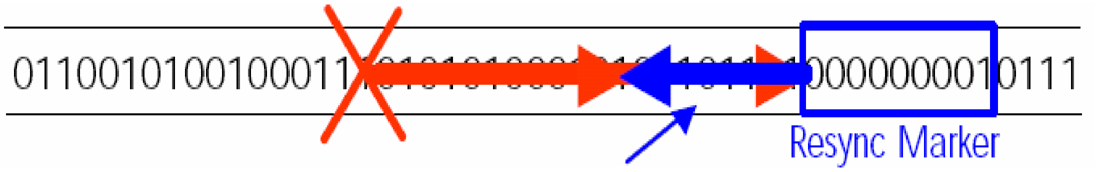


Figure 4: RVLCs enable both backward and forward unique decoding, limiting thus the number of lost bits.

The introduced complexity by RVLC is not prohibitive in manners such as coding efficiency, in contrary to what was originally believed. RVLC can be designed with near perfect entropy coding efficiency, thus easy implementation. Nevertheless, knowledge gained through practical use has shown that RVLC may as well prove to be more efficient than VLC for certain applications, providing greater error resilience. RVLC has been adopted by both MPEG-4 and H.263 in conjunction with insertion of synchronization markers.

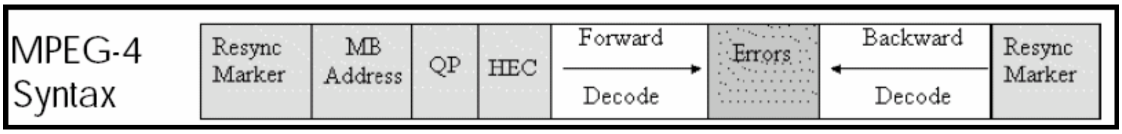


Figure 5: RVLC as implemented in the MPEG-4 Syntax.

2.1.3 Data Partitioning

The essence in data partitioning lies in the observation that the closer a bit is located to the resynchronization marker the highest is the possibility that this bit is accurate. Minding that not all data bits carry equal information, the most important bits are placed immediately after the resynchronization marker –MV's, shape info, DC coefficients- and less important bits –AC coefficients afterwards.

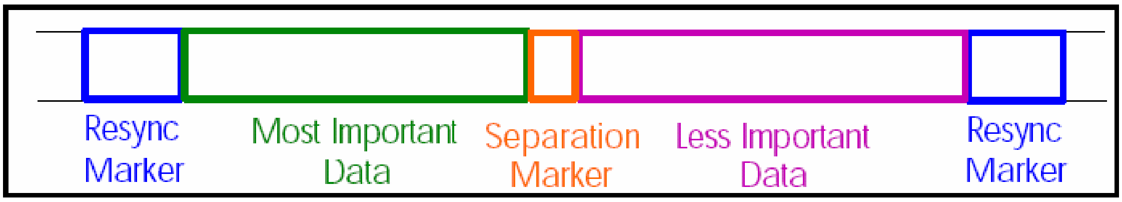


Figure 6: A typical data partitioning structure of a macroblock.

2.2 Incorrect State and Error Propagation

The presence of an error originates the incorrect reconstruction of the frame (state) at the decoder. Once the decoder is found in a different state than the encoder, subsequent frames reconstructions referencing this incorrect state will also result in error. This situation can produce significant error propagation. To address this issue, some form of re-initialization must take place in the prediction loop in order to limit extend of error propagation.

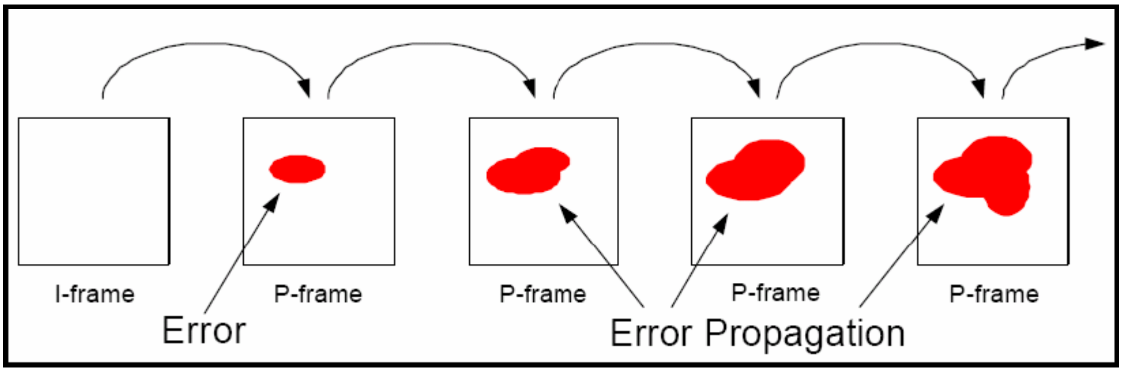


Figure 7: The presence of an error propagates to consequent frames.

2.2.1 Insertion of Intra-Block or Frames

The exclusive use of I-Frames in the computation of forward frames would eliminate error propagation. However, such approach is not considered due to the poor compression involved. Periodic insertion of I-frames (for example one every fifteen frames) limits the effect to a single GOP (Group of Pictures). Still, compression is relatively poor, constituting periodic insertion of I-frames incompatible with delay constraints imposed by video applications and are introduced by this measure.

On the other hand, the insertion of intra coded MBs has proved to be a considerable and effective technique towards the limitation of error propagation. In addition, the complexity introduced aggravates the encoder, keeping the decoder simple. Thus, choosing which, when and how many intra coded MBs are going to be inserted has to be decided. Intra coding however, involves higher bit rate than inter coding. Consequently, the use of intra-coding is not unlimited and should be employed wisely. Several approaches have been proposed throughout the years but none significantly outperforms some other.

i. Periodic intra-coding of all MBs, intra codes different MBs in each frame in some predefined order so that after a certain number of frames all MBs have been intra coded at least once.

ii. Pre-emptive intra-coding is based on previous knowledge of channel loss model, which allows the estimation of which MBs are most vulnerable to errors. Similarly we can place the intra-coded MBs in areas of highest activity.

iii. Random placement has also proved to perform quite well.

It is obvious that channel characteristics directly impact on the number of intra-coded MBs used. Channel knowledge may be obtained by a number of ways, one of which is point to point communication with back channel, which we consider later in Section 4.

2.3 Unequal Error Protection (UEP) – Layered Coding

Layered Coding (LC) codes a video into a base layer and one to many enhancement layers. Base layer provides limited but of satisfactory quality video while each of the enhancement layers incrementally improves quality. In order for LC to be constituted an efficient error resilience tool, it must be paired with UEP, so that the base layer is protected more strongly. Corruptions occurring in enhancement layers are not of equal importance with corruptions emerging in the base layer coding of video. Typically the base layer includes vital information, loss of which may result in having no video at all. LC structure and philosophy supports users of different bandwidth capacities and decoder capabilities to access same video data at varying qualities. Several techniques may be applied when dividing a video signal to two or more layers. One way is to down sample the video data and code the low frame rate into the base layer. The enhancement layers will then include the error between the original video and the up sampled video constructed from the low frame coded video. Similarly, the spatial resolution may be divided in such a way that the base layer includes a small frame-size video and incorporating DCT coefficients of each block with a coarse quantizer. Enhancement layers define the error between the original DCT coefficients and the coarsely quantized value. Furthermore, base layer can also contain the header and motion information, with the remaining information found in the enhancement layers. This approach is adopted by the MPEG and H.263 protocols where the terminology used for the first three options are temporal, spatial and SNR scalability whereas the last data partitioning. LC uses hierarchical, de-correlate decomposition.

2.4 Multiple Description Video Coding

As we have seen, conventional video coders have a single state which is generated by the previous coded frame and any loss may lead to error propagation. Multiple description video coding (MD) codes video data source into a number of descriptions, correlated and of roughly equal importance. The underlying idea is that receiving a single stream provides adequate quality, while the presence of multiple description streams suggests high quality video. Descriptions are correlated and this characteristic is what enables the decoder to tell when a description is corrupted or not and thus provide a satisfactory quality level out of every description. Nevertheless, MD provides a reliable sub channel even in very lossy networks. Opposite to LC, MD employs a non-hierarchical, correlating decomposition. MD video coding approaches include multiple threads with resynchronization [Wenger], predictive MD quantizer [Vaishampayan, John], MD transform coding [Reibman, Wang, Orchard, Puril] and multiple states [Apostolopoulos].

3. Error Concealment Techniques

It is very likely that transmission errors will result in loss of information. Error concealment techniques objective is to try and provide an accurate estimation of the missing data, concealing the presence of an error. Based on the fact that video presents a noteworthy correlation among its spatial and temporal dimensions, most techniques are developed taking advantage of that characteristic. Consequently, performing some kind of spatial or temporal interpolation, they attempt to recover lost data based on correctly received information. Spatial interpolation, Temporal interpolation and Motion compensated temporal interpolation techniques are discussed in this Section.

3.1 Spatial Interpolation

A simple method used for recovering corrupted data is spatial interpolation. An estimation of a missing or damaged pixel is derived by extrapolating surrounding correctly received pixels. However, due to the fact that all blocks or MBs of the same row are usually placed in the same packet, the only available adjacent blocks are those of the rows above and below, not a representative sample of the damaged pixels in most cases. As a result, only the boundary pixels in neighbouring blocks are used for interpolation. Even this way, correctly recovering missing pixels is extremely difficult. Instead, the DC (average) value is estimated and used to replace every corrupted pixel. Computing the DC proves efficient enough. One approach to spatial interpolation is by an interleaved packetization mechanism so that the loss of each packet only affects every other block or MB.

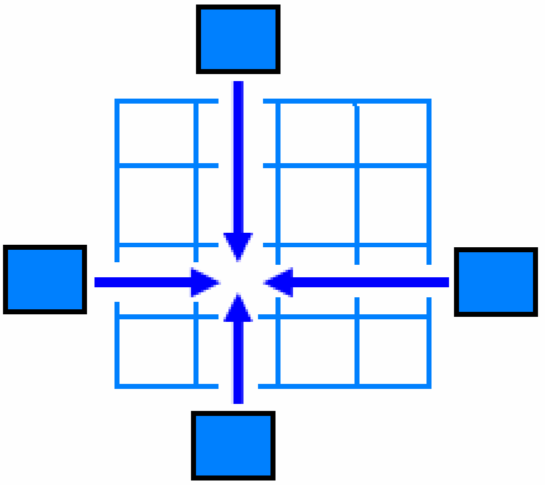


Figure 8: Spatial Interpolation.

3.2 Temporal Interpolation

Another approach to recover a corrupted MB is MC temporal prediction which copies the pixels at the same spatial location in the previous frame (freeze frame). This approach is effective when there is no motion involved but susceptible to problems in the presence of motion. To address potential problems, the motion vector (MV) is computed as well if it is also damaged. MC temporal techniques generally provide better results than spatial interpolation techniques. Combination of both techniques works better for the estimation of the MV.

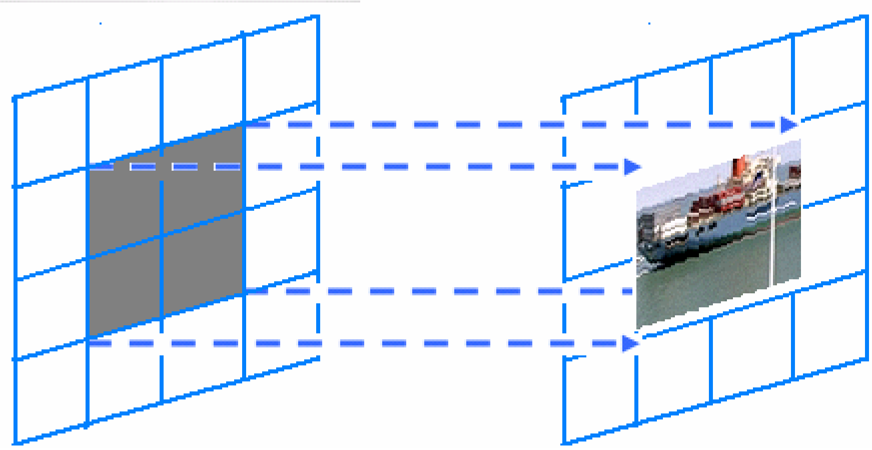


Figure 9: Temporal Interpolation.

3.3 Motion Compensated Temporal Interpolation

Motion vector (MV) is the corner stone for a plethora of the algorithms discussed since inter frames computation is based upon the knowledge of the MV in combination with the DCT coefficients of the prediction error. Consequently, corruption of the MV significantly reduces decoding ability for quality video data reconstruction before an intra frame is inserted. To achieve decoder error concealment, data partition may be employed at the encoder. As described above, data partition packs important data such as the MV and employed mode and transmits them with increased error protection, limiting thus the probability that this data will be damaged or lost. This mode is employed by both MPEG-4 and H.263. However, there exists considerable possibility that an error may occur and therefore actions must be taken to confront them. In such a case, both coding mode and MV need to be estimated. A simple way to compute the coding mode is to assume that the macro block (MB) is coded in the intra-mode and therefore use only spatial interpolation to recover the affected blocks. Another is to derive the damaged MB by collecting statistics of the surrounding MBs and select the most likely MB. For MV estimation several approaches exist which include among others: (a) assume the lost MVs to be zeros, (b) use the MVs of the corresponding block in the previous frame, (c) use the average of the MVs from spatially adjacent blocks, (d) use the median of MVs from the spatially adjacent blocks, (e) re-estimate the MVs. Typically, when a MB is damaged, its horizontally adjacent MBs are also damaged, and hence the average or mean is taken over the MVs above and below.

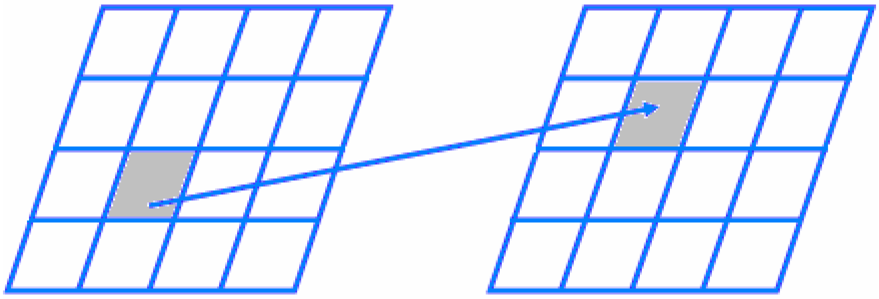


Figure 10: Motion compensated temporal interpolation.

4. Joint Encoder-Decoder Error Control Techniques

Instead of acting independently, encoder and decoder are eligible to set up a communication channel which can be then used to tackle error control more effectively. Decoder can sent feedback to encoder regarding the lost information – such as the position of the corrupted data - and the encoder can then adapt to these conditions to limit transmission errors. This Section encompasses two such approaches which are described below: Error tracking based on feedback information and choosing which frame to use based on feedback information.

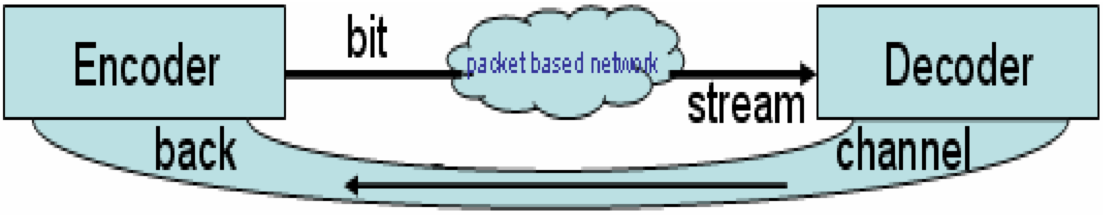


Figure 11: Encoder – Decoder communication with feedback channel.

4.1 Error Tracking based on feedback information

Upon notification by the decoder that an error has occurred, encoder confronts the situation by initializing one of the following methods:

i. Reinitialize prediction using an I-frame. Simple and relatively straightforward approach, however it suffers from the fact that it employs higher bit rate for intra coding.

ii. Avoid using the error affected area in the prediction of subsequent frames.

iii. Employ the same error concealment technique used by the decoder for the affected frames. In that way, encoder’s and decoder’s reference pictures will match when coding the next frame.

The first two techniques require only the knowledge of the error’s location by the encoder whereas the last one incorporates the duplication of the decoder error concealment procedure for the number of the damaged frames. All techniques cause error propagation to stop.

4.2 Choosing which frame to use based on feedback information

An alternative way to make use of the feedback arriving at the encoder carrying information regarding the location of the occurred error is by employing some kind of mechanism as to which frame to use as reference based on the received information. That is, instead of using the most recent reference frame for coding the next frame, use an older reference picture which is available at the decoder and known to be clean. For the utilization of this method, encoder and decoder need to store multiple previously coded frames. Note that this does not necessarily add delay at the decoder. Nevertheless, when compared to using an I-frame for the coding of the next frame, this approach is significantly more efficient when not too old pictures are used. Which previously coded frame is used as reference for prediction is decided by the encoder.
This approach is employed by both MPEG-4 and H.263 named NewPred and Reference Picture Selection (RPS) respectively. Two modes of operations which can be found in the MPEG-4 specification are ACK and NACK:

i. In the ACK mode the encoder only encounters acknowledged (i.e. correctly received) frames for prediction. By doing so, it minimizes error propagation but is susceptible to the use of relatively “old” (further apart) frames which leads to poor compression. This happens because acknowledgements may arrive late at encoder.

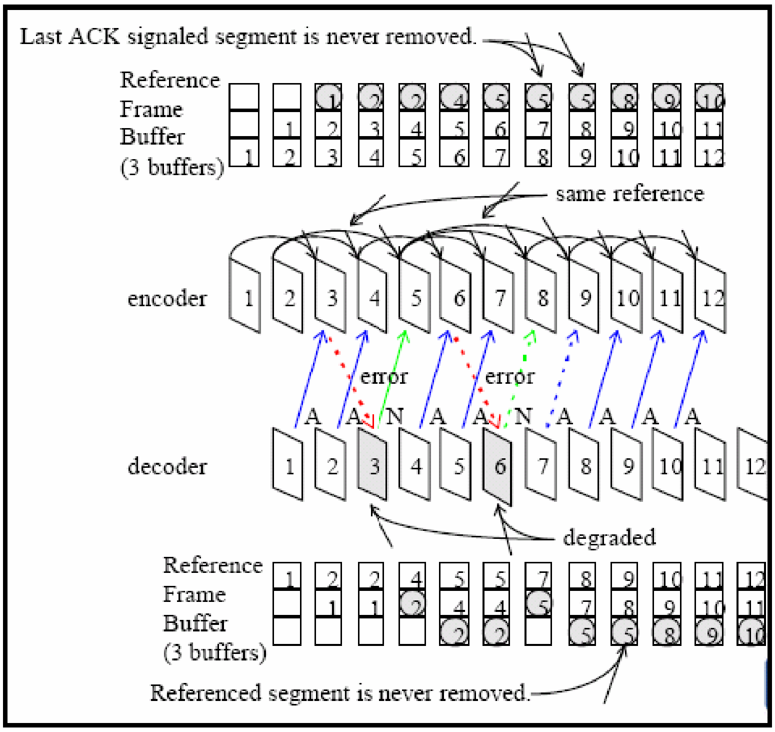

Last ACK signaled segment is never removed.
Reference Frame Buffer (3 buffers)
same reference
encoder
error
error
A
A
N
A
A
N
A
A
A
A
decoder
degraded
Reference Frame Buffer (3 buffers)
Referenced segment is never removed.

Figure 12: MPEG-4 Syntax of NewPred ACK mode of operation.

ii. On the other hand, NACK mode uses the last coded frame as reference for prediction, unless negative acknowledgement is received. In that way, the most recent coded frames are used. However, in the event of an error with delayed negative acknowledgement, error propagation increases.

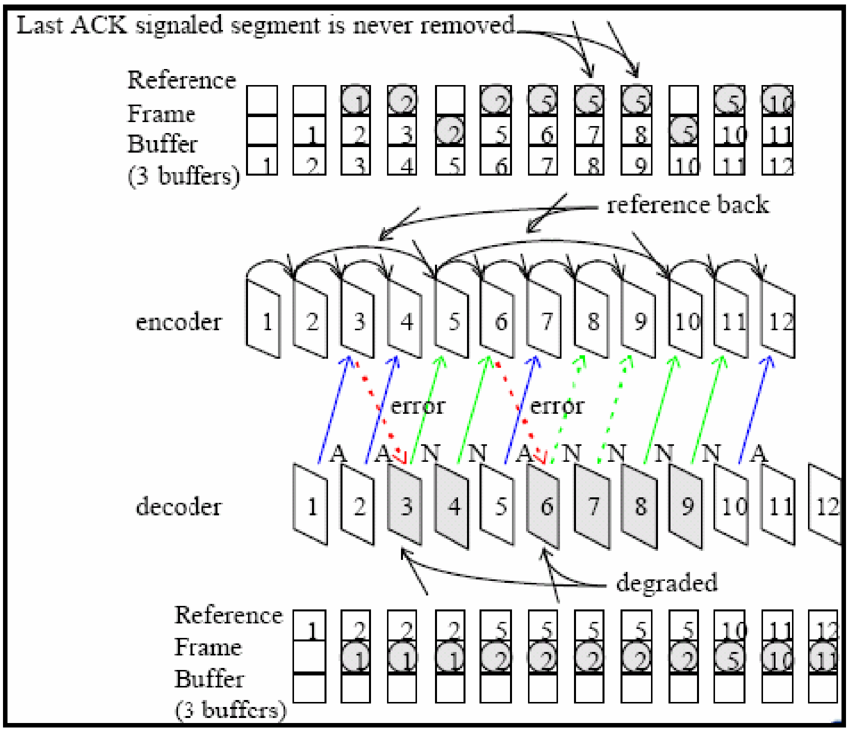


Figure 13: MPEG-4 Syntax of NewPred ACK mode of operation (ctd).

Availability of feedback can significantly improve error handling however, is strictly depended on round trip delay (RTD). In general, effectiveness decreases as RTD increases constituting this approach inapplicable for applications such as broadcast, multicast or pre-encoded video. Conversely, is suitable for applications which include video phone or conferencing.

5. Content Adaptation Techniques

Content Adaptation Techniques (CATs) aim to the adaptation of content to the desirable rate without the need for re-encoding or regenerating (transmission of information in multiple layers).

The objective of video coding was to optimize video quality at a given bit rate. Due to the new network video applications, the objective has somewhat changed. There is an increasing interest for multimedia streaming services over the Internet like multipoint video teleconferencing, distance learning, telemedicine, digital libraries (e.g. video database browsing with multi-resolution playback), video on demand etc. Their availability depends strongly on communication network infrastructure. Existing communication networks are very heterogeneous. Rapid development of multimedia services is stimulating interests in video transmission through heterogeneous communication networks that are characterized by various available levels of Quality of Service (QoS). New services demand leads the research for controlling the video/audio transmission in order to ensure QoS requirements.

In a streaming video system a source encodes video content and transmits the encoded video stream over a data network (wired or wireless) where one or more receivers can access, decode, and display the video to users in real-time (*Figure 14*). The presence of the network, which allows the source to be physically far from the receivers, differentiates streaming video from pre-recorded video used in consumer electronic devices such as DVD players.

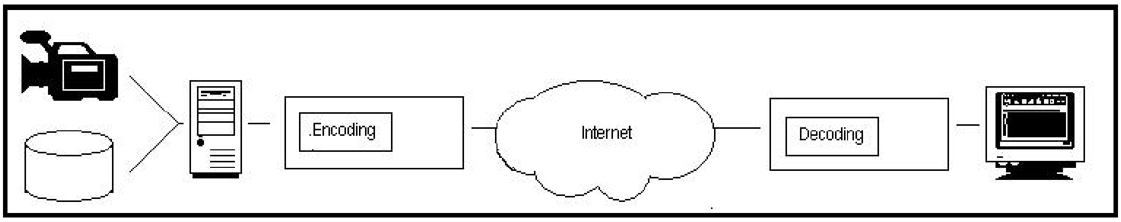


Figure 14: Streaming video over the Internet.

Given that uncompressed video has very large bandwidth demands, the need for efficient video compression is paramount. A unique problem that streaming presents for the compression technique is that in many applications the network cannot guarantee the bandwidth that is available. Users may also possess decode and display devices with a range of features and capabilities. Much of the work in streaming research has been motivated in finding ways to overcome the limitations of the network. Some of the techniques that improve video streaming systems include error control and reduction, and the development of scalable video compression techniques. An ideal data network is capable of transferring any amount of information without delay or loss; unfortunately practical networks do not possess such characteristics. In fact, if an ideal network were to exist, video streaming would be a trivial problem. Practical networks can introduce errors (bit errors), deletions (packet or bit loss) and insertions (cross-talk) and they have delay and latency, and finite bandwidth. These problems can also vary over time. Any traffic in the network, whether it is a video stream or some other data, is subject to the constraints of the network. The performance issues of a network are:

1. **Bandwidth** - Bandwidth is the amount of data that can traverse through the network or a part of the network at any given time. Network bandwidth is a shared, limited resource and will vary with time. Networks may carry multiple video streams simultaneously or carry non-video data. A network may not be able to guarantee that the required bandwidth for transporting video will be available.
2. **Delay, Jitter, and Latency** - Streaming video is subject to delay constraints since the video must be decoded and displayed in real-time. If video data spends too much time in the network, it is useless even if it arrives at the receiver. Buffering can reduce the effect of jitter (timing errors) but does not reduce delay. Latency can also be an issue when two-way communication is necessary.
3. “**Errors**” - The network may cause errors, loss or deletions, and insertions. Data can be lost in the network for a variety of reasons, including congestion, rejection due to excessive delay, and network faults. It is obvious that the streaming video system cannot ignore the possibility of data errors or loss during network transmission. These problems can cause unnatural deformation of the video, such as missing frames, lines, or blocks. Also, heavy network congestion can cause the video playback to be stopped until the receiver can resynchronize. The effective loss experienced by a user can be greater than the actual loss by the network. For example, if the first packet of data

corresponding to a video frame is lost or corrupt, the data in all subsequent packets of that video frame may not be decode-able, even if the packets were successfully delivered by the network. Furthermore, errors arising from lost data can affect multiple video frames by temporal error propagation.

The above are the primary QoS issues for any network. Most networks do not have QoS control, or mechanisms to prioritize network resources for streams that carry high-priority, time-sensitive data. Networks that have QoS control can maintain "guaranteed" resources by rejecting new users if QoS restrictions are violated using "smart" routing and scheduling algorithms, or by using a reservation method for network resources. Typical QoS parameters include minimum available bandwidth, maximum end-to-end delay, maximum bit or packet loss rate, and jitter. Because QoS control is not available for most networks, streaming video systems are usually end system-based, which implies that the network is not expected to provide any support for ensuring reliable transmission.

In addition to the QoS issues, two other network issues also affect the delivery of streaming video: heterogeneity and time-variance. A heterogeneous network is a network whose parts (sub-networks) may have vastly unequal resources. For example, some parts of a heterogeneous network may have abundant bandwidth and excellent congestion control while other parts of the network are overloaded and congested by overuse or by a lack of physical network resources. Different receivers on a heterogeneous network can experience different performance characteristics. When streaming video over a heterogeneous network, the video stream should be decode-able at optimal quality for users with a good network connection, and at useable quality for users with a poor connection. Time-variance implies bandwidth, delay, loss, or other network characteristics can vary significantly over time, sometimes changing drastically in a matter of seconds. When streaming over a network with time-variance, the steaming video source should be able to adjust its parameters to changing network conditions (adaptability).

The Internet is often the target network for streaming video. It is certainly not the only network that could be used for streaming (for example, cellular and wireless networks.) The Internet is a difficult network for transporting real-time data; it is a prime example of a heterogeneous, time-varying, network with no QoS control. The video data must be formatted in a way that it can be transported by and routed through the network. Most streaming applications assume a packet switched network that uses the Internet Protocol (IP). (Hence streaming is sometimes referred to as "video over IP".) Data is usually sent through the network using TCP/IP. This transport is very well suited to non real-time data and provides reliable communication through the use of retransmission. Retransmission is almost never possible for video traffic and hence "connectionless" protocols such as UDP are used.

For Real-time transport of live or stored video Internets best effort is not enough, UDP (User Datagram Protocol) is more suitable for real time applications since it has lower overhead and lower delay than TCP, but its unreliable nature requires that care must be taken to conceal the errors that are introduced due to dynamic network congestion that is caused by packet losses and jitter. So, the main problem is the variation in network bandwidth constraints cause (ranging from 28.8 Kbps modems links to 622 Mbps OC-12 links - Stands for Optical Carrier Level 12. OC-12 is a circuit that transmits 622 megabits per second). Different levels of QoS are mostly related to different available transmission bit-rates. On the other hand, the service providers demand that the data are broadcasted once to a group of users accessed via heterogeneous links that end in heterogeneous client terminals (*Figure 15*). For this purpose, the transmitted bit-stream has to be partitioned into some layers, i.e. the encoder should be scalable. In the simplest case, there are two layers: the base layer and the enhancement layer. The base layer is decode-able independently from the enhancement layer and it represents a video sequence with reduced spatial resolution, temporal resolution or Signal-to-Noise Ratio (SNR). The enhancement layer bit-stream provides additional data needed for reproduction of pictures with original quality. The respective functionality is called spatial, temporal or SNR scalability.

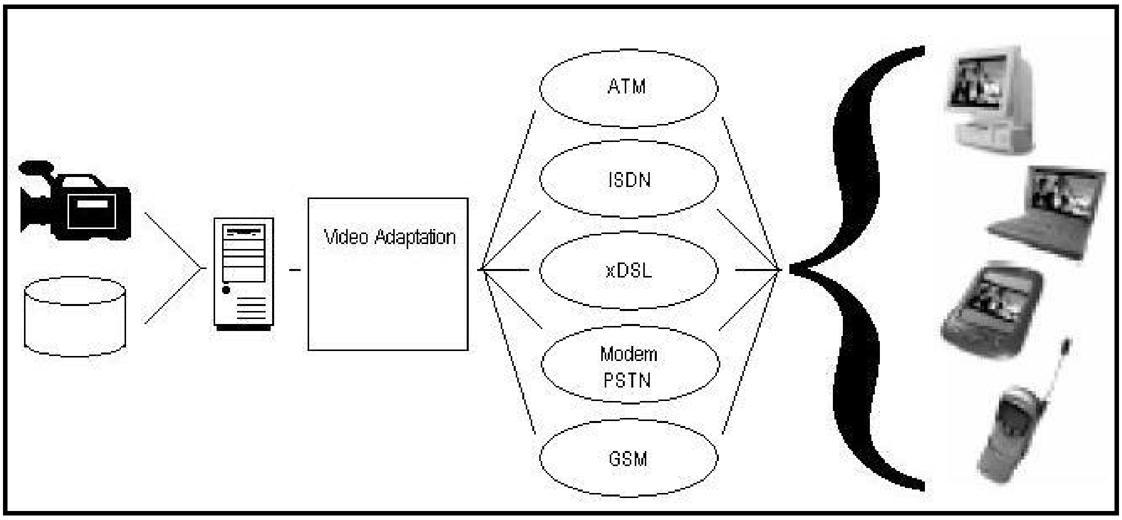


Figure 15: The role of Video Adaptation is to support heterogeneous networks and terminals.

Other protocols, such as RTP (Real Time Protocol), can be used with UDP for real-time applications. When multiple receivers wish to access common video content this can be accomplished by sending each receiver a separate set of packets known as a unicast stream. In unicast the network delivers an independent copy of the stream from the source to each receiver. This obviously can overwhelm the network if thousands of users are requesting content.

Multicast is an IP protocol that has broadcast-like capability and could be used to transport video. Multicast can significantly reduce the bandwidth required to simultaneously deliver a video stream to multiple receivers and is very useful for video conferencing and streaming of content to a large audience. Multicast is not as useful for video-on-demand applications, where different receivers are viewing different streams or at different points along a common stream. In multicast, the video is delivered from the source in such a way that additional copies of the video stream are created only when necessary. The routers in a multicast network accomplish this feat by establishing a multicast tree from the source to all receivers. Only a single copy of the video stream is sent along each link of the tree, in particular if the link is common in the paths from the source to two or more receivers. Multicast routing is not a trivial problem and the methods by which the tree is constructed and maintained are amongst the current research areas in multicast. Other multicasting issues include adding QoS control and resilience against network failure. It should be noted that when using multicast each receiver does not receive a unique stream, this has interesting implications with respect to security.

This problem can be solved with an end-to-end adaptive mechanism and multi-layer video encoding (Content Adaptation Techniques). As we mentioned before, Content Adaptation Techniques (CATs) aim to the adaptation of content to the desirable rate without the need for re-encoding or regenerating (transmission of information in multiple layers). We are focusing on video because video requires larger bandwidth (100 kbps - 15 Mbps) than audio (8 kbps – 128 kbps) and humans are more sensitive to loss of audio than video. In multi-layered encoding the video information is encoded into several layers. A scalable video compression algorithm allows extraction of coded visual information at varying rates from a single compressed stream. There are two primary advantages to using multi-layered video encoding in multicast-capable networks. First is the ability to perform graceful degradation of video quality when loss occurs. Because each video layer is prioritized, a network experiencing congestion may discard packets from low priority layers, thereby protecting the important base layer and higher priority enhancement layers from corruption. The second advantage, which is related to the first, is the ability to support multiple destinations with different bandwidth constraints or end-system capabilities. For each source-to-destination path with a unique bandwidth constraint, an enhancement layer of video may be generated.

Layers may be independent or hierarchical. When the layers are hierarchical, the decoding of higher layers depends on having properly received and decoded all lower layers, thus transmission of hierarchical layers can be inefficient on lossy networks, such as mobile wireless networks. If a network packet for the lowest layer is lost or corrupted, the now useless packets for the other layers still consume network bandwidth. If the layers were independent, then these remaining packets would still carry useful information.

When one compresses a video sequence the following parameters must be determined: frame size, frame rate, data rate, rate control, and de-compressed quality. One of the problems with many video compression methods is that these parameters are fixed at the encoding time and cannot be easily changed. For example, suppose one encodes a standard definition video sequence with MPEG-2 at 6 Mbps and stores it on a network video server that will stream it over a network. At a later time if a receiver requests this video information but does not want it at 6 Mbps, but at 4 Mbps, the server would have to transcode the sequence to meet the new rate requirement, which is usually computationally intense. Another way to address this problem is to store multiple copies of the compressed sequence at the video server that is then able to satisfy different users' needs; this can be expensive. Similarly a viewer may want the sequence at a different spatial resolution (i.e., different frame size) or at a different frame rate. This is an extremely important issue for a media producer who may have to provide content to be delivered at different resolution (temporal, spatial and/or rate) levels depending on the receivers' capability as well as the users' choice. An important question relative to scalability that impacts its use in video streaming is how often and fast can the compression parameters be changed. We shall say that a compression scheme is dynamically scalable if the compression parameters can be changed many times during the transmission process, i.e., during the streaming of the sequence. The network could then use this when congestion occurs to change the compression parameters on the fly and hence reduce the bandwidth being used. In this paper we will emphasize the use of rate-scalable compression.

Rate-scalable compression encodes the video such that the compressed video stream can be decoded at multiple rates, with increasing quality as the data rate increases. The video provider requires only one rate scalable video stream to support all the users in a heterogeneous network; the users with high bandwidth receive and decode the entire compressed video stream while those with lower bandwidth may only receive and decode a portion of the same video stream. Rate-scalable compression also allows the system to handle time-variance in the network, as each receiver receives the video stream at a rate that the network will allow at a given moment. A disadvantage for rate-scalable compression is compression efficiency.

There are several strategies for rate-scalability, including layered scalability, embedded coding, and hybrid layer/embedded coding. The scalability modes of most compression standards use layered scalability; this includes MPEG-4 and H.263+ [21, 22]. In layered scalability, the compressed video stream describes a base layer and one or more enhancement layers. The base layer is encoded at the minimum rate necessary to decode the video stream, and its decoding results in the lowest quality version of the video. Successive enhancement layers improve the quality of the video beyond that of the base layer. For example, a video source encoded using three layers, 64 Kbps base layer, 128 Kbps enhancement layer, and an additional 128 Kbps second enhancement layer can be decoded at 64 Kbps (base layer only, lowest quality), 192 Kbps (base + first enhancement, intermediate quality), and 320 Kbps (base + full enhancement, best quality). To decode at a given enhancement level, the receiver requires access the base layer and all enhancements layers up to the given level.

An alternative to using layer scalability is the use of an embedded coder. In an embedded coding scheme, the compressed video data is coded as a single unit without the use of distinct layers. For each frame, the decoder receives the compressed video stream from the picture header up till the available data rate, at which point the decode process stops and the video quality corresponding to that rate is achieved. To achieve the best performance, the most important information about the picture is placed at the beginning of the compressed video stream, followed by information of decreasing importance. Hybrids layer/embedded scalability technique combines layered and embedded coding, such as the Fine Grain Scalability (FGS) mode of MPEG-4. Similar to layer scalability, the compressed video stream consists of a base layer and enhancement layer. However, instead of using multiple enhancement layers to produce decode-able versions of intermediate quality, a single enhancement layer is coded using an embedded coding strategy. In MPEG-4 FGS, the MPEG-4 base layer is enhanced using bit-plane coding of the DCT coefficients of the residual.

From the previous discussions, it is clear that the network issues present a daunting challenge for streaming video. If a network such as the Internet is used to deliver streaming video, additional design elements beyond an encoder, network interface, and decoder can be introduced to overcome the network delivery issues (*Figure 16*). The video and audio sources are compressed and multiplexed into a single binary stream, adding synchronization and control information necessary for de-multiplexing and playback. The binary stream is then packetized. During the packetization, forward error correction can be added into the binary stream, as well as framing and timing information.

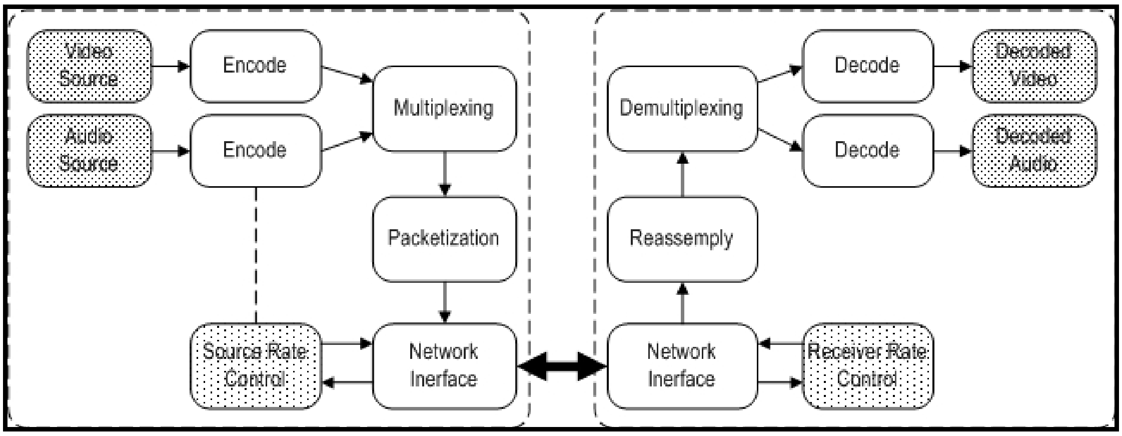


Figure 16: Video Streaming System.

Many scalable video-coding techniques have been proposed over the past few years for real-time Internet applications by several video compression standards such as MPEG-2/4 and H.263/263+. The types of scalability which are defined in these standards can be categorized as temporal, spatial, SNR, and object (only for MPEG4) scalability. All these types of scalable video consist of a Base Layer (BL), which is the minimum amount of data needed for decoding the video stream and one or more Enhancement Layers (EL). Both the base layer and the enhancement layer can be composed of I-P-B pictures which are the three generic picture types used in the above-mentioned standards.

Reviewing architecture for video streaming, we can examine the logical units that implement the video layering mechanism that is embedded in the encoding function. Different modalities of scalability are specified by video coding standards like MPEG-2 and MPEG-4. These standards aim to ensure interoperability amongst different manufacturers and to encourage cooperation, competition and increased choice. Scalable video coding has been an interesting topic. In MPEG-2 and MPEG-4, several layered scalability techniques, namely, SNR scalability (quality-level resolutions), temporal scalability, spatial scalability (and Fine Granularity scalability) have been included. ISO/IEC MPEG-1 did not provide any scalability mechanisms. The rest of the paper focuses on the scalability that MPEG-2 and MPEG-4 introduced and also presents other proposed scalability techniques.

5.1 MPEG-2 Standard for Video Encoding

Studies on MPEG-2 started in 1990 with the initial target to issue a standard for coding of TV-pictures with CCIR Rec. 601 resolution at data rates below 10 Mbps. In 1992 the scope of MPEG-2 was enlarged to suit coding of HDTV, thus planned MPEG-3 phase was abandoned (MPEG-3 idea was to have a separate scheme for HDTV). (The CCIR 601 is an international standard for digital sampling as it applies to NTSC, PAL, and SECAM standards. Although the international standards body has changed the name to ITU-R BT.601, the recommendations are popularly known as CCIR-601). The DIS (Draft International Standard) for MPEG-2 video was issued in early 1994 and it’s the coding method currently being used for digital TV broadcasting. It is a greatly expanded superset of MPEG-1, intended principally for high-quality entertainment video and audio. The video coding scheme used in MPEG-2 is again generic and similar to that of MPEG-1, however with further refinements and special consideration of interlaced sources. Furthermore, much functionality such as “scalability” was introduced. In order to keep implementation complexity low for products not requiring the full video input formats supported by the standard, so called “Profiles”, describing functionalities, and “Levels”, describing resolutions, were introduced to provide separate MPEG-2 conformance levels.

Flexibly supporting multiple resolutions is of particular interest for interworking between HDTV and Standard Definition Television (SDTV), in which case it is important for the HDTV receiver to be compatible with the SDTV product. Compatibility can be achieved by means of scalable coding of the HDTV source and the wasteful transmission of two independent bit streams to the HDTV and SDTV receivers can be avoided.

MPEG-2 describes a range of profiles and levels that provide encoding parameters for a range of applications. A profile is a subset of the full MPEG-2 syntax that specifies a particular set of coding features. Each profile is a superset of the preceding profiles. Within each profile, one or more levels specify a subset of spatial and temporal resolutions that can be handled. The profiles defined in the standard are shown in *Table 2*. Each level puts an upper limit on the spatial and temporal resolution of the sequence, as shown in *Table 3*. Only a limited number of profile/level combinations are recommended in the standard, as summarized in *Table 4*. Only SNR and spatial scalability are defined in the profiles.

Profiles	*Features*
Simple	4:2:0 sampling, I/P frames only, no scalable coding
Main	As above, plus B frames
SNR	As above, plus SNR scalability
Spatial	As above, plus spatial scalability
High	As above, plus 4:2:2 sampling

Table 2: MPEG-2 Profiles.

Level	*Maximum Resolution*
Low	352 x 288 luminance samples, 30Hz
Main	720 x 576 luminance samples, 30Hz
High-1, 440	1440 x 1152 luminance samples, 60Hz
High	1920 x 1152 luminance samples, 60Hz

Table 3: MPEG-2 Levels.

	Level			
Profile	*Low*	*Main*	*High-1, 440*	*High*
Simple		✓		
Main	✓		✓	✓
SNR	✓	✓		
Spatial		✓	✓	✓
High		✓	✓	✓

Table 4: Recommended Profile/Level Combinations.

Particular profile/level combinations are designed to support particular categories of applications. Simple profile/main level is suitable for conferencing applications, as no B pictures are transmitted, leading to low encoding and decoding delay. Main profile/main level is suitable for most digital television applications; the majority of currently available MPEG-2 encoders and decoders support main profile/main level coding. The two high levels are designed to support HDTV applications; they can be used with either non-scalable coding (main profile) or spatially scalable coding (spatial/high profiles). Note that the profiles and levels are only recommendations and that other combinations of coding parameters are possible within the MPEG-2 standard.

The term scalability is associated with the manipulation of a compressed stream that satisfies constraints on parameters, such as bit rates, display, resolutions or frame rates. The video stream is encoded into a set of cumulative sub-streams (layers), where each layer is a refinement of the previous layers - i.e. layered coding. The so-called base layer contains the most important data of the video. Additional layers are called enhancement layers. The base layer is fundamental for video decoding. Decoding only the base layer, results in a playable video of low quality. Scalable compression allows the compressed stream to be manipulated, even after compression. This is very important, since various applications do not know in advance, during the compression stage, the constraints on resolution, bit rate or decoding complexities. The perceptual quality of a frame-based video sequence depends on the frame rate, frame resolution and frame quality. By scaling each one of them we achieve temporal, spatial and quality scalability respectively. Temporal scalability is the most common scalability technique used in several video compression standards, such as H-263 and the MPEG family. These standards support combinations of such scalabilities. Therefore, new hybrid scalable schemes can emerge which would be more efficient for specific applications. A common hybrid scalable scheme is spatial-temporal scalability providing 2D space for scaling.

The scalability tools standardized by MPEG-2 support applications beyond those addressed by the basic main profile coding algorithm. The intention of scalable coding is to provide interoperability between different services and to flexibly support receivers with different display capabilities. Receivers either not capable or willing to reconstruct the full resolution video can decode subsets of the layered bit stream to display video at lower spatial or temporal resolution or with lower quality. Another important purpose of scalable coding is to provide a layered video bit stream, which is amenable for prioritized transmission. The main challenge here is to reliably deliver video signals in the presence of channel errors, such as cell loss in ATM based transmission networks or co-channel interference in terrestrial digital broadcasting.

MPEG-2 provides two layers, each layer supporting video at a different scale, i.e. a multi-resolution representation can be achieved by downscaling the input video signal into a lower resolution video (down-sampling spatially or temporally). This is general philosophy of a multi-scale video coding scheme (*Figure 17*). The downscaled version is encoded into a base layer bit stream with reduced bit rate. The up-scaled reconstructed base layer video (up-sampled spatially or temporally) is used as a prediction for the coding of the original input video signal. The prediction error is encoded into an enhancement layer bit stream. If a receiver is either not capable or willing to display the full quality video, a downscaled video signal can be reconstructed by only decoding the base layer bit stream. It is important to notice, however, that the display of the video at highest resolution with reduced quality is also possible by only decoding the lower bit rate base layer. Thus scalable coding can be used to encode video with a suitable bit rate allocated to each layer in order to meet specific bandwidth requirements of transmission channels or storage media.

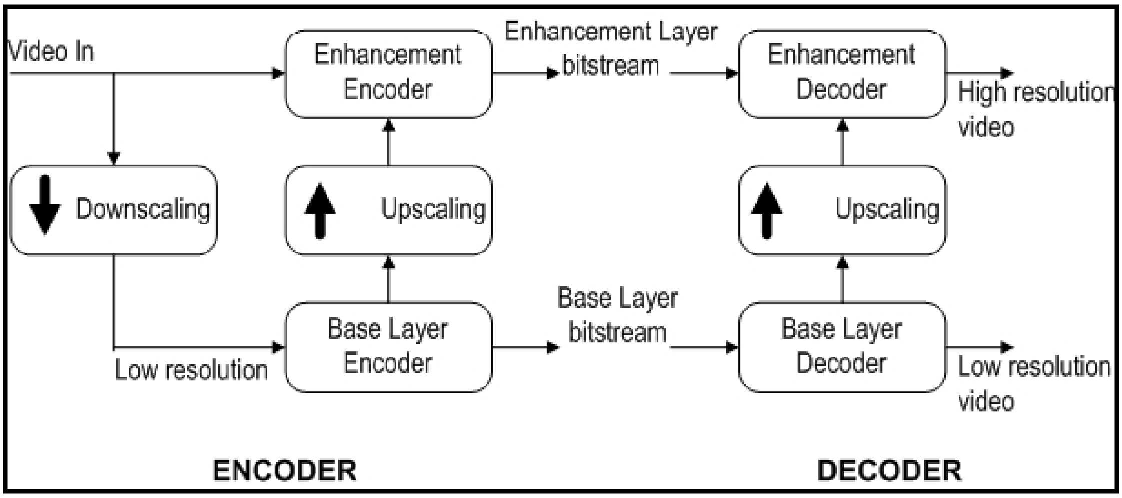


Figure 17: Scalable coding of video.

During the MPEG-2 standardization phase it was found impossible to develop one generic scalable coding scheme capable to suit all of the diverse applications requirements envisaged. While some applications are constricted to low implementation complexity, others call for very high coding efficiency. Four types of scalability are identified in MPEG-2, known as basic scalability: Spatial, SNR (Signal-to-Noise Scalability), Temporal and Data. Combinations of these tools are also supported and are referred-to as hybrid scalability. In the basic scalability, two layers of video referred to as the base (lower) layer and the enhancement layer are allowed. In hybrid scalability up to three layers are supported.

5.1.1 Spatial Scalability

Spatial scalability involves generating two spatial resolution video layers from a single video source such that the lower layer is coded by itself to provide the basic spatial resolution and the enhancement layer starting from the spatially interpolated lower layer restores the full spatial resolution of the input video source. Spatial scalability offers flexibility in choice of video formats to be employed in each layer. Also the codec can be made more resilient to channel errors by protecting the lower layer data against channel error.

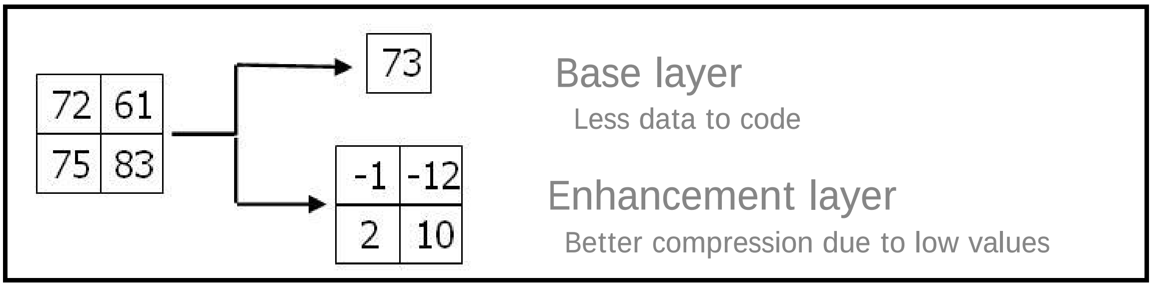


Figure 18: Spatial Scalability scheme.

Spatial Scalability has been developed to support displays with different spatial resolutions at the receiver - lower spatial resolution video can be reconstructed from the base layer. This functionality is useful for many applications including embedded coding for HDTV/TV systems, allowing a migration from a digital TV service to higher spatial resolution HDTV services. The algorithm is based on a classical pyramidal approach for progressive image coding. Spatial Scalability can flexibly support a wide range of spatial resolutions but adds considerable implementation complexity to the main Profile coding scheme.

Spatial scalability is a technique to code a video sequence into two layers at the same frame rate, but different spatial resolutions. The base layer is coded at a lower spatial resolution. The reconstructed base-layer picture is up-sampled to form the prediction for the high-resolution picture in the enhancement layer.

The MPEG-2 spatial scalable decoder uses as prediction a weighted combination of up-sampled reconstructed frame from the base layer and the previously reconstructed frame in the enhancement layer, while the MPEG-4 spatial scalable decoder allows a "bi-directional" prediction using up-sampled reconstructed frame from the base layer as the "backward reference" and the previously reconstructed frame in the enhancement layer as the "forward reference".

Spatial scalability is analogous to the hierarchical coding mode in JPEG, where each frame is encoded at a range of resolutions that can be "built up" to the full resolution.

5.1.2 Signal-to-Ratio Scalability

SNR scalability is a tool for use in video applications involving telecommunications, video services with multiple qualities, standard TV and HDTV, i.e. video systems with the primary common feature that a minimum of two layers of video quality are necessary. SNR scalability involves generating two video layers of the same spatial resolution but different qualities from a single source. The lower layer is coded by itself to provide a basic quality picture, while the enhancement layer is generated from the difference signal between the decoded basic picture and the non-coded input, and coded independently. When added back to the base layer the enhancement signal creates a higher quality reproduction of the input video.

This tool has been primarily developed to provide graceful degradation (quality scalability) of the video quality in prioritized transmission media. If the base layer can be protected from transmission errors, a version of the video with gracefully reduced quality can be obtained by decoding the base layer signal only. The algorithm used to achieve graceful degradation is based on a frequency (DCT-domain) scalability technique. The method is implemented as a simple and straightforward extension to the main Profile MPEG-2 coder and achieves excellent coding efficiency.

Signal to noise ratio (SNR) scalability is similar to the successive approximation mode of JPEG, where the picture is encoded in two layers, the lower of which contains information to decode a "coarse" version of the video and the higher of which contains enhancement information needed to decode the video at its full quality.

SNR scalability is also widely used especially in video transmission over a Diffserv network. In SNR scalability studies, BL is formed of I, P, and B pictures with a coarser quality. One of the drawbacks of this approach is that when one of the Enhancement Layer P frames is lost, the following Enhancement Layer P frames quality will degrade.

5.1.3 Temporal Scalability

A temporally scalable video coding algorithm allows extraction of video of multiple frame rates from a single coded stream. It is a tool intended for use in a wide range of video applications, from telecommunications to HDTV, in which migration to a higher temporal resolution system from lower temporal resolution may be necessary. In many cases the lower temporal resolution video source may be either an existing standard or a less expensive early generation system with the built-in idea of gradually introducing more sophisticated versions over time. In temporal scalability the basic layer is coded at a lower temporal rate and the enhancement layer is coded with temporal prediction with respect to the lower layer.

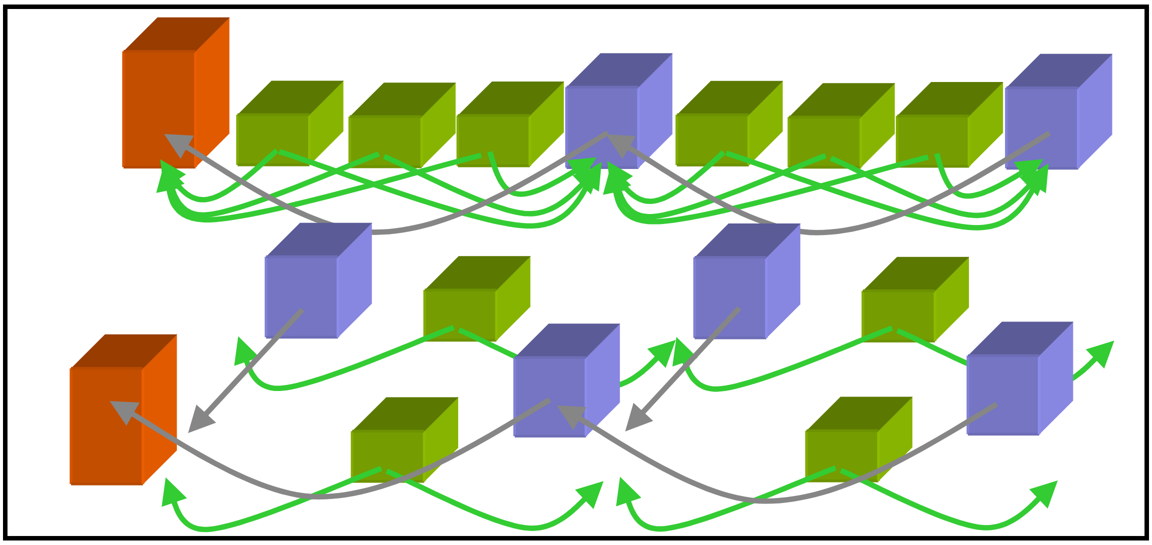


Figure 19: Temporal Scalability scheme.

Temporal scalability is a technique to code a video sequence into two layers at the same spatial resolution, but different frame rates. The base layer is coded at a lower frame rate. The enhancement layer provides the missing frames to form a video with a lighter frame rate. Coding efficiency of temporal scalability is high and very close to nonscalable coding. Temporal scalability is provided when the decoded video can vary in frame rate. Temporal scalability is a hierarchical coding mode in the temporal domain. The base layer is encoded at a lower frame rate. To give the full frame rate, the intermediate frames are interpolated between successive base layer frames. The difference between the interpolated frames and the actual intermediate frames is encoded as a second layer.

Involves partitioning of video frames into layers, whereas the lower layer is coded by itself to provide the basic temporal rate and the enhancement layer is coded with temporal prediction with respect to the lower layer, these layers when decoded and temporal multiplexed to yield full temporal resolution of the video source.

5.1.4 Data Partitioning

The bit-stream of the codec is partitioned between channels, such that its critical components (such as headers, motion vectors, DC coefficients) are transmitted in the channel with the better error performance. Data Partitioning is intended to assist with error concealment in the presence of transmission or channel errors in ATM, terrestrial broadcast or magnetic recording environments.

Less critical data such as higher DCT coefficients are transmitted in a channel with poorer error performance, but which is likely to be correspondingly less expensive. Because the tool can be entirely used as a post-processing and pre-processing tool to any single layer coding scheme it has not been formally standardized with MPEG-2, but is referenced in the informative Annex of the MPEG-2 DIS document [MPEG2]. The algorithm is, similar to the SNR Scalability tool, based on the separation of DCT-coefficients and is implemented with very low complexity compared to the other scalable coding schemes. To provide error protection, the coded DCT-coefficients in the bit stream are simply separated and transmitted in two layers with different error likelihood.

5.2 MPEG-4 Standard for Video Encoding

We referred to MPEG-4 standard in the previous deliverable (D1.1) so in this deliverable we will provide some differences between MPEG-4 and MPEG-1, MPEG-2. Moreover we are going to investigate scalability issues in MPEG-4 as well as Fine Granularity Scalability approach.

5.2.1 Differences between MPEG-4 and MPEG-1, MPEG-2

MPEG-1 and MPEG-2 are standards that focus on the compression and decompression of audio and video streams. Both standards address the needs of audio and video transport and synchronization. MPEG-1 was designed to provide a compression standard for media such as Video CD and CD-ROM, which have a typical playback rate of 1.2 Mbps. MPEG-2 was designed to provide higher quality for transmission applications, focusing mainly on Digital TV applications.

The major difference between MPEG-4 and MPEG-1 and 2 is the way MPEG-4 relates to the application level. MPEG-4 defines content that needs to be delivered over a network as a framework of media objects and scene descriptions. While MPEG-1 and MPEG-2 relate only to audio-video streams, MPEG-4 allows for the inclusion of other types of content such as animation, computer generated objects as well as video and audio. In MPEG-4, each component that comprises a multimedia scene is considered a media object. Each media object has spatial and temporal attributes that govern its behavior and location in the multimedia scene.

In addition to the concept of media objects, the MPEG-4 standard specifies that the transport mechanism of the multimedia stream need not be defined by the standard, but by the service provider or application developer. In contrast to MPEG-1 and 2, MPEG-4 defines streaming, synchronization and content rendering so as to accommodate bursty and scalable content delivery and to enable interactivity. Such requirements are intended to address the streaming of rich media over heterogeneous networks at bit-rates as low as 24 Kbps.

Although MPEG-4 covers more or less the same encoding range as MPEG-1 and MPEG-2, its target applications are different. MPEG-4 defines interactivity, scalability and streaming of rich media. Thus content compressed according to the MPEG-4 standard can be streamed over the broad or narrowband Internet, used in Interactive TV applications or streamed to wireless appliances such as cellular phones and PDAs (Personal Digital Assistants).

5.2.2 Scalability in MPEG-4

There are several scalable coding schemes in MPEG-4: spatial scalability, temporal scalability, fine granularity scalability and object-based spatial scalability. Spatial scalability supports changing the spatial resolution. Object-based spatial scalability extends the 'conventional' types of scalability towards arbitrary shape objects, so that it can be used in conjunction with other object-based capabilities. Thus, a very flexible content-based scaling of video information can be achieved. This makes it possible to enhance SNR, spatial resolution, shape accuracy, etc, only for objects of interest or for a particular region, which can be done dynamically at play-time.

Fine granularity scalability (FGS) was developed in response to the growing need on a video coding standard for streaming video over the Internet. FGS and its combination with temporal scalability address a variety of challenging problems in delivering video over the Internet. FGS allows the content creator to code a video sequence once and to be delivered through channels with a wide range of bitrates. It provides the best user experience under varying channel conditions. It overcomes the "digital cut-off" problem associated with digital video. In other words, it makes compressed digital video behave similarly to analogue video in terms of robustness while maintaining all the advantages of digital video.

5.2.3 Fine Granularity Scalability

SNR scalability as well as the other types of scalability (temporal, spatial) that were defined in MPEG-2 standard, has drawbacks. One of the drawbacks is that when one of the Enhancement Layer-P frames is lost, the Enhancement Layer-P quality will drastically degrade.

To provide more flexibility in meeting different demands of streaming (e.g., different access link bandwidths and different latency requirements), a new scalable coding mechanism, called fine granularity scalability (FGS), was proposed in MPEG-4.

In FGS, there is no temporal relation among the frames in the Enhancement Layer. Since in FGS the Enhancement Layer is formed of bitplane blocks which are DCT coded, bandwidth may be utilized more efficiently.

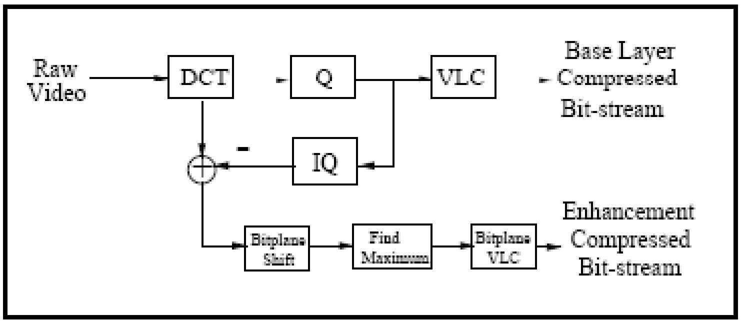

Raw Video
DCT
Q
VLC
Base Layer Compressed Bit-stream
IQ
-
Bitplane Shift
Find Maximum
Bitplane VLC
Enhancement Compressed Bit-stream

Figure 20: FGS Encoder.

As shown in above *Figure 20*, an FGS encoder compresses a raw video sequence into two substreams, i.e., a base layer bit-stream and an enhancement bit-stream. Compared with an SNR scalable encoder, an FGS encoder uses bitplane coding to represent the enhancement stream as shown in the following *Figure 21*. With bitplane coding, an FGS encoder is capable of achieving continuous rate control for the enhancement stream. This is because the enhancement bitstream can be truncated anywhere to achieve the target bit-rate.

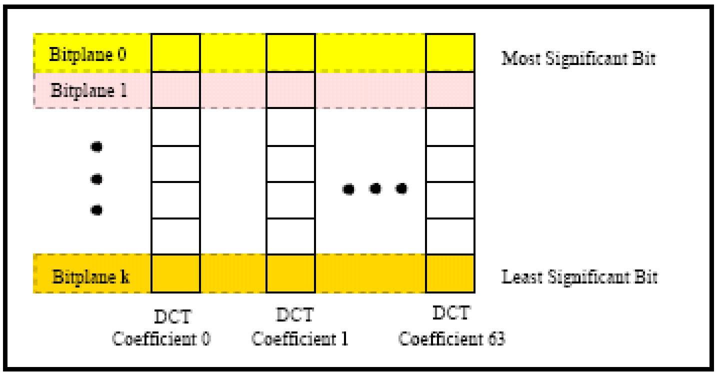


Figure 21: Bitplane coding.

As it was stated in MPEG-2 scalability capabilities description, in a layered scalable coding technique, a video sequence is coded into a base layer and an enhancement layer. The enhancement layer bitstream is similar to the base layer bitstream in the sense that it has to be either completely received and decoded or it does not enhance the video quality at all. As shown in the following *Figure 22*, layered scalability techniques, mentioned in MPEG-2, change the non scalable single staircase curve to a curve with two stairs. The base-layer bit rate determines where the first stair is and the enhancement layer bit rate determines the second stair. The two curves shown in the figure for layered scalability have different characteristics. The first one has a poor performance for the base layer and a good performance for the enhanced video. The second is the opposite, namely, poor performance for the enhancement layer and good performance for the base layer. As shown in *Figure 22*, the desired objective of video coding for Internet streaming video is to achieve the continuous curve that parallels the distortion-rate curve with a single bitstream. This is the objective of the fine granularity scalability (FGS) video-coding technique in MPEG-4.

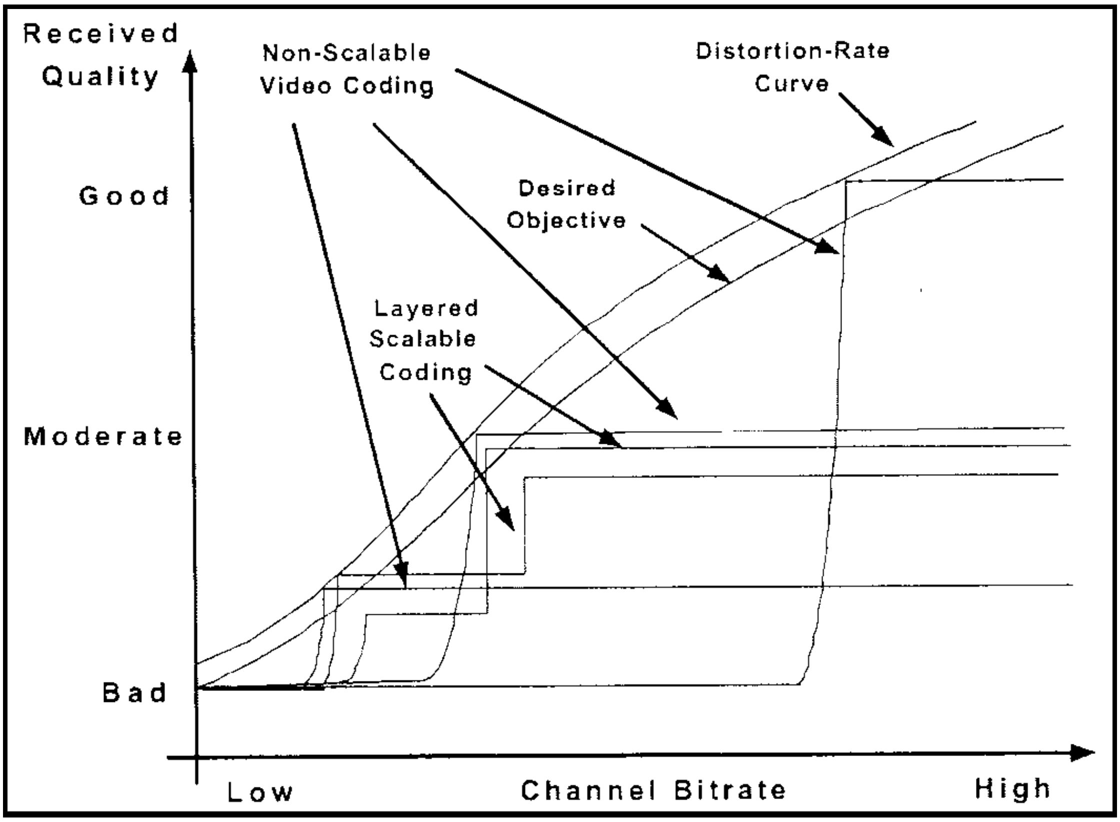


Figure 22: Comparisons between layered scalability techniques.

The horizontal axis in above figure indicates the channel bit rate, while the vertical axis indicates the video quality received by a user. The distortion-rate curve indicates the upper bound in quality for any coding technique at any given bit rate. The three staircase curves indicate the performance of an optimal non scalable coding technique. Once a given bit rate is chosen—either low, medium, or high—the non scalable coding technique tries to achieve the optimal quality indicated by having the upper corner of the staircase curve very close to the distortion-rate curve. If the channel bit rate happens to be at the video-coding bit rate, the received video quality is the best. However, if the channel bit rate is lower than the video coding bit rate, a so-called "digital cut-off" phenomenon happens and the received video quality becomes very poor. On the other hand, if the channel bit rate is higher than the video-coding bit rate, the received video quality does not become any better.

The spatial scalability decoders defined in MPEG-2 and MPEG-4 use two prediction loops, one in the base layer and the other in the enhancement layer. The MPEG-2 spatial scalable decoder uses as prediction a weighted combination of up-sampled reconstructed frame from the base layer and the previously reconstructed frame in the enhancement layer, while the MPEG-4 spatial scalable decoder allows a “bi-directional” prediction using up-sampled reconstructed frame from the base layer as the “backward reference” and the previously reconstructed frame in the enhancement layer as the “forward reference”.

The major difference between FGS and the layered scalable coding techniques is that, although the FGS coding technique also codes a video sequence into two layers, the enhancement bit-stream can be truncated into any number of bits within each frame to provide partial enhancement proportional to the number of bits decoded for each frame. Therefore, FGS provides the continuous scalability curve illustrated in the previous figure.

5.3 Hybrid Temporal-SNR Fine Granular Scalability for Internet Video

The hybrid temporal-SNR scalability has been adopted in the MPEG-4 standard in order to support video-streaming applications.

A limitation of the original MPEG-4 FGS framework is that only the image quality of the base-layer pictures can be enhanced (i.e., it provides only SNR scalability). However, if clients with very different connection capabilities need to access the same video sequence, tradeoffs should be made between the frame rate (motion smoothness) and image quality (SNR) of each individual frame. Therefore, the frame rate of the transmitted video sequence has to be enhanced in conjunction with the individual image quality.

Building upon the MPEG-4 FGS approach, the proposed framework provides a new level of abstraction between the encoding and transmission process by supporting both SNR and temporal scalability through a single enhancement layer. This abstraction is important, since the transmission bandwidth is not known at encoding time and thus, the optimal tradeoffs cannot be made a priori.

Moreover, depending on the individual user preference, which cannot be anticipated at encoding time, the individual image quality or the motion smoothness can be enhanced. With the proposed solution, which employs a fine-granular single layer for both SNR and temporal scalability, these decisions can be easily performed at transmission time depending on the user, decoder or server requirements. Another advantage of the framework presenting this technique is its reduced decoder complexity, requiring minimal addition to the original MPEG-4 FGS (SNR) implementation.

The hybrid temporal-SNR FGS scalable coding is based on a scalable technique for coding B-frames. This method is especially beneficial for devices with limited computational resources (e.g., mobile phones, wireless gadgets, etc.) that cannot guarantee the full decoding of non scalable B-frames. With the proposed scalable algorithm, the video quality is enhanced even if the B-frames are only partially decoded due to the limited computational resources.

Therefore, the single-layer FGS hybrid temporal-SNR scalability, extends the flexibility specific to FGS to the hybrid temporal-SNR scalability scheme. The hybrid temporal-SNR scalability provides total flexibility in supporting:

- SNR scalability while maintaining the same frame rate;
- Temporal scalability by increasing only the frame rate;
- Both SNR and temporal scalabilities.

The following figure shows the proposed hybrid scalability structure.

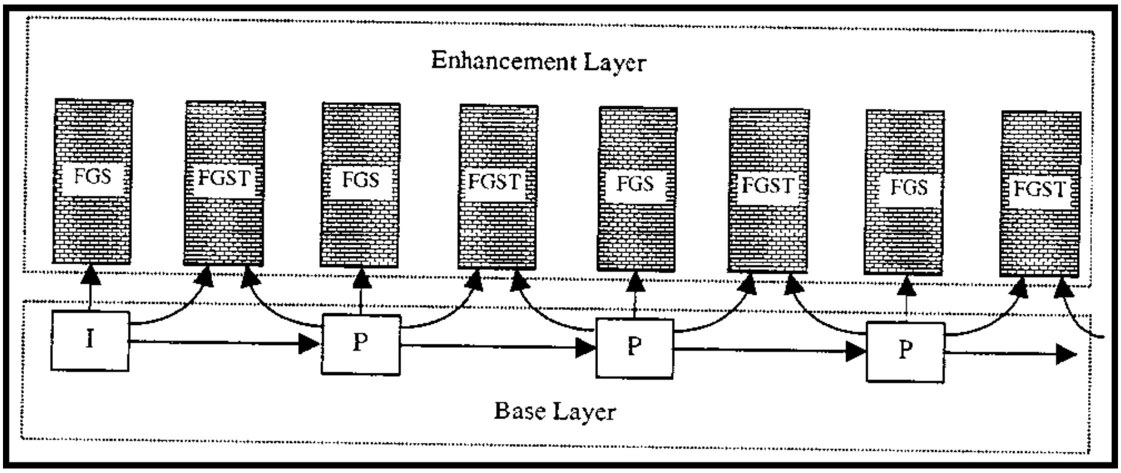


Figure 23: Proposed hybrid scalability structure.

The most important characteristic of the proposed method is that the encoding and transmission process are separated, allowing the tradeoffs to be performed in real time, depending on the available bandwidth, packet losses, or user preference. The presented scheme is also very resilient to packet-losses, since unequal error-protection can be easily employed to provide enhanced protection of the base layer and limited or no protection to the hybrid temporal-SNR enhancement layer.

Spatio – Temporal Scalability

The existing and standardized solutions for spatial scalability are not satisfactory; therefore new approaches are very actively explored.

The application of the MPEG-2 spatial scalability is mostly related to non acceptably high bitrate overheads as compared to single-layer MPEG-2 encoding of video. This additional overhead for MPEG-2 spatial scalability is about 60%–70% of total bit rate. By many test sequences, the total bit-stream is not much smaller than sum of bitstreams obtained for simulcast transmission with two different resolutions.

There were many attempts to improve the scheme of spatial scalability by application of subband decomposition. The idea is to split each image into four spatial subbands (*Figure 24*). The subband of lowest frequencies constitutes a base layer while the other three subbands are jointly transmitted in an enhancement layer.

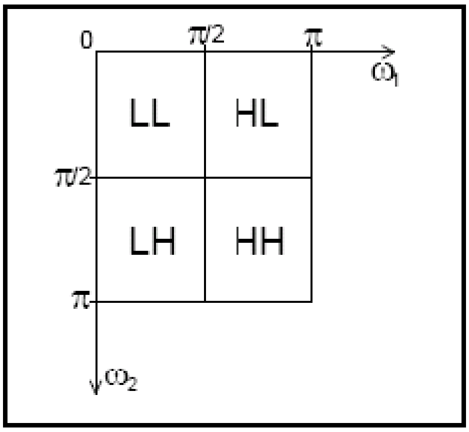


Figure 24: Subband decomposition.

Embedding of subband decomposition into a motion-compensated coder leads to in-band or out-band compensation performed on individual subbands or whole image, respectively. Some experimental results show that the latter is more efficient.

Unfortunately, direct application of the above scheme does not allow controlling flexibly the bitstreams of the base and enhancement layers. Therefore hybrid spatio-temporal scalability was proposed in order to obtain further data reduction in the base layer.

The idea was to combine spatial and temporal scalability. Following assumptions have been made for the solutions proposed:

- The base layer coder is fully MPEG-2 - compatible.
- The coder produces similar bitstreams in the base layer and the enhancement layer.
- Base layer represents video with reduced both temporal and spatial resolutions.

The enhancement layer is used to transmit the information needed for restoration of the full spatial and temporal resolution. Two coder versions have been considered by the authors.

Systems of the first type incorporate B-frames into enhancement layer. Base layer consists of the subband LL from each even frame. Enhancement layer includes subbands LH, HL, HH from each even frame as well as each whole odd frame which is assumed to be a B-frame .

Base layer: I B P B P B P

Enhancement layer: I B B B P B B B P B B B P

Another proposal is to use systems with three-dimensional subband analysis. The input video sequence is analyzed in a three-dimensional (3-D) separable filter bank, i.e. there are three consecutive steps of analysis: temporal, horizontal and vertical.

Temporal analysis results in two subbands Lt and Ht which are partitioned into four spatial subbands (LL, LH, HL and HH) each. For spatial analysis, both horizontal and vertical, separable filters are used. The three-dimensional analysis results in eight spatio-temporal subbands. Three high-spatial-frequency subbands (LH, HL and HH) in the hightemporal-frequency subband Ht are discarded as they correspond to the information being less relevant for the human visual system. Therefore five subbands are encoded:

- In a base layer - the spatial subband LL of the temporal subband Lt .

- The enhancement layer includes the spatial subbands LH, HL and HH from the temporal subband Lt and the spatial subband LL of the temporal subband Ht. The base layer is produced by an MPEG-2 motion compensated coder.

6. Conclusion

Growing demand for Quality of Service (QoS) over today's unreliable packet based networks has motivated the development of a number of considerable error resilient techniques, some of which have already been established in recent coding standards. In this project we review the most worth noticing error resilient approaches in an attempt to describe the generic framework through which error resilience is achieved. We discuss encoder's side means to constitute the compressed bit resilient to errors while balancing the introduced redundancy. We also examine a number of decoder based techniques for reconstructing a corrupted picture providing thus error concealment. Nevertheless, we review some of the latest error control techniques based on encoder-decoder communication, where encoder regulates its operations according to feedback send by the decoder. Despite their obvious effectiveness, these latter approaches suffer from delay constraints constituting them unacceptable for wide usage. Having done so, we conclude that error resilience is an area which has been studied thoroughly over the past years and this is evident by the fact that two of the most recent and widely used coding standards, MPEG-4 and H.263, have included a notable number of error resilient tools (MPEG-2 is mostly used so far). With H.264 just released, it is likely to absorb even more research in the future.

Furthermore, we investigated some Content Adaptation Techniques in order to adapt the content to the desirable rate without the need for re-encoding or regenerating. This can be done xxxxxxxxxx. In fact multi-layered video is not by itself sufficient to provide ideal network bandwidth utilization or video quality, however. To improve the bandwidth utilization of the network and optimize the quality of video received by each of the destinations, the source must respond to constantly changing network conditions by dynamically adjusting the number of video layers it generates as well as the rate at which each layer is transmitted. For the source to do this, it must have congestion feedback from the destinations and the network. This issue will be investigated through the next deliverable D2.1.

To realize video streaming over the Internet with good performance, people have to successfully deal with the problems of heterogeneity and packet loss of the Internet.

For video coding, the new challenge primarily is to achieve good scalability and error resilience performance, while maintaining good compression rate.

In recent years, there have appeared several new transport protocols to improve the QoS support of the Internet for real-time media streaming applications. Which one will finally become the widely accepted industrial standard depends more on market evolvement than on technological development.

Instead of single server-based systems, to effectively serve a large number of clients distributed multi-server based delivery mechanisms (such as RLM that will be mentioned in deliverable D2.1) will be practically applied, which poses a variety of video coding problems. Finding efficient solutions of these problems will be increasingly important for further progress of the media streaming technology.

With the growth of the Internet infrastructure and intensive deployment of distributed media delivery networks, the research focus is now shifting from the simple server-client transmission problem to the optimization of the overall QoS of the server-server-client transmission. In particular, the encoding algorithm, transport protocol, and post-processing method should be designed jointly to achieve the best end-to-end service quality.

The focus of our future work is to investigate some Network Adaptation Techniques (NATs). The basic requirements of NATs are (1) to provide accurate information on the network load, (2) to distinguish between core congestion and wireless link errors, (3) to recognize a change in the possible bandwidth due to changes in the wireless link conditions, and (4) to adapt accordingly the transmission rate at the source.

These decisions will be based on feedback obtained by the receiver. We will examine the possibility of extracting useful information from feedback about the objective quality at the receiver. We expect that the estimation of objective quality will be made possible with the right feedback since the objective quality has a direct relationship with the nature of errors, the actual information content and the capability of concealment and reconstruction at the wireless mobile terminal.

References

[1] Chowdary Adsumilli, and Yu Hen Hu. “Adaptive Wireless Video Communications: Challenges and Approaches, Packet VideoWorkshop”, PA, USA, 2002.

[2] T. Wiegand, N. Färber, K. Stuhlmüller and B. Girod “Error-Resilient Video Transmission Using Long-Term Memory Motion-Compensated Prediction”, IEEE Journal on Selected Areas in Communications, Vol. 18, No. 6, Jun 2000.

[3] Y. Wang, S. Wenger, J. Wen, and A. K. Katsaggelos “Error resilient video coding techniques”, IEEE Signal Proc. Mag., vol. 17, pp. 61--82, July 2000.

[4] R. Yan, F. Wu, S. Li, and R. Tao “Error resilience methods for FGS video enhancement bitstream”, The First IEEE Pacific-Rim Conference on Multimedia (IEEE-PCM 2000), Dec. 13-15, 2000 Sydney, Australia.

[5] J.G. Apostolopoulos “Video Communications and Video Streaming”, May 2001.

[6] J. Kim, R.M. Mersereau and Y. Altunbasak “Error-Resilient Image and Video Transmission Over the Internet Using Unequal Error Protection”, IEEE Transactions on Image Processing, VOL. 12, NO. 2, Feb 2003.

[7] P. Arankalle “Survey on Error Handling and Packet Loss Recovery for Video Streaming in Heterogeneous Networks”, 2003.

[8] W.Y. Kung, H.S. Kong, A. Vetro and H. Sun “Error Resilient Methods for Real-Time MPEG-4 Video Streaming”, TR-2004-049 June 2004.

[9] S. Praveenkumar, H Kalva and B. Furht “An Efficient Application of Video Error Resilience Techniques for Mobile Broadcast Multicast Services (MBMS)”, IEEE Workshop on Wireless Multimedia, Dec 2004.

[10] H.J. Chiou, Y.R. Lee and C.W. Lin “Error Resilient Transcoding using Adaptive Intra Refresh for Video Streaming”.

[11] H.Lee, “Standard Coding for MPEG-1, MPEG-2 and Advanced Coding for MPEG-4”, Report for EE8205, June 1997.

[12] A.Puri, A.Eleftheriadis, “MPEG-4: An object-based multimedia coding standard supporting mobile applications”, Mobile Networks and Applications 3, 1998.

[13] J.Liang, “New Trends in Multimedia Standards: MPEG4 and JPEG2000”, Informing Science – Special Issue on Multimedia Informing Technologies, Vol. 2 No.5, 1999.

[14] F.H.P.Fitzek, M.Reisslein, “MPEG-4 and H-263 Video Traces for Network Performance Evaluation”, Telecommunication Networks Group, October 2000.

[15] D.Wu, Y.T.Hou, W.Zhu, Y.Zhang, J.M.Peha, “Streaming Video over the Internet: Approaches and Directions”, IEEE Transactions on Circuits and Systems for Video Technology, V.11 N.3, March 2001.

[16] E.Gurses,G.B.Akar,N.Akar, “Selective Frame Discarding for Video Streaming in TCP/IP Networks”.

[17] W.Li, “Overview of Fine Granularity in MPEG-4 Video standard”, IEEE Transactions on Circuits and Systems for Video Technology, V.11 N.3, March 2001.

[18] M.Van Der Schaar, H.Radha, “A Hybrid Temporal-SNR Fine-Granular Scalability for Internet Video”, IEEE Transactions on Circuits and Systems for Video Technology, V.11 N.3, March 2001.

[19] M.Domanski, A.Luczak, S.Mackowiak, “Spatio-Temporal Scalability for MPEG Video Coding”, IEEE Transactions on Circuits and Systems for Video Technology, V.10 N.7, October 2000.

[20] M.Domanski, A.Luczak, S.Mackowiak, R.Swierczynski, “Hybrid coding of video with spatio-temporal scalability using subband decomposition”, Poznañ University of Technology, Institute of Electronics and Telecommunications.

[21] M.Domanski, A.Luczak, S.Mackowiak, R.Swierczynski, U. Benzler, “Spatio-Temporal Scalable Video Codecs with MPEG-Compatible Base Layer”, Poznañ University of Technology, Institute of Electronics and Telecommunications.

[22] M.Domanski, A.Luczak, S.Mackowiak, “SPATIO-TEMPORAL SCALABILITY USING MODIFIED MPEG-2 PREDICTIVE VIDEO CODING”.

[23] Thomas Sikora “Digital Video Coding Standards and Their Role in Video Communications”.

[24] E. T. Lin, C. I. Podilchuk, T. Kalker, and E. J. Delp, “Streaming video and rate scalable compression: What are the challenges for watermarking?”, Proceedings of the Security and Watermarking of Multimedia Contents III, January 22-25, 2001, San Jose, CA, Vol. 4314, pp. 116-127.

[25] Panagiotis Papadimitriou, Sofia Tsekeridou, Vassilis Tsaoussidis “Multimedia Streaming over the Internet”, Democritus University of Thrace, Electrical & Computer Engineering Department.

[26] Adam Fuczak, Sawomir Maækowiak, Marek Domañski “Spatio-Temporal Scalability Using Modified MPEG-2 Predictive Video Coding”, Poznañ University of Technology, Institute of Electronics and Telecommunications, Piotrowo 3A, 60-965, Poznañ, POLAND (Internal Paper Number: 4314-16).

[27] Gregory J. Conklin, Sheila S. Hemami “Evaluation of Temporally Scalable Video Coding Techniques”, School of Electrical Engineering Cornell University, Ithaca, NY 14853.

[28] Rosa M. Figueras, Ventura, Pierre Vandergheynst and Pascal Frossard “LOW RATE AND SCALABLE IMAGE CODING WITH REDUNDANT REPRESENTATIONS”, Swiss Federal Institute of Technology Lausanne (EPFL) Signal Processing Institute Technical Report TR-ITS-03.02 Submitted to IEEE Transactions on Image Processing, June 2003. Revised January 2004.

[29] Eren Gurses, Gozde Bozdagi Akar, Nail Akar “Impact of Scalability in Video Transmission in Promotion-Capable Differentiated Services Networks”, Middle East Technical University & Bilkent University.

[30] A.Panayides, M. S. Pattichis, C. S. Pattichis, A. Pitsillides, “A Review of Error Resilience Techniques in Video Streaming,” International Conference on Intelligent Systems and Computing: Theory and Applications (ISYC06), Agia Napa, Cyprus, July 2006, pp. 39-48.