PACKETIZATION AND CONCEALMENT SCHEMES FOR DELIVERING H.263+ CODED VIDEO OVER INTERNET

by

Thomas Reino Huitika

B.Sc. (Electrical Engineering), Lakehead University, 1989

A thesis submitted in partial fulfillment of
the requirements for the degree of

MASTER OF APPLIED SCIENCE

In Department of Electrical and Computer Engineering

We accept this thesis as conforming to the required standard

[redacted]

Dr. P. F. Driessen, Supervisor (Department of Electrical and Computer Engineering)

[redacted]

Dr. P. Agathoklis, Departmental Member (Department of Electrical and Computer Engineering)

[redacted]

Dr. R. N. Horspool, Outside Member (Department of Computer Science)

[redacted]

Dr. M. H. M. Cheng, External Examiner (Department of Computer Science)

Supervisor: Dr. P. F. Driessen

ABSTRACT

Real time streaming H.263+ coded QCIF video for non-interactive security surveillance, unicast and multicast over the Internet using the RTP/UDP/IP transport protocol, must use a packetization scheme that has a high payload efficiency and real time error recovery scheme for lost datagrams due to network congestion.

Presently every 256 bytes of the coded video stream are encapsulated into a datagram without considering, the payload's efficiency, the network's MTU, and an error recovery scheme for datagram loss. Also a ratio of four P-pictures to I-pictures is used for error resilience without considering the network path and conditions.

The first part of the thesis addresses the encapsulation problem by investigating two existing packetization schemes that have efficient payloads, which are based upon the RFC2425 document. The Internet's datagram loss process is then simulated to study the two existing packetization schemes robustness against lost datagrams. Simulation is achieved by using a Markov chain and measured datagram statistics from a known network path and conditions that affect decoding of non-interactive real time streaming video. Simulation of the datagram loss process also determines the number of 'Decoder Error Concealment' failures at each ratio of P-pictures to I-pictures (five to fifteen) using five sets of datagram statistics.

The second part of the thesis addresses the error recovery problem for datagram loss by using 'Decoder Error Concealment'. 'Decoder Error Concealment' is used since it is independent of the various sub-network channel resources and can be simple to

implement. 'Decoder Error Concealment' schemes that motion compensates the previous picture with predicted motion vectors are only considered.

In this thesis a concealment scheme for the picture's missing odd or even slices is proposed. The proposed scheme uses two techniques for concealment. The first technique copies previous picture macroblocks for missing macroblocks that do not contain a lot of motion. The second technique uses a fast motion estimation algorithm for missing macroblocks that contain a lot of motion. The decision between the two techniques is determined by comparing a computed threshold value to a set threshold value. Copying is performed when the computed threshold value is less than or equal to the set value. Motion estimation is performed when the computed threshold value is greater than the set value. The fixed threshold value provides flexibility between the two types of techniques used for concealment. The proposed scheme time is compared to the existing schemes by using PSNR concealment result and CPU time.

Comparison between the concealment results, CPU times, and robustness against datagram loss allows the selection of a packetization and the corresponding concealment scheme for the transport of non-interactive real time streaming video over the Internet.

Examiners:

[redacted]

Dr. P. F. Driessen, Supervisor (Department of Electrical and Computer Engineering)

[redacted]

Dr. P. Agathoklis, Departmental Member (Department of Electrical and Computer Engineering)

[redacted]

Dr. R. N. Horspool, Outside Member (Department of Computer Science)

[redacted]

Dr. M. H. M. Cheng, External Examiner (Department of Computer Science)

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

GLOSSARY OF TERMS

Access Control (Pg. 9): Decides if a 'connection request' is allowed to be carried out by the network. The main considerations behind this decision are current traffic load, current QoS, requested QoS, requested traffic profile, requested pricing, and other policy considerations. The role of the access control algorithm is to ensure that admittance of a new request into a resource constrained network does not violate service commitments made by the network to requests that have already been admitted.

Conditional Probability (Pg. 31): A method used to measure a datagram loss burst. The conditional probability is, assuming datagram n is lost, the probability that n+1, n+2, etc are also lost.

Dynamic Routing (Pg. 9): A routing strategy that is used with the Resource ReServation Protocol (RSVP) that allows the network path followed by packets of a given transmission to change when the network path changes.

Interlaced Scanning (Pg. 4): A scan that traces the odd and even number lines in a picture separately.

Integrated Services Digital Network (ISDN) (Pg. 1): The standardization of user interfaces implemented as a set of digital switches and paths, which supports a broad range of traffic types and processing services.

Motion Compensation (Pg. 22): Process of compensating for the displacement of moving objects from one picture to another.

Motion Estimation (Pg. 5): Process of finding corresponding pixels among pictures.

Progressive Scanning (Pg. 4): A scan that traces the total number of lines in a picture sequentially.

LIST OF SYMBOLS

a, b, c (Pg. 30): Parameters used with the Markov chain model, which express the probability of staying in the indicated states for one more time step.

1-*b*, 1-*c*, *d*, *e* (Pg. 30): Parameters used with the Markov chain model, which express the state transition probability.

b_u (Pg. 54): Upper one-pixel wide boundary of a macroblock.

b_l (Pg. 54): Lower one-pixel wide boundary of a macroblock.

D (Pg. 35): A matrix whose diagonal components are the eigenvalues arranged according to the ordering of the columns of X^*.

f (Pg. 30): Parameter used with the Markov Chain model, which is the mean datagram loss rate.

$\hat{f}_{1,2}$ (Pg. 37): Estimated recorded failure for packetization scheme 1 and 2.

$\hat{f}_1$ (Pg. 36): Estimated recorded failure for packetization scheme 1.

$\hat{f}_2$ (Pg. 36): Estimated recorded failure for packetization scheme 2.

g (Pg. 31): Parameter used with the Markov Chain model, which is the mean datagram out-of-order rate.

h (Pg. 44): Pixel height of the picture.

l (*x*, *y*) (Pg. 54): Sum of the Absolute Differences (SAD) for the lower one-pixel wide boundary of the current picture missing macroblock and the previous picture macroblock located at the same spatial location.

l (Pg. 44): Pixel length of the picture.

m (Pg. 54): Size of the macroblock.

m_v (Pg. 6): Actual value of the motion vector.

m_v_d (Pg. 6): Differently coded motion vector.

m_v_p (Pg. 6): Predictor motion vector.

m_v_1 (Pg. 7): Previous picture's motion vector.

m_v_2 (Pg. 7): Current pictures directly above motion vector.

m_v_3 (Pg. 7): Current pictures above right motion vector.

m_v_a (Pg. 63): Above macroblock motion vector.

m_v_b (Pg. 63): Below macroblock motion vector.

m_v_r (Pg. 63): Right macroblock motion vector.

$m\_v_{(n)}$ (Pg. 55): Current picture's motion vector.

$m\_v_{(n-1)}$ (Pg. 55): Previous picture's motion vector.

k (Pg. 55): Picture's total number of motion vectors.

$picture_{(concealed)}$ (Pg. 44): Decoded concealed picture.

$picture_{(uncorrupted)}$ (Pg. 44): Decoded uncorrupted picture.

$\hat{p}_{1,2}$ (Pg. 38): Estimated probability of failure for packetization scheme 1 and 2.

$\hat{p}_1$ (Pg. 43): Estimated probability of failure for packetization scheme 1.

$\hat{p}_2$ (Pg. 43): Estimated probability of failure for packetization scheme 2.

$p\_d_{1,2}$ (Pg. 38): Overall datagram discard probability for packetization scheme 1 and 2.

p_d_1 (Pg. 40): Overall datagram discard probability for packetization scheme 1.

p_d_2 (Pg. 40): Overall datagram discard probability for packetization scheme 2.

P (Pg. 33): Markov Chain transition matrix.

p_s (Pg. 55): Normalized SAD of the pixels between two consecutive pictures.

q (Pg. 37): Number of trials to improve the estimated probability of failure.

r (Pg. 37): Number of datagrams produced by the ratio of P-pictures to I-pictures.

s (Pg. 34): Steady state distribution vector of P.

s_1, s_2, s_3 (Pg. 34): The states of the steady state distribution vector $\boldsymbol{s}$.

t_h (Pg. 54): Computed threshold value that is determined by finding the maximum value between $u\ (x, y)^*$ and $l\ (x, y)^*$.

t_s (Pg. 26): Time step value for the Markov chain.

$u\ (x, y)$ (Pg. 54): SAD for the upper one-pixel wide boundary of the current picture's missing macroblock and the previous picture's macroblock located at the same spatial location.

v_s (Pg. 55): Averaged SAD of the motion vectors between two consecutive pictures.

(x, y) (Pg. 44): Denotes the coordinates of the upper left corner of the macroblocks.

X (Pg. 35): A matrix whose columns are the eigenvectors of P.

$x(t_n)$ (Pg. 27): Random variable index by time parameter t_n.

λ (Pg. 33): Eigenvalue of matrix P.

$\sigma_{1,2}$ (Pg. 38): Standard deviation for packetization schemes 1 and 2 estimated probability of failure.

ACKNOWLEDGEMENTS

I would like to thank the following individuals:

- Dr. Peter F. Driessen for his overall comments on the thesis.
- Gary Robb (president) of Audio-Visual-Telecommunications (AVT) for the one-year Natural Sciences and Engineering Research Council of Canada (NSERC) Industrial Postgraduate Scholarships (IPS) sponsorship.
- Hyunho Jeon, Nyeongkyu Kwon, and Chengdong Zhang for their invaluable technical support regarding video codecs and for their friendship.
- Rafael Parra Hernandez for his positive encouragement, guidance, and friendship.
- Professor Rodney L. Kirlin for his review and comments regarding the chapter on Datagram Loss Model.
- Dr. Abdul-Imam Al-Sammak for his review and comments regarding the thesis outline.
- Professor Fayez Gebali for his technical support regarding Markov chains.

DEDICATION

For all the long hours I spent working on my class work and the thesis, I am dedicating this thesis to my wife Heather for her support and understanding.

1. INTRODUCTION

One of the many commercial applications for digital video, which is growing in interest, is the broadcast of real time streaming security surveillance video over the Internet to single and multiple clients. The video data requires the appropriate compression standard to be delivered over low and high bitrate networks [29]. Delivery of real time streaming video data over unicast or multicast networks also requires the appropriate transport protocol standard [28]. Section 1.1 addresses the compression topic by providing basic information on the selected compression standard relating to Chapter 2 and 3 simulations. Section 1.2 addresses the protocol topic by providing a description of the selected real time transport protocol. The problems of transmitting compressed video over the Internet using the real time transport protocol is provided in section 1.3. Section 1.4 provides a literature review on previous work that was done to address the problems of section 1.3 and section 1.5 provides the main contributions in the thesis. Chapter 2 simulates the Internet's datagram loss process by a Markov chain using measured datagram statistics. Chapter 3 simulates and compares concealment schemes while Chapter 4 provides an overall summary and direction for future work.

1.1 H.263+ (VERSION 2) VIDEO CODING STANDARD

The ITU-T H.263+ low bit rate lossy video compression standard was developed to facilitate videoconferencing and videophone services over the Integrated Services Digital Network (ISDN). The H.263+ standard is based on the configuration of the ITU-T H.261 standard. The H.263+ standard has a number of basic and optional coding modes added for improved performance and increased functionally [3,18]. More details on the H.263+ coding standard may be found in [3,18,29]. With the specific application of delivery of

video for security monitoring, the H.263+ coding standard does not need to consider bandwidth reservation for speech and control.

1.1.1 Luminance and Chrominance

The tri-stimulus theory of color [29] states that almost any color can be reproduced by appropriately mixing the three additive primaries, red (*R*), green (*G*), and blue (*B*). The luminance component (*Y*) contains the gray information, while the chrominance components (*Cr* and *Cb*) contain the color information of the video images [29].

To reduce the psycho-visual redundancy in the video images, the H.263+ standard chrominance components are sub-sampled by two in both directions. This is done because the activity of the human visual system drops off with increasing spatial frequency. As a result the chrominance components contain half as many pixels compared to the luminance component. For all picture formats the sub-sampling is done so that their block boundaries coincide with *Y* block boundaries shown in Figure 1 [18,29]. For every four *Y* pixels, there are only one *Cr* and one *Cb* pixel. The video picture pixel can have any of the 255 decimal values of an unsigned 8-bit number (2^8) [18].

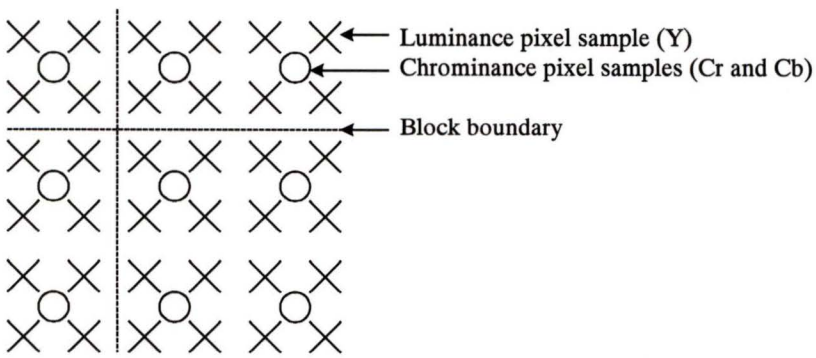

Figure 1: H.263+ Positioning of Luminance and Chrominance Pixel Samples

1.1.2 Hierarchy of Coded Video

The H.263+ standard video picture coding has the following hierarchy:

- Pixels are at the lowest level.
- They are followed by 8 by 8 pixel blocks.
- They are followed by macroblocks that contain four 8 by 8 pixel blocks (16 by16 pixels).
- They are followed by a Group of Blocks (GOB) that contains one or more horizontal rows of macroblocks that equals a slice.
- Finally, they are followed by a picture that contains the entire slice's GOBs shown in Figure 2 [3,18,23].

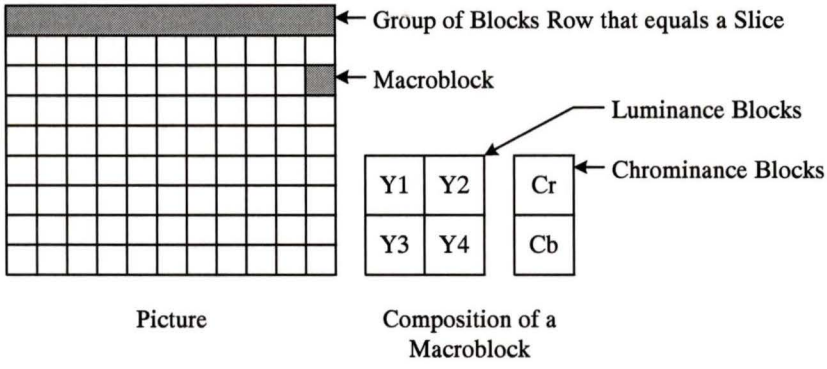

Figure 2: H.263+ Hierarchy of Coded Video

1.1.3 Picture Format

Table 1 shows the picture sizes that are supported by H.263+ standard. The coder determines the format to be used [3,18].

Format	*Y-Picture Size*	*Cr and Cb-Picture Size*
Sub-QCIF	128×96	64×48
QCIF	176×144	88×72
CIF	352×288	176×144
4CIF	704×576	352×288
16CIF	1408×1152	704×576

Table 1: H.263+ Picture Formats

A problem in defining an international standard for video conferencing is that there are two different line and picture rate television standards. North America and Japan systems (NTSC) use 525 lines per interlaced picture at 30 pictures per second while other countries' systems (PAL and SECAM) use 625 lines per interlaced picture at 25 pictures per second. The Common Intermediate Format (CIF) was adopted to eliminate the problem of interoperability among systems with different formats for video conferencing. CIF is based on 352 pixels per line, 288 progressive lines per picture at 30 pictures per second [3,18,29]. Video coders may use the Quarter-CIF (QCIF) for low bit rate applications because it has half the pixels per line and half the progressive lines per picture required for CIF [3].

1.1.4 Picture Types

The H.263+ standard uses the following three types of picture that offer flexibility for coding efficiency and error resilience: Intrapicture, Predicted Picture, and Bi-directional Predicted Picture.

1.1.5 Intrapicture (I-Picture)

Intrapictures (I-pictures) take advantage of the high degree of spatial redundancy between the surrounding pixels for compression. Intracoding is done for each 8 by 8 block in the video picture by using the discrete cosine transform (DCT) algorithm. The 8 by 8 dimension is chosen because it provides a good compromise between efficient spatial redundancy reduction and computation time [3,18,29]. The I-picture coding is similar to the Joint Photographic Experts Group (JPEG) coding standard for still images [3].

1.1.6 Predicted Picture (P-Picture)

Predicted pictures (P-pictures) use a hybrid inter and intracoding scheme so they have a higher amount of compression efficiency when compared to I-pictures. Intercoding uses the high degree of temporal redundancy between the current and previous picture in the video sequence. The objective of intercoding is to determine the macroblock in the current picture that best matches (according to a given criterion) the characteristics of the macroblock in the previous picture by using a block motion estimation process. The motion estimation process produces a motion vector value for each macroblock. Once the best match is found, the difference between the current and previous macroblocks is done to obtain the residual block (prediction error). Then each 8 by 8 block is DCT coded (intracoded) [3,18,29]. The macroblock dimension is chosen because it provides a good compromise between efficient temporal redundancy reduction and computation time when compared to using individual pixels [3].

It is impractical to search the entire picture for the best match so the search is restricted to a search area of $\pm n$ pixels horizontally and vertically from the upper left corner of the macroblock in the current picture [3]. The value of n is typically fifteen. The block motion estimation process may not detect objects that rotate about their center, or zoom in, or vary shading. This may reduce the reliability of the motion vector prediction results [29].

1.1.7 Predicted Bi-Directional Picture (PB-Picture)

Predicted Bi-directional pictures (PB-pictures) are two pictures coded as one unit. The P-picture is the first picture in the unit that is predicted from the previous I-picture or P-picture. The B-picture is the second picture that is predicted from the P-picture in the unit

and the previous I-picture or P-picture. The B-picture name was chosen because parts of the B-picture may be bi-directionally predicted from the previous I-picture or P-picture and the P-picture in the unit. PB-pictures have the highest amount of compression when compared to I-pictures and P-pictures [3,18]. PB-pictures cause coding delays since two pictures need to be buffered and coded. As a result of the coding delay, PB-pictures are not considered because of the time restriction [3].

1.1.8 Precision of Motion Estimation

The H.263+ standard motion estimation process has a half-pixel accuracy for improved prediction results but it requires more computations when compared to integer pixel accuracy. The half-pixel accuracy is found by interpolating the integer-pixel grid by a factor of two (bi-linear interpolation) as shown in Figure 3 [3,18].

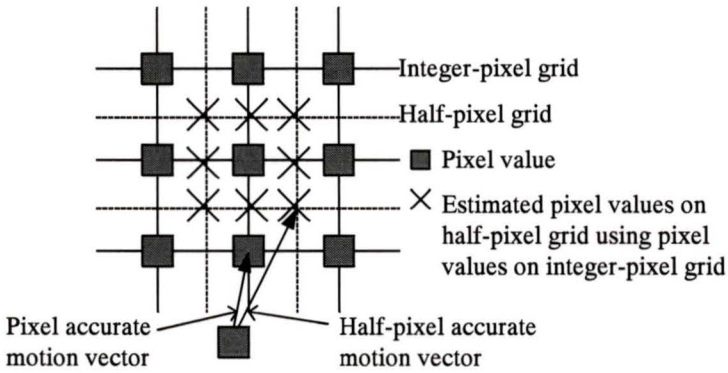

Figure 3: H.263+ Accuracy of Motion Vector Estimation

1.1.9 Motion Vector Coding

The H.263+ motion vectors are differentially coded as given in (1.1) where m_v is the actual value of the motion vector, m_v_p is the predictor motion vector, and m_v_d is the differentially coded motion vector.

$$m\_v_d = m\_v - m\_v_p \tag{1.1}$$

The predictor is the median of the motion vectors from the previously coded ones as given in (1.2) and shown in Figure 4 [3,18].

$$m\_v_p = \text{median}\left(m\_v_1, m\_v_2, m\_v_3\right) \tag{1.2}$$

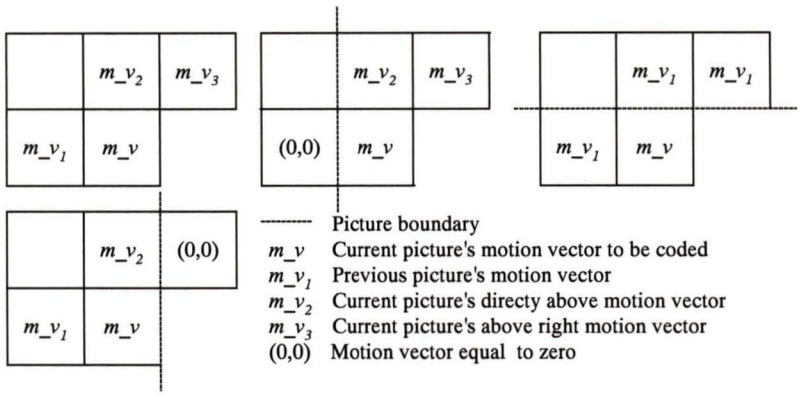

Figure 4: H.263+ Predictor Motion Vector

1.1.10 Optional Unrestricted Motion Vector Mode

Macroblocks at the border of the current picture will have sub-optimal prediction if the corresponding motion vectors are restricted to point at pixels located inside the previous picture border. The unrestricted motion vector mode occurs when the motion vector is allowed point outside the previous picture boundary to a larger previous picture boundary. As a result the unrestricted motion vector mode produces better prediction results for objects moving near the picture boundary. Better prediction results especially occur for smaller picture formats such as QCIF (extension from 176×144 to a new previous picture boundary of 192×160 pixels). The closest edge pixel is used for the pixel referenced by the motion vector that points outside the picture boundary. The unrestricted motion vector mode includes an extension of the motion vector range from {–16 15.5} to {-31.5 31.5} pixels horizontally and vertically so larger motion vectors can be used if the calculated predictor motion vector is outside the range of {-15.5 16} [3,18].

1.1.11 Optional Slice Structure Mode

Slice error resilience is achieved by having the data dependencies such as the coded prediction of the macroblock DCT coefficients and motion vectors confined to the slice boundaries or headers within the current picture. The slice headers located within the coded bit stream act as resynchronization points for the decoder. As a result the slice structure mode makes the coded video stream more agreeable to use with datagram delivery because it restricts error propagation to the boundary of the slice [18,33]. The H.263+ slice mode is an improvement over the H.263 GOB mode that requires the GOBs to be coded and decoded in order. The out-of-order slice decoding within a picture reduces the decoding delay [11,18].

1.1.12 Optional Rate Control Mode

The H.263+ standard supports input source picture rates from 1 to 30 pictures per second with and without rate control [18]. Lower source picture rates produce lower output bit rates and lower coded picture quality when compared to the original picture [3]. Rate control attempts to generate a constant bit rate (CBR) while no rate control generates a variable bit rate (VBR). VBR produces constant video quality, however a large number of factors influence the output bit rate (e.g., compression mode). As a result it is more difficult for a network to guarantee a certain Quality of Service (QoS) with VBR than it is with CBR because the high burstiness due to VBR leads to router buffer overflow and network congestion [14,31].

1.2 VIDEO TRANSPORT PROTOCOL

The Real Time Protocol/User Data Protocol/Internet Protocol (RTP/UDP/IP) datagram used for real time streaming video over the Internet has become very popular because IP

is the most widely used internetworking protocol [28]. IP has become popular since it is an inexpensive network service to operate [8] and it can provide service in many different types of networks. IP provides a connectionless or a datagram service between end systems (not a predefined route through the network), which makes IP robust. IP is robust because if a router fails or is congested, the subsequent datagrams may bypass that router [28]. IP provides no QoS mechanisms for datagram loss and orderly delivery. IP assumes that all traffic sources are well behaved.

A solution to the IP QoS problem is the Transmission Control Protocol (TCP). TCP provides QoS mechanisms for datagram loss and sequence delivery through retransmissions [8,28]. TCP is not considered in this thesis due to the real time restriction and multicast delivery over a vast number of sub-networks that may have uneven resources of bandwidth and processing [7,8,10].

Another solution to the IP QoS problem is to have routers support resource reservation such as the Resource ReSerVation Protocol (RSVP), access control techniques, and scheduling mechanisms (dynamic routing) within the networks in the Internet [8,28]. The above solutions are not considered in this thesis since all the routers on the path from the transmitter to the receiver may not have the mechanisms to support QoS. QoS networks will require tariffs, which is more expensive than the use of a service without QoS guarantees. It may also be a relatively long time before IP services with QoS will be readily available with a reasonable tariff [8].

IP datagrams are therefore susceptible to network congestion in routers. Network congestion occurs when datagrams arrive too fast for the router to process them (make routing decisions) so the router's buffer starts to fill up. Routers will try to discard or

reroute any incoming datagram for which there is no available buffer space. Rerouting leads to out-of-order deliveries and jitter [8,28,34].

The following steps are required for non-interactive real time streaming video to be delivered from the sender to the receiver. The video is:

1. Coded for delivery over low and high bitrate networks.
2. Segmented to permit the encapsulation of the various protocol layers.
3. Encapsulated into the RTP/UDP/IP layers for transportation through the network.
4. Transmitted through the Internet via network routers.
5. De-encapsulated from the RTP/UDP/IP layers to retrieve the data delivered from the network.
6. Decoded for viewing.
7. Viewed.

The layered architecture for Internet transportation is shown in Figure 5 [8].

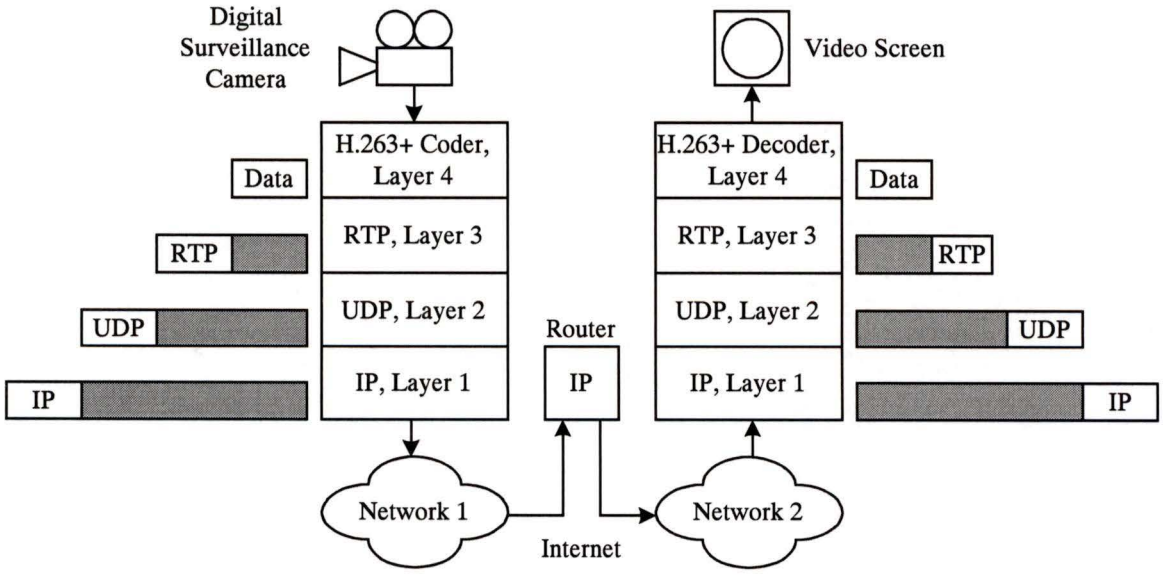

Figure 5: Layer Architecture for Video Transportation

1.2.1 Internet Protocol (IP, Layer 1)

IP's major function is to interface with the higher protocol layer for the sending or receiving of a datagram. Some of the relevant IP header fields are:

- Source address (Internet work address of sending IP entity).
- Destination address (Internet work address of destination IP entity).
- The protocol (recipient protocol entity such as UDP).
- Don't fragment identifier (indicates whether IP can fragment data to accomplish delivery).
- Data length (length of data being transmitted).

IP has a checksum. The checksum only applies to the IP header and not the data field. It is noted that IPv6 (version 6) is available. IPv4 (version 4) is the protocol considered in this thesis since IPv4 is currently more widely used when compared to IPv6 [28]. A detailed IPv4 header format and the corresponding field descriptions are provided in [28]. Similar to IP, UDP and RTP do not provide mechanisms for lost datagrams and sequence delivery [9,28].

1.2.2 User Data Protocol (UDP, Layer 2)

UDP operates above IP and its major function is to provide source and destination port addressing for IP. UDP also has a checksum. The checksum applies to the entire UDP/IP segment (UDP header and user payload) plus a pseudoheader prefixed to the UDP header at the time of calculation. The pseudoheader includes the source and destination Internet address protocol fields from the IP header. The pseudoheader therefore protects against a misdelivery by IP. A checksum will not be made on the user payload if a UDP checksum is not performed. UDP checksum is therefore used in this thesis instead of the IP

checksum. A detailed UDP header format and the corresponding field descriptions are provided in [28]. The details of the checksum calculation are provided in the appendix.

1.2.3 Real Time Protocol (RTP, Layer 3)

RTP operates above UDP and its major function is to provide end-to-end delivery services for data with real time characteristics such as voice and video. Some of the relevant RTP header fields are:

- Marker bit (indicates datagram stream events; it is set to 1 when the current datagram carries the end of current picture, 0 otherwise).
- Payload type (identifies the format and the application of the payload).
- Sequence number (ordering of datagrams for the detection of possible losses).
- Time stamp (performs synchronization and measures jitter), identification of the synchronized source(s) (ensures that no two synchronized sources within the same RTP session will have the identical identifier).
- Contributing sources (identifies the contributing sources for the payload in the datagram).

Version 2 is used in this thesis since it is the latest version. A detailed RTP header format and the corresponding field descriptions are provided in [9].

1.2.4 Payload Efficiency

The combined header size of a RTP/UDP/IP datagram is:

- 20 bytes for IP.
- 8 bytes for UDP.
- 12 bytes for RTP.

The RTP/UDP/IP datagram header size of 40 bytes requires high payload efficiency, however it must take into account the network's Maximum Transfer Unit (MTU). IP has a MTU of 65,536 bytes (if 'don't fragment identifier' is enable) because the IP datagram length field is 16 bits $\left(2^{16}\right)$ [10,28]. The IEEE 802.3 Frame Format for the Ethernet Local Area Network (LAN) standard has a MTU of 1,500 bytes [7,10,28]. Datagrams transmitted over the Ethernet LAN that exceed 1,500 bytes will be split into at least two or more separate packets depending on original datagram's size. The datagram loss rate can increase by the factor of the split since the entire datagram will be lost if any of the corresponding split packets are lost during the transmission over the Ethernet LAN [7]. The maximum size of a RTP/UDP/IP datagram is therefore 1,500 bytes, which allows a payload size of 1,450 bytes or 11,600 bits [7,10].

RTP/UDP/IP header compression is possible for datagrams sent over the Point-to-Point Protocol (PPP) if each end of the link agrees on a set of configuration parameters for the compression scheme [6]. Header compression is not considered in this thesis because it may not work on networks not using the required set of configuration parameters. Header compression is also not considered since the payload data is selected to be as large as possible for efficiency [6,10].

1.3 PROBLEMS

The current situation has every 256 bytes of the H.263+ coded QCIF video stream encapsulated into a RTP/UDP/IP datagram. The current encapsulation scheme does not have high payload efficiency for the RTP/UDP/IP datagram header size of 40 bytes and the network's MTU of 1500 bytes. It also does not work in conjunction with the H.263+

standard's optional slice structure mode for packetization and error recovery for datagram loss [10].

An error recovery scheme is not presently implemented for datagram loss. Erasure error caused by datagram loss will greatly reduce the quality of the decoded P-pictures since they are intercoded. The erasure error will appear and propagate through the remaining P-pictures until the next I-picture. Error recovery schemes must take into account multicast delivery of real time streaming video in a packet-switching network using the RTP/UDP/IP transport protocol. RTP/UDP/IP provides no QoS mechanisms for datagram loss and sequence delivery [33]. Error recovery schemes must also take into account the size and complexity of the Internet where sub-networks may have uneven resources such as bandwidth and processing [13,34]. It is difficult therefore to provide an efficient and flexible error recovery scheme for the above transport environment.

Intracoding may be used for error resilience because it will stop error propagation [7,10,32,34]. A video coder and the corresponding decoder, which takes into account prior knowledge of the packetization and error recovery scheme and datagram loss rates for intracoding, is proposed in [10]. It was shown that the above combination produced good results for datagram loss rates of 20% and when only error recovery for the P-picture is required. The I-picture is presently used for error resilience because it is simple to implement compared to the scheme proposed in [10]. A ratio of four P-pictures to I-pictures with rate control is used without considering the network path or conditions.

Typically the H.263+ standard's rate control algorithm is designed to operate at a high ratio of P-pictures to I-pictures since I-pictures have lower compression efficiency when compared to P-pictures. I-pictures may be inserted at any time, however the rate

control may be exceeded if the ratio of P-pictures to I-pictures is too low. Quantization step size and picture skipping (temporal sub-sampling) are two parameters that may be varied to control the rate of coded video. An increased quantization step size results in fewer bits allocated to each picture there-by reducing the quality of the coded picture. Skipping complete pictures is equivalent to temporal sub-sampling [3,18,29]. Using the I-picture for error resilience therefore may greatly reduce the visual quality of the coded pictures if the ratio is too low with rate control [18,29]. As a result there is a tradeoff between using the I-picture for error resilience and picture quality with rate control.

1.4 PREVIOUS WORK

Section 1.4 addresses the problems of error recovery for datagram loss and payload efficiency by providing:

- A description of an error recovery scheme for datagram loss.
- A description of two types of packetization schemes proposed in [10] based upon the RFC2425 document that specifies the RTP payload format for encapsulating H.263+ coded video [5].
- A review of error recovery schemes for the two types of packetization schemes proposed in [10].
- A description of the de-packetization and decoding process for the two types of packetization and error recovery schemes.
- A description of a model used to simulate the datagram loss process of the Internet. The model is used to study the effects of datagram loss on the two types of packetization and error recovery schemes.

1.4.1 Error Recovery Schemes

An excellent review for error recovery schemes of real time streaming video transported over a variety of unreliable networks is provided in [32,33,34]. 'Forward' and 'Decoder Error Concealment' schemes have been developed for the situation of real time streaming video sent over the Internet using a transport protocol that provides no QoS mechanisms for datagram loss and sequence delivery.

'Forward' schemes combat the effect of erasures due to datagram loss by adding redundancy to the coded video such as Forward Error Correction (FEC) codes. The H.263+ standard has an optional Bose, Chaudhuri, and Hocquengem (BCH) FEC code and framing pattern mode [18]. Many other types of FEC based schemes have also been widely examined and implemented for coded video transported over unreliable channels. Adding redundancy to coded video may be prohibitive for sub-networks that have uneven resources such as bandwidth and processing and becomes unnecessary when datagram loss do not occur.

'Decoder Error Concealment' or concealment uses the decoder to provide compensation for network errors by using only the original recovered data (not using any additional information from the coder). The advantages of concealment are that it does not increase the bit rate, it is performed only when datagram loss occur, it is independent of the various sub-network channels resources, and it is simple to implement for the real time restriction [1,13,32]. Concealment is therefore selected as the error recovery scheme for datagram loss.

Concealment must work in conjunction with error resilience. The primary goal of a video coder is to eliminate, as much as possible, the visual and statistical redundancy of

the video sequence to achieve the best compression gain. This approach makes the task of limiting the error propagation at the decoder difficult. A solution to this problem is error resilience. Error resilience is achieved by limiting the prediction within non-overlapping regions (i.e., sacrificing some compression by having the coder intentionally keep some of the redundancy) [32]. An excellent review for error resilient video coding techniques of real time streaming video transported over unreliable networks is provided in [32,33]. The H.263+ standard's optional slice structure error resilience mode is used in [10,11] for the specific situation of using the RTP/UDP/IP transport protocol for the delivery of H.263+ coded QCIF video over the Internet.

1.4.2 Packetization Schemes using RFC2425

Using low source picture rates, coding of the QCIF video test sequence permits the I-picture and the P-picture to be placed into a single datagram while still being below the maximum payload size of 11,600 bits. Packetization scheme 1 places each picture into a single datagram so the payload efficiency is high. Loss of a single datagram means the loss of an entire coded picture, which is not desirable for concealment [10]. Also placing the entire I-picture into a single datagram while being below the maximum payload size may not be possible at higher source picture rates for video sequences containing a lot of detail and motion.

It is more desirable to divide a coded picture into a large number of datagrams (i.e., a picture's single slice is placed into a single datagram) for concealment because the spatial area affected by a single datagram loss is reduced [10]. This process reduces payload efficiency [10]. Complexity for concealment will increase because of the number of different datagram loss combinations that result in different areas of the picture needing

to be recovered. Consecutive datagram loss may result in the loss of the picture's consecutive slices and reduce the concealment result. Packetization scheme 2 utilizes the H.263+ standard's error resilience slice mode by interleaving each picture's odd and even slices into separate datagrams, which is the optimal compromise between concealment and payload efficiency [10].

The Network Working Group (NWG) standard for RFC2425 is a document that specifies the RTP payload format for the 1998 version of the International Telecommunication Union-Telecommunication Standardization Sector (ITU-T) H.263+ coded video. The goals of the payload format are to specify an efficient way of encapsulating the H.263+ standard compliant bit stream and enhance resiliency (working with H.263+ error resilience tools) against datagram loss for transmission over the Internet. It is noted that RFC2425 is based upon and does not replace RFC2190 the standard for the encapsulation of H.263 coded video [5,38]. The following sections describe the NWG RFC2425 RTP payload format for packetization schemes 1 and 2 and the corresponding de-packetization process.

The H.263+ payload header is present for each RTP/UDP/IP datagram and is used to define the datagram's payload. The layout of the RTP H.263+ video datagram is shown in Figure 6 [5].

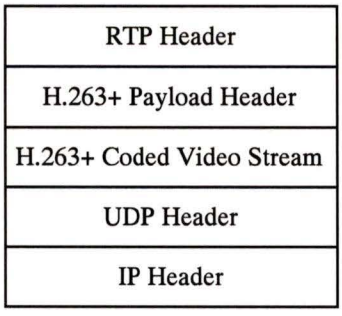

Figure 6: Layout of the RTP/UDP/IP H.263+ Video Datagram

When transmitting H.263+ coded video streams over the Internet, the output of the coder may be packetized directly with the exception of when the payload of the datagram begins with a picture or a slice. At the start of each coded picture, a picture header is created followed by a picture or a picture's slices. The picture header contains the necessary information for the decoder to decode the individual picture. Some of the information that the picture header contains is picture size (e.g., QCIF), picture coding (e.g., P or I), optional coding modes (e.g., unrestricted motion vector), and selected reference picture (e.g., P or PB). If the datagram containing a picture header is lost, the corresponding picture will use the default decoding parameters and may not be able to be decoded.

A H.263+ coder allows picture headers to be sent in an abbreviated form in order to prevent repetition of overhead information that does not change from picture to picture. For error resiliency sending a complete picture header (not abbreviated) by including the required data field information in the H.263+ coder [18] for each coded picture is done for environments where significant datagram loss occur at an increased cost of header information [5,10,18].

The general H.263+ payload header structure and the corresponding field description are shown in Figure 7 [5]. More detailed information on the payload's data fields are provided in [5].

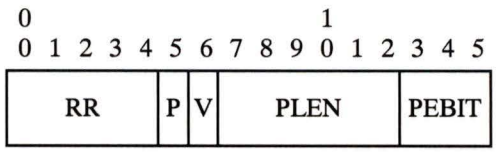

Figure 7: H.263+ Payload Header

The data fields in Figure 7 are defined as follows:

RR: Five reserved bits all set to zero.

P: A single bit indicating a picture or slice start, and the end of a video sequence. This bit allows the omission of the first two bytes of the start codes.

V: A single bit indicating the presence of an eight-bit field containing information for Video Redundancy Coding (VRC). For error resiliency VRC is not used in this thesis.

PLEN: Six bits indicating the length in bytes of the extra picture header for the packetization scheme that places each picture's odd and even slices into separate datagrams [10]. If no extra picture header is attached, PLEN is zero. If PLEN>zero the extra picture header is attached immediately following the rest of the payload header. The length reflects the omission of the first two bytes of the Picture Start Code (PSC).

PEBIT: Three bits indicating the number of bits ignored in the last byte of the picture header. If PLEN is not zero, the ignored bits shall be the least significant bits of the byte. If PLEN is zero then PEBIT shall be zero. It is used when an extra picture header is attached, as signified by PLEN>zero.

Example of packetization scheme 1 for a coded QCIF video picture is shown in Figure 8. Datagram 1 payload starts with the last six bits of the PSC ('100000'), adding the complete original picture header, and then adding the bit stream of the entire coded video picture [5].

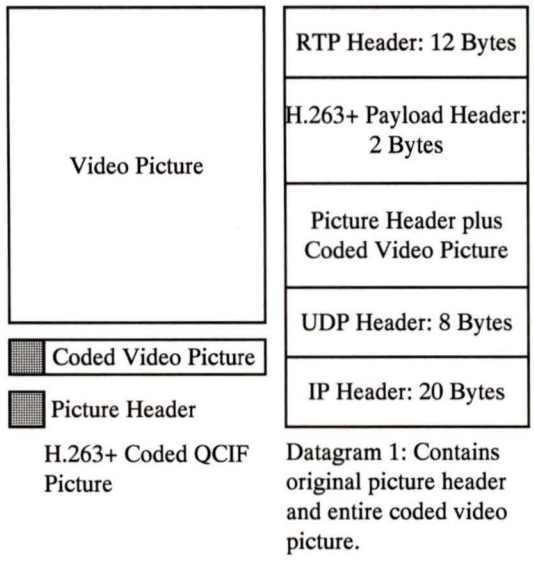

Figure 8: RFC2425 Packetization Scheme 1

Example of packetization scheme 2 for a coded QCIF video picture is shown in Figure 9 [10]. Datagram 1 payload follows the same procedure as datagram 1 payload of packetization scheme 1 except the bit stream of the coded odd slices are added instead of the entire video picture. Datagram 2 starts with the last six bits of the PSC, adding the redundant picture header, and then adding the bit stream of the coded even slices. The H.263+ payload header size of two bytes does not contribute to the overall datagram header size because the payload header omits the first two all zero bytes of the PSC. The first 2 bytes of the PSC are used for synchronization and are represented by the 'P' bit of the H.263+ payload header [5,10].

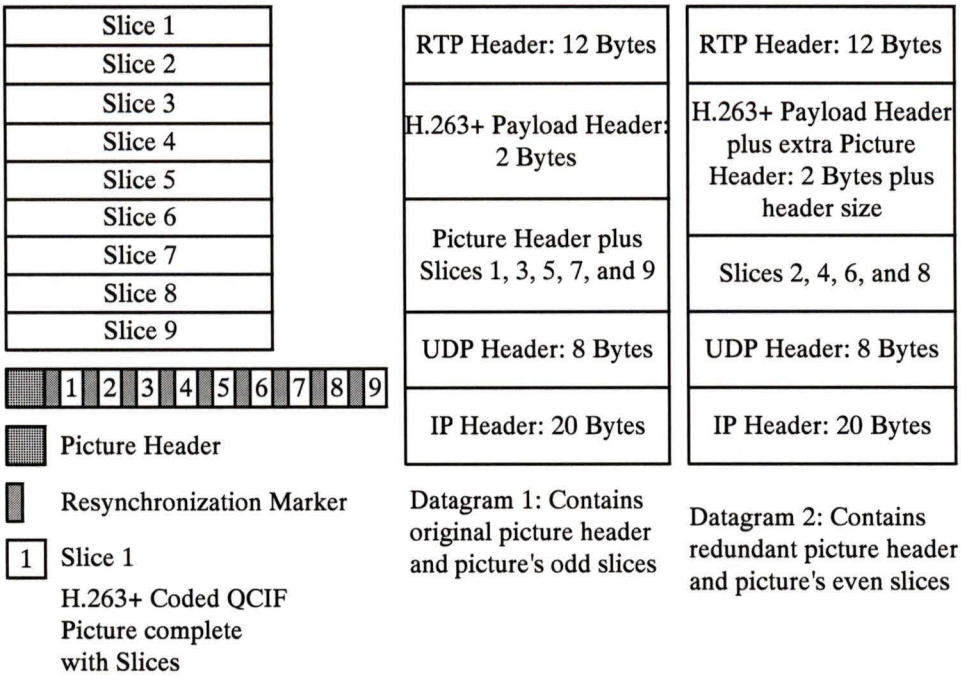

Figure 9: RFC2425 Packetization Scheme 2

Packetization schemes 1 and 2 proposed in [10] improved the current packetization situation, which simply encapsulates every 256 bytes of the coded video stream into a single datagram without considering payload efficiency and the network's MTU.

1.4.3 Concealment for Packetization Schemes

Concealment can be divided into the three following categories: spatial, temporal, and hybrid spatial/temporal. Spatial schemes exploit correlation within the current picture. Temporal schemes exploit correlation between consecutive pictures. Hybrid schemes exploit correlation between the current and consecutive pictures [1]. The type of scheme to be implemented depends upon the type data available for concealment.

Concealment schemes must take into account packetization schemes 1 and 2, which have efficient payloads. When the lost area becomes large, spatial concealment by exploiting the resemblance between neighboring pixels [12,32] or their transform coefficients [35] becomes useless. Improved concealment is achieved by copying the

missing parts from a motion compensated previous picture after predicting the lost motion vectors [10,27]. Concealment schemes that predict lost motion vectors are only considered in this thesis because of the large amount of lost spatial area due to a datagram loss and can be simple to implement for the real time restriction.

A lost macroblock motion vector can be estimated by the following schemes: zero, spatial, temporal, hybrid, and decoder motion vector estimation. The zeros scheme uses the previous picture macroblock located at the same spatial location of the missing macroblock (zero displacement). The spatial scheme uses the current picture, recovered, and neighboring macroblock motion vectors. The temporal scheme uses the previous picture, recovered, and neighboring macroblock motion vectors. The hybrid scheme uses either the zero, spatial, or temporal macroblock motion vector depending upon which scheme produces the lowest distortion value between the border of the concealed and the surrounding recovered macroblocks [1,32,33]. The decoder motion vector estimation scheme is a process similar to the motion estimation process performed at the coder [36].

Spatial macroblock motion vectors are not available for I-pictures since they are intracoded. The Moving Pictures Expert Group version No. 2 (MPEG-2) standard calculates and sends motion vectors as side information as a concealment scheme for the I-picture [32,36]. MPEG is the video compression standard for the storage of coded video on digital storage media. This concealment scheme is not considered because the H.263+ coder must be modified and it requires additional coding time. Also the redundant motion vector data becomes unnecessary when datagram losses do not occur. Therefore previous picture macroblocks, previous picture macroblock motion vectors, and decoder motion

vector estimation are the only existing schemes considered in this thesis for the estimation of the I-picture's missing macroblock motion vector [36].

P-picture concealment schemes using the spatial adjacent macroblock motion vectors are only considered because simulation results in [21] show that the missing macroblock motion vector is more correlated with the neighboring spatial macroblock motion vectors when compared to the neighboring temporal macroblock motion vectors. Therefore the above, median, average, and hybrid (i.e., temporal or spatial) are the only existing schemes considered in this thesis for the estimation of the P-picture's missing macroblock motion vector [32].

1.4.4 De-packetization and Decoding using RFC2425

Depending on the datagram loss situation two different situations may occur at the receiver for packetization scheme 1.

1. Datagram 1 is correctly received and detected by the RTP header. The coded bit stream is then sent to the decoder.
2. Datagram 1 is lost and detected by the RTP header. The decoder is notified of the situation. The decoder performs concealment by using the previous picture.

Depending on the datagram loss situation four different situations may occur at the receiver for packetization scheme 2 [10].

1. Datagrams 1 and 2 are correctly received and detected by the RTP header. The redundant picture header is discarded from datagram 2 and the coded bit stream is fed to the decoder.
2. Datagram 1 is correctly received while datagram 2 is lost and is detected by the RTP header. The decoder is notified of the situation. The even slices bit stream is

then sent to the decoder. Concealment is performed for the missing odd slices by using the recovered even slices and the previous picture.

3. Datagram 2 is correctly received while datagram 1 is lost and detected by the RTP header. The decoder is notified of the situation. The odd slices bit stream is then sent to the decoder. Concealment is performed for the missing even slices by using the recovered odd slices and the previous picture.
4. Datagrams 1 and 2 are both lost and are detected by the RTP header. The decoder is notified of the situation. Concealment is performed for the current missing picture by using the previous picture. The previous picture is used for concealment since the RTP/UDP/IP transport protocol does not provide any QoS mechanisms for current missing picture [10].

Packetization scheme 1 has two possible de-packetization and decoding situations while packetization scheme 2 has four possible de-packetization and decoding situations. Packetization scheme 1 is therefore simpler to implement when compared to packetization scheme 2.

1.4.5 Datagram Loss Modeling

Measured datagram loss statistics in [4,7,25] indicate that datagram loss occurs in bursts. Each datagram is also treated independently, with no reference to datagrams that have gone before [28]. The Internet's bursty datagram loss process is also mentioned in [10]. However no work has been done in [10] to study the effects (robustness against datagram loss) of the Internet's bursty datagram loss process on packetization and concealment schemes 1 and 2. Studying the affects of the Internet's bursty datagram loss process on packetization and concealment schemes 1and 2 can also be used to determine the optimal

ratio of P-pictures to I-pictures for a known network path. Using an appropriate model to simulate the Internet's datagram loss process may accomplish the above task.

A Markov chain (on/off model) is used in [2,14,24,37] and suggested in [25]. A Markov chain is therefore very popular and will be used in this thesis to model the Internet's bursty datagram loss process. It should be noted that the Pareto distribution, which is a heavy tail distribution, was used in [4] to model the bursty datagram loss process. A two-state Markov chain or the Gilbert model generally models the datagram loss process shown in Figure 10 [2,14,24,37].

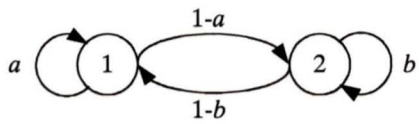

Figure 10: Datagram Loss Model using a Two-State Markov Chain

A single datagram that is correctly received from the network is represented by state one. A single datagram that is lost by the network is represented by state two. Parameters a and b express the probability of staying in the corresponding state for one time step. The increase of a and b probabilities value increases the probability of remaining in states one or two for another time step. Parameters 1-a and 1-b express the state transition probability for one time step [14,37].

A Markov chain is a special case of the Markov process where time is discrete, the discrete time instants are separated by t_s, and are indexed by $n = 0,1,...$ (i.e., $t_n = nt_s$). The time step value t_s depends upon the system being studied. For the particular case of modeling the Internet datagram loss process, the states of the Markov chain are updated with the arrival or non-arrival of the datagram from the Internet. This type of Markov chain is synchronous because it is consistently updated [14].

A Markov process is a random process where the model's future state depends only on its present state. This is called the memory-less property of a Markov process. The transition between states depends only on the transition probabilities. The conditional probability expression for the Markov process (the probability that event x_n occurred given that event $x(t_{n-1})$ took place) is given by (1.3) where $x(t_n)$ is a random variable indexed by the time parameter t_n.

$$p\left[x(t_n) = x_n \mid x(t_{n-1}) = x_{n-1}\right] \tag{1.3}$$

A two-state Markov chain used for modeling the video datagram loss process in [2,37] does not use measured datagram statistics. The accuracy of the model will improve by using measured datagram statistics that affect decoding of non-interactive real time streaming video.

1.5 MAIN CONTRIBUTIONS

In this thesis a concealment scheme for the QCIF I-picture and P-picture's missing odd and even slices is proposed. Comparison is done between the proposed and existing scheme's overall mean concealment results and corresponding CPU times.

In this thesis a Markov chain using measured datagram statistics that affect the decoding of non-interactive real time streaming video simulates the Internet's datagram loss process. Simulated datagram loss process is used to estimate the number of concealment failures for the two different packetization schemes proposed in [10] using increasing ratios of P-pictures to I-pictures. The estimated number of concealment failures improves the tradeoff between using the I-picture for error resilience and picture quality with rate control for a known network path. Simulated datagram loss process is also used to estimate the probability of a concealment failure for the two different

packetization schemes proposed in [10]. Estimated probability of a failure is used to compare the two different packetization schemes robustness against datagram loss.

Comparison between the estimated number of concealment failures, robustness against lost datagrams, concealment results, and CPU times allows the selection of a packetization and concealment scheme for the transport of non-interactive real time streaming H.263+ coded QCIF video over the Internet.

2. DATAGRAM LOSS MODEL

Chapter 2 is organized as follows:

- Section 2.1 describes the datagram statistics that affect decoding of non-interactive real time streaming video.
- Section 2.2 describes the development of the datagram loss model, the datagram loss process, concealment failure for packetization schemes 1 and 2, estimated probability of a concealment failure, and the datagram loss model summary.
- Section 2.3 provides the simulation results.

2.1 DATAGRAM STATISTICS

For a real time streaming video datagram to be considered useful it must be decoded immediately after it is received. As a result out-of-order delivery and jitter are considered as a datagram loss [7,34]. Datagram delays are not considered as a datagram loss since the delay is only noticeable at start-up and transparent after start-up. Non-interactive applications can tolerate delays of a second or more. Typical delay requirements for audio-visual applications [8] are:

- Tight: up to 200×10^{-3} s.
- Medium: 200 to 500×10^{-3} s.
- Loose: greater than 500×10^{-3} s.

The RTP/UDP/IP transport protocol has no QoS mechanisms for datagrams that are discarded by the network. Discarded datagrams are therefore considered as a datagram loss and occur when:

- Congested routers discard any incoming datagram for which there is no available buffer space [8,28,34].

- Noise on a network connection link changes one or more bits in the datagram. The bit change(s) are detected by the router's checksum and the datagram is discarded [37].

A corrupted datagram delivered to the H.263+ decoder is considered a datagram loss because the decoder may not be able to decode the corrupted data [25]. Datagram corruption occurs when the router's checksum does not detect a certain sequence of bit changes[1] caused by noise and the network connection delivers the corrupted datagram to the receiver [25,28].

The datagram loss process of the Internet channel for non-interactive real time streaming video can be related to the following datagram statistics:

- Out-of-order delivery and jitter (magnitude of delay variation).
- Discard due to congested routers and connection link noise.
- Corruption.

2.2 DATAGRAM LOSS MODEL

A three-state Markov chain using measured datagram statistics shown in Figure 11 is used to model the Internet datagram loss process of non-interactive real time streaming video. A single datagram that is correctly received from the network is represented by state one. A single datagram that is lost due to congested network routers is represented by state two. A single datagram that is lost due to out-of-order delivery is represented by state three. Parameters a, b, and c express the probabilities of staying in the corresponding states for one time step. Parameters 1-b, 1-c, d, and e express the state transition probabilities for one time step. Parameter f expresses the mean datagram loss

[1] See appendix for more details.

rate from state one to state two while parameter g expresses the mean datagram out-of-order rate from state one to state three [37].

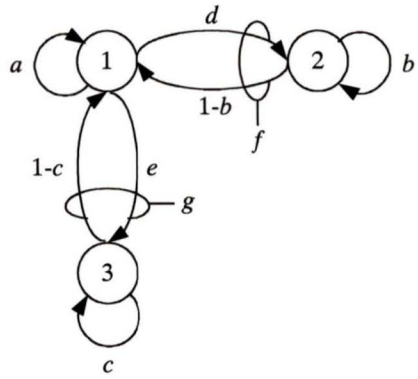

Figure 11: Datagram Loss Model using a Three-State Markov Chain

Measured Internet RTP/UDP datagram statistics such as loss rate, out-of-order delivery, and corresponding conditional probabilities were recorded in [7] for three sites, two data rates, and different periods of time. The video test sequence was approximately five minutes long and MPEG-1 coded at 30 pictures per second. The video test sequence also had a ratio of P-pictures to I-pictures equal to fifteen. The packetization scheme had I-pictures and P-picture's single row slices interleaved into separate datagrams and entire Bi-directional predicted pictures (B-pictures) placed into a single datagrams [7]. MPEG B-pictures are similar to H.263+ PB-pictures except B-pictures are coded as a single unit.

A TCP/IP datagram has the same checksum as a UDP data datagram [25]. The number of TCP/IP corrupted datagrams therefore can be used for the number of corrupted RTP/UDP/IP datagrams. The number of measured Internet TCP/IP datagrams corrupted in transit provided in [25] is approximately 1 in 5,000 or a rate of 0.02%. The corrupted datagrams statistic is negligible when compared to measure datagram loss rates from [7] of 1.5, 5.6, 9.0, 12.4, and 21.1%. The state for corrupted datagrams was therefore not included in the model.

Jitter was not included in the model since measured statistics could not be found for non-interactive real time streaming video. Datagrams discarded by network channel noise were not included in the model since measured statistics also could not be found.

2.2.1 Calculation of Model Parameters

The values of b are provided by the measured conditional probability curves for the f values of 1.5, 5.6, 9.0, 12.4, and 21.1%. The conditional probabilities were measured assuming an infinite buffer (a datagram was counted as received even though it was delayed). The conditional probability at a datagram distance of one is less than 35% for all the mean datagram loss rates curves. This indicates that the majority of datagram loss has a burst length of one. Parameter g is equal to f because it was found that the measured datagram out-of-order and mean loss rates almost have identical values for three different sites with two different data rates. States two and three are not connected because f and g statistics are recorded independently of each other. Parameter c is half of the value of g because it was found that the measured percentage out-of-order datagram curves for different number of datagram delays value decreased by approximately half at a datagram delay value of one [7].

The general 3 by 3 transition or Markov matrix for the three-state Markov chain is represented by (2.1). Element p_{ij} represents the transition probability from the present state j to the next state i (i.e., p_{12} represents the probability of the event making the transition from state 2 to state 1).

$$P = \begin{array}{c} \\ \text{Next State} \end{array} \begin{array}{c} \text{Present State} \\ \begin{array}{ccc} 1 & 2 & 3 \end{array} \\ \begin{array}{c} 1 \\ 2 \\ 3 \end{array} \begin{bmatrix} p_{11} & p_{12} & p_{13} \\ p_{21} & p_{22} & p_{23} \\ p_{31} & p_{32} & p_{33} \end{bmatrix} \end{array} \tag{2.1}$$

The Markov transition matrix is also termed as a column stochastic matrix because of the following three properties [14].

1. A Markov matrix is nonnegative because the transition probabilities can only have values of $0 \leq p_{ij} \leq 1$.
2. Since the columns of P represent the probability transitions out of the given state, the sum of each column must be 1 because each column covers all the possible transition events out of the state thus $\sum_{i=1}^{3} p_{ij} = 1$.
3. The magnitude of all the eigenvalues of the Markov matrix obeys the condition of $|\lambda_i| \leq 1$.

With the recorded RTP/UDP datagram statistics provided in [7] for b, c, f, and g and with the transition matrix P constructed from Figure 11 given by (2.2), the remaining parameters a, d, and e can be found.

$$P = \begin{bmatrix} a & 1-b & 1-c \\ d & b & 0 \\ e & 0 & c \end{bmatrix} \tag{2.2}$$

The formula for f is given by (2.3) [37]. With f and b values provided in [7] parameter d is found by manipulation and is given by (2.4).

$$f = \frac{d}{d+1-b} \tag{2.3}$$

$$d = \frac{(1-b)f}{(1-f)} \tag{2.4}$$

The formula for g is given by (2.5) [37]. With g and c values provided in [7] parameter e is found by manipulation and is given by (2.6).

$$g = \frac{e}{e+1-c} \tag{2.5}$$

$$e = \frac{(1-c)g}{(1-g)} \tag{2.6}$$

The sum of each column must equal one and is given by (2.7) [14]. With d and e values given by (2.4) and (2.6) parameter a is found by manipulation and is given by (2.8).

$$a + d + e = 1 \tag{2.7}$$

$$a = 1 - (d + e) \tag{2.8}$$

2.2.2 Calculation of Steady State Distribution Vector

Once the parameters are substituted into P, the steady state distribution vector $\boldsymbol{s}$ is calculated. The distribution vector probabilities have reached steady state when (2.9) is satisfied.

$$P\boldsymbol{s} = \boldsymbol{s} \tag{2.9}$$

The value of $\boldsymbol{s}$ is an eigenvector of P with the corresponding eigenvalue being equal to one ($\lambda = 1$). The steady state distribution vector is defined, as the probability of being in state n that will not change with time and is given by (2.10) [14].

$$\boldsymbol{s} = \begin{bmatrix} s_1 & s_2 & s_3 & \dots & s_n \end{bmatrix}^t \tag{2.10}$$

The eigenvector technique is used for finding $\boldsymbol{s}$ because P is expressed numerically and the 3 by 3 size is reasonable for a mathematical package like MATLAB® to find the eigenvector [14]. The other techniques for finding $\boldsymbol{s}$ are difference equations, Z-transform

(generating functions), direct numerical techniques to solve a system of linear equations, and iterative numerical techniques to solve a system of linear equations. Examples of all the above techniques for finding $\boldsymbol{s}$ are provided in [14]. The eigenvector technique ensures that P is diagonalizable by first finding distinct eigenvalues given by (2.11).

$$P = XDX^{-1} \tag{2.11}$$

Matrix X given by (2.12) columns is the eigenvectors of P.

$$X = \begin{bmatrix} x_{11} & x_{12} & x_{13} \\ x_{21} & x_{22} & x_{23} \\ x_{31} & x_{32} & x_{33} \end{bmatrix} \tag{2.12}$$

The diagonal matrix D given by (2.13) components is the eigenvalues arranged according to the ordering of the columns of X.

$$D = \begin{bmatrix} \lambda_1 & 0 & 0 \\ 0 & \lambda_2 & 0 \\ 0 & 0 & \lambda_3 \end{bmatrix} \tag{2.13}$$

The first column of X is $\boldsymbol{s}$ because it corresponds to the column of D for which $\lambda = 1$ and is given by (2.14).

$$\begin{aligned} \begin{bmatrix} x_{11} & x_{21} & x_{31} \end{bmatrix}^t &\leftrightarrow \lambda_1 = 1 \\ \begin{bmatrix} x_{12} & x_{22} & x_{32} \end{bmatrix}^t &\leftrightarrow \lambda_2 \\ \begin{bmatrix} x_{13} & x_{23} & x_{33} \end{bmatrix}^t &\leftrightarrow \lambda_3 \end{aligned} \tag{2.14}$$

The eigenvalues of $\boldsymbol{s}$ must be normalized to ensure that its summation is equal to one. Summing the first column of X and dividing the column by the summation accomplishes the normalization and the value of $\boldsymbol{s}$ and is given by (2.15) [14].

$$\begin{aligned} \boldsymbol{s} &= \begin{bmatrix} x_{11} & x_{21} & x_{31} \end{bmatrix}^t \Big/ \sum_{i=1}^{3} x_{i1} \\ \boldsymbol{s} &= \begin{bmatrix} s_1 & s_2 & s_3 \end{bmatrix}^t \end{aligned} \tag{2.15}$$

2.2.3 Datagram Loss Process

When a single datagram is correctly received from the network (represented by state 1) the output is '0'. When a single datagram is lost due to being discarded by network routers (represented by state two) the output is '1'. When a single datagram is lost due to network routers out-of-order delivery (represented by state three) the output is '1'. The output of a random number generator uniformly distributed between {0 1} is used to simulate the datagram loss process. The random number is compared with the value of *s* for state one (s_1). The output of the Markov Chain model is '0' if the random number is less than or equal to the value of s_1. The output of the Markov chain model is '1' if the random number is greater than the value of s_1. The actual overall datagram loss rate is the sum of s_2 (datagram loss) and s_3 (out-of-order) probabilities.

2.2.4 Concealment Failure

A concealment failure for packetization scheme 1 ($\hat{f}_1$) occurs when two consecutive datagrams are lost. This is considered a failure because the previous picture that is used as concealment for the current missing picture is corrupted. An example of a recorded concealment failure for packetization scheme 1 is shown in Figure 12.

A concealment failure for packetization scheme 2 ($\hat{f}_2$) is when either one or both datagrams containing the current picture's odd and even slices are lost *and* either one or both datagrams containing the previous picture's odd and even slices are lost. This is considered a failure because the previous picture that is used concealment for either or both the current pictures' missing slices is corrupted. An example of a recorded concealment failure for packetization scheme 2 is shown in Figure 13.

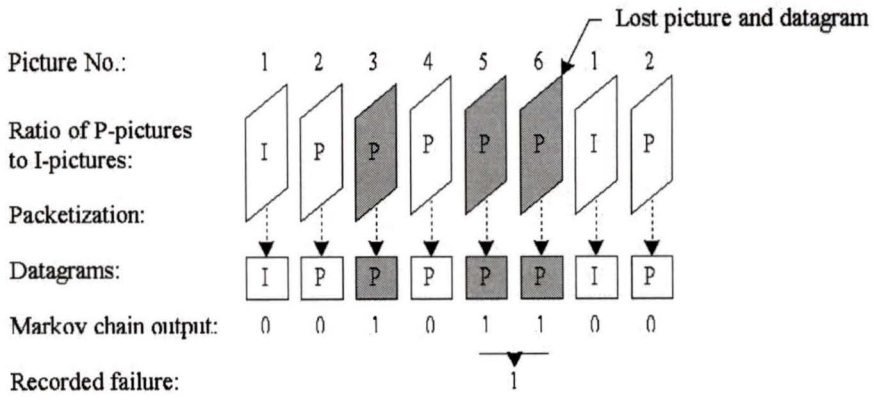

Figure 12: Packetization Scheme 1 Recorded Failure

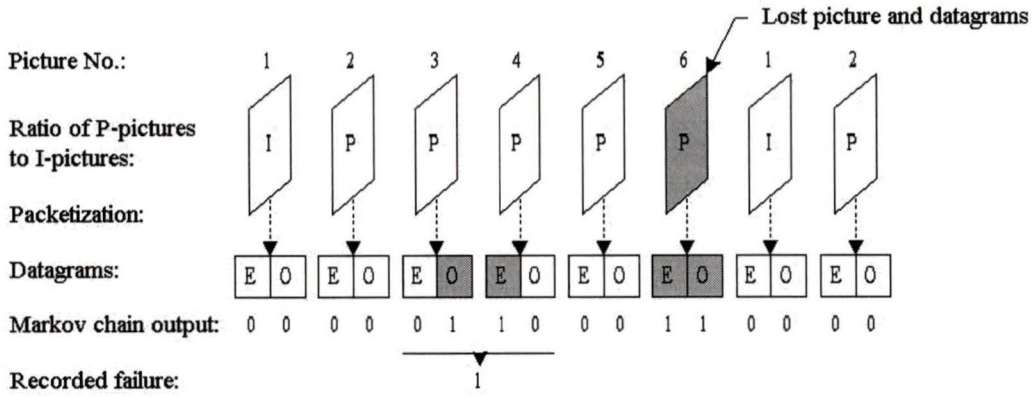

Figure 13: Packetization Scheme 2 Recorded Failure

2.2.5 Estimated Probability of a Concealment Failure

Estimated probability of a concealment failure for the two different packetization schemes is calculated by first finding the estimated number of concealment failures. The estimated number of concealment failures for the two different packetization schemes ($\hat{f}_{1,2}$) are found by summing the number of times a failure occurs in the number of datagrams produced from the P-pictures to I-pictures ratio (r). A number of trials (q) are used to improve the estimate for the number of failures and is given by (2.16).

$$\hat{f}_{1,2} = \frac{1}{q}\sum_{i=1}^{q}\sum_{j=1}^{r} failure(i,j) \tag{2.16}$$

The estimated probability of a concealment failure for the two different packetization schemes ($\hat{p}_{1,2}$) is found by summing the number of times a failure occurs in r and dividing this summation by r. The same q is used to improve the estimate of the probability of a failure and is given by (2.17).

$$\hat{p}_{1,2} = \frac{1}{q}\sum_{i=1}^{q}\frac{1}{r}\sum_{j=1}^{r} failure(i,j) \tag{2.17}$$

The standard deviation ($\sigma_{1,2}$) for $\hat{p}_{1,2}$ could be calculated because the failures follow a Bernoulli distribution [20] and is given by (2.18). The standard deviation provides a measure of the spread about the mean for $\hat{p}_{1,2}$ [20].

$$\sigma_{1,2} = \sqrt{\frac{1}{q}\hat{p}_{1,2}\left(1-\hat{p}_{1,2}\right)} \tag{2.18}$$

Calculated overall datagram loss rate ($p\_d_{1,2}$) for the two different packetization schemes is found by summing the number of times a datagram loss occurs in r and dividing this summation by r. The same q is used to improve the calculated overall loss rate and is given by (2.19).

$$p\_d_{1,2} = \frac{1}{q}\sum_{i=1}^{q}\frac{1}{r}\sum_{j=1}^{r} datagram\_discard(i,j) \tag{2.19}$$

2.2.6 Datagram Loss Model Summary

In the datagram loss model the value of r is equal to each of the different ratios of P-pictures to I-pictures for packetization scheme 1. The value of r is equal to twice the different ratios of P-pictures to I-pictures for packetization scheme 2. The value of q is set to a value of 5,000. Parameter d is calculated by (2.4) for each of the different f and b values provided in [7]. Parameter e is calculated by (2.6) for each of the different g and c

values provided in [7]. Parameter a is calculated by (2.8) for each of the different d and e values. The parameters (shown in Table 2) are then substituted into P given by (2.2). X and D are calculated by (2.11) using the MATLAB® 'eig' function. Vector $\boldsymbol{s}$ is then calculated by (2.15).

Creating a matrix by using the MATLAB® 'rand' function whose values are randomly uniformly distributed between {0 1} is used to simulate the datagram loss process of the Internet. The number of columns in the matrix is equal to q while the number of rows is equal to r. The random numbers are compared to s_1 for each r-value of the matrix. The output is '0' (datagram correctly received) if the random number is less than or equal to s_1. The output is '1' (datagram discarded) if the random number is greater than s_1.

Any two consecutive '1's in r is a recorded failure of packetization scheme 1 (Figure 12). Any '1's occurring in both consecutive pairs in r is a recorded failure of packetization scheme 2 (Figure 13). The $\hat{f}_{1,2}$ values are calculated by (2.16), the $\hat{p}_{1,2}$ values are calculated by (2.17), the $\sigma_{1,2}$ values are calculated by (2.18), and the $p\_d_{1,2}$ values are calculated by (2.19).

2.3 SIMULATION RESULTS

To confirm the operation of the Markov chain model and the value of q, the value of $p\_d_{1,2}$ is compared to the actual overall datagram loss rate (s_2+s_3). The value of $p\_d_{1,2}$ was found by taking the mean of $p\_d_{1,2}$ calculated at each ratio of P-pictures to I-pictures (five to fifteen). The mean was calculated for the network's five measured and calculated datagram statistics (different network paths and conditions) listed in Table 2.

Measured Datagram Statistics	1	2	3	4	5
$f\%$	1.500	5.600	9.000	12.40	21.10
$b\%$	33.00	23.00	20.00	20.00	25.00
$g\%$	1.500	5.600	9.000	12.40	21.10
$c\%$	0.7500	2.800	4.500	6.200	10.55
Calculated Parameters	1	2	3	4	5
$a\%$	97.47	89.67	82.64	75.40	56.02
$d\%$	1.020	4.570	7.910	11.32	20.06
$e\%$	1.510	5.770	9.450	13.28	23.92
$(s_2 + s_3)\%$	***2.960***	***10.61***	***16.51***	***22.06***	***34.85***
$p\_d_1\%$	***2.970***	***10.55***	***16.47***	***22.01***	***34.70***
$p\_d_2\%$	***2.970***	***10.63***	***16.56***	***22.05***	***34.95***

Table 2: Overall Datagram Loss Probability

Since the values of $p\_d_{1,2}$ are approximately equal to the values of $s_2 + s_3$, it is concluded that the Markov chain model is operating correctly and that the value of 5,000 for q is acceptable for the following simulations.

2.3.1 Estimated Number of Concealment Failures

The second set of simulations was done to find $\hat{f}_{1,2}$ at each ratio of P-pictures to I-pictures (five to fifteen) using the five sets of datagram statistics. The $\hat{f}_1$ value is approximately half of the $\hat{f}_2$ value and both values approximately linearly increase with the ratio increase shown by Figure 14 and Figure 15. Both estimates improve the tradeoff between using the I-picture for error resilience and picture quality with rate control if the network path is known (i.e., intercontinental or transcontinental).

As an example the overall datagram loss rate for a transcontinental or a busy network path (worst-case scenario) [7,25] is 34.85%. For the above network path a ratio of nine P-pictures to I-pictures may be used for packetization scheme 1 since the corresponding $\hat{f}_1$ is less than one.

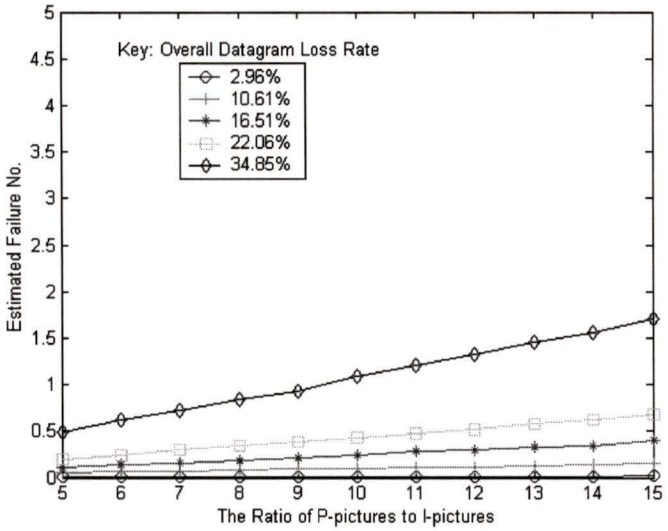

Figure 14: Number of Failures vs. Ratio for Packetization Scheme1

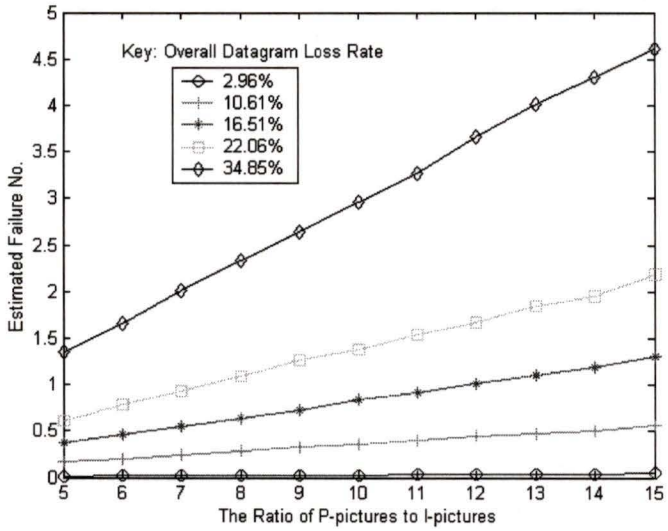

Figure 15: Number of Failures vs. Ratio for Packetization Scheme 2

2.3.2 Estimated Probability of a Concealment Failure

The third set of simulations was done to find $\hat{p}_{1,2}$ and $\sigma_{1,2}$. The values of $\hat{p}_{1,2}$ and $\sigma_{1,2}$ were found by taking the mean of $\hat{p}_{1,2}$ and $\sigma_{1,2}$ calculated at each ratio of P-pictures to I-pictures (five to fifteen) using the five sets of overall datagram loss rate statistics. The

ratio does not affect $\hat{p}_{1,2}$ because of the division by r. A larger ratio however, produces a better $\hat{p}_{1,2}$.

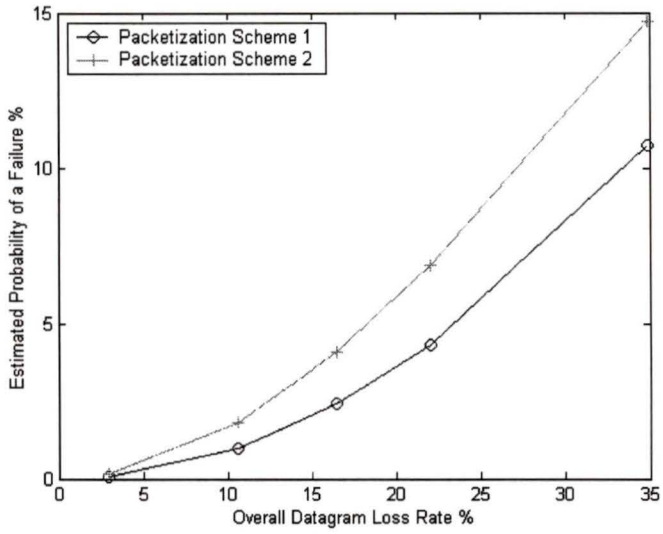

Figure 16: Probability of a Failure vs. Loss Rate for Packetization Schemes 1 and 2

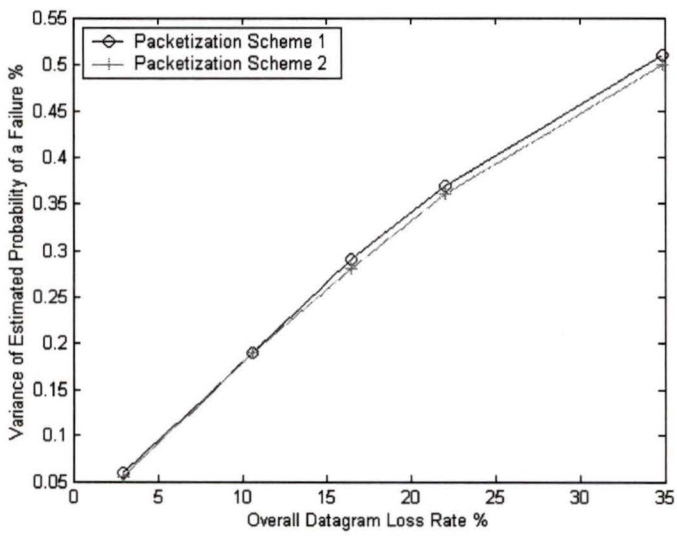

Figure 17: Variance of Probability of a Failure for Packetization Schemes 1 and 2

The $\hat{p}_1$ value is approximately equal to the $\hat{p}_2$ value shown by Figure 16. This occurs because parameter r for packetization scheme 2 is twice as large as parameter r for packetization scheme 1 and $\hat{p}_{1,2}$ is $\hat{f}_{1,2}$ divided by r.

The two different packetization schemes are robust against a concealment failure because $\hat{p}_1$ is 10.72% and $\hat{p}_2$ is 14.73% for highest overall datagram loss rate of 34.85% as shown by Figure 16. The $\hat{p}_{1,2}$ are good since $\sigma_{1,2}\%$ are less than 0.5500% as shown by Figure 17.

The accuracy of the Markov chain model is improved since it uses measured datagram statistics that affect the decoding of non-interactive real time streaming video. The Markov chain simulation results cannot be considered as 'typical' because jitter was not included in the model and as the Internet increases in size and complexity, measuring and modeling its datagram statistics vary from site-to-site and for different periods of time [7,25].

The estimated number of concealment failures for the two different packetization schemes would decrease if audio datagrams were included in the stream because the audio data can occupy 25% of the channel's bandwidth (one audio datagram for every four video datagram) [11]. This situation is not considered because of the particular case of streaming video over the Internet for security viewing purposes only (no audio is included). The estimated number of failures would also decrease if the transmitted datagrams are shuffled or interleaved [22]. Shuffling is not considered because the received datagrams are required to be buffered for proper re-ordering and for a real time streaming video datagram to be considered useful, it must be decoded immediately after it is received [7].

3. CONCEALMENT SCHEMES

3.1 EVALUATION METHODS

Section 3.1 provides a description of the evaluation methods, simulation environment, and video test sequences used to compare the various concealment schemes.

3.1.1 Peak Signal-To-Noise Ratio (PSNR)

A measure for signal quality is the signal-to-noise ratio (SNR) and may be expressed in decibels (dB) by (3.1) [3].

$$SNR = 10\log_{10}\frac{signal\ \ energy}{noise\ \ signal\ \ energy} \tag{3.1}$$

For the particular case of error concealment Peak Signal-to-Noise Ratio (PSNR) is used instead of SNR. The signal energy is defined as the pixel's maximum decimal value of 255, while the noise signal energy is defined as the difference between the decoded uncorrupted picture and the decoded concealed picture. The PSNR value is expressed in dB by (3.2) [3,29].

$$PSNR = 10\log_{10}\frac{255^2}{\frac{1}{hl}\sum_{y=1}^{h}\sum_{x=1}^{l}\left(picture_{(uncorrupted)}(x,y) - picture_{(concealed)}(x,y)\right)^2} \tag{3.2}$$

l is the picture's pixel length, h is the picture's pixel height, $picture_{(uncorrupted)}$ is the decoded uncorrupted picture, and $picture_{(concealed)}$ is the decoded concealed picture, and (x,y) denotes the coordinates from the upper left corner of the pictures.

A high PSNR value does not always correspond to a perceptually high quality concealed picture. Another method to measure the quality of error concealment is to use a five-point scale and the corresponding mean score for the subjective assessment of the

quality of the concealed picture [3]. This subjective assessment will not be implemented in this thesis but it is proposed for future work.

3.1.2 Central Processor Unit (CPU) Time

The CPU time is the time in seconds that has been used by the concealment scheme with no other programs running concurrently on the CPU (e.g., screen saver) [16]. The CPU time is used for a complexity comparison between the various schemes for the real time restriction and is used in conjunction with the PSNR value to evaluate the overall performance of the various concealment schemes.

3.1.3 Real Time Streaming Application

A real time streaming application refers to the decoding of a video sequence by a computer at the same speed, as it would occur in real life. A real time video decoder must be able to perform the following tasks:

- Decode the recovered picture data.
- Provide concealment for the missing picture data if necessary.
- Display the complete picture results at the same speed as the real life video sequence.

The H.261 standard specifies a maximum coding delay of 150×10^{-3} seconds since it has been determined that delays exceeding 150×10^{-3} seconds do not give the viewer the impression of direct visual feedback for bi-directional video communication [18,29]. Processing time may be reduced via efficient algorithms implemented in hardware [3] and the Pentium Processor MMX programming language [17].

3.1.4 Simulation Environment

The simulations in this chapter were done using MATLAB® software. MATLAB® was used because of its versatility and numerous built in functions. MATLAB® was also used because of the number of required concealment simulations and because it permits the calculation of the overall mean concealment PSNR value and CPU time through the different portions of the test video sequences. The MATLAB® software simulations will help select the concealment scheme to be implemented in the actual video decoder 'C' software. The 'C' software simulations are much faster when compared to the MATLAB® simulations so therefore the CPU time for the selected concealment scheme will be lower in the actual 'C' video decoder.

The PSNR result and CPU time for the various simulated motion vectors recovery schemes were performed on a Toshiba 'Satellite 2500 CDS' that has 64×10^{6} bytes of RAM and an 233 MHz MMX Intel Pentium processor.

3.1.5 Video Test Sequences

In the various video test sequences the QCIF video picture format and a source picture rate of 10 pictures per second were applied to the coder with a fixed bit rate output. The source picture rate with the rate control does not affect the concealment simulations since the simulations compare the decoded video pictures to the concealed decoded video pictures.

An I-picture was used at the beginning of the sequence, while P-pictures were used for the remaining pictures. The P-picture's unrestricted integer motion vectors were written to a file. Concealment simulations were done only using the *Y* component of the picture since it is equal to the picture format and reduce the number of concealment

simulations from three to one. Concealment on the *Cr* and *Cb* components of the picture is done once the desired scheme is selected.

The 'Suzie' video scene that is used for videoconferencing simulations [3] has an individual talking on a telephone as shown in Figure 18. All of the movement is in the head and the shoulder area and there is no background scenery. The 'Suzie' video sequence is used because the slow and rapid motion portions, and the detail will extensively test the various concealment schemes [26]. Pictures 1 to 105 of the 'Suzie' video sequence are used.

Figure 18: First Picture of 'Suzie' Video Test Sequence

Concealment results from 'Suzie' 1 were used in the development of the proposed I-picture and P-picture schemes. Every third picture in the 'Suzie' 1 video sequence was coded. The total number of pictures coded was 35. Concealment simulations were performed on every even picture in the last 34 pictures (i.e., 17 pictures) while every odd picture was uncorrupted. Temporal sub-sampling was done at the source input (i.e., every third picture coded) for low bit rate applications [3]. Temporal sub-sampling is the worst-case scenario for concealment since it increases the magnitude of scene change between the current picture and the previous picture, which is used for concealment. Three additional simulations are done to further test the various concealment schemes.

'Suzie' 2 has the same coding criteria as 'Suzie' 1 with the exception that the temporal sampling rate was set to one. The number of coded pictures was 105. The concealment simulations were performed on every even picture in the last 104 pictures (i.e., 52 pictures) while every odd picture is uncorrupted.

The third and fourth set of simulations used the 'Car Phone' video scene that has an individual sitting in the back seat of a car whose mouth, head, hands, and torso produces abrupt and large amount of motion as shown in Figure 19. The 'Car Phone' video scene also has a rapid background scene change in the side window of the car. The individual's abrupt and large amount of motion, and the constant background scene change will provide a different simulation environment when compared to the 'Suzie' video sequence. It will also extensively test the various concealment schemes [26]. The 'Car Phone' has the same coding criteria as 'Suzie' 1 and 2 with the exception that pictures 1 to 207 are used.

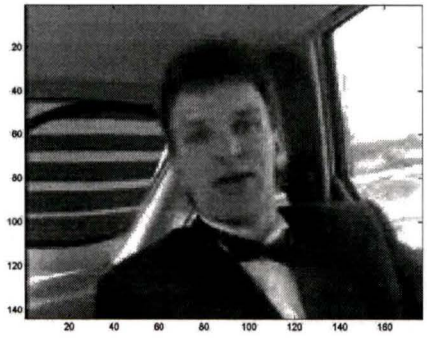

Figure 19: First Picture of the 'Car Phone' Video Test Sequence

'Car Phone' 1 has a temporal sampling rate of three so the number of coded and decoded pictures is 69. The concealment simulations were performed on every even picture in the last 68 pictures (i.e., 34 pictures) while every odd picture is uncorrupted.

'Car Phone' 2 has a temporal sampling rate of one so the number of coded and decoded pictures is 207. The concealment simulations were performed on every even picture in the last 206 pictures (i.e., 103 pictures) while every odd picture was uncorrupted.

3.2 CONCEALMENT SCHEMES BACKGROUND

An incorrect estimation of the missing motion vector leads to an incorrect displacement of the uncorrupted previous picture macroblock. This leads to a poor concealment result that propagates through the remaining P-pictures until the next I-picture [1]. A better estimation of the missing macroblock motion vector improves the concealment result, however it requires a more complex estimation scheme. The complex scheme increases the CPU time, which is not desirable for the real time restriction. The objective of the following simulations is to compare the various schemes by using overall mean concealment results and corresponding CPU times. The sections are organized as follows:

- Section 3.3 consists of the concealment schemes and the corresponding results for the I-picture's odd and even slices.
- Section 3.4 consists of the concealment schemes and the corresponding results for the P-picture's odd and even slices.
- Section 3.5 consists of the complete simulation results for the I-picture's and the P-picture's odd and even slices.
- Section 3.6 consists of the concealment schemes and the corresponding results for the entire missing picture.

3.3 I-PICTURE CONCEALMENT SCHEMES

Concealment of the I-picture is very important because the propagation of the concealment error is more visually noticeable through the remaining P-pictures when compared to the concealment error of the P-picture. The following three existing schemes are used to estimate the picture's missing odd or even slices due a single packet loss.

1. Copy the previous picture slice located in the same spatial location as the missing slice. This scheme has the lowest complexity and works well in macroblocks that contain a small amount of motion [1,32,33] as shown in Table 3.
2. Copy the previous picture macroblock located in the same spatial location as the missing macroblock that is motion compensated with its corresponding motion vector [1,32,33]. This scheme has the second lowest complexity and produces better overall mean concealment results compared to scheme 1 shown in Table 3.
3. Perform a full search on the previous picture for the best match of the missing macroblock upper and lower one pixel wide boundary (if available). Concealment is achieved by copying the previous picture macroblock that best matches the pixel boundary of the current picture lost macroblock. This is a process that is similar to the motion estimation process performed at the coder and is called Decoder Motion Vector Estimation (DMVE) [36]. The Sum of the Absolute Differences (SAD) is used for the calculation of the minimum distortion point. SAD is preferred for real time applications when compared to the Sum of the Squared Differences (SSD) since SAD does not use multiplication, which increases the CPU time [21]. A one-pixel wide boundary is used instead of two and eight because simulations in [36] found that the mean concealment value did

not increase significantly (0.2600 dB) with a large increase of CPU time (multiplication factor of 3.500). To match the coder, an unrestricted search area is used. For the DMVE simulations the unrestricted search area is equal to ±15 pixels horizontally and vertically. Single pixel accuracy is used instead of half pixel accuracy to further reduce the CPU time. The concealment results will be slightly reduced by using the above restrictions [3,18,36]. This scheme has the highest amount of complexity and produces the best overall mean PSNR concealment results compared to schemes 1 and 2 as shown in Table 3.

3.3.1 Proposed Concealment Scheme

The proposed scheme uses scheme 1 for macroblocks that contain a small amount of motion and scheme 3 for macroblocks that contain a large amount of motion but with the following modification. The previous picture's motion vector is used as a predictor or a starting point for the One-at-a-Time Search (OTS) algorithm. The OTS is a fast search motion estimation algorithm. The full search motion estimation algorithm is computationally expensive because all the points in the specified area are searched. Fast motion estimation algorithms are desirable for real time video coding because the number of points searched is reduced thus reducing the CPU time. Reduction in the number of points searched means that fast motion algorithms may become trapped in the local minima of the search area and will produce a lower prediction result [3,21].

To avoid being trapped in the local minimum, a proposed algorithm using the median value of the spatial neighboring macroblock motion vectors is used as the starting point for the OTS algorithm [21]. Simulation results in [21] show that the current missing macroblock motion vector is more correlated with the spatial neighboring when

compared to the temporal neighboring macroblock motion vectors. Simulation results also show that 70% of the optimal motion vector falls into a 3 by 3 window using the H.263+ predicted motion vector and therefore a fast motion estimation algorithm that requires a low number of search points in a localized area was investigated in [21].

OTS was selected when compared to the Three Step Search (TSS), Direct Search (DS), and Logarithmic Search (LS) fast motion algorithms because it was found that OTS had the lowest number of search points in an area search of less than seven pixels with the same prediction PSNR results [21].

Spatial neighboring motion vectors are not available for the I-picture so the temporal neighboring motion vectors must be used as the starting point for the OTS algorithm. To further reduce the CPU time without significantly reducing the concealment result, the previous picture's macroblock motion vector located at the same spatial location as the missing macroblock is used as the starting point rather than the median of the temporal neighboring motion vectors. The previous picture motion vector was used since it produced good overall mean concealment results as shown in Table 3.

The OTS algorithm is where each axis search is done separately to find the minimum distortion value (SAD). Figure 20 describes an OTS. A first search is done to find the minimum point in the x direction, indicated by '1'. Starting from the minimum x point, a second search is done to find the minimum point in the y direction, indicated by '2'. Starting from the x and y minimum points, a third search is done to find the minimum point in the x direction and it is determined that the new x and y points are the minimum points in the search area, indicated by '3'. The motion vector in this example has the final location of x equal to +3 and y equal to –3. Simulation results in [21] show that the

optimum number of adjacent pixels to be searched is two. This is done to ensure that the OTS does not become trapped in the local minima with a slight increase in the CPU time.

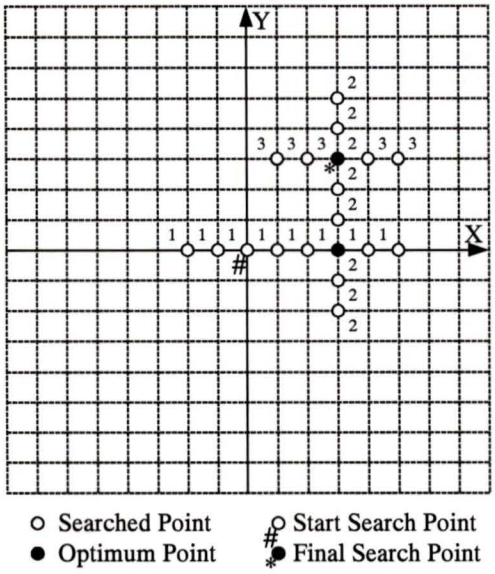

Figure 20: Example of an OTS

Comparing a computed threshold value to a set threshold value determines the decision between the two following schemes used for concealment:

- Copying the previous picture macroblock located at the same spatial location as the missing macroblock.
- Using an OTS that has the previous picture macroblock motion vector located at the same spatial location as the missing macroblock as a starting point.

If the computed value is less than or equal to the set value, the previous picture macroblock is used for concealment because of the small amount of motion in the missing macroblock. If the computed value is greater than the set value, the modified OTS is used for concealment because of the large amount of motion in the missing macroblock.

The computed threshold value is determined by first finding the SAD separately for the upper and lower one-pixel wide boundary given by (3.3) and (3.4). The computed threshold value is then determined by finding the maximum value of (3.3) and (3.4) using (3.5) [13].

$$u(x,y) = \sum_{i=x}^{x+m-1} \left| b_{u(n)}(i, y-1) - b_{u(n-1)}(i, y-1) \right| \tag{3.3}$$

$$l(x,y) = \sum_{i=x}^{x+m-1} \left| b_{l(n)}(i, y+m) - b_{l(n-1)}(i, y+m) \right| \tag{3.4}$$

$$t\_h = \max\left[u(x,y) \quad l(x,y)\right] \tag{3.5}$$

The calculation of the computed threshold value is shown in Figure 21. m is the size of the macroblock, picture (n) is the current picture, picture (n-1) is the previous picture, b_u is the upper pixel boundary, b_l is the lower pixel boundary, and (x, y) denotes the coordinates of the upper left corner of the macroblocks. The positive direction for the x-axis is right while the positive direction for the y-axis is down from the upper left corner of the macroblocks [3]. The computed threshold value is determined by (3.3), (3.4), and (3.5).

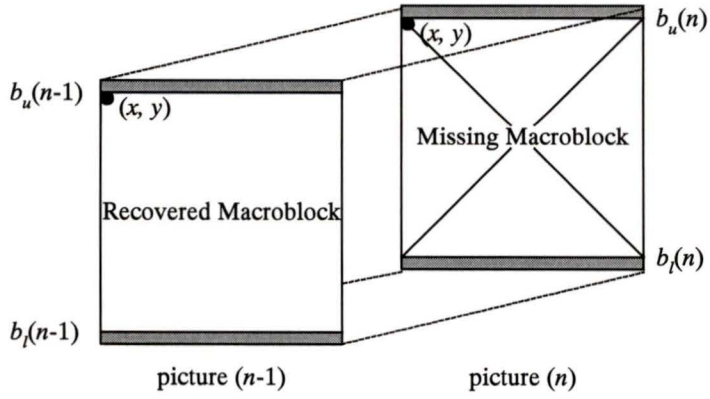

Figure 21: Computed Threshold Value

The set threshold value depends upon the specific picture format and can be found through experimentation (trial and error) [13]. Instead of determining the set threshold value through experimentation, the set threshold value was calculated by first finding the amount of motion in the 'Suzie' 1 test sequence. The amount of motion in the video sequence was computed by the two following methods:

- Normalize the SAD of the pixels (p_s) between two consecutive pictures given by (3.6) [29].
- Average the SAD of the macroblock motion vectors (v_s) between two consecutive pictures given by (3.7) [30].

$$p\_s = \frac{1}{hl}\sum_{y=1}^{h}\sum_{x=1}^{l}\left|picture_{(n)}(x,y) - picture_{(n-1)}(x,y)\right| \tag{3.6}$$

$$v\_s = \frac{1}{k}\sum_{i=1}^{k}\left|m\_v_{(n)}(i) - m\_v_{(n-1)}(i)\right| \tag{3.7}$$

l is the picture's pixel length, h is the picture's pixel height, k is the picture's total number of v_s, $picture_{(n)}$ is the current picture, $picture_{(n-1)}$ is the previous picture, $m\_v_{(n)}$ is the current picture's motion vector, and $m\_v_{(n-1)}$ is the previous picture's motion vector. The smallest amount of motion in the video sequence was used for the calculation of the set threshold value and occurred between the first two pictures as shown by Figure 22. These two pictures therefore are used for the calculation of the set threshold value. The set threshold value was determined by first computing the PSNR value between the first two consecutive pictures in the video sequence given by (3.8) [29].

$$PSNR = 10\log_{10}\frac{255^2}{\frac{1}{hl}\sum_{y=1}^{h}\sum_{x=1}^{l}\left(picture_{(n)}(x,y) - picture_{(n-1)}(x,y)\right)^2} \tag{3.8}$$

The calculated PSNR value of 36.16 dB was subjectively acceptable so the fixed threshold value was calculated by finding the mean of the second to the eighth slice between the first two pictures using (3.5). The calculated mean value was 49.17 so a set threshold value of 50 was used in the first set of 'Suzie' 1 simulations.

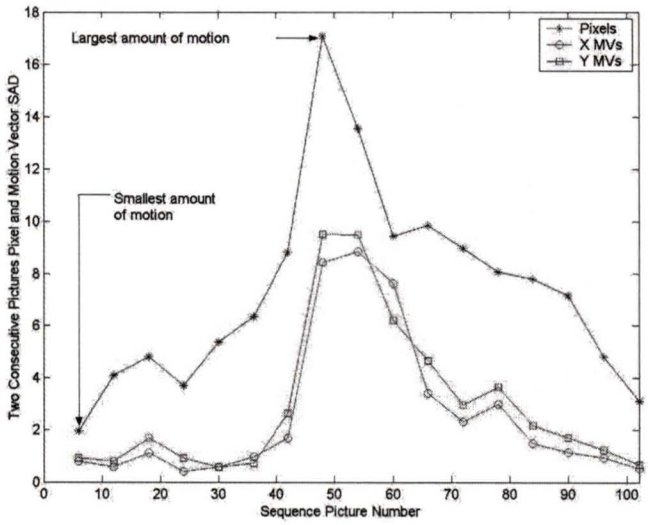

Figure 22: Two Consecutive Pictures Pixel and Motion Vector SAD vs. Picture No. Using 'Suzie' 1

Figure 23 is the flow chart of the proposed concealment scheme algorithm.

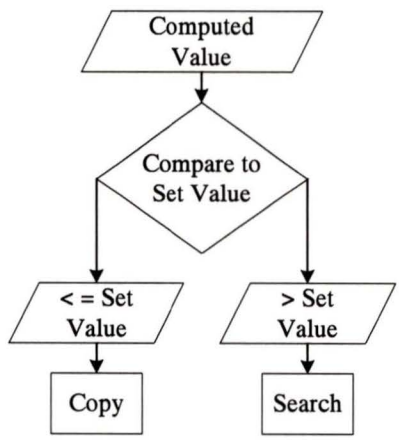

Figure 23: Flow Chart of Proposed Concealment Scheme

3.3.2 Simulation Results

The simulations using the 'Suzie' 1 video sequence for schemes 1, 2, and 3, and proposed scheme were done separately for the picture's missing odd and even slices. The results for the concealment schemes are shown in Table 3, Table 4, and Figure 24 to Figure 29. Scheme 1 concealment results are used as a reference for the comparison between schemes 2 and 3 and the proposed concealment results as shown in Figure 25 and Figure 28.

Concealment for Odd Slices	*Mean Y-PSNR (dB)*	*Mean CPU Time (seconds)*
Scheme 1	28.66	0.0141
Scheme 2	29.65	0.0301
Scheme 3	31.07	29.20
Proposed Scheme	31.02	0.7110
Concealment for Even Slices	*Mean Y-PSNR (dB)*	*Mean CPU Time (seconds)*
Scheme 1	30.75	0.01120
Scheme 2	31.90	0.02120
Scheme 3	34.28	28.46
Proposed Scheme	33.71	0.7380

Table 3: Comparison of I-Picture Schemes 1, 2, 3, and Proposed

Concealment for Odd Slices	*Mean Y-PSNR (dB) Difference*	*CPU Time Multiplication Factor*
Scheme 1	+2.364	50.43 increase
Scheme 2	+1.370	23.62 increase
Scheme 3	-0.05410	41.08 decrease
Concealment for Even Slices	*Mean Y-PSNR (dB) Difference*	*CPU Time Multiplication Factor*
Scheme 1	+2.955	65.00 increase
Scheme 2	+1.806	34.34 increase
Scheme 3	-0.5684	39.09 decrease

Table 4: Comparison between I-Picture Schemes 1, 2, 3, and Proposed

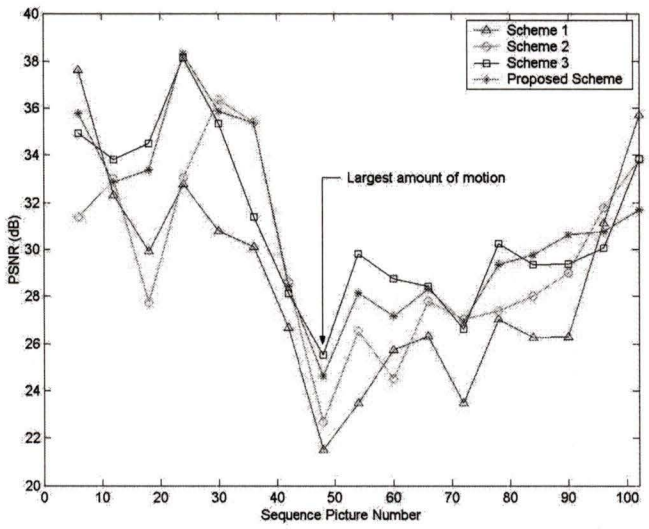

Figure 24: I-Picture Odd Slice Concealment PSNR vs. Picture No.

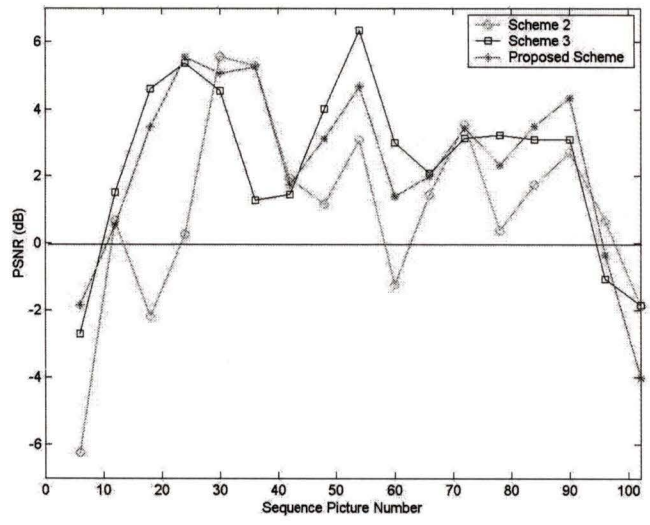

Figure 25: I-Picture Odd Slice Concealment PSNR Comparison between Scheme 1 and Schemes 2, 3, and Proposed vs. Picture No.

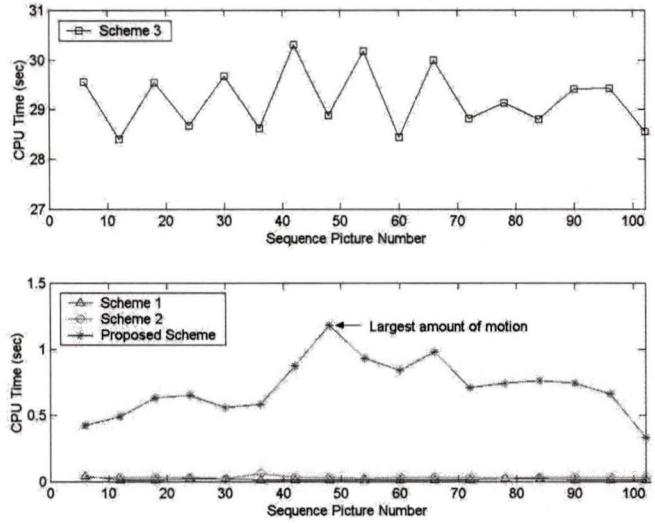

Figure 26: I-Picture Odd Slice Concealment CPU Time vs. Picture No.

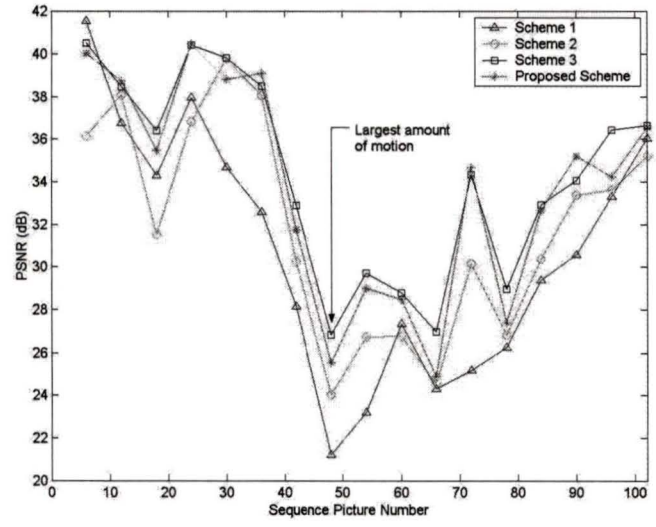

Figure 27: I-Picture Even Slice Concealment PSNR vs. Picture No.

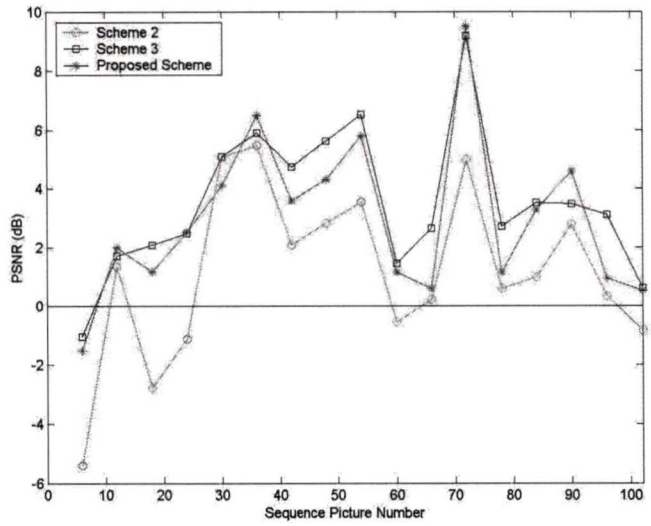

Figure 28: I-Picture Even Slice Concealment PSNR Comparison between Scheme 1 and Schemes 2, 3, and Proposed vs. Picture No.

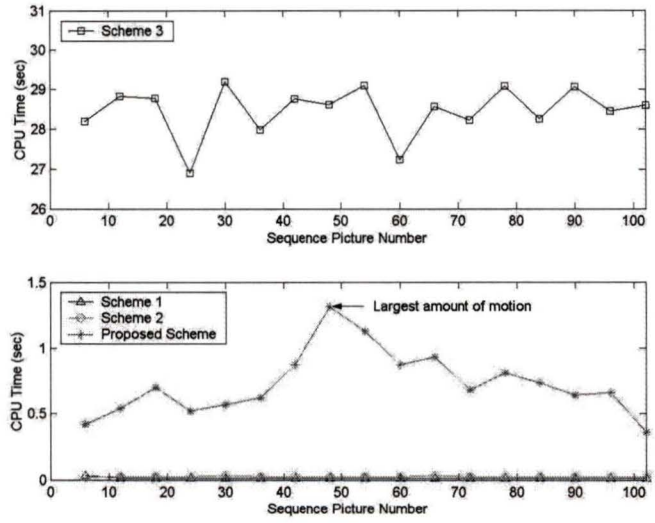

Figure 29: I-Picture Even Slice Concealment CPU Time vs. Picture No.

Concealment results are lower for the missing odd slices when compared to the missing even slices because a QCIF video picture consists of five odd and four even slices. Concealment results are greatly reduced for the schemes that use motion vectors and copying the previous picture slices for the large amount of motion that occurs between

pictures 49 and 52 in the video sequence as shown by Figure 24 and Figure 27. Concealment using motion vectors is less effective for macroblocks that contain a lot of motion because of the increase magnitude of the motion vector used for concealment and the slice/macroblock area (block artifacts effects) to conceal due to a single packet loss (all odd or even slices). Concealment by copying the previous picture macroblocks is less effective for macroblocks that contain a lot of motion because of the large scene change between two consecutive pictures and the slice/macroblock size area (block artifacts effects) to conceal due to a single packet loss (all odd or even slices). The large amount of motion also increases the CPU time for the proposed scheme (increased search concealment) shown by Figure 26 and Figure 29.

The DMVE scheme's CPU time is not acceptable for the real time application because all the points in the specified area are searched. The proposed scheme produced similar concealment results to the DMVE scheme with a large reduction in the CPU time because of the use of the copying and the modified OTS search technique. Concealment results can be reduced for the proposed and DMVE schemes since the block motion estimation process may not detect objects that rotate about their center, zooming, and shading [3].

3.4 P-PICTURE CONCEALMENT SCHEMES

The following four existing schemes are used to estimate the picture's missing odd or even slices due a single packet loss.

1. The macroblock's motion vector located above the missing macroblock when available. If not available use zero [10].

2. The median value of directly above, above right, and below macroblock motion vectors when available [1,32,33]. If not available use the available adjacent to improve concealment results (shown in Figure 30) [18]. The use of the adjacent motion vector is a similar format used for the calculation of the predictor motion vector.
3. The integer average between the macroblock motion vectors directly above and below the current missing macroblock when available. If not available use the available adjacent to improve concealment results (shown in Figure 30) [18]. The integer average between motion vectors directly above and below was used because it has been found that the median value between three macroblock motion vectors produced better concealment results when compared to the average value [32].
4. The macroblock motion vector that produces the smallest boundary error is used for concealment (hybrid) between: the above schemes 2 and 3, the macroblock motion vector located at the same spatial location in the previous picture, and setting the macroblock motion vector to zero (copying the previous picture macroblock). The boundary error is the SAD between the one-pixel wide boundary between the concealed macroblock and the recovered one directly above and below (if available). This method usually produces better visual concealment results because the pixels in the video picture are highly correlated and the continuity across the block boundaries is used to access the quality of concealment [1,32,33].

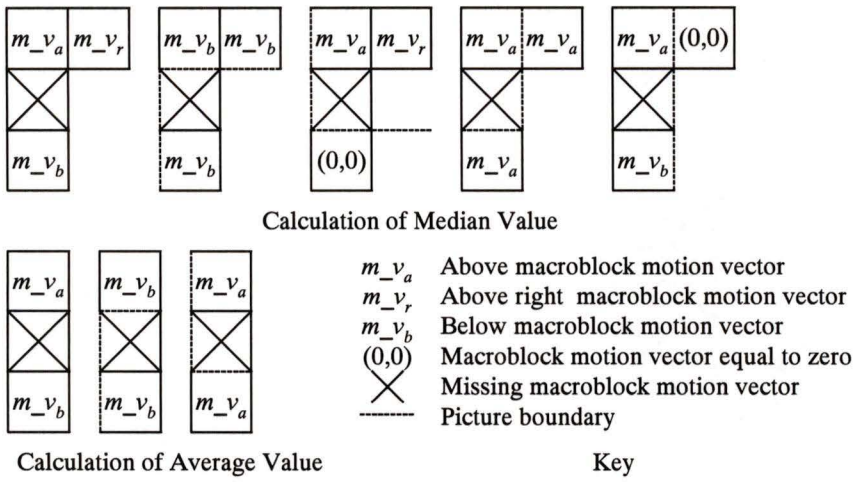

Figure 30: Calculation of Missing Macroblock Motion Vector from Spatial Values

3.4.1 Proposed Concealment Scheme

The proposed P-picture concealment scheme uses the same algorithm as the proposed I-picture concealment scheme in section 3.3 with the exception that scheme 3 motion vector result is used as the starting point for the OTS when the calculated threshold value is greater than the set value. Scheme 3 was used as the starting point because the concealment simulations produced the highest overall mean concealment results with the second lowest overall mean CPU time when compared to schemes 1, 2, and 4 for the picture's missing odd and even slices as shown in Table 5.

3.4.2 Simulation Results

The simulation results for schemes 1, 2, 3, and 4 and proposed scheme were done separately for the picture's missing odd and even slices. The simulation results are shown in Table 5, Table 6, and Figure 31 to Figure 36. Scheme 1 concealment results are used as a reference for the comparison between schemes 2, 3 and 4 and the proposed concealment results and is shown in Figure 32 and Figure 35.

Concealment for Odd Slices	*Mean Y-PSNR (dB)*	*Mean CPU Time (seconds)*
Scheme 1	30.92	0.03120
Scheme 2	31.33	0.2091
Scheme 3	31.34	0.07060
Scheme 4	31.18	0.5094
Proposed Scheme	31.52	0.7340
Concealment for Even Slices	*Mean Y-PSNR (dB)*	*Mean CPU Time (seconds)*
Scheme 1	33.56	0.02420
Scheme 2	34.03	0.2085
Scheme 3	34.17	0.07480
Scheme 4	34.00	0.4836
Proposed Scheme	34.32	0.6926

Table 5: Comparison of P-Picture Schemes 1, 2, 3, 4, and Proposed

Concealment for Odd Slices	*Mean Y-PSNR (dB) Difference*	*CPU Time Multiplication Factor*
Scheme 1	+0.6048	23.53 increase
Scheme 2	+0.1944	3.510 increase
Scheme 3	+0.1815	10.40 increase
Scheme 4	+0.3395	1.441 increase
Concealment for Even Slices	*Mean Y-PSNR (dB) Difference*	*CPU Time Multiplication Factor*
Scheme 1	+0.7629	28.62 increase
Scheme 2	+0.2951	3.322 increase
Scheme 3	+0.1524	9.259 increase
Scheme 4	+0.3207	1.432 increase

Table 6: Comparison between P-Picture Schemes 1, 2, 3, 4, and Proposed

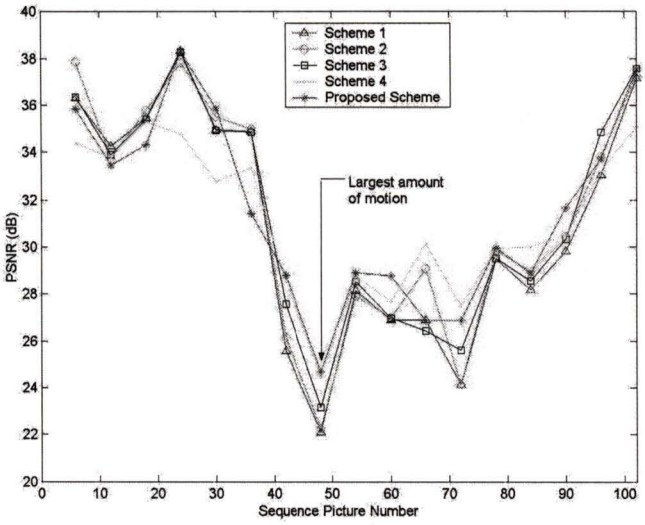

Figure 31: P-Picture Odd Slice Concealment PSNR vs. Picture No.

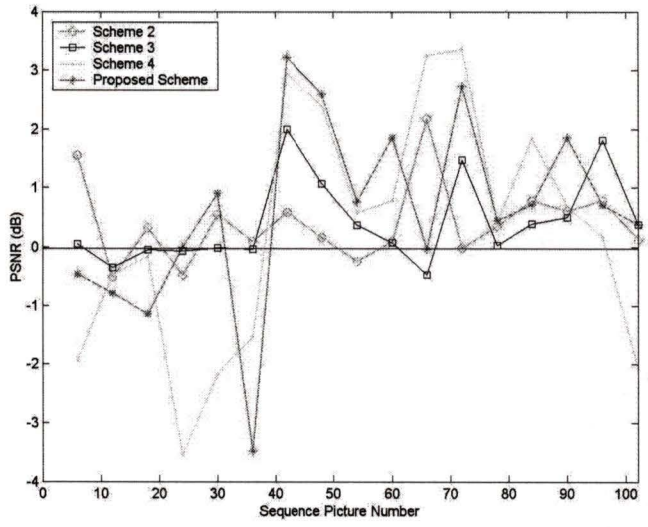

Figure 32: P-Picture Odd Slice Concealment PSNR Comparison between Scheme 1 and Schemes 2, 3, 4, and Proposed vs. Picture No.

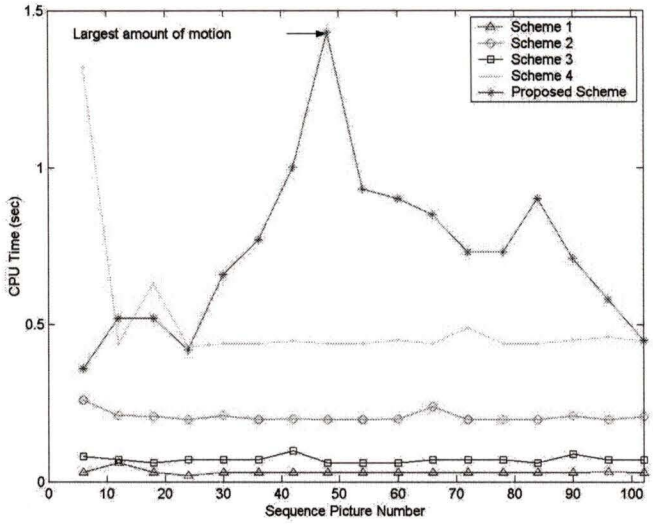

Figure 33: P-Picture Odd Slice Concealment CPU Time vs. Picture No.

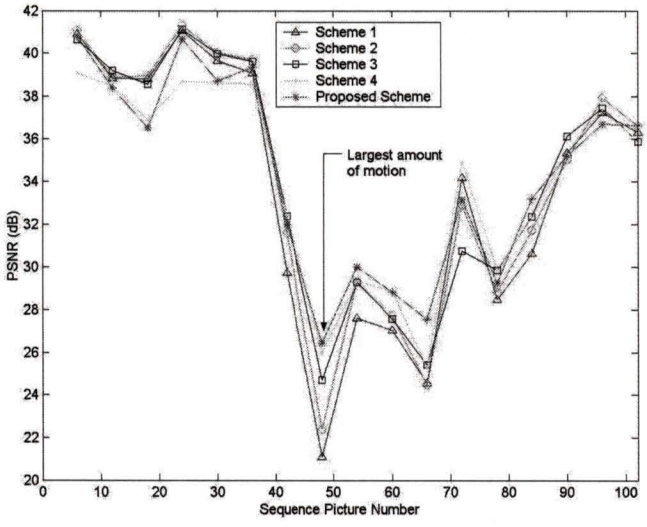

Figure 34: P-Picture Even Slice Concealment PSNR vs. Picture No.

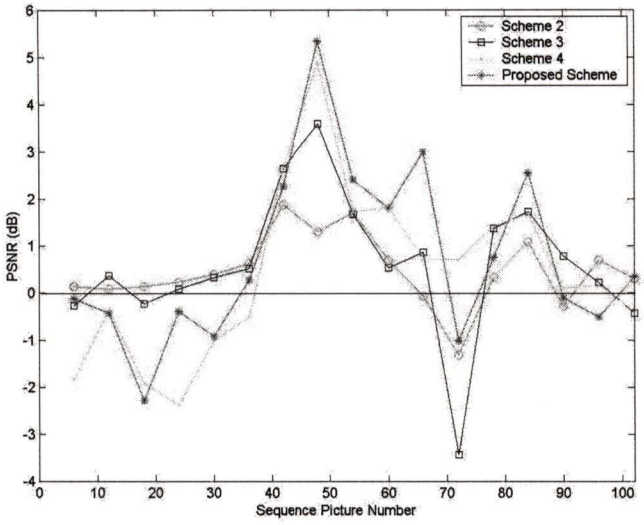

Figure 35: P-Picture Even Slice Concealment PSNR Comparison between Scheme 1 and Schemes 2, 3, 4, and Proposed vs. Picture No.

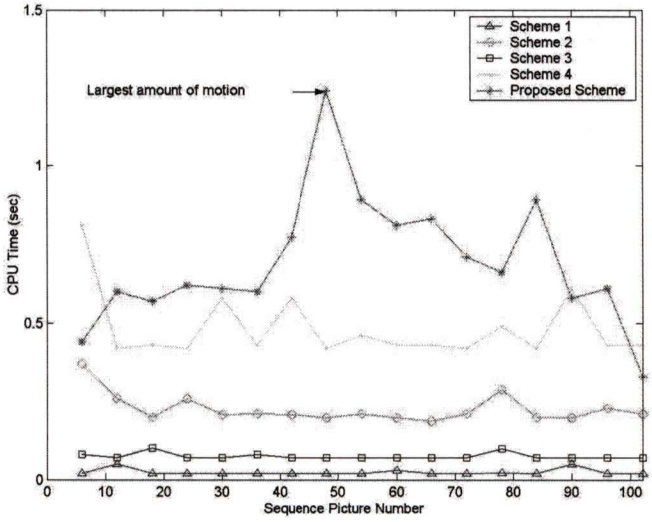

Figure 36: P-Picture Even Slice Concealment CPU Time vs. Picture No.

The spatially neighboring macroblock motion vectors overall mean concealment results are similar but slightly lower than the proposed P-picture scheme overall mean concealment results. Performed simulations and the simulation results provided in [21]

shows that the current missing macroblock's motion vector is highly correlated with the spatially neighboring located macroblock motion vectors.

3.5 COMPLETE SIMULATION RESULTS

Three additional simulations were done to further test the various concealment schemes for the video picture's missing odd slices. 'Suzie' 2 has the set threshold value equal to 150. The set threshold value was increased from 50 to 150 because the temporal sampling rate was reduced from three to one, which results in more temporal correlation between consecutive pictures [3]. 'Car Phone' 1 had a temporal sampling rate of three and the set threshold value equal to 50. 'Car Phone' 2 had a temporal sampling rate of one and the set threshold value equal to 150.

The amount of motion in the 'Car Phone' 1 video test sequence, which was computed by normalizing the SAD of the pixels [29] and averaging the SAD of the macroblock motion vectors [30] between two consecutive pictures given by (3.6) and (3.7) is shown in Figure 37. Figure 37 indicates that 'Car Phone' 1 has a large amount of abrupt motion between consecutive pictures. Figure 22 indicates that 'Suzie' 1 video sequence also has a large amount of motion, however it approximately increases and decreases linearly (i.e., not abrupt). The abrupt scene change in 'Car Phone' 1 significantly reduces all the scheme's overall mean concealment results, especially the schemes that use an estimated motion vector as shown in Table 7 and Table 8. The I-picture and P-picture's complete overall mean concealment results and CPU times are shown in Table 7 and Table 8.

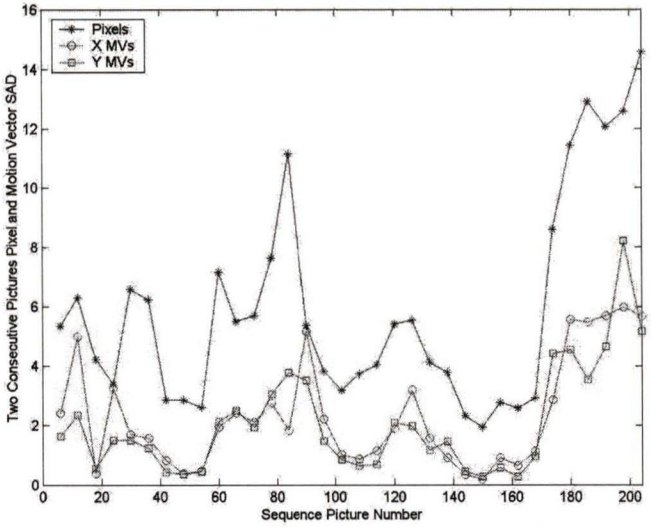

Figure 37: Two Consecutive Pictures Pixel and Motion Vector SAD vs. Picture No. Using 'Car Phone' 1

Concealment for 'Suzie' 1	*Mean Y-PSNR (dB)*	*Mean CPU Time (seconds)*
Scheme 1	28.66	0.01410
Scheme 2	29.65	0.03010
Scheme 3	31.07	29.20
Proposed Scheme	31.02	0.7110
Concealment for 'Suzie' 2	*Mean Y-PSNR (dB)*	*Mean CPU Time (seconds)*
Scheme 1	34.48	0.01790
Scheme 2	36.58	0.03540
Scheme 3	34.48	29.10
Proposed Scheme	35.77	0.1871
Concealment for 'Car Phone' 1	*Mean Y-PSNR (dB)*	*Mean CPU Time (seconds)*
Scheme 1	30.69	0.01500
Scheme 2	28.32	0.04820
Scheme 3	26.69	30.20
Proposed Scheme	30.34	0.7579
Concealment for 'Car Phone' 2	*Mean Y-PSNR (dB)*	*Mean CPU Time (seconds)*
Scheme 1	34.36	0.01060
Scheme 2	32.67	0.03990
Scheme 3	30.11	30.05
Proposed Scheme	34.12	0.2596

Table 7: Complete Comparisons of I-Picture Schemes 1, 2, 3, and Proposed

The I-picture proposed scheme produced either the first or the second highest overall mean concealment result with the third highest CPU time when compared to schemes 1, 2, and 3. The overall mean concealment results for the schemes 2 and 3, and proposed scheme that use an estimated motion vector for concealment decreased significantly using both 'Car Phone' video sequences because of the abrupt and large amount of motion between consecutive pictures. Schemes 2 and 3, and proposed scheme produced good overall all mean concealment results for both 'Suzie' video sequences. Scheme 1 that uses the previous picture slice produced the highest overall mean concealment results using both 'Car Phone' video sequences, the lowest and the second lowest overall mean concealment results using 'Suzie' 1, 2 video sequences.

Concealment for 'Suzie' 1	*Mean Y-PSNR (dB)*	*Mean CPU Time (seconds)*
Scheme 1	30.92	0.03120
Scheme 2	31.33	0.2091
Scheme 3	31.34	0.07060
Scheme 4	31.18	0.5094
Proposed Scheme	31.52	0.7340
Concealment for 'Suzie' 2	*Mean Y-PSNR (dB)*	*Mean CPU Time (seconds)*
Scheme 1	35.23	0.04650
Scheme 2	35.69	0.2300
Scheme 3	35.56	0.08150
Scheme 4	36.00	0.4831
Proposed Scheme	35.78	0.2035
Concealment for 'Car Phone' 1	*Mean Y-PSNR (dB)*	*Mean CPU Time (seconds)*
Scheme 1	28.50	0.03380
Scheme 2	29.98	0.2356
Scheme 3	28.60	0.08320
Scheme 4	30.53	0.5285
Proposed Scheme	30.08	0.7665
Concealment for 'Car Phone' 2	*Mean Y-PSNR (dB)*	*Mean CPU Time (seconds)*
Scheme 1	31.66	0.04370
Scheme 2	32.68	0.2329
Scheme 3	31.65	0.07710
Scheme 4	33.09	0.5102
Proposed Scheme	33.83	0.2788

Table 8: Complete Comparison of P-Picture Schemes 1, 2, 3, 4, and Proposed

The P-picture proposed scheme produced either the first or the second highest overall mean concealment results with the highest and second highest overall mean CPU time (depending on the set threshold value) when compared to schemes 1, 2, 3 and 4. The overall mean concealment results for the schemes 1, 2, 3, 4, and proposed decreased significantly for both 'Car Phone' video sequences and were less than the copying technique that is used for concealment of the I-picture. Schemes 1, 2, 3, 4, and the proposed produced good overall mean concealment results for both 'Suzie' video sequences.

3.6 ENTIRE PICTURE CONCEALMENT SCHEMES

The following two existing schemes were used to conceal the entire missing picture.

1. Copy the previous picture for the current missing picture. This scheme is simple to implement, requires no CPU time because the decoder simply replays the previous decoded picture, and works well in video sequences that contain a small amount of motion [1,32,33].
2. Copy the previous picture macroblock located in the same spatial location as the current picture missing macroblock that is motion compensated with its own motion vector [1,32,33].

The simulations for schemes 1 and 2 were done on every even picture while every odd picture is uncorrupted for 'Suzie' 1, 2 and 'Car Phone' 3, 4.

3.6.1 Simulation Results

The simulation results for schemes 1 and 2 for the entire missing picture are given in Table 9.

Concealment for 'Suzie' 1	*Mean Y-PSNR (dB)*	*Mean CPU Time (seconds)*
Scheme 1	26.45	-
Scheme 2	27.51	0.1388
Concealment for 'Suzie' 2	*Mean Y-PSNR (dB)*	*Mean CPU Time (seconds)*
Scheme 1	32.19	-
Scheme 2	36.99	0.1394
Concealment for 'Car Phone' 1	*Mean Y-PSNR (dB)*	*Mean CPU Time (seconds)*
Scheme 1	27.95	-
Scheme 2	25.32	0.1371
Concealment for 'Car Phone' 2	*Mean Y-PSNR (dB)*	*Mean CPU Time (seconds)*
Scheme 1	31.64	-
Scheme 2	29.86	0.1321

Table 9: Comparisons of Missing Picture Schemes 1 and 2

The two existing schemes overall mean concealment results significantly increased from using 'Suzie' 1, 'Car Phone' 1 to 'Suzie' 2, 'Car Phone' 2 video sequences because of the increased temporal correlation between consecutive pictures [3]. Using 'Suzie' 1 and 2, scheme 2 when compared to scheme 1 produced an overall mean concealment result of +1.060 dB ('Suzie' 1) and +4.800 dB ('Suzie' 2) with an overall mean CPU time of 0.1388 seconds ('Suzie' 1) and 0.1394 seconds ('Suzie' 2). Using 'Car Phone' 1 and 2, scheme 2 when compared to scheme 1 produced an overall mean concealment result of –2.630 dB ('Car Phone' 1) and –1.780 dB ('Car Phone' 2) with an overall mean CPU time of 0.1371 seconds ('Car Phone' 1) and 0.1321 seconds ('Car Phone' 2). Using the complete 'Suzie' 1, 2 and 'Car Phone' 1, 2 results, scheme 2 when compared to scheme 1 has an overall mean concealment result of +1.440 dB.

Concealment for the missing second picture in the ratio of P-pictures to I-pictures will be the same for both schemes since there are no previous picture macroblock motion vectors available for concealment (I-picture).

4. SUMMARY AND FUTURE WORK

4.1 SUMMARY

This thesis provided a general description of the H.263+ compression standard and the RTP/UDP/IP transport protocol used to send non-interactive real time streaming QCIF video for security surveillance. The thesis then presented the following issues:

- Every 256 bytes of coded video stream are encapsulated into a single datagram without considering the combined header size of 40 bytes for the RTP/UDP/IP datagram and the network's MTU of 1500 bytes (payload efficiency).
- Error recovery for datagram loss is not provided for the RTP/UDP/IP transport protocol that has no QoS for loss and sequence delivery and the multicast delivery over sub-networks that may have uneven resources.
- A ratio of four P-pictures to I-pictures with rate control is used for error resilience against datagram loss, which does not consider the network path and conditions, and quality of the coded picture.

The encapsulation and error recovery problem were solved by packetization and concealment schemes 1 and 2 proposed in [10]. Simulating the Internet datagram loss process using a Markov chain and datagram statistics that affect decoding of non-interactive real time streaming video in Chapter 2 showed packetization and concealment schemes 1 and 2 are robust against datagram loss. Packetization and concealment schemes 1 and 2 are robust against datagram loss because for the highest overall datagram loss rate of 34.85% the probability of a concealment failure for packetization scheme 1 is 10.72% and the probability of a concealment failure for packetization scheme 2 is 14.73%. Simulating the Internet datagram loss process also showed that packetization

scheme 1 estimated number of concealment failures is approximately one half when compared to packetization scheme 2 estimated number of concealment failures. The estimated number of concealment failures at each ratio of P-pictures to I-pictures (five to fifteen) and the five overall datagram loss rates provides an indication of the ratio of P-pictures to I-pictures to be used for a known network path and condition. The estimated number of failures for packetization and concealment schemes 1 and 2 improved the present situation that simply uses a ratio of four without considering the network path and conditions.

In Chapter 3 concealment scheme for the QCIF I-picture and P-picture's missing odd and even slices is proposed. The comparison between the proposed and existing scheme's overall mean concealment results and corresponding CPU times using two video test sequences and temporal sub-sampling of one and three showed the proposed scheme produced first or second highest concealment results. The proposed scheme's set threshold value provides means of selecting between two types of schemes used for concealment for the I-picture and P-picture's missing odd or even slices. Increased set threshold value permits the copying technique to be used more when the temporal sampling rate is decreased from three to one and the CPU time is reduced. Concealment results for the entire missing picture are lower compared to concealment for a picture's missing odd or even slices because of the higher amount of lost spatial area [10].

Chapter 2 and 3 comparisons of the estimated number of concealment failures, robustness against lost datagrams, concealment results, and CPU times provides the necessary information to be used with the visual subjective assessment (described in the future work section) for the selection of a packetiztion and concealment scheme.

4.2 FUTURE WORK

Future work is proposed for further investigation into: packetization and concealment transport protocols, error resilience, and modeling the Internet's datagram loss process.

4.2.1 Packetization and Concealment

The amount of subjectively acceptable concealment error for the QCIF video picture's missing odd and even slices and for an entire missing picture is not presently available. The magnitude of the visual error propagation will depend upon the size of the concealment error and the ratio of P-pictures to I-pictures. Future work would implement the schemes using the RFC2425 RTP payload format into the actual decoder 'C' software code. Concealment would be done for the *Y*, *Cr*, and *Cb* components of the video picture. The CPU time would be noted for decoding and concealment to ensure that it is below the H.261 standard recommended time of 150×10^{-3} seconds [18,29].

Comparison of the accepted concealment schemes would follow the ITU-T standard for subjectively gauging the quality of digital video. Comparisons would be done using a group of approximately 30 people not technically familiar with digital video, a video test sequence that has extensive detail and rapid motion, and a five-point image impairment scale as shown in Table 10 [26].

Image Impairment	*Score*
Imperceptible	5
Perceptible, but not annoying	4
Slightly annoying	3
Annoying	2
Very annoying	1

Table 10: Impairment Scale for Subjective Assessment

The concealment PSNR value and CPU time would be noted for the packetization and concealment scheme that has the highest subjective quality score. This information would help determine the packetization and concealment scheme to be employed.

As stated in the introduction the present H.263+ coded QCIF video stream is segmented into blocks of 256 bytes and placed into separate RTP/UDP/IP datagrams for transportation over the Internet. Using the subjectively acceptable packetization and concealment scheme, future work would be actual implementation of coding, packetization, transportation, de-packetization, decoding, concealment (if required), and viewing of the video test sequence.

4.2.2 Transport Protocols

IPv4 is presently used, however IPv6[2] will eventually replace IPv4 because of some of the following important enhancements [28]:

- Address space expansion (128 bits compared to 32 bits).
- Addressing flexibility (anycast address).
- Resource allocation support (labeling datagrams as belonging to a particular traffic flow such as real-time video).

Future work would study and possibly implement IPv6. Presently, there are no security provisions for the datagrams transmitted over the Internet. The option parameter in the IPv4 and IPv6 header allows the addition of the security parameter that attaches a security label to the datagram (encryption) [28]. Future work would study and possibly implement the security header.

---

[2] Versions 1 through 3 were successively defined and replaced by 4. Version 5 is assigned to the stream protocol, a connection-oriented Internet layer protocol. Version 6 then replaces version 4 [28].

Real Time Transport Protocol (RTP), Real Time Control Protocol (RTCP), and the Real Time Streaming Protocol (RTSP) are the Internet Engineering Task Force (IETF: the protocol engineering and development arm of the Internet) protocols that support multimedia streaming over the Internet [15]. RTCP monitors the periodic delivery of RTP datagrams and some of the following features are [15]:

- Provides feedback on the quality of data distribution.
- Establishes identification for each receiver.
- Scales or limits the control of datagram transmission with the number of receivers to avoid excessive feedback traffic in large multicast groups.
- Provides minimal session control information by using the feedback data so the transmitter may adjust its rate accordingly.

The feasibility of RTCP would be reviewed since it may not applicable for receivers located in sub-networks that are separated by large distances and having different processing and bandwidth resources.

RTSP is used in a number of commercial streaming media applications. RTSP coordinates the delivery of multimedia files through exchanges of various control messages between the receiver and the transmitter. RTSP typically runs over TCP though UDP may also be used [15]. The feasibility of RTSP would also need to be reviewed since it is more complex to implement when compared to RTP and RTCP.

4.2.3 Error Resilience

To help solve the problem of using the I-picture for error resilience, future research is proposed to investigate if any work has been done on varying the ratio of QCIF P-pictures to I-pictures for different source picture rates with rate control. If no previous

research has been done, then work will be done to determine the lowest ratio for the source picture rate by comparing the quality of the coded picture to the original picture subjectively and using the PSNR calculation. If the desired coded picture quality cannot be obtained at acceptable low ratios and source rates, the feasibility of implementing the coder proposed in [10] will be studied.

4.2.4 Datagram Loss Model

If the implementation of RTCP or RTSP is not practical for non-interactive real time streaming video, future research is proposed to increase the accuracy of the Markov chain model. Datagram discard due to network channel noise would not be included since most wired network's Bit Error Rate (BER) is negligible [37] compared to the statistic of datagram discard and out-of-order delivery from network routers. Accuracy would be improved by recording simultaneously the following Internet's RTP/UDP/IP datagram statistics for specific network paths, data rates, and time periods:

- Loss rate.
- Out-of-order delivery.
- Jitter.

The amount of acceptable jitter for non-interactive real time streaming video would be determined. The corresponding conditional probabilities using the subjectively selected the packetization scheme would be calculated. Figure 38 shows the four states of the Markov chain.

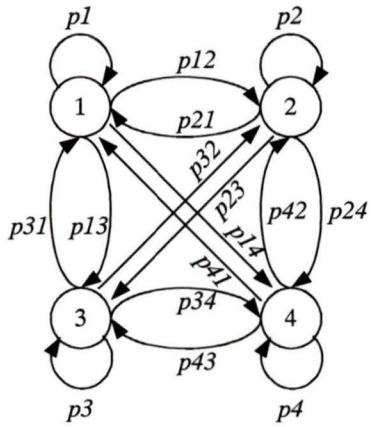

Figure 38: Internet Datagram Loss Process represented by a Markov Chain

A single datagram that is correctly received from the network is represented by state one (output is indicated by a '0'). A single datagram that is lost due to network routers is represented by state two (output is indicated by a '1'). A single datagram that is considered lost due to network out-of-order delivery is state three (output is indicated by a '1'). A single datagram that is considered lost due to network jitter is represented by state four (output is indicated by a '1'). The four states are connected together because the datagram statistics would be recorded simultaneously. Parameters *p1*, *p2*, *p3*, and *p4* express the conditional probability for a single time step. The remaining parameters (e.g., *p12*, *p21*, etc) express the state transition probabilities for a single time step. The improved Markov model would be used to determine the maximum ratio of P-pictures to I-pictures for picture quality and error resilience with rate control. The Markov model would also be used for future work on proposed concealment and recovery schemes.

BIBLIOGRAPHY

[1] M.E. Al-Mualla, C.N. Canagarajah, and D.R. Bull, 'Motion Field Interpolation for Temporal Error Concealment', *IEE Proceedings-Vision, Image, and Signal Processing*, vol. 147, no. 5, October 2000.

[2] Andrea Basso, Socrates Varakliotis, and Roberto Castagno, 'Transport of MPEG-4 over IP/RTP', *Packet Video Workshop*, Cagliari, Italy, May 2000.

[3] Vasudev Bhaskaran and Konstantinos Konstantinides, *Image and Video Compression Standards Algorithms and Architectures*, 2nd Edition, Kluwer Academic Publishers, Copyright 2000, Norwell Massachusetts U.S.A.

[4] Michael S. Borella, Debbie Swider, Suleyman Uludag, and Gregory B. Brewster 'Internet Packet Loss: Measurement and Implications for End-to-End QoS', *Proceedings of the 1998 Workshops on Architectural and OS Support for Multimedia Applications/Flexible Communications Systems/Wireless Networks and Mobile Computing*, pp. 3-21.

[5] C. Bormann, L. Cline, G. Deisher, T. Gardos, C. Maciocco, D. Newell, J. Ott, G. Sullivan, S. Wenger, and C. Zhu, *Network Working Group Request for Comments: 2425*, Standards Track for RTP Payload Format for the 1998 Version of ITU-T Rec. H.263+ Video, October 1998.

[6] C. Bormann, S, S. Casner, and M. Engan, *Network Working Group Request for Comments: 2509*, Standards Track for IP Header Compression over PPP, February 1999.

[7] Jill M. Boyce and Robert D. Galianello, 'Packet Loss Effects on MPEG Video Sent Over the Public Internet', *ACM Multimedia 98 – Electronic Proceedings*.

[8] Georg Carle, and Ernst W. Biersack, 'Survey of Error Recovery Techniques for IP-based Audio-Visual Multicast Applications', *IEEE Network Magazine*, pp. 2-14, November 1997.

[9] S. Casner, GMD. Fokus, R. Frederick, V. Jacobson, and H. Schulzrinne, *Network Working Group Request for Comments: 1889*, Standards Track for RTP: A Transport Protocol for Real-Time Applications, January 1996.

[10] Guy Côté and Stephan Wenger, 'Using RFC2429 and H.263+ at Low to Medium Bit-rates for Low-latenay Applications', *Packet Video Workshop 1999, New York*, U.S.A., May 1999.

[11] Guy Côté, Shahram Shirani, and Faouzi Kossentini, 'Optimal Mode Selection and Synchronization for Robust Video Communications over Error-Prone Networks', *IEEE Journal on Selected Areas in Communications*, Vol. 18, No. 6, June 2000.

BIBLIOGRAPHY

[1] M.E. Al-Mualla, C.N. Canagarajah, and D.R. Bull, 'Motion Field Interpolation for Temporal Error Concealment', *IEE Proceedings-Vision, Image, and Signal Processing*, vol. 147, no. 5, October 2000.

[2] Andrea Basso, Socrates Varakliotis, and Roberto Castagno, 'Transport of MPEG-4 over IP/RTP', *Packet Video Workshop*, Cagliari, Italy, May 2000.

[3] Vasudev Bhaskaran and Konstantinos Konstantinides, *Image and Video Compression Standards Algorithms and Architectures*, 2nd Edition, Kluwer Academic Publishers, Copyright 2000, Norwell Massachusetts U.S.A.

[4] Michael S. Borella, Debbie Swider, Suleyman Uludag, and Gregory B. Brewster 'Internet Packet Loss: Measurement and Implications for End-to-End QoS', *Proceedings of the 1998 Workshops on Architectural and OS Support for Multimedia Applications/Flexible Communications Systems/Wireless Networks and Mobile Computing*, pp. 3-21.

[5] C. Bormann, L. Cline, G. Deisher, T. Gardos, C. Maciocco, D. Newell, J. Ott, G. Sullivan, S. Wenger, and C. Zhu, *Network Working Group Request for Comments: 2425*, Standards Track for RTP Payload Format for the 1998 Version of ITU-T Rec. H.263+ Video, October 1998.

[6] C. Bormann, S, S. Casner, and M. Engan, *Network Working Group Request for Comments: 2509*, Standards Track for IP Header Compression over PPP, February 1999.

[7] Jill M. Boyce and Robert D. Galianello, 'Packet Loss Effects on MPEG Video Sent Over the Public Internet', *ACM Multimedia 98 – Electronic Proceedings.*

[8] Georg Carle, and Ernst W. Biersack, 'Survey of Error Recovery Techniques for IP-based Audio-Visual Multicast Applications', *IEEE Network Magazine*, pp. 2-14, November 1997.

[9] S. Casner, GMD. Fokus, R. Frederick, V. Jacobson, and H. Schulzrinne, *Network Working Group Request for Comments: 1889*, Standards Track for RTP: A Transport Protocol for Real-Time Applications, January 1996.

[10] Guy Côté and Stephan Wenger, 'Using RFC2429 and H.263+ at Low to Medium Bit-rates for Low-latenay Applications', *Packet Video Workshop 1999, New York*, U.S.A., May 1999.

[11] Guy Côté, Shahram Shirani, and Faouzi Kossentini, 'Optimal Mode Selection and Synchronization for Robust Video Communications over Error-Prone Networks', *IEEE Journal on Selected Areas in Communications*, Vol. 18, No. 6, June 2000.

[12] Victor DeBrunner, Linda DeBrunner, Longji Wang, and Sridhar Radhakrishnan, 'Error Control and Concealment for Image Transmission', *IEEE Communications Surveys and Tutorials*, First Quarter, 2000, vol. 3, no. 1.

[13] Constantinos Dovrolis, Damon Tull, Parameswaran Ramanathan, 'A Loss Concealment Algorithm for Packet Video Applications', *Proceedings of the 9th International Packet Video Workshop*, New York NY, May 1999.

[14] Fayez Gebali, *Computer Communication Networks Analysis and Design Part 1*, NorthStar Digital Design, Copyright 2001, Victoria B.C. Canada.

[15] Stephane Gruber, Jennifer Rexford, and Andrea Basso, 'Protocol Considerations for a Prefix-Caching Proxy for Multimedia Streams', *to appear in Infocom 2000*.

[16] Duane Hanselman and Bruce Littlefield, *Version 5 of the Student Edition of MATLAB®, Prentice Hall*, Upper Saddle River New Jersey, 1997.

[17] P. Hsu and K.J.R Liu, 'Software Optimization of H.263 Video Encoder on Pentium Processor with MMX Technology', Proc. *IEEE International Conference on Multimedia and Expo*, pp. 103-106, New York City U.S.A., August 2000.

[18] *ITU-T Recommendation H.263 version 2*, Video Coding for Low Bit Rate Communication, 02/98.

[19] Randy H. Katz, *Contemporary Logic Design*, The Benjamin/Cummings Publishing Company Inc., Copyright 1994, Redwood City, California U.S.A.

[20] Erwin Kreyszig, *7th Edition Advanced Engineering Mathematics*, John Wiley and Sons Inc., Copyright 1993, U.S.A.

[21] Nyeongkyu Kwon and Peter Driessen, 'Efficient and Fast Predictive Block Based Motion Estimation for Low Bit Rate Video Coding', *PACRIM'01*, University of Victoria, Victoria, British Columbia, Canada, August 2001.

[22] Kin K. Leung, Peter F. Driessen, Kapil Chawla, and Xiaoxin Qiu, 'Performance Improvement for Streaming Services by Link Adaptation and Power Control in EGPRS Wireless Networks', *IEEE Multiaccess, Mobility and Teletraffic for Wireless Communications*, Florida, December 2000.

[23] Max Luttrell, John Villasenor, Jeong-Hoon Park, and Dong-Seek Park, 'A Data Partitioning Based Video Coding Algorithm for Error Prone Channels', *International Packet Video Workshop*, Columbia University, New York City U.S.A., April 1999.

[24] Teruko Miyata, Harumoto Fukuda, Satoshi Ono, 'Characteristics of Successive Loss Process', *Proceedings of the 15th International 2001 Conference on Information Networking*, pp. 663-667.

[25] Vern Paxson, 'End-to-End Internet Packet Dynamics', *IEEE/ACM Transactions on Networking*, vol. 7, no. 5, pp. 277-292, June 1999.

[26] M. Angela Sasse and Anna Watson, 'Measuring Perceived Quality of Speech and Video in Multimedia Conferencing Applications', *ACM Multimedia 98 – Electronic Proceedings.*

[27] T. Shanableh and M. Ghanbari, 'Interframe Loss Concealment Techniques for Bursty Packet Losses in IP Environments', *International Packet Video Workshop, PVW-2000*, Cagliari Italy, May 2000.

[28] William Stallings, *Data and Computer Communications*, 6th Edition, Prentice-Hall, Copyright 2000, Upper Saddle River New Jersey U.S.A.

[29] A. Murat Tekalp, *Digital Video Processing*, Prentice Hall PTR, Copyright 1995, Upper Saddle River, New Jersey U.S.A.

[30] S. Tsekeridou, F. Alaya Cheikh, M. Gabbouj, and I. Pitas, 'Motion Field Estimation by Vector Rational Interpolation for Error Concealment Purposes', *Acoustics, Speech, and Signal Processing (ICASSP 1999)*, vol. 6, pp. 3397-3400, Phoenix Arizona U.S.A., March 1999.

[31] Deepak Turaga and Tsuhan Chen, 'Modeling of Dynamic Video Traffic', *The 2000 IEEE International Symposium on Circuits and Systems (ISCAS 2000)*, vol. 4, pp. 293-296, Geneva.

[32] Yao Wang and Qin-fan Zhu, 'Error Control and Concealment for Video Communication: A Review', *Proceedings of the IEEE*, vol. 86, no. 5, May 1998.

[33] Yao Wang, Stephan Wenger, Jiangtao Wen, and Aggelos K. Katsaggelos, 'Review of Error Resilient Coding Techniques for Real-Time Video Communications', *IEEE Signal Processing Magazine*, July 2000.

[34] Dapeng Wu, Yiwei Thomas Hou, and Ya-Qin Zhang, 'Transporting Real-Time Video over the Internet: Challenges and Approaches', *Proceedings of the IEEE, Vol. 88*, No. 12, December 2000.

[35] Kook-yeol Yoo, 'Low Complexity Error Concealment Method for the Transmission of H.263+ Coded Bit Stream over Mobile Channels', *Packet Video Workshop*, New York, U.S.A., May 1999.

[36] Jian Zhang, John F. Arnold, Micheal R. Frater and Mark R. Pickering, 'Video Error Concealment Using Decoder Motion Vector Estimation', *IEEE 10th Annual Conference on Speech and Image Technologies for Computing and Telecommunications*, vol. 2, pp. 777-780, 1997.

[37] Liren Zhang, David Chow, and Chee Hock Ng, 'Cell Loss Effects on QoS for MPEG Video Transmission in ATM Networks', *IEEE Conference of Communication*, vol. 1, pp. 147-157, 1999.

[38] C. Zhu, *Network Working Group Request for Comments: 2190*, Standards Track for RTP Payload Format for the 1998 Version of ITU-T Rec. H.263 Video, September 1997.

APPENDIX

A.1 UDP CHECKSUM

The UDP checksum is an error detection code based on a summation operation performed on the bits to be checked. The following steps show how the transmitter generates a checksum:

- Checksum data field itself is cleared.
- UDP/IP segment and the pseudoheader are broken into sixteen bit message blocks.
- One's complement sum [31] is computed for the entire sixteen bit message blocks.
- One's complement of this sum is placed in the checksum field.

The router performs the same calculation but includes the checksum. If the result is all zeros bits the check succeeds and if not, the datagram is discarded by the network router [8]. If two or any even number of bits located in the same column is erroneously inverted during transmission, an undetected error occurs and produces a corrupted datagram. The use of the checksum therefore is not completely foolproof since noise impulses are sometimes often long enough to erroneously invert more than one bit, especially at high data rates [2].

VITA

Surname: Huitika Given Names: Thomas Reino

Place of Birth: Fort Frances, Ontario, Canada

Educational Institutions Attended:

University of Victoria	1999 to 2002
Lakehead University	1985 to 1989
RCC School of Technology	1983 to 1985

Degrees Awarded:

B.Sc. in Electrical Engineering	Lakehead University	1989

Honours and Awards:

Natural Sciences and Engineering Research Council of Canada (NSERC) Industrial Postgraduate Scholarship (IPS-1)	2001 to 2002
University of Victoria President's Research Scholarship	2001 to 2002

Publications:

Thomas R. Huitika, Hyunho Jeon, Nyeongkyu Kwon, and Peter Driessen, 'A Concealment Scheme for H.263 Coded Video transported over the Internet using the RTP/UDP/IP Protocol', *2001 IEEE Pacific Rim Conference on Communications, Computers and Signal Processing (PACRIM 2001)*, Volume II, pp. 465-468, August 26-28, 2001, University of Victoria, Victoria BC Canada

UNIVERSITY OF VICTORIA PARTIAL COPYRIGHT LICENSE

Title of Thesis/Dissertation:

Packetization and Concealment Schemes for Delivering H.263+ Coded Video over Internet

Author [redacted]

Thomas Reino Huitika

May 2, 2002