Using Redundancy to Repair Video Damaged by Network Data Loss

Yanlin Liu and Mark Claypool
{yanlin,claypool}@cs.wpi.edu

Computer Science Department
Worcester Polytechnic Institute
100 Institute Road
Worcester, MA 01609-2280

Abstract

With rapid progress in both computers and networks, real-time multimedia applications are now possible on the Internet. Since the Internet was designed to support traditional applications, multimedia applications on the Internet often suffer from unacceptable delay, jitter and data loss. Among these, data loss often has the largest impact on quality. In this paper, we propose a new forward error correction technique for video that compensates for lost packets, while maintaining minimal delay. Our approach transmits a small, low-quality redundant frame after each full-quality primary frame. In the event the primary frame is lost, we display the low-quality frame, rather than display the previous frame or retransmit the primary frame. To evaluate our approach, we simulated the effect of network data loss on MPEG video clips and repaired the data loss by using the redundancy frame. We conducted user studies to experimentally measure users' opinions on the quality of the video streams in the presence of data loss, both with and without our redundancy approach. In addition we analyzed the system overhead incurred by the redundancy. We find that video redundancy can greatly improve the perceptual quality of video the presence of network data loss. The system overhead that redundancy introduces is dependent on the quality of the redundant frames, but a typical redundancy overhead will be approximately 10% that of primary frames alone.

1. Introduction

Emerging new technologies in real-time operating systems and network protocols along with the explosive growth of the Internet provide great opportunity for distributed multimedia applications. Multimedia data transmission on the Internet often suffers from delay, jitter, and data loss. Data loss in particular can be extremely high on the Internet, often as much as 40% [GBC98, Ma94, Pax99]. Unlike traditional applications, multimedia applications can tolerate some data loss. A small gap in a video stream may not significantly impair media quality, and may not even noticeable to users. However, too much data loss can result in unacceptable media quality.

There has been much work has been done to find effective techniques to repair audio streams damaged by loss [PHH98]. These techniques have proven to be effective for audio stream data loss, but may have yet to be applied to video. Most of the previous work in data loss recovery for video has focused on media scalability, error concealment and retransmission. However, most existing media scaling techniques have limitations, such as special network requirements. Error concealment is limited in how much loss can be hidden. Retransmission can serve for all types of networks, but it is not appropriate for some multimedia applications that can tolerate only short end-to-end delays.

In our research, we apply an existing forward error correction technique used for audio [HSH+95] and propose a means to piggyback low bandwidth video redundancy at the sender. A lost packet is replaced by the redundancy transmitted within the next packet. When the redundancy fails to repair the lost packet, a repetition based error concealment technique is used to fill the gap left. Figure 1.1 shows how the proposed scheme works.

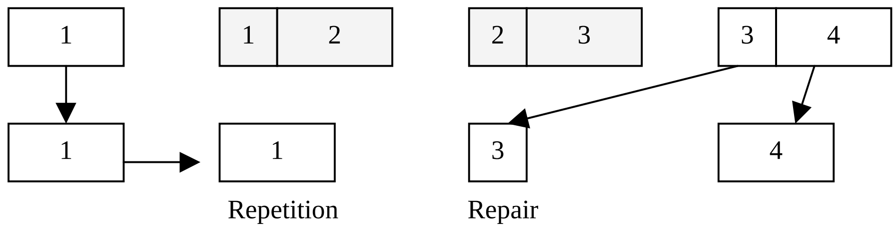

Figure 1.1 Our Approach – Combine Media Specific FEC and Packet Repetition

Many video compression techniques (such as MPEG) use lossy compression (some of the original image data is lost during encoding), by adjusting the quality and/or compression rate at encoding time. The higher the quality, the lower the compression rate, and vice versa. We encode the original video frames into two versions, one with high quality and one with low quality, as depicted in Figure 1.2. The high quality frames are sent as primary frames, while the low quality ones are considered secondary frames and are piggybacked with the next primary frame for repair in case of original primary frame is lost.

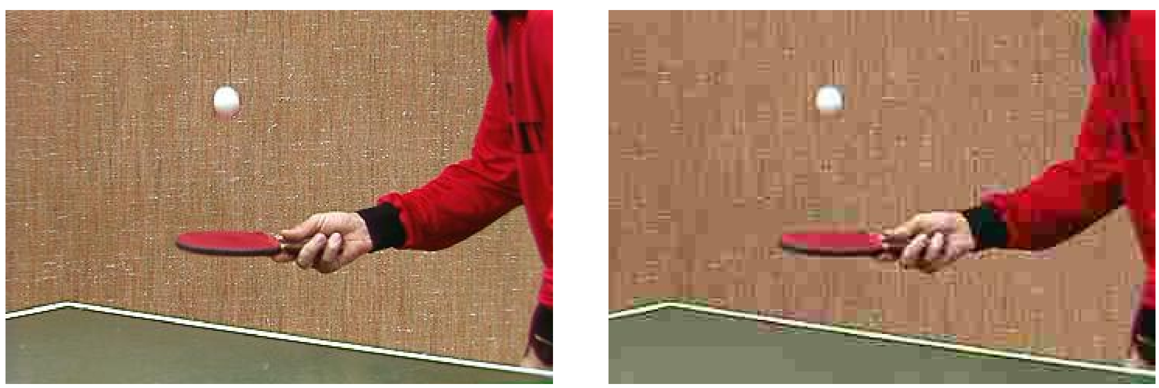

Figure 1.2 Two Frames with Different Quality and Compression Rates
Both of the two frames are compressed from the same original frame. The left frame is compressed with high quality, but has a low compression rate. The size of this frame is 19K bytes. The right frame is compressed with low quality, but has a high compression rate. The size of this frame is 3K.

To evaluate our approach, we first examine the effects our technique has on Perceptual quality, a measure of the performance of multimedia from the user's perspective. Since the redundancy added to the video stream requires extra processing time and network bandwidth, we also analyze the system overhead.

The remainder of the paper is as follows. In Section 2, we discuss related work. In Section 3, we present details of our approach to the problem of packet loss, describe our methodology for testing Perceptual quality, and discuss the user study results. In Section 4, we analyze the system overhead of the redundancy. In Section 5, we briefly discuss possible future work and in Section 6, we draw our conclusions.

2. Related Work

The goals of this section are to present related research in multimedia repair and present some fundamental concepts to better understand this work. The topics include audio and video loss repair, and MPEG encoding.

2.1. Multimedia Repair

Most video frames are larger then audio frames, but since audio has similar real-time requirements as video, we build our work upon past research in audio over the Internet. There are two types of audio repair techniques: *sender-based* and *receiver-based*, as depicted in Figure 2.1 [PHH98]. Sender-based repair techniques require the addition of repair data from the sender to recover lost packets. Receiver-base repair techniques rely only on information in the correctly received packets to ameliorate the effect of lost packets.

Sender-based repair techniques can be split into two categories: *passive channel coding* and *active retransmission*. With passive channel coding, the sender sends the repair data. The sender is not informed whether or not the loss is repaired. If it is not, the sender will have no further intention to repair it. Passive channel coding techniques include forward error correction (FEC) and interleaving-based schemes. Forward Error Correction can be further divided into Media Independent FEC, which uses block or algebraic codes to produce additional repair packets for transmission, and Media Dependent FEC, which uses the knowledge of the contents of the transmitted packets to aid repair. Interleaving attempts to reduce the effect of the loss by spreading it out. With active retransmission, if there is still time for repair, the sender will be informed of the loss and required to assist in recovering from the loss. Active retransmission techniques can be used when the application can tolerate larger end-to-end delays. A widely deployed multicast scheme based on the retransmission of lost packets is Scalable Reliable Multicast (SRM) [VEF98].

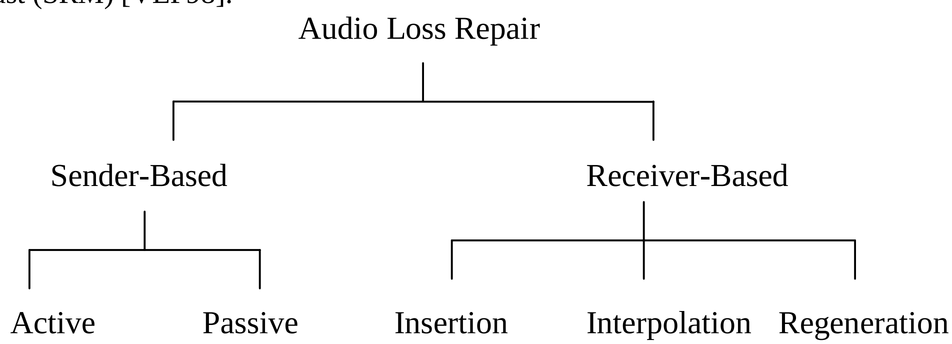

Figure 2.1 Taxonomy of Audio Loss Repair Techniques

Receiver-based repair techniques are also called *error concealment*. These techniques can be initiated by the receiver of an audio stream without the assistance of the sender. If the sender-based repair scheme fails to recover all loss, or when the sender is unable to participate in the recovery, receiver-based repair techniques can be used to make the loss of the packet less noticeable to the user. As shown in Figure 2.5, there are three kinds of receiver-based data loss repair techniques: *insertion*, *interpolation*, and *regeneration* schemes.

Some research in video data transmission over a network proposes to reduce the data loss by controlling the network congestion, or to provide a way to recover lost video frames [LPP+97]. Kanakia et al. dynamically change the video quality level during network congestion [KMR93]. The research by Gringeri et al deal with network data loss on an ATM network by using hierarchical coding and scalable syntax [GKL+98]. Hierarchical coding allows reconstruction of useful video from pieces of the total bit stream. This technique can ensure the base quality level of video transmission, but requires specific QoS services available in ATM networks. We seek to improve the quality of video streams with the existence of data loss on the Internet.

2.2. MPEG Video Encoding

Since video data are usually too large for raw transmission or storage, most video streams are compressed. The Motion Picture Expert Group (MPEG) standard is in popular use today [MP96]. MPEG compression is lossy, which means to achieve a higher compression rate some information in the original image may be lost during the compression and cannot be recovered when decoded. Thus, the compressed video streams may have lower quality than the original ones. The higher the compression rate, the lower the size of the frame, and vice versa.

To achieve a high compression rate, temporal redundancies of subsequent pictures must be exploited. MPEG distinguishes three main frame types of image coding for processing: I-frames, P-frames, and B-frames. To support fast random access, intra-frame coding is required. I-frame stands for Intra-coded frame. They are self-contained. The compression rate of the I-frames is the lowest within MPEG. P-frame stands for Predictive-coded frame. The encoding and decoding of P-frames requires the information of previous I-frames and/or all previous P-frames. Compression rates for P-frames are higher than that of I-frames. B-frame stands for Bi-directionally predictive-coded frame. The encoding and decoding of B-frames requires the information of the previous and following I- and/or P-frame, but achieves the highest compression rate. Their dependency relationship is illustrated in Figure 2.2.a. The encoding pattern of this stream is IBBPBBPBB, where the last two B-frames depend on both the second P-frame and the next I-frame.

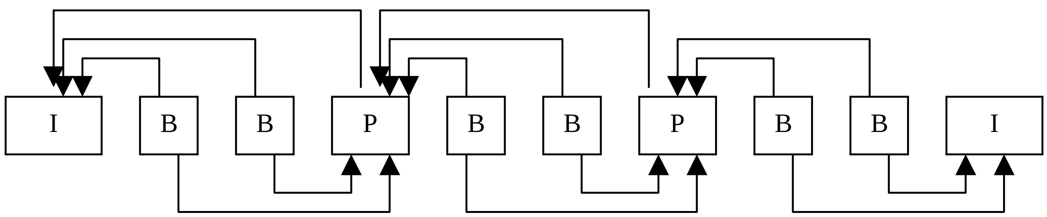

Figure 2.2.a MPEG Frame Dependency Relationship

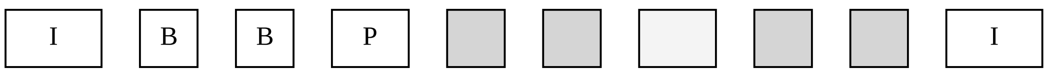

Figure 2.2.b Loss of the Second P-Frame

Shown in Figure 2.2.b, the loss of one P-frame can make some other P- and B-frames useless, while the loss of one I-frame can result in the loss a sequence of frames. In MPEG encoded video streams, I-frames and P-frames are more important than B-frames.

3. Perceptual Quality

In this section, we simulate the effects of our technique on MPEG video streams in the presence of packet loss by building movies that repeat frames if there is no redundancy and use a low quality frame when using redundancy. We use these streams in a user study in which we gather the opinion of the users, and draw conclusions on to what extent this technique can improve the perceptual quality of the video streams with loss.

3.1. Research approach

We use a repetition technique to compensate for the loss by repeating the frame that is received immediately before the lost one. If the lost frame is an important frame (I-frame or P-frame), the subsequent frames will appear lost as well since they are dependent upon the lost frame. By playing the previous frame again, the perceptual quality of the video may decrease. The end users may notice some sudden stop during the display, the screen momentarily frozen, followed by a big jump from one scene to a totally different one.

To solve this problem, we propose a method to include redundancy for video repair in the presence of packet loss during network transmission. Before transmission, the encoder generates the two versions of compressed frames, one with high quality and a low compression rate, the other with low quality and a high compression rate. The high quality frames we call *primary frames* and refer to them as H*i*. The low quality frames we call *secondary frames* and refer to them as L*i*. For each frame *i*, H*i* will be transmitted first and L*i* will be piggybacked with H*i+1*. At the receiver side, if H*i* is received successfully, it will be played to the end user directly and L*i* will be discarded upon its arrival. If, unfortunately, H*i* is lost or totally corrupted during the transmission, the decoder will wait for the next packet. L*i* will be extracted and take the place of the lost (or corrupted) H*i*. Figure 3.1 shows how our redundancy scheme may be incorporated into a video application.

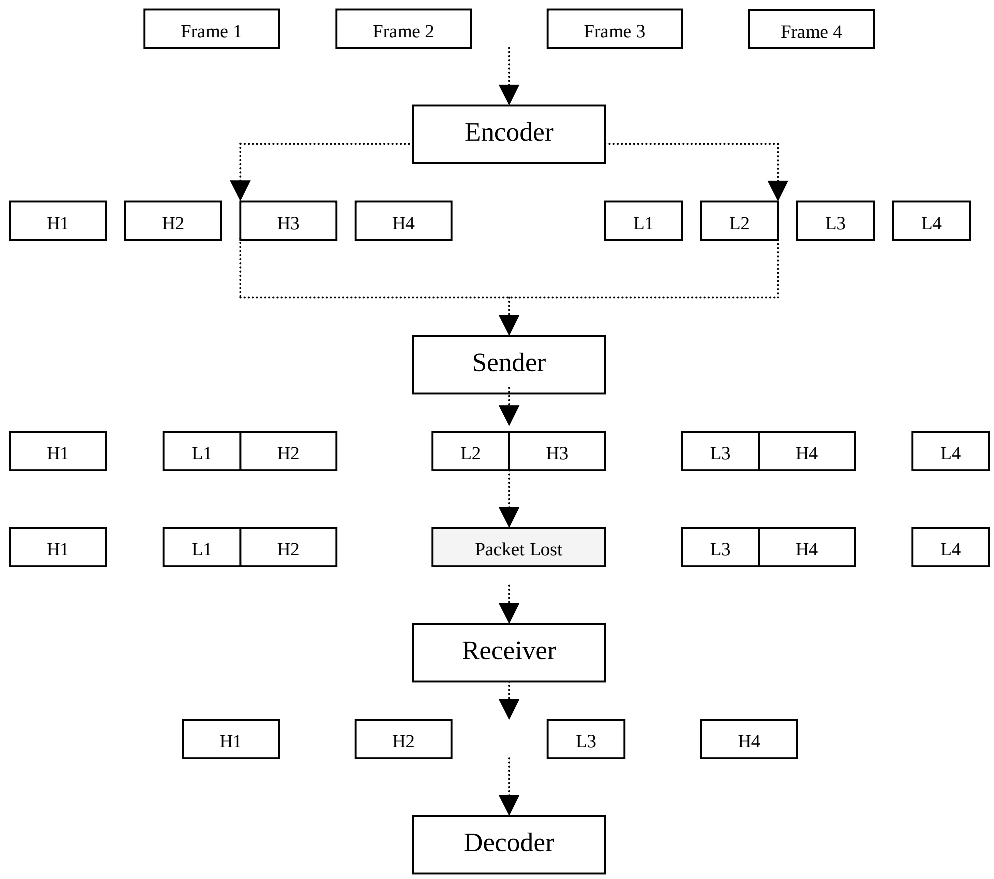

Figure 3.1 Video Redundancy Architecture

In this figure, each box represents a frame. The boxes with H*i* represent high quality frames and the boxes with L*i* represent low quality frames. Each low quality frame is piggybacked with the next high quality frame during transmission.

In a network with bursty loss, the secondary frame might also not be able to reach the receiver. In this case, not all the losses can be repaired. If neither H*i* nor L*i* manage to survive the network transmission, we use repetition to conceal the loss. Although the redundancy can make the video look better, sudden stops and abrupt jumps may still exist in the presence of heavy loss. Part of our user study examines to what extent consecutive frame loss has on repaired video streams.

3.2. Simulation

In our research, we simulated the network data loss by using repetition or redundancy as appropriate. In this subsection, we describe in detail the methodology we used to build movies that simulate lost frames.

The encoding tool we used is Berkeley MPEG Video Encoder [bm]. It contains the following tools that we used for this simulation: `mpeg_encode`, and `ppmtoeyuv`. The decoding tools we used are Berkeley MPEG Player [bm] and the Microsoft Media Player [mmp]. We wrote a Perl script to automate building the video clips used in the user studies:

- First we break the original `.mpg` file into separate `.ppm` files, one file for each frame in the video stream and convert each `.ppm` file into a `.yuv` file (EYUV format), which can be accepted by the MPEG encoder and has a smaller file size.
- Then we adjust the perceived frame rate from 30 fps to 5 fps. Since the encoder can only accept frame rates of 24 or more fps, and a typical frame rate through a WAN is at most 5 fps, we simulate 5 fps by duplicating some the frames in the video stream and dropping others. In our clips, the frame rate was set to be 30 fps with the duplicate rate 6, which means each frame is played 6 times and only 5 different frames are played within one second. For example, in an original 30 fps MPEG file, the first 12 frames are:
  ```
  F0  F1  F2  F3  F4  F5  F6  F7  F8  F9  F10 F11
  ```
  In our simulated stream, the frames become:
  ```
  F0  F0  F0  F0  F0  F0  F6  F6  F6  F6  F6  F6
  ```
  Although the stream will be played out at 30 fps, the frame rate perceived by the user will be 5fps.
- Next, we adjust the IPB pattern. In this simulation we used the common MPEG Group of Pictures (GOP) pattern: IBBPBBPBB.
- Then, we adjust the loss rate. In order to realistically simulate packet loss, we relied upon work in [GBC98] which gathered the data of 102 network data transmissions over the Internet, across the USA and New Zealand. Each transmission represented a 200-second trace of a multimedia connection using UDP. The contents transmitted included MPEG video data with different IPB pattern (only I-frames, or only I- and P-frames, or I-, P-, and B-frames) or CBR audio. Figure 3.2.a shows a histogram of the recorded loss rate and Figure 3.2.b shows a histogram of the recorded consecutive packet losses.

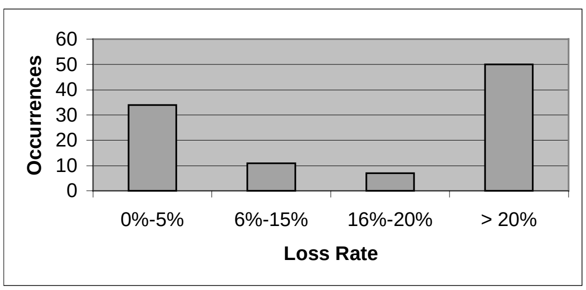

Figure 3.2.a Loss Rate Distribution

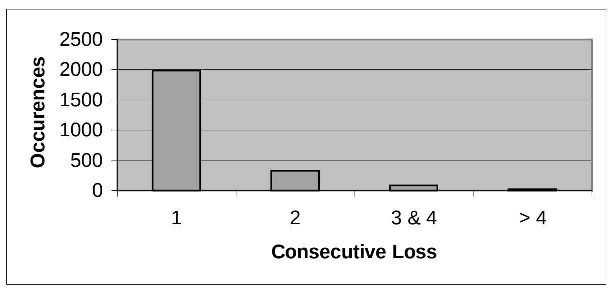

Figure 3.2.b Consecutive Loss Distribution

From Figure 3.2.a we can see that 50 of these transmissions had a loss rate greater than 20%. Of those that had a loss rate less than or equal to 20%, most of them are within the range of 0% to 5%. For a transmission with a loss rate greater than 20%, the quality is bound to suffer greatly under all kinds of repair, so we focused our attention to loss regions where repair techniques may acceptably improve the video quality.

From these results we concluded that for low loss rates (0% to 10%), most loss is of single consecutive packet. As you can see from Figure 3.2.b, that the total number of consecutive losses is much less than that of single loss.

Thus, in our experiment, we choose 3 loss rates for examination: 1%, 10%, and 20%, which we call the *raw* loss rate. For example, if 10 out of 100 frames are lost through the network, the *raw* loss rate is 10%. However the loss of I- or P-frame can leave the frames that are dependant on it useless (see Section 2.2), which results in a even higher loss rate as perceived by the user.

- Lastly, we adjust the consecutive loss parameter to study how bursts of packet losses can affect the repair result. In some circumstances, the network can introduce bursty loss to the video stream, with 2 or more consecutive lost frames. Most of the consecutive losses are from the transmission with loss rates greater than 10%. However, Figure 3.2.b shows that 4+ packet consecutive loss do occasionally occur. In this case, both the primary and redundancy frames will be lost. Three different numbers are used for this study: 1, 2 and 4. Therefore, the combinations of loss rate and loss pattern we used are:

Loss Rate:	1	10	20	20	20
Loss Pattern:	1	1	1	2	4

- Our last step is to simulate packet loss. Since a B-frame relies on the I- and/or P-frame both before and after, it is impossible to play a B-frame without first receiving the frames it depends upon. Thus the compression sequence and transmission sequence for the frames are different from the IPB pattern we specified. For the pattern IBBPBBPBB, the transmission sequence will be IPBBPBBIBB. So even if the two frames are lost in a sequence during the transmission, during playback, they are not necessarily played adjacent to each other.

We encoded the primary frames with an MPEG-quality of 1, which is the best quality. For the secondary frames, we encoded it with an MPEG-quality of 25 (out of 31). With this quality level, the encoded frames have a much higher compression rate. Although the frame quality of such encoding is quite coarse, under typical Internet loss, the low quality frames have fewer chances to be displayed, so only a small percent of frames are low quality. We chose a quality level of 25 for the secondary frames based on pilot tests that indicated users could notice the degradation of the clearness, but the frames still conveyed the basic content information.

3.3. User Study

We build upon the user-study techniques of other researchers study [RH97, GT98, WS98] in designing our user study. Using the techniques given in Section 3.2 for simulating packet loss in video streams, we generated MPEG files for our user study. Twenty-two unique video clips were chosen for the study. Two are without loss, ten are redundancy repaired with the five combinations of loss rate and loss pattern, and ten are of the same five combinations that simulate the effect of normal packet loss with repetition.

The study was done on two 600MHz Alpha machines running Windows NT version 4.0. The player used was Microsoft Media Player 6.0 [mmp]. The average frame rate achieved was 30 fps, which matched the frame rate specified during the generation of the video clips.

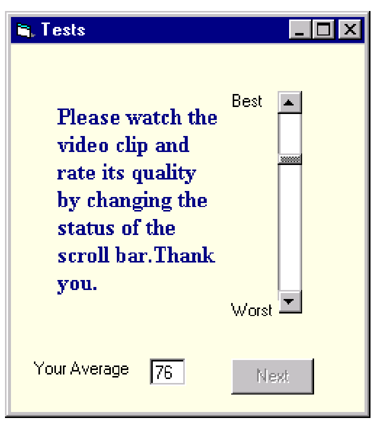

Figure 3.3 Screen Shot for the Message Box for Entering Perceptual Quality Scores

We designed and developed a Visual Basic program to assist the user study. We record the user information, such as the computer familiarity and video watching frequency and the scores that the user gives to each video clip. After the information is entered, we show a perfect video clip to “prime” all users equally. The 22 clips were ordered such that the video clips with high loss were not clustered together.

We provided a slider for the users to enter perceptual quality scores between 0 and 100. Figure 3.3 shows the message box displayed to the users after a video clip was displayed. The text box in the bottom of this message box shows the current user’s average score given for the video clips that have been viewed. The initial value of the slider is set to this value, so that the user can easily move it up if they find the current video has a quality above average, or down if they find the current video quality below average. The study lasted for two weeks and included 42 users.

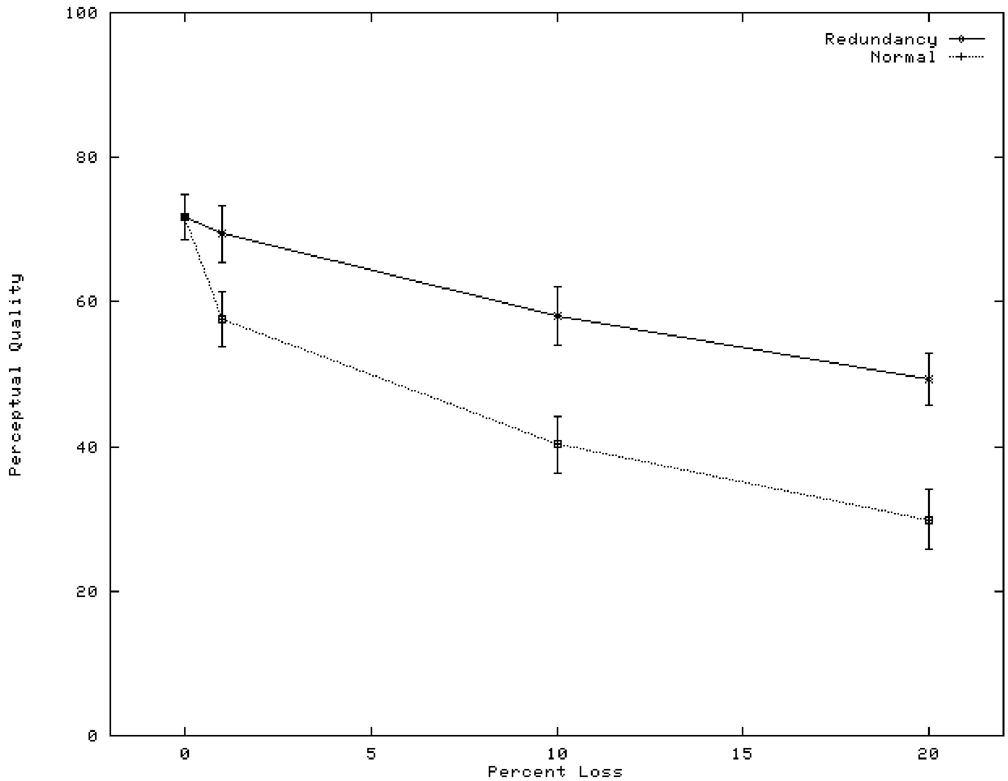

Figure 3.4 Effects of Loss Rate to the Perceptual Quality
The x-axis represents percent loss, ranging from 0% to 20%. The y-axis represents average score of the perceptual quality we gathered from the user study. The error bars represent 95% confidence intervals around the mean.

After gathering all the scores from the users, we examine the data to compare the average scores for redundancy repaired video clips and unrepaired video clips. Figure 3.4 plots the average quality scores for the videos that have no consecutive loss. We calculated a 95% confidence interval for each data point, depicted with an error bar.

We can see that the redundancy repair technique improves the quality of the video by 20% in the presence of low loss (1% raw loss rate). With high raw loss rate (20%), this technique can improve the quality of the video by 65%. As shown in Figure 3.4, the average score for 0% loss, which is considered as perfect video, is 72. With the increase in the percent loss, the quality for both redundancy repaired videos and unrepaired videos decreases exponentially. However, the perceptual quality with redundancy repair decreases much less than without. For a 1% frame loss, the average score for redundancy repaired video is 69, which is very close to the scores for video with no loss. Figure 3.4 shows that the average score for 1% loss with redundancy repair falls within the confidence interval range of the average quality for video with no loss. The difference between the qualities of these two kinds of videos is small and cannot be noticed in some cases. Without the redundancy repair technique, the quality of the video decreases dramatically to 58, which shows a significant difference between 0% and 1% loss. Users can easily notice the degradation in quality from the seemingly small amount of loss.

With the increase of the percent loss, the difference between redundancy repaired videos and unrepaired videos becomes larger. With 10% loss and no redundancy, the average quality score is 30, while the average quality score with redundancy is 49. While this is far from the perceptual quality score of video with no loss, it is still far better than the score for 10% loss without redundancy. With the same percent loss, there is no overlap between the confidence intervals of those with redundancy and those without.

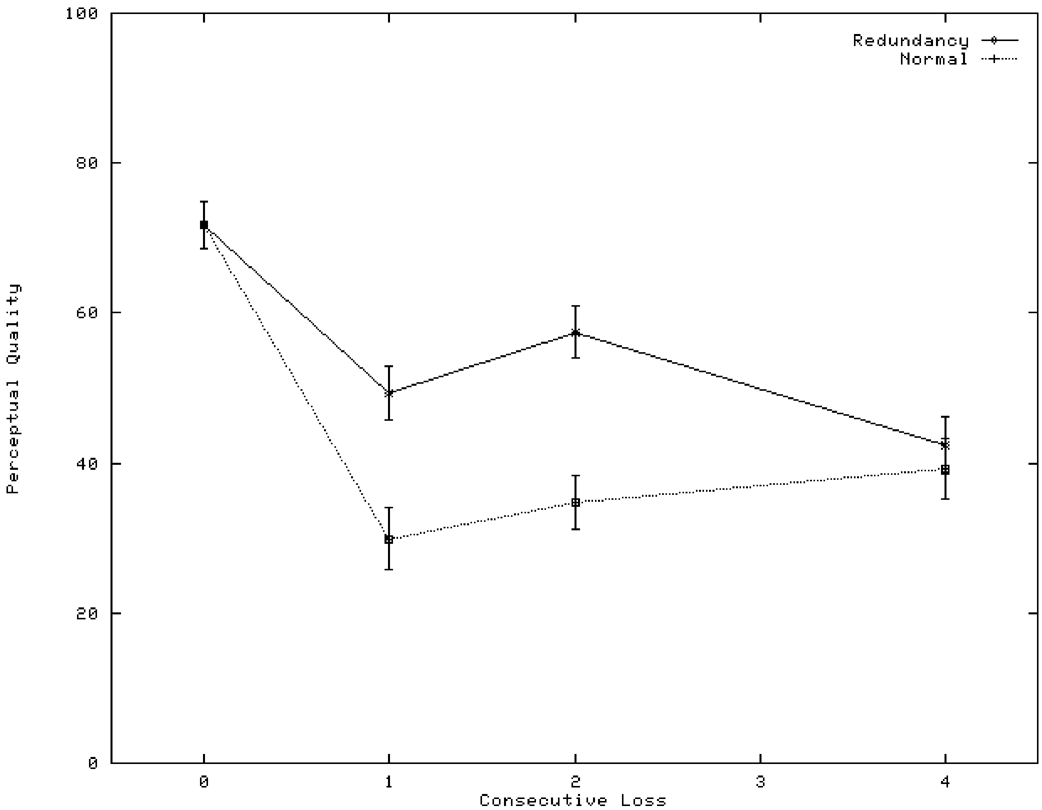

Figure 3.5 Effects of Loss Pattern to the Perceptual Quality

The x-axis represents the number of losses in a sequence. The y-axis represents the average score of perceptual quality. The error bars represent 95 % confidence intervals around the mean.

Figure 3.5 shows the average perceptual quality of video clips with the same loss rate, 20%, but with different loss patterns. The x-axis represents the number of the lost frames in a sequence, 1, 2, or 4. The y-axis represents the average perceptual quality.

Note that the average quality increases as the number of consecutive packet losses increases. We believe this is because with higher consecutive number and same loss rate, there are few gaps within the stream than within the single losses. Thus, fewer dependent frames are lost because of the loss of other frames.

As shown in Figure 3.6.a, a P-frame and a B-frame are lost in a sequence. Another three B-frames all depend on the P-frame, and cannot be reconstructed if the P-frame is lost. Even if the B-frame is not lost, it will become useless as well. The loss of the B-frame does not affect other frames. The perceptual loss to the user is 5 frames. In Figure 3.6.b, two P-frames are lost. Each one has some other frames dependent on it. The first lost P-frame left 4 B-frames useless, while the second lost P-frame left another

5 B-frames useless. The perceptual loss to the user is 10 frames. Although these two streams have the same raw loss rate, the loss pattern determines the final streams visible to the user. Usually with the same frame rate, single losses may result in a greater number of perceptually lost frames.

Consecutive loss also makes redundancy less useful. As shown in Figure 3.5, the average perceptual quality for redundancy repaired video clips increases when the consecutive loss number changes from 1 to 2, but decreases when consecutive loss increases. For single losses, the redundancy can always be received and thus the loss can always be repaired. With the existence of consecutive loss, the redundancy can be lost with the primary frame in a sequence. With a consecutive loss number of 2, it is likely that no important frames (I- or P-frames) are lost. However, in 4 consecutive frame loss, there will always be one I- or P-frame within the lost frames and the chance of repair is small. The average quality scores for videos with and without redundancy with 4 consecutive losses are very close. In fact, their confidence intervals overlap. Figure 3.5 shows that with a consecutive loss number of 4, there is no statistical difference between the average perceptual qualities of redundancy repaired and unrepaired video clips and there is no obvious advantage of using the redundancy repair technique when a large amount of bursty loss exists.

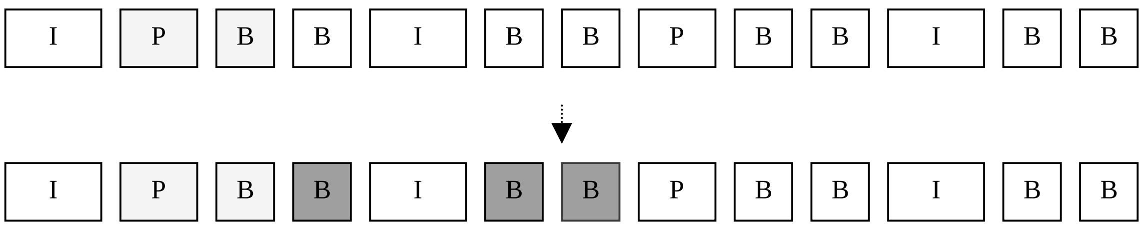

Figure 3.6.a Two Frames Lost in a Sequence

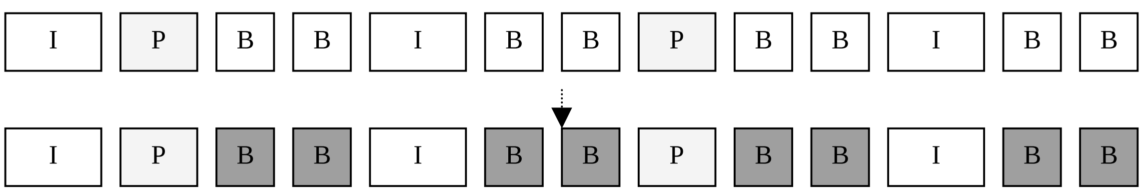

Figure 3.6.b Two single losses

Each box represents one frame. The boxes that have the color of light gray are considered to be lost while those have the color of dark gray are considered useless due to the loss of other frames.

4. System Analysis

Although our user study indicated that redundancy can improve the perceptual quality of video in the presence of packet loss, the secondary frames require extra buffer space and processing time. In this section, we analyze the size overhead that the low quality redundancy adds to the system. See [Liu99] for details on the processing overhead.

With our approach, there is no extra packet during network transmission since the redundancy data are piggybacked with the primary frames. This is significant as congestion on the Internet is often determined by the number of packets. However, the redundancy adds to the overall bytes transmitted. We compare the sizes of high quality frames vs. the low quality frames. Examinations are based on the type of video and the average sizes of the I-, P-, and B-frames. As in the user study, in this analysis, the high quality frames have the MPEG-quality of 1 and the low quality frames have an MPEG-quality of 25.

Figures 4.1–4.3 capture the size overhead of quality 1 versus quality 25 frames. Figure 4.1 compares the sizes of primary and secondary frames for one particular video stream, Figure 4.2 gathers the information of overhead ratio to the primary frames, and Figure 4.3 compares the size differences for different kinds of videos.

4.1. Frame Size Differences

Figure 4.1 compares the encoded sizes of primary frames vs. secondary frames with the same type of movie, a news clip. The GOP we used was IBBPBBPBB. Information about 51 I-frames, 100 P-frames and 300 B-frames for each of these two quality levels were gathered to derive the result. From Figure 4.1.1, we can see that the average size of primary (quality number 1) I-frames is 29.2 Kbytes, while the average size of secondary (quality number 25) I-frames is 4.1 Kbytes. The average ratio of secondary I-frame size over primary I-frame size is about 14.25%. The 95% confidence interval is between 14.01% and 14.49%.

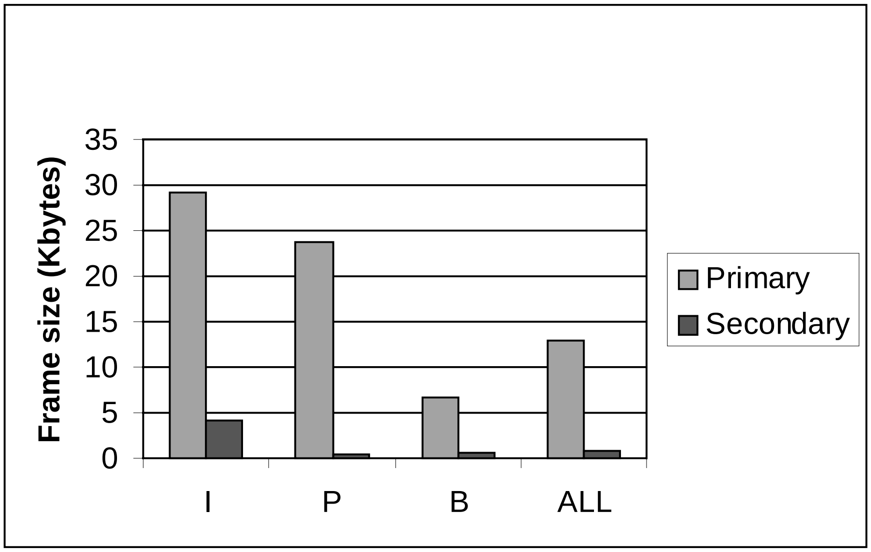

Figure 4.1 Frame Size Difference for Primary Frames and Secondary Frames
The x-axis represents the type of frames to be examined. Information of three kinds of MPEG frames, I-, P-, B-, as well as the average of all the frames is shown in the figure. The y-axis represents the average size of each type of frame. This experiment was done on a news video.

Similarly, we gathered data for P-frames and B-frames. The average size of primary P-frames is 23.8 Kbytes, which is slightly smaller than that of primary I-frames. However, the average size of secondary P-frames is 0.46 Kbytes, almost 10% of average size of secondary I-frames. The average ratio of secondary P-frame size over primary P-frame size is only about 1.8%, which is extremely low, with the probability confidence of 95%, the confidence interval is between 1.5% and 2.0%.

Average size of primary B-frames is 6.6 Kbytes, which is about ¼ the size of primary I-frames. The average size of secondary B-frames is 0.53 Kbytes. The average ratio of secondary B-frame size over primary B-frame size is 12.8%, similar to that of I-frames, with the probability confidence of 95%, the confidence interval is between 11.6% and 14.1%. Thus, the size of secondary B-frames is about 13% of the primary counterparts.

After gathering the information of all the frames, we conclude the average primary frame size is 13 Kbytes, the average secondary frame size is 10 Kbytes, and the average ratio of secondary frame size over primary frame size is 10.5%. Therefore, for this video clip, the average overhead that the redundancy (secondary frames) added to the video stream is about 10% of the primary frames.

4.1.2. Video Contents

In the above section, we examine the frame size differences for different quality numbers for one particular video (a news clip). It is also possible that the contents in the video affect the encoding and the frame sizes. Figure 4.2 compares the frame size differences for four kinds of videos: animations, sports, sitcoms, and news.

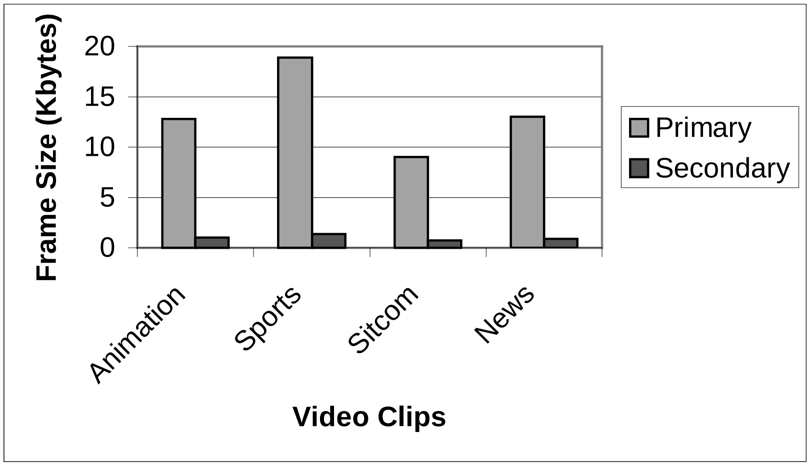

Figure 4.2 Frames Size Differences for Four Videos
The x-axis represents different types of videos. Four basic types are examined here, animation, sports, sitcom, and news. The y-axis represents the average size of the all frames.

Figure 4.2 shows the frame size differences for all the I-, P-, and B-frames. The primary frames range from 9.0 Kbytes to 19.0 Kbytes, and the secondary frames range from 0.72 Kbytes to 1.4 Kbytes. The four secondary frame size vs. primary frame size ratios are 22%, 11%, 9%, and 11%. The 22% one results from the high ratio of I-frames and B-frames. P-frames for the animation video are quite similar to other videos.

Our research shows that ratios for sports, sitcom, and news are quite similar to each other. The overall average ratio, I-frame ratio and B-frame ratio are all between 9% and 14%. The ratios for all the P-frames are very low. All of them are below 5% with most of them below 2%. P-frames are relatively important within the MPEG frames and the overhead for secondary P-frames is very low suggesting that they are excellent candidates for redundancy.

Figure 4.3 shows the average ratios of the size of the overhead over the size of the primary frames. We computed 95% confidence intervals for I-frames, P-frames, B-frames, and all I-, P-, B-frames. We see that for I- and P-frames, there is little variance in the ratios. The ratios for different video clips tend to be close to each other, while there can be a lot of variance with the ratio of B-frames, which makes the overall ratio vary dramatically. However, since B-frames are the least important frames with MPEG encoding scheme, the repair of lost B-frame is not as beneficial as the repair of lost I-frames and lost P-frames. One option to reduce the overhead the secondary frames added to the system and network transmission is to transmit secondary frames only for I-frames and P-frames. Since the overhead sizes of these two kinds of frames are more predictable and the main purpose of this repair technique can be fulfilled by the modified version of the our approach, the large variance in B-frame overhead size is not of much importance. However, if we do choose to transmit the secondary frames for B-frames, the absolute overhead size for each frame is about 343 bytes, which is very low compared to I- and P-frame's overhead.

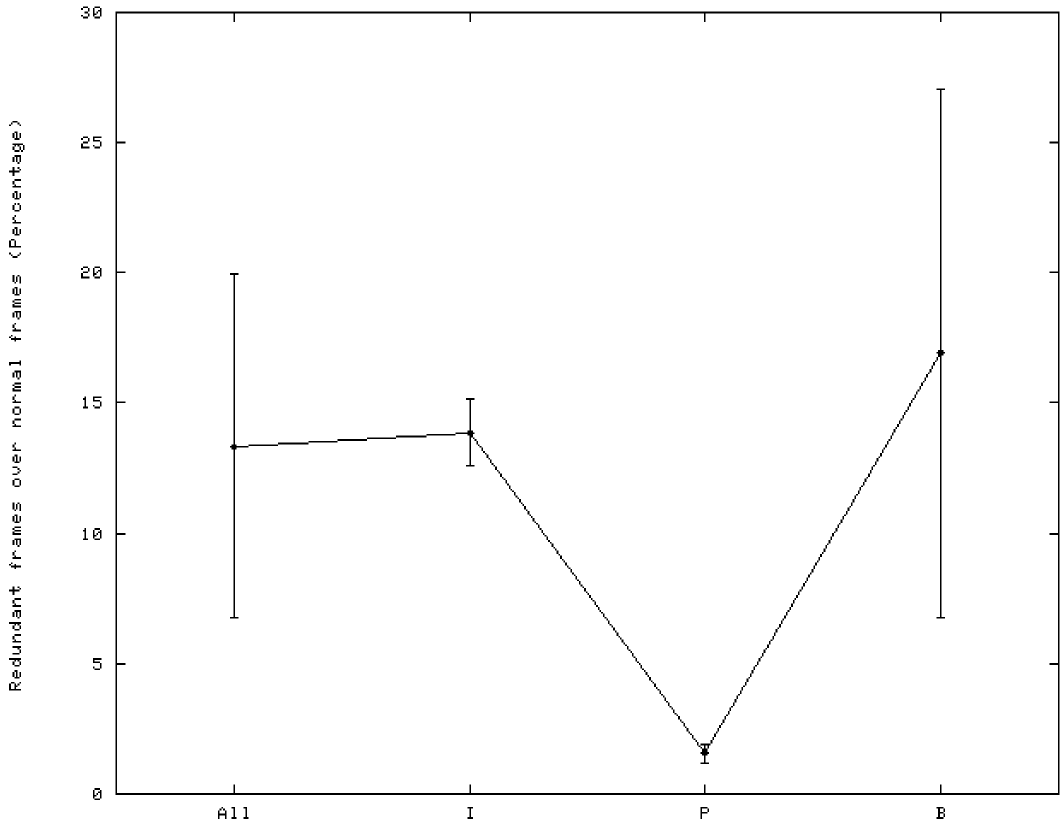

Figure 4.3 Ratios of Overhead over Frame Size
In this Figure, *All* in x-axis represents all the I-, P-, B-frames. *I* represents all the I-frames, *P* represents all the P-frames and *B* represents all the B-frames.

5. Future Work

With our technique, if each frame is to be decoded and played as soon as it arrives, there will be added delay during the display when one packet is lost until the secondary frame is extracted, decoded and played. Analysis on how much jitter video redundancy introduces to the display and how to solve this problem can be an interesting issue for future research. Also, to what degree the overhead affects network congestion is another area for further study. When a larger end-to-end delay is tolerable, it is possible to piggyback the transmission of the secondary frame 3 or 4 or even more frames back in order to repair a video stream even in the presence of bursty loss. It is also possible to combine video redundancy with other repair techniques, such as interleaving.

6. Conclusions

In this paper, we present a solution to ameliorate the effects of network data loss that damages video stream. Our approach piggybacks redundant video frames within the transmitted video stream in order to repair lost frames. At the sender, images are compressed into two versions, one with high quality and a large frame size, the other with low quality and a small frame size. High quality frames are sent to the receiver as the primary frame, while low quality frames are piggybacked with the next primary frame as the redundant frames. In the case the primary frame is lost, the corresponding low quality redundant frame is used to replace it. We investigated the benefits of this approach to user perceptual quality, as well as the overhead on the system.

We find most single frame loss can be repaired by the redundancy added to the video stream. When the loss rate is 20%, the perceptual quality of repaired video is rated about 65% greater than that of unrepaired video. With 1% single loss, perceptual quality can drop as much as 20% below that of video with no loss, while the average quality with 1% single loss redundancy repaired video streams is statistically the same as that of perfect video. We conclude that video redundancy competely repairs the perceptual quality in the presence of single consequent packet loss, the major loss pattern in the Internet today.

We also examined how the loss pattern reduces the effectiveness of this technique. With the same loss rate, higher consecutive loss results in a better perceptual quality for unrepaired video streams. However, for redundancy repaired videos, this is not the case. When four consecutive frames are lost, at least one important frame (I- or P-frame) is lost and the chance of this particular frame can be repaired is low. It is unlikely that redundancy will make significant differences in the presence of bursty loss. However, video redundancy is a reasonable repair method when the loss rate is under 20% since a pattern of more than two frames lost in a sequence rarely happens when the loss rate is relatively low. At a very high loss rate, perceptual quality is bound to suffer under all repair schemes.

The advantage of video redundancy is that it improves video quality in the presence of most packet loss, while the disadvantage is that it adds overhead to the system and the network. Low quality frames need time and space to be read, multiplexed, transmitted, and extracted. The extra time and space depends mainly on the size of low quality frames. By piggybacking the redundant frames, we add no additional packets to the network. Our analysis shows that size of typical low quality frames is only between 9% and 10% that of high quality frames. If we choose to only repair I- and P-frames, the ratio can be further reduced and will add an even smaller overhead.

In summary, in this paper, we present a method to encode two different versions of frames for the same video at the sender. We designed a means to simulate the effect of data loss and repair the lost high quality data by low quality redundancy. We conducted a user study to measure the effects of this repair technique. Results of the user study indicate that with the addition of about 10% overhead, video redundancy can greatly improve the perceptual quality of video streams in the presence packet loss. The contributions of this work include: experiments which measure the effects of loss on the perceptual quality of video; analysis of the system overhead from the redundancy; a methodology for conducting perceptual quality user studies; analysis of the effects of video content on system overhead; and a redundancy repair technique applied to MPEG video.

References

[bm] *Berkeley MPEG encoder and player*, http://bmrc.berkeley.edu/frame/research/mpeg/.

[GBC98] J., W. Buchanan, M. Claypool, *MMlib – A Library for End-to-End Simulation of Multimedia over a WAN*, MQP CSMLC-ML98, May 1998.

[GKL+98] Steven Gringeri, Bhumip Khasnabish, Arianne Lewis, Khaled Shuaib, Roman Egorov, Bert Basch, *Transmission of MPEG-2 Video Streams over ATM,* In Proceedings of the IEEE Multimedia Systems Conference, Vol. 5, No. 1, Jan-Mar, p.58-71, 1998.

[GT98] G. Ghinea, J. Thomas, *QoS Impact on User Perception and Understanding of Multimedia Video Clips*, In Proceedings of the ACM Multimedia Conference, 1998.

[Ha97] Mark Handley, *An Examination of Mbone Performance*, USC Information of Science Institute Research Report: ISI/RR-97-450, 1997.

[HSH+95] Vicky Hardman, M. Angela Sasse, Mark Handley, Anna Watson, *Reliable Audio for Use over the Internet*, in Proceedings INET95, Ohahu, Hawaii, 1995.

[KMR93] Hemant Kanakia, Partho Mishra, Amy Reibman, *An Adaptive Congestion Control Scheme for Real-Time Packet Video Transport*, ACM SIGCOMM'93 – Ithaca, NY, pp. 20-31, September 1993.

[LPP+97] Xue Li, Sanjoy Paul, Pramod Pancha, Mostafa Ammar, *Layered Video Multicast with Retransmission (LVMR): Evaluation of Error Recovery Schemes*, Proceedings of NOSSDAV 97, St. Louis, MO, May 1997.

[Liu99] Yanlin Liu, *Video Redundancy - A Best-Effort Solution to Network Data Loss*, Master Thesis, Worcester Polytechnic Institute, May 1999.

[Ma94] Bruce Mah, *Measurements and Observations of IP Multicast Traffic*, Technical Report UCB/CSD-94-858, University of California, Berkeley, CA, December 1994.

[MM96] Matthew Mathis, Jamshid Mahdavi, *Forward Acknowledgement: Refining TCP Congestion Control*, ACM SIGCOMM Computer Communication Review,Vol. 26, No. 4, Oct, Pages 281-291, 1996.

[mmp] *Microsoft Media Player6.0*, http://www.freeware.com.

[MP96] Joan Mitchell, William Pennebaker, *Mpeg Video: Compression Standard*, Chapman & Hall, SBN: 0412087715, 1996.

[Pax99] Vern Paxson, *End-to-End Internet Packet Dynamics*, IEEE/ACM Transaction on Networking, Fall 1999 (to appear).

[PHH98] Colin Perkins, Orion Hodson, Vicky Hardman, *A Survey of Packet-Loss Recovery Techniques for Streaming Audio*, IEEE Network Magazine, September/October 1998.

[RH97] H. de Ridder and R. Hamberg, *Continuous Assessment of Image Quality*, SMPTE Journal, February Pages 123-128, 1997.

[SN95] Ralf Steinmetz, Klara Nahrstedt, *Multimedia: Computing, Communications & Applications*, Prentice-Hall Inc., ISBN 0-13-324435-0, 1995.

[VEF98] K. Varadhan, D. Estrin, S. Floyd, *Impact of Network Dynamics on End-to-End Protocols: Case Studies in TCP and Reliable Multicast*, In the Proceedings of the IEEE Symposium on Computers and Communications, July 1998.

[WS97] A. Watson & M. A. Sasse, *Multimedia Conferencing via Multicast: Determining the Quality of Service Required by the End-User*, Proceedings of the 1997 International Workshop on Audio-Visual Services Over Packet Networks 15-16 September 1997.

[WS98] A. Watson & M. A. Sasse, *Measuring perceived quality of speech and video in multimedia conferencing applications*, In Proceedings of ACM Multimedia'98, Bristol, UK, September 1998.