Using Redundancy and Interleaving to Ameliorate the Effects of Packet Loss in a Video Stream

Yali Zhu, Mark Claypool and Yanlin Liu
Department of Computer Science
Worcester Polytechnic Institute
100 Institute Road
Worcester, MA 01609, USA
{yaliz,claypool,yanlin}@cs.wpi.edu

ABSTRACT

Traditionally, the Internet had been dominated by text-based applications such as file transfer, electronic mail and recently the Web. With the rapid improvement in computer and network technologies, high-bandwidth, interactive streaming multimedia applications are now possible on the Internet. However, the Internet does not provide the necessary Quality of Service (QoS) guarantees needed to support high-quality, real-time multimedia transmission, causing Internet multimedia applications to suffer from delay, jitter and loss. Among these, loss, typically caused by network congestion, degrades the perceptual quality of multimedia streams the most.

We propose two new techniques to repair video damaged by network data loss: redundancy and interleaving. The redundancy approach transmits a small, low-quality 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. Redundancy works well in case of single loss, but becomes less effective when multiple consecutive losses occurs in the video stream. To address this problem, we propose a video interleaving approach that ameliorates the effects of loss by spreading out bursty packet losses. The sender first re-sequences data before transmission to help distribute packet loss, and returns the data to their original order at the receiver. We apply both approaches to MPEG and evaluate the benefits to perceptual quality with user studies. Our results show that with approximately 10%-15% bandwidth overhead, both interleaving and redundancy significantly improve the perceptual quality of Internet video, while interleaving performs better in the case of consecutive loss, at the expense of increased latency.

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. However, the Internet does not provide the necessary Quality of Service (QoS) guarantees that are needed to support high-quality, real-time video transmission. Multimedia data transmitted over the Internet often suffers from delay, jitter, and data loss. Data loss in particular can be extremely high on the Internet, often as high as 40% [3, 11]. 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 be noticeable to users. However, too much data loss can result in unacceptable media quality.

A number of techniques exist to repair packet loss in a media stream [9]. 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 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 delay.

In our research, we develop two video repair techniques: redundancy and interleaving. Redundancy is an forward error correction technique that has typically been used for audio repair only. We propose a means to piggy-back low bandwidth redundancy to the video stream at the sender. A lost packet is replaced by the redundancy transmitted within the next packet [5]. When the redundancy fails to repair the lost packet, a repetition based error concealment technique is used to fill the gap. Figure 1 shows how our proposed scheme works. The original stream at the top contains primary frames piggy-backed with redundant frames. In the case that packet 2 and 3 are lost, the resulting stream at the bottom uses repetition of the primary frame 1 and repairs the primary frame 3 with redundancy.

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 compres-

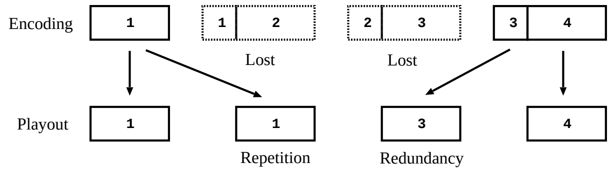

Figure 1: Our Video Redundancy Approach - Combine Media Specific Forward Error Correction and Packet Repetition

sion rate at encoding time. The higher the quality, the lower the compression rate, and vice versa. In redundancy, we encode the original video frames into two versions, one with high quality and one with low quality, as shown in Figure 2 and Figure 3. The high quality frames are sent as primary frames, while the low quality ones are considered secondary frames and piggy-backed with the next primary frame to repair any loss of a primary frame.

Figure 2: High Quality

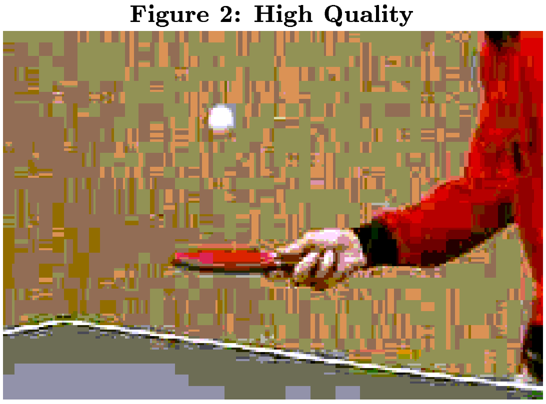

Figure 3: Low Quality

While redundancy may improve perceptual quality for single loss, consecutive losses can still result in a big gap in the video playout. In Figure 1, the 2nd and 3rd frames are lost, so both the original frame 2 and the low quality secondary frame 2 are lost. This gap in the playout stream can be compound if the lost frame has multiple frames encoded based upon it.

We propose a video interleaving approach to further enhance repair from loss. Interleaving assumes that better perceptual quality can be achieved by spreading out bursty packet losses in a media flow. In other words, several small gaps are better than a big gap in a multimedia flow. For example, assume there is a frame consisting of several characters of information:

WorcesterPolytechnicInstitute

Assume that during transmission several characters in the frame get lost:

terPolytechnicInstitute

The first word is then very hard to reconstruct. However, the original frame can be interleaved as:

otlhnuWsocItreynstcrtiteePeci

After applying the same loss to the interleaved frame, the frame then can be reconstructed as:

WrceserPoytecnicIstitute

It is much easier to "interpolate" the missing letters. The same idea has been applied to audio streams as a loss recovery technique [8]. However, it is known that the human visual system is less sensitive than the human auditory system, thus a small gap in a video stream maybe less noticeable than a small gap in an audio stream, suggesting that interleaving may be more effective for video than for audio.

In our research, we design and implement an interleaving strategy for video streams. The sender re-sequences the video stream before transmitting, so that original adjacent units are separated by a guaranteed distance in the transmitted stream, and the receiver returns them to their original order.

In [6], we presented our initial work in video redundancy. In this paper, we revisit our redundancy approach, present our new contribution of video interleaving, and compare the two approaches, in terms of benefits to video quality and system overhead.

To evaluate our work, we design a methodology for measuring the effects of packet loss on video and the possible benefits from our repair techniques from user's perspective. Both of our video repair techniques are applied to raw frames instead of encoded MPEG files, in order to make them compression-independent without worrying about the detail of each compression standard. We starting from some sample MPEG clips recorded from TV, and decompress them to get raw frames. We apply our repair approach on these raw frames, simulate loss and build movie clips, which are then shown to users. Users evaluates each movie clip based on its perceptual quality (PQ). We also analyze the system overhead, including bandwidth and delay caused by redundancy and interleaving.

The rest of this paper is organized as follows. We review some of the related work in section 2. In section 3, we propose our video redundancy and video interleaving approaches respectively, with analysis on system overhead. Section 4 describes our methodology for testing PQ, and discuss the user study results. In section 5, we draw our conclusions and briefly discuss possible future work.

2. RELATED WORK

Although multimedia applications can tolerate a limited amount of data loss, studies show that perceptual quality for packet audio or video traffic without any form of loss repair is degraded at loss rates as low as 1%-5%, and the limits to comprehensibility are reached at around 10% loss [5]. However, experiments have shown that packet drop rates between 7%-15% on the Internet are common, with occasional drop rates as high as 50% [2]. Recovery from losses in a multimedia application becomes a key problem to achieve better multimedia performance on the Internet. In scenarios where loss rates are relatively low (less than 20%), error control or recovery schemes can effectively improve perceptual quality. However, high loss rates limit the effectiveness of theses error recovery schemes. In our evaluation on video repair techniques, we focus on the effects of those techniques at loss rates no higher than 20%.

Most video frames are larger than audio frames, but since audio has similar real-time requirements to video, we build our work upon past research in the repair of audio over the Internet. There are two types of possible audio repair techniques: sender-based and receiver-based as depicted in Figure 4 [9]. Sender-based repair techniques require the addition of repair data from the sender to recover lost packets. Receiver-based repair techniques rely only on the correctly received packets.

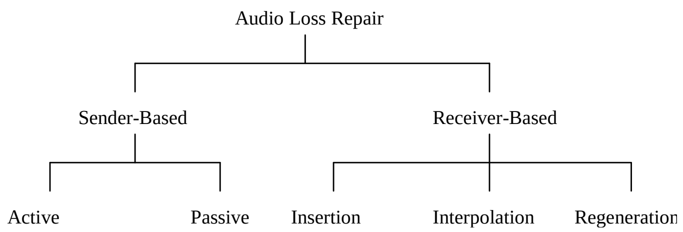

Figure 4: Taxonomy of Audio Loss Repair Techniques

Sender-based repair techniques can be split into two categories: passive channel coding and active retransmission, as depicted in Figure 4. With passive channel coding, the sender sends the repair data, but is not informed of whether or not the loss is repaired. If the loss is not repaired, 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 code to produce additional repair packets from transmission, and media dependent FEC, which uses the knowledge of the contents of the transmitted packets. 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 the loss. Active retransmission techniques can be used when larger end-to-end delay can be tolerated.

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 schemes fail to recover all loss, or when the sender is unable to participate in the recovery, these techniques can be used to make the loss of the packet less noticeable to the user. As shown in Figure 4, there are three kinds of receiver-based data loss repair techniques: insertion based, interpolation based and regeneration based. Insertion based schemes repair losses by inserting a simple fill-in packet. Repetition is one of insertion based schemes, in which lost units are replaced by previous consecutive units, and it performs reasonable well, as compared to other insertion based schemes, while having low computational complexity. Interpolation based repair techniques produce the replacement for a lost packet by interpolating from packets surrounding the loss, and regeneration based schemes use the knowledge of the audio compression algorithm to derive codec parameters, such that audio in a lost packet can be synthesized [9].

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. [4] proposes a method that uses hierarchical coding and scalable syntax. 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 available in ATM networks. In our research, we seek to improve the quality of video streams with the existence of data loss on the Internet. Our research extend the idea of media dependent FEC and interleaving, and apply them to video streams. We also use the receiver-based repetition scheme when lost frames cannot be repaired.

Multimedia streams are compressed before being transmitted over the network. The MPEG (Motion Picture Expert Group) achieves high compression rate by exploiting temporal redundancies of subsequent pictures. MPEG distinguishes 3 main frame types of image coding: I-frame, P-frame, and B-frame. To support fast random access, intra-frame coding is required. I-frame stands for Intra-coded frame which are self-contained, and the compression rate of the I-frames is the lowest. 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 of P-frames are higher than that of I-frames. B-frame stands for Bi-directionally predictive-coded frames. 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. The dependency relationship between the frames is illustrated in Figure 5. The encoding pattern of this stream is IBBPBBPBB.

During transmission of an MPEG video stream over network, one frame lost may mean that other frames in the same Group of Picture (GOP) cannot be decoded, if they are encoded/decoded based on the lost frame, leaving a big

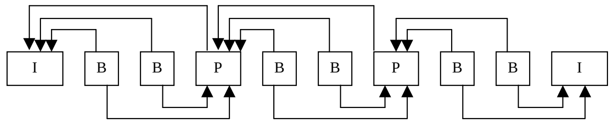

Figure 5: MPEG Frame Dependency Relationship

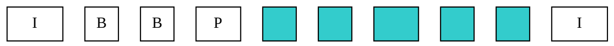

Figure 6: Loss of the Second P-Frame

gap in the video stream. For example, if we use the same GOP pattern in Figure 2.1, the loss of the first P-frame in a GOP will make all the frames after the first I-frame lost, and even the loss of the second P-frame will result in the loss of 5 consecutive frames, as shown in Figure 6. This loss propagation characteristic of the MPEG stream has the same effects as bursty loss, although only one frame is truly lost by the network.

3. VIDEO REPAIR APPROACHES

We designed and implemented two new video repair techniques: redundancy and interleaving. Both techniques had been proposed primarily for audio streams [9]. In our research work, we extend these two approaches to video, and simulate the effects of our techniques on MPEG video streams in the presence of frame loss. The video redundancy approach transmits a small, low-quality redundant frame after each full-quality primary frame. The video interleaving approach re-sequences the video stream before transmitting, so that original adjacent units are separated by a guaranteed distance in the transmitted stream, and returned to their original order at the receiver. Both approaches result in some amount of system overhead. Redundancy adds extra low-quality frames to the video stream, while interleaving changes the inter-frame dependence relationship, and results in larger P or B frames. In this section, we present both approaches and then analyze the system overhead of both approaches, and we further propose a method for decreasing the overhead for these repair techniques, and test it on interleaving.

3.1 Video Redundancy

To help solve the problem of loss, 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 higher compression rate. The high quality frames we call *primary frames* and refer to them as Hi. The low quality frames we call *secondary frames* and refer to them as Li. For each frame i, Hi will be transmitted first and Li will be piggy-backed with Hi+1. At the receiver side, if Hi is received successfully, it will be played to the end user directly and Li will be discarded upon its arrival. If, unfortunately, Hi is lost or totally corrupted during the transmission, the decoder will wait for the next packet. Li will be extracted and take the place of the lost (or corrupted) Hi.

Redundancy can perform well in case of single frame loss. However, it becomes less effective in the case of consecutive frame loss, which can occur either from bursty loss in the network, or because of loss propagation of MPEG video streams. In the situation of consecutive losses, the secondary frame might also not reach the receiver. In such a case, not all the losses can be repaired by using redundancy only. If neither Hi nor Li 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. If the lost frame is an I frame, the frames dependent upon it will also be lost, resulting in a big gap in the playout.

3.2 Video Interleaving

To repair video in the presence of multiple frame losses, we propose another video repair technique called interleaving. The basic idea of interleaving is to spread out one big gap in the media stream into several small gaps, which are separated by guaranteed distance. In this way the effect of the loss of multiple consecutive frames will be ameliorated, and the perceptual quality will be increased.

One important factor in interleaving is the granularity on which the interleaving algorithm should be operating. In a MPEG video stream, the interleaving could be a GOP, or several frames, as in audio interleaving [9]. High compression rates in MPEG are gained by using the inter-frame encoding, the techniques which take advantage of the similarity among consecutive frames. Therefore, a finer granularity of interleaving will result in less overhead. In our video interleaving approach, instead of interleaving based on several frames, we propose a technique to interleave based on one frame. At the sender, frames in a video stream are first interleaved, with the original consecutive frames being separated by a specific distance that is given by the interleaving algorithm. After arriving at the receiver, frames are then reconstructed to their original order. If consecutive loss occurs in the interleaved stream during transmission, or as a result of single loss propagation, after reconstruction at the receiver, a big gap in the stream caused by the consecutive loss or propagated loss will be spread out into several small gaps that are separated by the distance value.

A parameter to interleaving is the distance the smaller gaps are separated by the interleaving scheme. For example, with distance=2, and a GOP pattern of IBBPBBPBB (GOP size=9), the interleaving stream will look like the following sequence, in which a number indicates the position of one frame in a video stream:

1 3 5 7 9 11 13 15 17 2 4 6 8 10 12 14 16 18 ...

And with distance=5, the interleaved frame will look like the following:

1 6 11 16 21 26 31 36 41 2 7 12 17 22 27 32 37 42 ...

In the above sequence, each underscore line indicates a group of 9 frames generated by adding the value of *distance* to the number of a previous frame in the stream. This method is repeated until there are 9 frames in the group. After that, a new group starts and the number of the next frame is then picked from the smallest available frame number. The next frame is obtained by adding the value of distance to the number of the current frame, until the new group fills up with 9 frames. The number "9" is chosen here because it is the number of frames in a GOP. The reason for doing this it that in the interleaved stream, even if all 9 frames in a GOP are lost, in the reconstructed stream, those losses are guaranteed to be non-consecutive and separated by the value of *distance*.

As in our video redundancy approach, our interleaving approach is also combined with repetition error-recovery, in which a lost frame is recovered by repeating the previous consecutive frame. For example, with the GOP pattern IBBPBBPBB, Figure 7 demonstrates whole-interleaving with a distance=2 and the result of the first P-frame lost within a video stream, and Figure 8 shows the situation of the same P-frame lost in the same video stream without interleaving.

3.3 System Overhead

Both redundancy and interleaving add bandwidth and processing overhead to the video stream.

For video redundancy, the secondary frames require extra buffer space, and the bandwidth overhead is approximately 10% that of the original frames. Interleaving may seem to have no bandwidth overhead since it does not add redundant data directly to the video stream. However, by re-sequencing frames in a video stream, the video compression may be less effective (adding more bandwidth) because it separates originally consecutive frames with a larger distance (as shown in Figure 9 with distance=2), and therefore the similarity between the consecutive frames in the interleaved streams becomes less than in the original streams. The key aspect of moving picture compression is the similarity between frames in a sequence. As a general rule, less activity (smaller differences) between consecutive frames leads to better compression and smaller file sizes, and vice versa. In our video interleaving, the larger interleaving distance value will result in a larger overhead and bandwidth, since originally consecutive frames are further separated by the value of *distance*. At interleaving distance=2, we get a bandwidth overhead of about 15%, and as the distance goes higher to 5, the bandwidth increases to about 20%.

We explore a method to decrease overhead of both redundancy and interleaving, yet at the expense of little quality degradation for the encoded video stream by reducing the MPEG quality. The basic premise is that a repaired stream of slightly lower quality but with little overhead will induce less network loss than a repaired stream with higher overhead, and appear better than an unrepaired, if high quality, stream. The MPEG quality, which is the peak signal-to-noise ratio, is defined as:

$$20log_i \frac{255}{\sqrt{MSE}}$$

where MSE is the mean square error. Larger quality numbers give better compression (smaller file size), but worse perceptual quality. This MPEG quality can be tuned by choosing different combinations of I-frame, P-frame and B-frame encoding quality scales. In our video interleaving implementation, we encode the non-interleaved stream with a certain quality Q1, and the interleaved stream with a slightly lower quality Q2, by doing so the bandwidth overhead drops to less than 5%. This method is only effective when the difference between the two qualities Q1 and Q2 are too close to be distinguished by users. Human eyes have the limitation to distinguish between the quality of different video clips if the difference is too small. This is confirmed by our user study on perceptual quality, which will be discussed in next section. The bandwidth overhead of redundancy can be similarly reduced by re-encoding the high quality primary frame at lower quality when redundancy is added to the original stream.

When using redundancy, the sender needs to generate a lower quality frame for each original frame, and attach it to the next high quality frame, while the receiver, in case of frame losses, replaces lost frames with its lower quality version attached in the next frame, or uses repetition when one frame cannot be repaired by redundant frames. Since only one frame delay is needed, the small delay overhead makes redundancy suitable for interactive video applications.

To apply interleaving, delay in the video stream is necessary, and the number of delayed frames depends on the value of interleaving distance. The original frames are resequenced into a different order at the sender, and reordered back at the receiver. Figure 9 shows the process of re-sequencing frames at the sender, with interleaving distance=2 and 9 frames in a GOP. From the figure we can see that totally 2 x 9 frames need to be delayed to complete the resequence process at the sender. At 30 frames per second, this is nearly 500ms of delay. Generally speaking, the amount of delay overhead for interleaving is distance * GOPsize frames. The larger interleaving distance value will result in higher delay. The delay overhead makes interleaving less attractive to interactive video applications, as compared to redundancy.

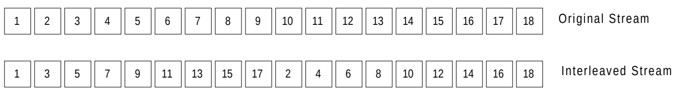

Figure 9: Process of Interleaving frames at the sender (top: original stream, bottom: interleaved stream

Both sender and receiver require additional processing on the video stream, which cause extra delay overhead. The tasks of sender and receiver can be done off-line, if it is not for interactive applications. However, for real-time or interactive video applications, the delay overhead can degrade perceptual quality.

4. EVALUATION

Since it is the end-user who will determine whether a service or application is a success, it is vital to carry out subjective assessment of the multimedia quality afforded by our repair approaches [1]. In our work, we evaluate the effects of our redundancy and interleaving video repair techniques on the quality of video streams through studies in which users evaluate the video quality.

In this section, we first describe the implementation of our

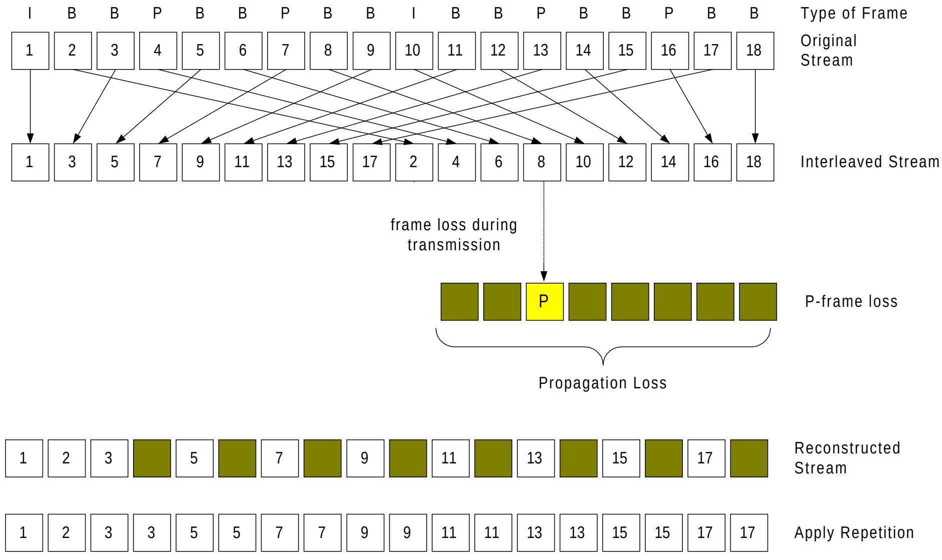

Figure 7: Effects of the first P-frame lost when using interleaving

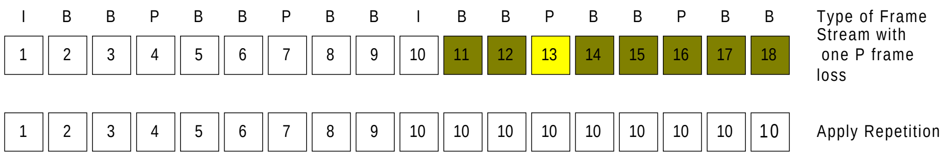

Figure 8: Effects of the first P-frame lost without interleaving

repair techniques, and the process of simulating loss and repair. We then present our methodology to evaluate the effects of video repair by user studies on perceptual quality, followed by the analysis on the results.

4.1 Simulation

The idea of redundancy and interleaving are quite different, but the process of applying the two video repair techniques to video streams are rather similar. A process for applying redundancy and interleaving is shown in Figure 10.

For our system, video clips used in our implementation were recorded from TV and then encoded into MPEG-1 format. The encoding tool we used was Berkely MPEG-1 Video Encoder [7], and the decoding tools we used was Berkeley MPEG-2 player [7] and the Microsoft Media Player. As the process demonstrated in Figure 4.1, we first break the original .mpg file into separate .ppm files using the MPEG decoder, one file for each frame in the video stream. In the next step, we apply our redundancy or interleaving algorithm as appropriate to the .ppm sequences. Then, we encode the modified .ppm sequences using MPEG encoder to generate the .mpg file for transmission over the network. We apply a randomly generated loss rate to the video stream between sender and receiver, simulating a lossy network. After the receiver gets the video stream, the redundancy or interleaving algorithm is again applied to the video stream to recover from any packets lost.

We first implemented video redundancy, and chose three loss rates for examination based on previous Internet traces [3]: 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 an I- or P-frame can leave the frames that are dependant on it useless, which results in an even higher loss rate shown to the end user. For the loss simulation with interleaving, we used similar loss rates of 2%, 5%, 10% and 20%.

4.2 User Study

We did user studies for redundancy and interleaving separately. Totally 42 users participated in the study on redundancy, and 32 for the study on interleaving. We recorded each user's background information, such as the computer familiarity and computer-based video watching frequency. Although the two user sets were different, all users were graduate or undergraduate students with similar backgrounds.

Using the techniques described in section 4, we generated MPEG files for our user studies. Users were required to watch a sequence of mpeg movie clips and evaluate the quality of the movie clip with scores between 0, the worst quality, and 100, the best quality.

The study for redundancy was done on two Alpha(600MHz) machines running Windows NT version 4.0, and the player used was Microsoft Media Player 6.0. The study for interleaving was carried on an Intel PIII (800 MHz) running

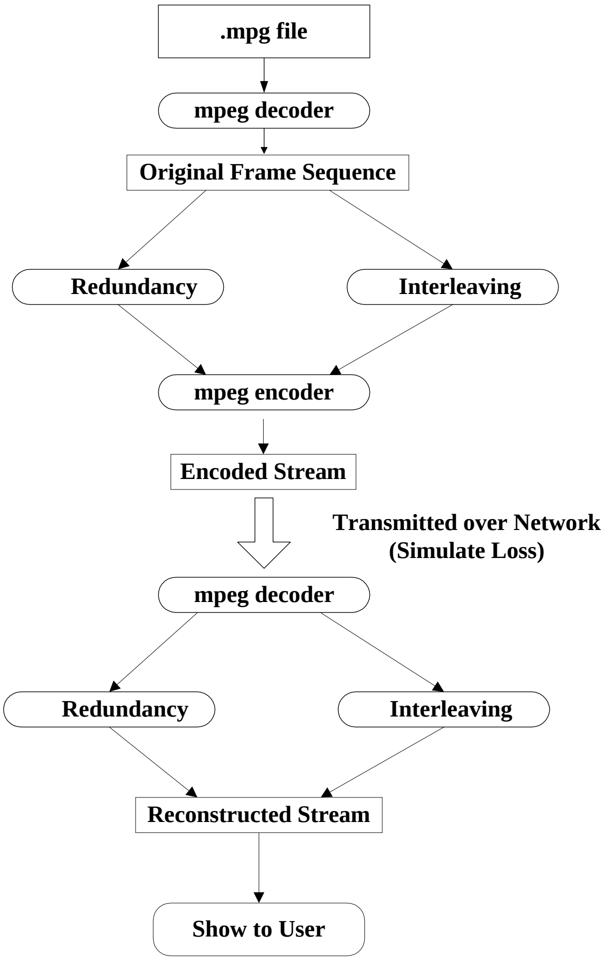

Figure 10: Video Repair

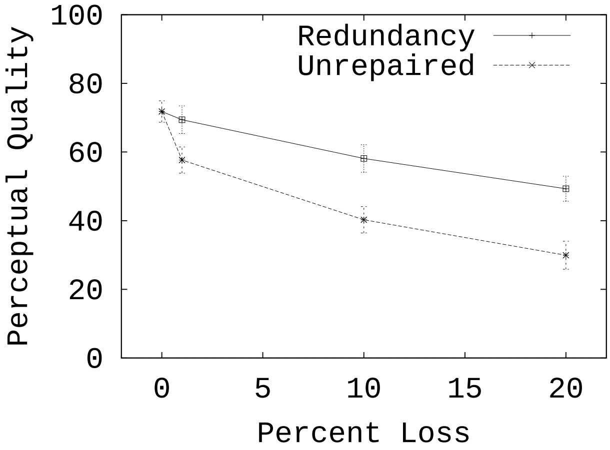

Figure 11: Effects of Loss Rate to Perceptual Quality for Redundancy

SuSE Linux 6.4 i686, and the player used was the Berkeley MPEG-2 player. Both client machines could play MPEG clips at 30 frames per second.

4.3 Results Analysis

After gathering all the scores from the users, we examine the data and analyze the effects of video redundancy and video interleaving on the perceptual quality of video streams. Figure 11 compares the average quality scores for redundancy repaired video clips and unrepaired clips that have no consecutive loss. The x-axis represents percent loss, and the y-axis represents the average perceptual quality score. We also calculated 95% confidence intervals for each data point, depicted with an error bar.

From Figure 11, we can see that 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 improves the quality of the video by 65%. The average score for 0% loss, which is considered as perfect video, is 72. With the increase of 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 videos is 69, which is very close to the score given to the perfect clip. The figure shows that the average point for 1% loss with redundancy repair falls within the range of the confidence interval of the average quality for perfect videos. The difference between the qualities of these two kinds of videos is small and cannot be noticed in some cases. Without the repair technique, the quality of the frame decreases dramatically to 58, which shows a big difference between 0% and 1% loss. Apparently, users can easily notice the seemingly small degradation in quality.

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

Figure 12 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 and 4. The y-axis represents the average perceptual quality. Note that the average quality increases as the number of consecutive losses increases. We believe it is because with a higher consecutive number and the same loss rate, there are fewer gaps within the stream than within the single losses. Thus fewer dependent frames are lost because of the loss of other frames.

However, consecutive loss does make redundancy less useful. As shown in Figure 12, 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 further. For single losses, the redundancy can always be received and thus the loss can always be repaired. With the existence of consecutive

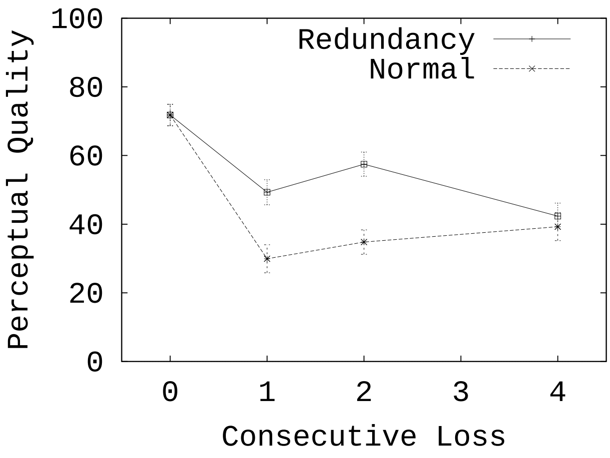

Figure 12: Effects of Loss Pattern to Perceptual Quality

loss, the redundancy can be lost with the primary frame in a sequence. In a sequence of 4-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 video with and without redundancy with 4 consecutive losses are very close. In fact, their confidence intervals overlap, so there is hardly any difference between the average perceptual quality of redundancy repaired and normal video clips and there is no obvious advantage of using the redundancy repair technique when a large amount of bursty loss exists.

As described in Section 3.3 (system overhead), our interleaving approach has about a 15% increase in bandwidth. We propose a method to reduce this overhead by encoding the interleaved stream with a slightly lower MPEG quality. This method decreases the overhead of interleaving to less than 5% at the expense of little quality difference that can hardly be distinguished by end users. To evaluate this method in our user study on interleaving, we encode non-interleaved video clips with relatively higher quality, and interleaved video clips with slightly lower quality. The same method could be used to reduce the overhead of video redundancy but we do not evaluate the impact on quality here.

Figure 13 shows the comparison between two video clips at loss rate of 0% and 10%, in which quality number 1 indicates the relatively higher MPEG quality, and quality number 2 represents the slightly lower quality. The x-axis indicates the loss rate, while the y-axis is the average score for a certain video clip as given by the users. For each loss rate, the left column represents the clip encoded with quality number 1 and the right with quality number 2. The small error bar on each column indicates the 95% confidence interval of the mean value. We can see that the average perceptual quality score of the two MPEG quality number versions are statistically the same, thus trading off bandwidth for quality may be effective.

Figure 14 shows the comparison between interleaving repaired video clips and unrepaired clips that have consecutive loss. At loss rates of 5%, 10% and 20%, interleaving improves the quality of the video clips. At 5% loss, the average score for unrepaired clips is 59, by using interleaving, the average score improved to 76 (about 25%). And this improvement goes even higher to 39% at a loss rate of 10% and 38% at a loss rate of 20%. At a loss rate of 2%, although the average score for the interleaved clips is higher than that of the non-interleaved ones, the 95% confidence intervals for the two values overlap. A larger user study with more participants may be necessary to statistically separate the perceptual quality of the two loss rates. Unlike redundancy, interleaving still performs well at a loss rate of 20% in the presence of consecutive loss. Interleaving repairs packet loss by spreading out big gap in video stream into smaller gaps separated by interleaving distance, thus it is more robust in the case of consecutive loss than redundancy.

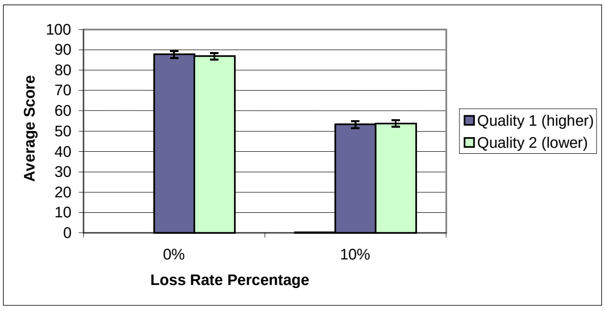

Figure 13: Comparison with Interleaving Quality 1 and Quality 2.

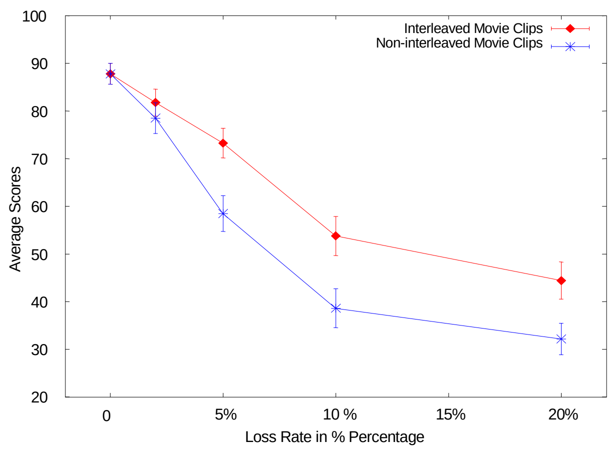

Figure 14: Effects of Loss Rates to Perceptual Quality Using Interleaving

In our user study on interleaving, we also chose two interleaving distances, 2 and 5, to test the effects of the distance value on interleaving. At loss rates of 5% and 20%, movie clips are interleaved using a distance of 2 and a distance of 5. The comparison of those two interleaving distances, together with the non-interleaved clips, is shown in Figure 15. For both loss rates, the unrepaired clips have worse perceptual quality, with interleaving with distance=2 having the highest average score, and interleaving with distance=5 being ranked in the middle. At a loss rate of 5%, the distance=5 interleaving seems only slightly helpful, since the average scores of unrepaired and distance=5 interleaving are almost the same, with the latter one slightly higher. At a loss rate of 20%, the distance=5 interleaving does have some effect on improving perceptual quality, but the distance=2 still performs better. These results show that increasing the interleaving distance value may not be helpful to the performance of the interleaving algorithm. Moreover, increasing the interleaving distance dramatically increases latency, as described in Section 3.3.

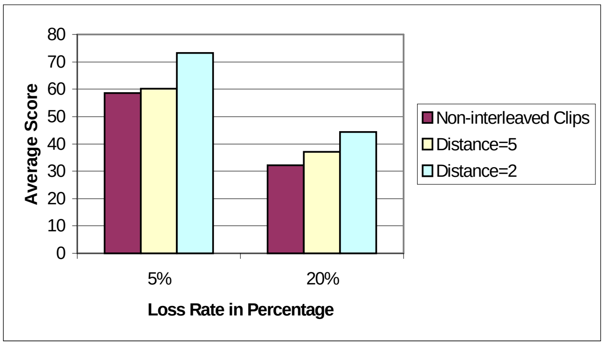

Figure 15: Effects of Interleaving Distance on Perceptual Quality

5. CONCLUSIONS AND FUTURE WORK

In this paper we present video redundancy and interleaving as two repair techniques to ameliorate the effects of network data loss for video. We design a methodology to investigate the benefits of our approaches to users. We also analyze the bandwidth and delay overhead for both approaches, and propose a method to reduce bandwidth overhead and decrease the amount of network traffic by trading off encoded video quality.

Our video redundancy approach piggy-backs redundant video frames within the transmitted video stream in order to repair lost frames. At the sender, frames 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 piggy-backed 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. When a lost frame cannot be repaired by a redundant frame, repetition is used, in which the lost frame is replaced by repeating the previous frame. We find that 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 20% from video with no loss, while the average quality with 1% single loss redundancy repaired video streams is very close to that of perfect video. We conclude that our video redundancy approach completely repairs the video in the presence of single consequent packet loss, the major loss pattern in the Internet today.

While video redundancy may perform well in case of single loss, it becomes less effective when lost frames are consecutive. To address this problem, we propose a video interleaving approach. Our interleaving approach re-sequence units in a video stream at the sender, transmits them through network to the receiver, and reconstruct the video to its original order. In this way, big gaps in a video stream due to consecutive frame losses can be spread out to several small gaps, thus improving the perceptual quality of the video stream. In the situation of random consecutive loss, our interleaving algorithm can improve the perceptual quality by about 25% at a loss rate of 5%, and about 40% at loss rates of 10% and 20%. We also examined the interleaving distance, which is the distance of the small gaps after one big gap is spread out. We find interleaving with distance 2 has much better performance than interleaving with distance 5.

Both video repair approaches add bandwidth and delay overhead. By adding redundant frames to the video stream, redundancy adds about 10% bandwidth overhead. Interleaving changes the similarity relationship between consecutive frames in a video stream, and results in about 15% bandwidth overhead. We propose a approach to decrease this overhead by trading off some quality. For high quality video, the quality difference between two videos is often undistinguishable while the size can vary significantly. This feature is used to decrease the overhead for interleaving by encoding interleaved streams with a quality number slightly lower than non-interleaved stream. The same strategy can be used to decrease the bandwidth overhead of video redundancy. We investigate the effect of this method on interleaving by a user study. As a result, the perceptual qualities of those two videos with slightly different qualities are evaluated as equal by users, suggesting that our method may be effective for some video.

Redundancy requires a delay of only one frame, while interleaving requires a delay of the interleaving distance times the number of frames between I frames. This difference makes redundancy more attractive to interactive or real-time video applications. Thus interleaving is more robust for consecutive loss, while redundancy has low latency which makes it more suitable for real-time video applications.

Future work can combine redundancy and interleaving together, to design a video repair technique that are effective to consecutive loss, as well as suitable for real-time applications. The method of decreasing bandwidth overhead by trading off quality can be utilized to tune the tradeoff between quality and bandwidth overhead based on the current network bandwidth, which can be reflected in the loss rate [10]. When loss rate becomes higher, quality can be slightly decreased to get less bandwidth overhead.

Another area for future study is to compare different video loss recovery techniques, such as redundancy, retransmission and interleaving. Since each technique results in different overhead in delay and bandwidth, the effects of this overhead on the network, for example network congestion, link utilization and goodput, remains uncertain and worth further evaluation.

Redundancy is a media dependent FEC, which used the knowledge of the contents of the transmitted packets. With different types of media, such as audio or other video compression techniques, the strategy may be different. Media independent FEC, which uses block, or algebraic code to produce additional repair packets for transmission, does not require information on the media content, but often requires more processing than media dependent FEC. A performance comparison between these two types of FEC is another area for possible future work.

6. REFERENCES

[1] A. Watson, M. A. Sasse. Measuring Perceived Quality of Speech and Video in Multimedia Conferencing Applications. *In Processing of ACM Multimedia '98, Bristol, UK*, September, 1998.

[2] G. Carle, E. W. Biersack. Survey of Error Recovery Technologies for IP-based Audio-Visual Multicast Applications. *IEEE Network*, 1997.

[3] J. Gerek and W. Buchanan. MMLIB – A Library for End-to-End Simulation of Multimedia over a WAN. Technical Report Major Qualifying Project MQP-MLC-ML98, Worcester Polytechnic Institute, May 1998. Advisor Mark Claypool.

[4] S. Gringeri, G. Khasnabish, A. Lewis, K. Shuaib, R. Egorov, and B. Basch. Transmission of MPEG-2 Video Streams over ATM. *IEEE Multimedia*, 5(1):58 – 71, Jan-Mar 1998.

[5] V. Hardman, M. A. Sasse, M. Handley, and A. Watson. Reliable Audio for Use over the Internet. In *Proceedings of Internet Society's International Networking Conference (INET)*, 1995.

[6] Y. Liu and M. Claypool. Using Redundancy to Repair Video Damaged by Network Data Loss. In *Proceedings of IS&T/SPIE/ACM Multimedia Computing and Networking (MMCN)*, January 25-27 2000. (short paper).

[7] U. of California Berkeley. Berkeley MPEG-2 Encoder and Player. Interent site `http://bmrc.berkeley.edu/frame/research/mpeg/`.

[8] C. Perkins and O. Hodson. Options for Repair of Streaming Media. *The Internet Society RFC 2354*, 1998.

[9] C. Perkins, O. Hodson, and V. Hardman. A Survey of Packet-Loss Recovery Techniques for Streaming Audio. *IEEE Network Magazine*, Sep/Oct 1998.

[10] S. Floyd, M. Handley, J. Padhye and J. Widmer. Equation-Based Congestion Control for Unicast Applications. In *Proceedings of ACM SIGCOMM Conference*, pages 45–58, 2000.

[11] V. Paxson. End-to-End Internet Packet Dynamics. *IEEE/ACM Transactions on Networking*, Fall 1999.