Mosaicing videos to stream over multiple independent channels

Chris Boehnen, Allison Regier, Deborah Thomas, Surendar Chandra and Patrick Flynn
Department of Computer Science and Engineering, University of Notre Dame
Notre Dame, IN 46556, USA
{cboehnen,aregier,dthomas,surendar,flynn}@nd.edu

ABSTRACT

Streaming high fidelity multimedia objects requires large amounts of network bandwidth resources. Sometimes these resources are achieved by aggregating a number of independent and lower capacity network channels. Network level aggregation schemes can stream the single video across all the network links. However, splitting multi-layer encoded video streams are not resilient to network failures on individual links because enhancement layers are not independent and depend on the availability of base layers. In this paper, we investigate several multiple description coding mechanisms that split the stream into multiple independent sub-streams. Our mechanisms attempt to retain the spatial and temporal redundancy inherent in the original stream in order to achieve good compression efficiency. We examine the impact of our approach on changes in peak transmission requirements, overall transmission size and stream quality. We show that the sub-streams are able to sustain substantial data loss while still providing a viewable stream. We also show the object size overhead for the various mechanisms.

Keywords

Multiple Description Coding

1. INTRODUCTION

This work was motivated by application scenarios that operate on high fidelity video streams. The resource requirements of these streams can overwhelm a single communication link. Sometimes, it is preferable to use multiple network links to transmit the stream. For example, remote tele-medicine systems such as Tavarua [7] utilized multiple cellular links to transmit the video streams.

One way to transmit the stream is to use a network level aggregation mechanism that transmits across the multiple links. However, video stream contents are not independent and depend on other parts of the stream. For example, multi-layer encodings depend on the availability of base layers to successfully decode the enhancement layers while MPEG streams require preceding I frames in order to decode subsequent P and B frames. Losing packets on any one of these links can have a catastrophic effect. Though some systems can adapt to lossy networks by reducing the future stream fidelity, transmissions during the actual loss on any of the links can seriously degrade the entire stream. Retransmitting these lost packets on other channels adds latency. In order to stream the media through these independent links, we desire multiple description coding mechanisms (MDC) that can split the original stream into multiple streams, each of which can be adapted to the capacity of the individual channel. We also prefer mechanisms that can continue to operate with the remaining links, albeit at proportionally lower quality levels. The ability to operate through link failures requires MDC mechanisms that create independent set of streams.

In this paper, we explore various ways of splitting a single stream into multiple independent streams. Each of the independent streams can be encoded and transmitted separately. Ideally, each of the sub-streams should add little overall compression overhead. However, modern encoding schemes such as MPEG achieves high compression ratios by exploiting the spatial and temporal redundancy inherent in videos. Unless care is taken in the splitting process, the new sub-streams may have less spatial and temporal redundancy as compared to the original stream, leading to a decrease in compression efficiency (and a corresponding increase in stream size).

In this paper, we explore schemes that retain some of the spatial and temporal redundancy present in the original image. We describe a scheme that maintains spatial redundancy by choosing nearby pixels for the various streams, a scheme that maintains temporal redundancy by choosing nearby frames and a quadrant based approach which maintains relative spatial and temporal redundancy of a smaller portion of the original image. Our experiments show that sub-streams are able to achieve good error resiliency; the received quality proportionally depends on the number of streams that are successfully received.

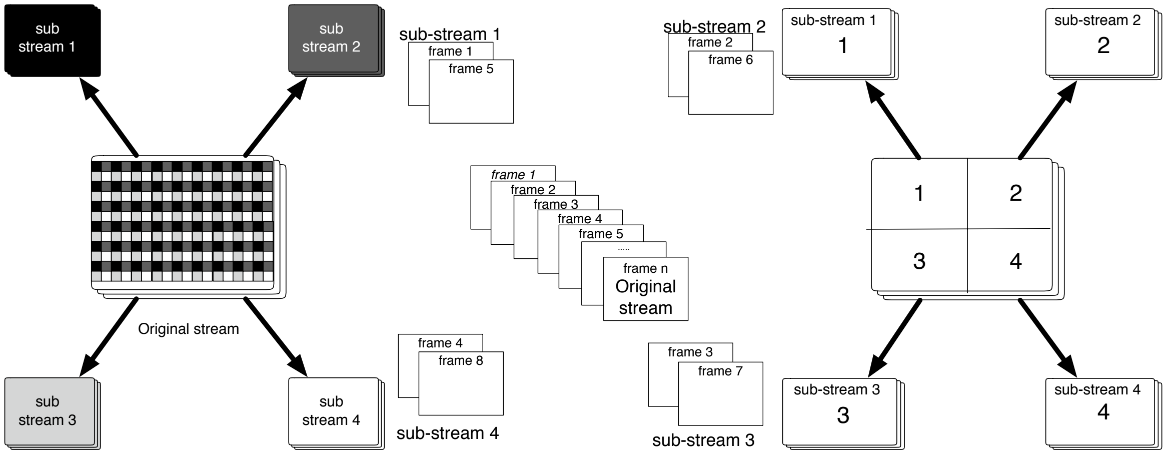
2. OUR APPROACH

2.1 Objectives

Our goal is to split a stream S to k independent streams $s_1, s_2 \ldots s_k$. Though k can be arbitrarily large, we restrict ourselves to analyzing the behavior of the system using four sub-streams. Ideally, each channel might sport different capacities. For simplicity, we assume that all the channels have equal capacity. Suppose the compressed object sizes of the original and the various sub-streams are N and $n_1, n_2 \ldots n_k$, respectively. We prefer schemes that are:

1. *proportional quality:* quality of the output stream is proportional to the number of sub-streams that were successfully received. We used PSNR to measure stream quality.
2. *space efficient:* $N \leq \sum_{i=1}^{k} n_i$
3. *fair size:* equal space requirements for the various sub-streams: $n_1 == n_2 == n_3 == n_4$

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
NOSSDAV '07 Urbana, Illinois USA
Copyright 2007 ACM 978-1-59593-746-9/06/2007 ...$5.00.



(a) Spatial separation (b) Temporal separation (c) Quadrant-based

Figure 1: Our approach to split a single stream into four sub-streams

We focus our attention on MPEG-4 encoded streams. MPEG-4 compression exploits the spatial redundancy using intra-coded frames (I frames) and temporal redundancy using inter-coded frames (P and B frames). The choice of the number of I, P and B frames depend on the motion characteristics of the particular stream. Streams with high temporal redundancy can achieve high compression ratios using few I frames. However, the loss of intra coded 'I' frames are catastrophic for successfully decoding subsequent P and B frames. Hence, streamed videos incorporate additional 'I' frames in order to achieve good resiliency. We explore mechanisms that create sub-streams that retain some of the spatial and temporal redundancy found in the original stream. Next we outline our proposed mechanisms (Figure 1).

2.1.1 Spatial Separation

The spatial separation mechanism (Figure 1(a)) retains much of the spatial redundancy in the original stream. Each of the four pixels in every 2x2 pixel block of each frame is assigned to a separate stream. Nearby pixels are expected to continue to retain any spatial correlation from the original image (every other pixel in the original stream becomes neighbors in the sub-stream). The sub-streams are spatially reduced by a factor of two on both the dimensions. One advantage of this approach is the easier opportunity for error correction; lost pixels from one or more sub-streams can be estimated through linear interpolation of the neighboring pixels from successfully received streams.

2.1.2 Temporal separation

The next approach (Figure 1(b)) maintains the temporal redundancy in the original stream by assigning each of four consecutive frames (starting from the first frame) to the different sub-streams. The resulting sub-streams retain the spatial dimensions of the original stream while the frame rates are reduced by a factor of four. The I-frames of the resulting compressed sub-streams can be expected to retain their size in the original stream. Retaining the same inter I-frame distance as the original stream has the effect of effectively quadrapuling the inter I-frame distance on the original stream. Chakareski [2] used a similar scheme for their analysis.

2.1.3 Quadrant-based approach

The last approach (Figure 1(c)) is a hybrid that retains the temporal and spatial redundancy of the original stream by assigning the four quadrants of each frame of the original to a sub-stream. The resulting sub-streams were spatially scaled by half on each dimension as the original stream while still retaining the original frame rates. If one of the sub-stream is lost, then a whole quadrant of the original stream will be lost. Depending on the temporal characteristics of the original image, each of the sub-streams might retain differing amounts of temporal redundancy. For example, a sub-stream of a newscast might show high motion in the quadrant where it shows a news clip inlay while exhibiting little motion in other sub-streams. Qureshi used a similar scheme in Tavarua [7].

3. EXPERIMENT SETUP

Next, we describe the experiment setup: streams used, evaluation metrics and experimental setup. We discuss our experimental observations in the next section.

3.1 Video clips used

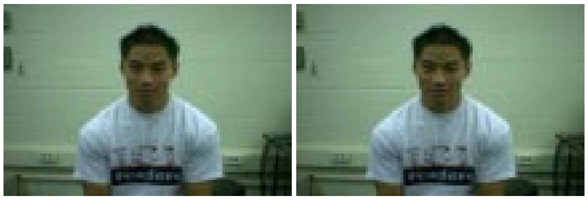
Our dataset consisted of six video clips.

- *NDSet:* Three clips were acquired through the Computer Vision Research Lab[1] data acquisition process. The clips were captured using an iSight webcam (640x480), Canon camcorder (720x480) and JVC HD camera (1280x720). These clips consisted of a subject sitting in a chair and uttering an unique phrase. There was little spatial movement in the clip. These clips were representative of application scenarios such as tele-medicine and video conferencing wherein the foreground subject might be talking and gesturing against a relatively static background.

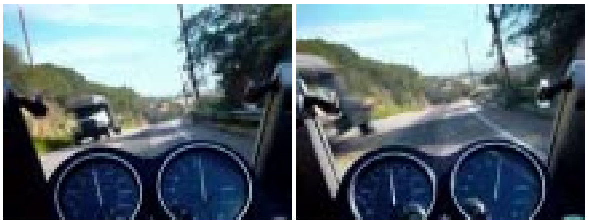
- *MotorCycleSet:* The other three video clips were downloaded from Motorcycle Online[2]. These clips (320x240) show a motorcycle racing sequence. As the camera was rapidly panned to follow the motorcycle these clips exhibited high motion with the background changing constantly.

Figure 2 shows two frames that were captured five frames apart from each of the sequences used in our experiments. The MotorCycleSet frames show high motion as the motorcycle moves at a high speed along the highway.

[1] `www.nd.edu/~cvrl/UNDBiometricsDatabase.html`
[2] `www.motorcycle.com/mo/mcvideos/videos.html`



(a) NDSet



(b) MotorCycleSet

Figure 2: Illustrative frames that were five frames apart

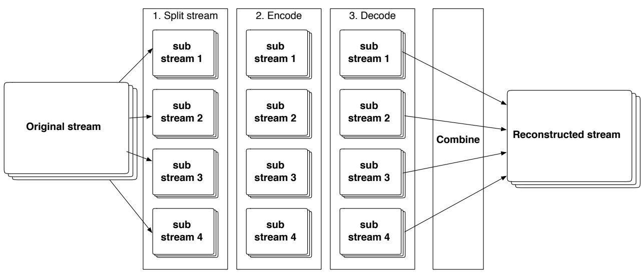


Figure 3: Evaluation architecture

3.2 Evaluation architecture

For our experiments (illustrated in Figure 3), we split the video clips into four sub-streams, compressed each of the sub-streams independently, decompressed them and then combined them to reconstruct the original stream. We analyzed the total size of the sub-streams as well as the quality of the reconstructed stream. We experimented with five different encoders that were included with the *ffmpeg* [1] software package. We used FFmpeg/ffdshow ISO MPEG-4 (FMP4), ffmpeg 1/2 (MPG2), Flash Sorenson Video (FLV1), Motion JPEG (MJPG), and Sorenson v1 (SVQ1). We specified the target bitrate for the different streams. We observed that the MPEG4 encoder consistently achieved higher compression efficiencies; both for the original stream as well as for the sub-streams. We report our experiences with MPEG-4 encoding in this paper. We forced a staggered start for the various sub-streams (similar to Qureshi et. al. [7]). We experimentally varied the number of frames between I-frames from 3 to 90 frames and plotted the PSNR and compression efficiency in Figure 4. We observed that the streams exhibited high compression efficiency and PSNR for choosing values over 12 frames. Hence, we chose a fixed inter I-frame distance of 12 frames for the rest of the experiments.

3.3 Evaluation Metrics

We used three performance metrics to evaluate our approach. First, we examined the impact of file size and PSNR for transmitting using the sub-streams. We also analyzed the impact of data loss during transmission. In order to have repeatable performance, we manually corrupted some compressed data from the various sub-streams before decompressing and attempting to combine the various sub-streams and recreate the full stream. For our experiments, we simulated data loss of the I-frames. This represents the worst case behavior in terms of data loss; losing P or B frames are expected to be much less disruptive. We determined the file offsets of the various I-Frames and replaced the first 1500 bytes (typical MTU size) with zeros in order to simulate data loss during transmission. We varied the number of lost I-frames among the sub-streams. The modified MPEG4 files were decoded by ffmpeg (using its error compensation mechanisms) before the streams were combined to recreate the original stream. Missing data in our spatial separation mechanism was recreated by using a linear intrapolation of surrounding pixels. We also examined the peak transmission requirements of the various streams over time. The magnitude of the peaks represents the burstiness of the different methods.

4. RESULTS

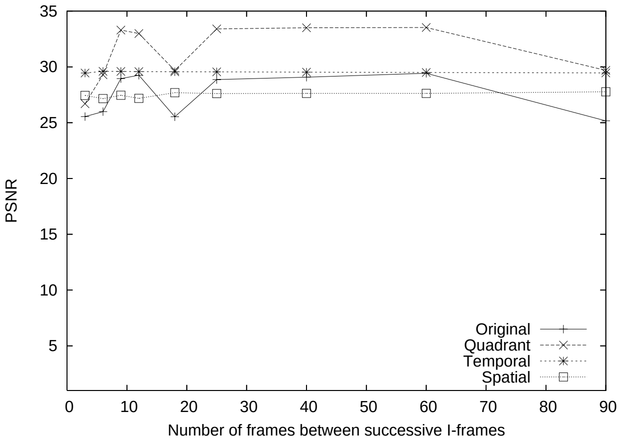
4.1 Resiliency to loss on individual links

The primary motivation behind our sub-stream mechanism was to achieve resiliency for loss over any of the network links. The original MPEG-4 stream has relationship among the various I, B and P frames and so could not independently recover from loss on the different links. For our experiments, we modified the first 1500 bytes (typical MTU size) of the I-frames to zero. Distorting the I-frames has a significant effect on the stream quality because the error propagates to subsequent P and B frames. We performed experiments distorting all or the first half of the I-frames. For the sub-streams, we repeated experiments distorting one, two, three or all the sub-streams. We allowed the decoders to recover from the distortion. For the spatial separation method, we also used our linear interpolation to recover from lost sub-streams. We plot the PSNR values for losing various number of sub-streams among the NDSet and MotorCycleSet in Figure 5. Ideally, we prefer PSNR values that degrade gracefully with increasing data loss.

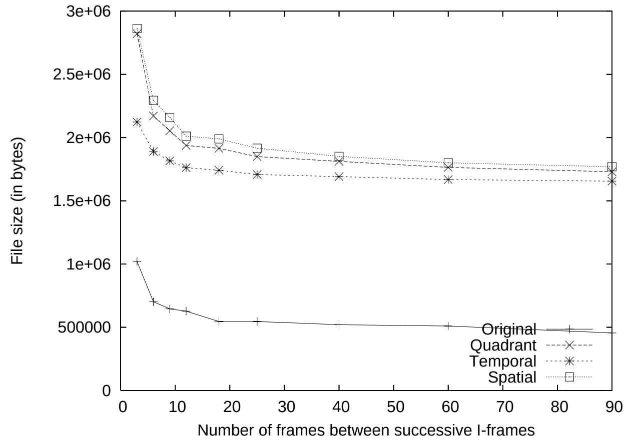
At a high level, we note that the sub-streams achieve graceful degradation of quality with increasing data loss. For the spatial and the quadrant-based mechanisms, the sub-streams have the same frame rate as the original; distorting each I-frame corresponds to four times the data loss (6000 bytes vs 1500 bytes per I-frame). Still, even with I-frame loss in all the links for half the I-frames, the spatial and quadrant based mechanisms are competitive for NDset. Similarly, distortion in all I-frames achieves the same quality loss for NDSet and MotorCycleSet. Our linear interpolation significantly improved the PSNR for both the data sets. The temporal separation faced similar data loss as the original. These results were sensitive to compression parameters that are discussed in further detail in Section 4.2.

4.2 Sub-stream characteristics

The previous section showed that our approach can create independent streams that gracefully lost quality on data loss on any of the network links. MPEG-4 streams offer many configurable parameters; our goals are to understand the specific range of parameters. For example, the original stream was compressed for a target of 175 kbps. For our experiments, for each of the sub-streams, we configured the encoder to target streams of bandwidth requirements of 5, 10, 25, 50, 75, 100, 125, 150 and 175 kbps. For each of these streams, we plot the total object size (sum of the four sub-streams) against the stream PSNR (by comparing with the original stream against the recreated stream). We plot the data for NDSet using a DV camera (720x480), high definition camera (1280x720) as well as the MotorCycleSet (320x240) in Figure 6. From Fig-

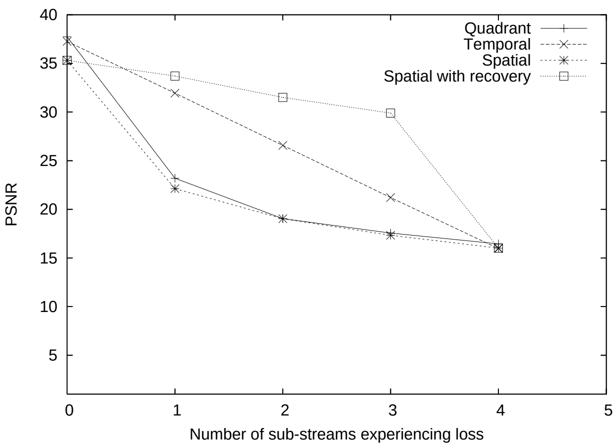


(a) PSNR

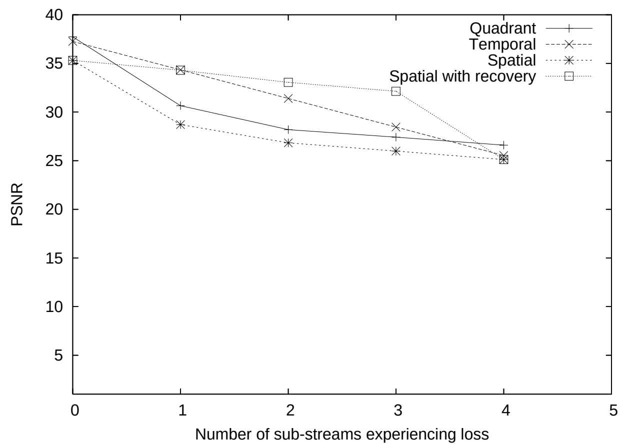


(b) Video file size

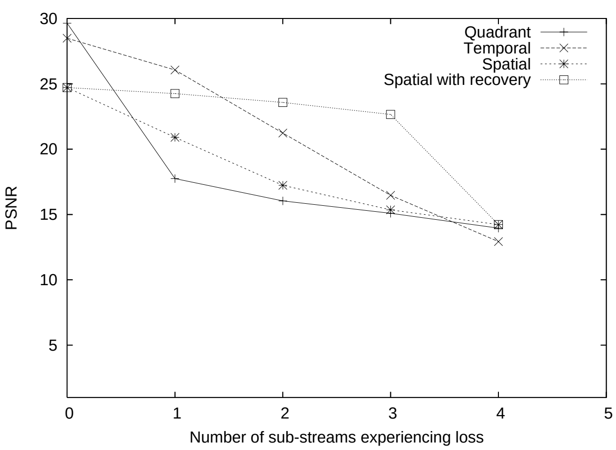
Figure 4: Effects of varying the inter I-frame distance



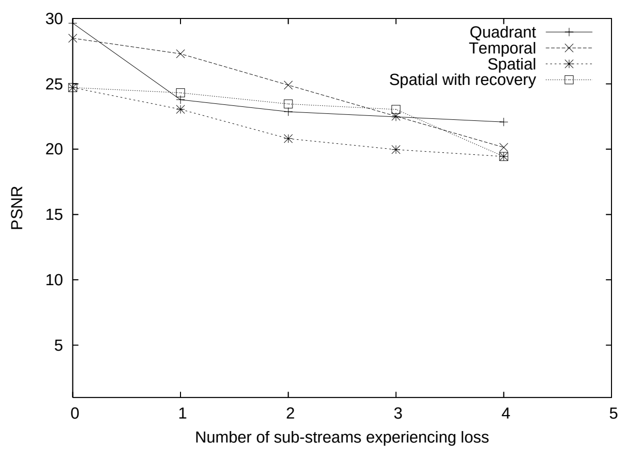
(a) NDSet: distortion of all I-frames. For comparison, PSNR of original: no distortion = 38.3, distortion of all I-frames = 16.3



(b) NDSet: distortion of half the I-frames. For comparison, PSNR of original: no distortion = 38.3, distortion of half the I-frames = 26.5

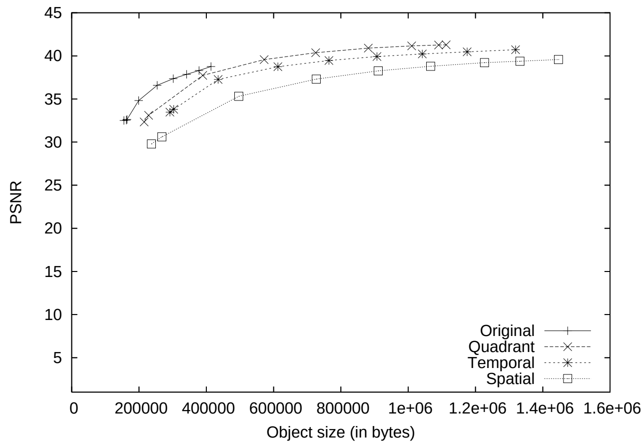


(c) MotorCycleSet: distortion of all I-frames. For comparison, PSNR of original: no distortion = 31.1, distortion of all I-frames = 12.9

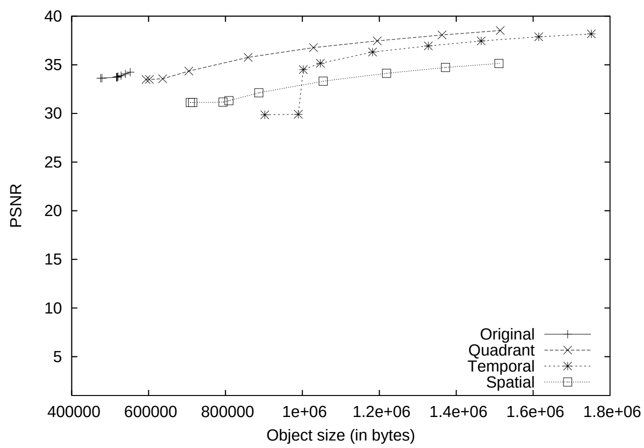


(d) MotorCycleSet: distortion of half the I-frames. For comparison, PSNR of original: no distortion = 31.1, distorting half I-frames = 22.4

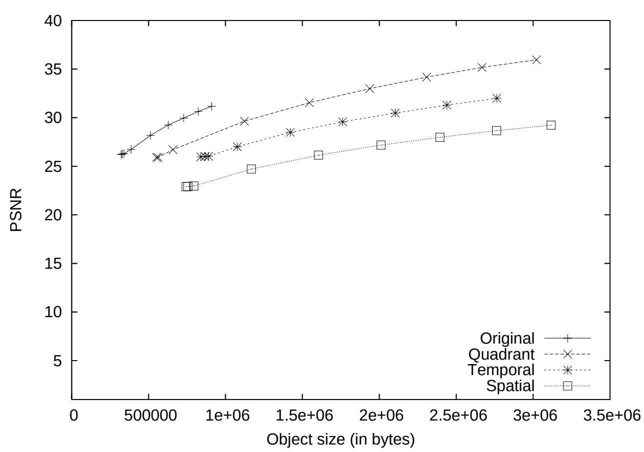
Figure 5: Distortion of I-frame data. PSNR calculated by comparing the original stream against the recreated stream. Note that the initial PSNR of the streams was not the same due to differences in encoding parameters. Hence, the different plots may not be directly compared, even though the relative differences within a separation mechanism are significant



(a) NDSet DV camera

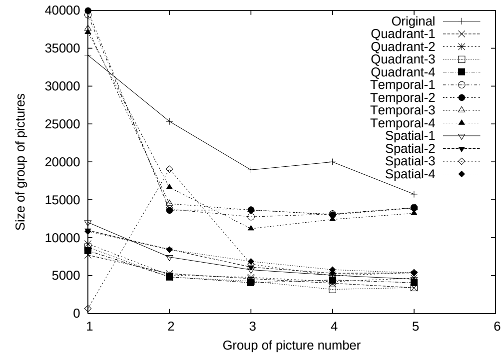


(b) NDSet high definition camera

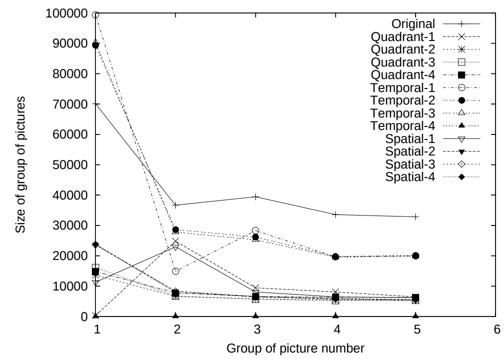


(c) MotorCycleSet

Figure 6: Combined size and PSNR characteristics of split sub-streams



(a) NDSet



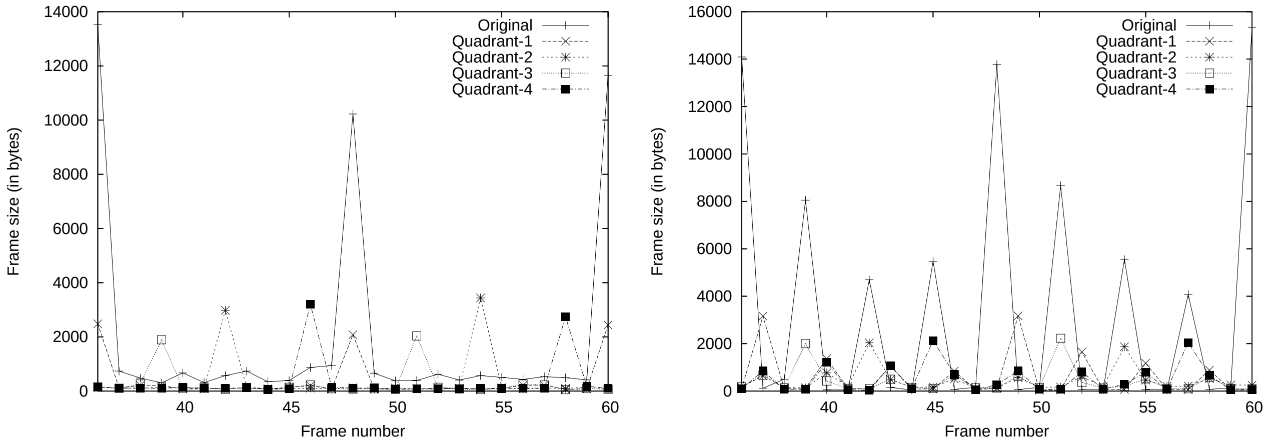
(b) MotorCycleSet

Figure 7: Stream size between successive I-frames

ure 6, we observe that the sub-stream mechanism can add significant overhead, especially when creating a higher quality stream. For example, for the NDSet stream that uses a DV camera (Figure 6(a)), the original, quadrant, temporal and spatial streams consumed about 171 kB, 210 kB, 250 kB and 395 kB of total size to achieve a PSNR value of 33.1, respectively. We noticed similar overhead across all our streams and mechanisms. Sub-streams appear less efficient from a compression perspective. In order to further understand the different sub-stream mechanisms, we plot the total stream size between successive I-frames (GOP size). We plotted the values for the original stream as well as the various individual sub-streams for the NDSet and MotorCycleSet in Figure 7. Ideally, we prefer values for the sub-streams which are a quarter of the original stream. As can be seen from Figure 7, the GOP size of the individual stream depended on the actual stream. Sometimes, the sub-stream GOP size is even larger than the original stream. Future work will investigate mechanisms to predict this behavior so that automatic choices can be developed to choose the appropriate sub-stream mechanism. Also, in Section 4.1, we chose a encoding target rate of 150 kbps for the original stream and 25 kbps for the sub-streams in the NDSet and 175 kbps for original, 50 kbps for spatial and quadrant and 75 kbps for temporal for sub-streams in the MotorCycleSet.

4.3 Peak stream requirement

One of the benefits of splitting the streams is that the sub-streams are expected to have smaller I-frame sizes, especially for the quadrant and spatial separation schemes; temporal separation mechanisms retains the original spatial dimensions and hence can be expected to have similar I-frame size. Systems such as [7] specifically used stream splitting for this purpose. We plot the peak stream



(a) NDSet (b) MotorCycleSet

Figure 8: Peak frame size analysis

requirements for the NDSet and MotorCycleSet in Figure 8. We observed that I-frames of the sub-streams were relatively smaller. This smoothing effect can also be preferable in scenarios where a stream is split and transmitted over a single link.

5. RELATED WORK

Goyal [5] describe various MDC mechanisms for image, audio and video objects. Chakareski [2] used a MDC scheme that created two descriptions by choosing alternate frames for the sub-stream. Qureshi et. al. [7] used quadrant separation mechanism for streaming H.264 encoded streams over multiple cellular links. Kim et. al. [6] describe a unbalanced MDC mechanism that provided unequal error protection. In general, earlier work addressed the issue of streaming MPEG over IP networks. Feng et. al. [3, 4] and Rexford et. al. [8] analyzed the streaming behavior and investigated buffer management mechanisms to smooth the effects of streaming MPEG objects. Wang et. al. [9] used reference frames to enhance the streaming performance. Recently, Xu et. al. [10, 11] investigated mechanisms to adapt a multi-layer encoding for transmission across multiple independent channels. Our work adds to these systems by investigating MDC mechanisms to stream MPEG-4 video over multiple links.

6. CONCLUSIONS

Using multiple independent and lower capacity links to achieve higher performance is becoming popular. We investigated application level mechanisms to stream MPEG-4 streams over these links. Our experimental analysis showed that our sub-streams lose quality gracefully with a corresponding cost in increase in total transmission requirements. The key challenge is to choose the specific compression parameters and the separation mechanism based on the expected object motion characteristics.

Acknowledgments

This was was supported in part by the U.S. National Science Foundation (CNS-0130839 and CNS-0447671). Regier was supported by a Lilly fellowship.

7. REFERENCES

[1] BELLARD F. Ffmpeg multimedia system. http://ffmpeg.sourceforge.net/index.php.

[2] CHAKARESKI, J., HAN, S., AND GIROD, B. Layered coding vs. multiple descriptions for video streaming over multiple paths. In *Proceedings of the eleventh ACM international conference on Multimedia (MULTIMEDIA '03)* (New York, NY, USA, Nov. 2003), ACM Press, pp. 422–431.

[3] FENG, W., JAHANIAN, F., AND SECHREST, S. An optimal bandwidth allocation strategy for the delivery of compressed prerecorded video. *Multimedia Syst. 5*, 5 (1997), 297–309.

[4] FENG, W., AND REXFORD, J. A comparison of bandwidth smoothing techniques for the transmission of prerecorded compressed video. *infocom* (1997).

[5] GOYAL, V. K. Multiple description coding: compression meets the network. *IEEE Signal Processing Magazine 18*, 5 (Sept. 2001), 74–93.

[6] KIM, J., MERSEREAU, R., AND ALTUNBASAK, Y. Distributed video streaming using multiple description coding and unequal error protection. *IEEE Transactions on Image Processing 14*, 7 (July 2005), 849–861.

[7] QURESHI, A., CARLISLE, J., AND GUTTAG, J. Tavarua: video streaming with wwan striping. In *MULTIMEDIA '06: Proceedings of the 14th annual ACM international conference on Multimedia* (New York, NY, USA, 2006), ACM Press, pp. 327–336.

[8] REXFORD, J., AND TOWSLEY, D. Smoothing variable-bit-rate video in an internetwork. *IEEE/ACM Trans. Netw. 7*, 2 (1999), 202–215.

[9] WANG, Y., CLAYPOOL, M., AND KINICKI, R. Impact of reference distance for motion compensation prediction on video quality. In *SPIE Multimedia Computing and Networking (MMCN)* (San Jose, CA, Jan. 2007).

[10] XU, W., AND HEMAMI, S. S. Efficient partitioning of unequal error protected mpeg video streams for multiple channel transmission. In *International Conference on Image Processing* (2002), vol. 2, pp. II–721–II–724.

[11] XU, W., AND HEMAMI, S. S. Spectrally Efficient Partitioning of MPEG Video Streams for Robust Transmission Over Multiple Channels. In *International Workshop on Packet Video* (Pittsburg, PA, Apr. 2002).