Segmentation and Compression of Video for Delay-flow Multimedia Networks

Yuan-Chi Chang and David G. Messerschmitt

Department of Electrical Engineering and Computer Sciences
University of California at Berkeley
Berkeley, California 94720, USA

ABSTRACT

Digital video coding has traditionally used frame-by-frame synchronous reconstruction. The transport must then be delay-jitter-free, forcing the modern integrated service packet network such as the Internet to operate in an inefficient "circuit emulation" mode. This mode results in a jitter-free delay representative of the worst-case network delay, which is problematic for delay-sensitive interactive applications. In response, we have proposed and demonstrated a "delay cognizant" model of video coding (DCVC) that operates in an asynchronous reconstruction mode. DCVC minimizes the perceptual delay observed by the user, and still achieves good quality and high compression. Furthermore, the feasibility of asynchronous reconstruction is evidenced by vision science studies of spatiotemporal masking in human visual systems at temporal edges of video.

1. INTRODUCTION

With few exceptions, most work in digital video coding and processing has assumed frame-by-frame synchronous reconstruction and display. This model implicitly implies the "channel", where the video signal is carried, such as storage media and network transport, must function like a circuit, with a fixed delay. However, with the advance of integrated service packet networks, the transport layer no longer operates efficiently in the "circuit emulation" mode. This mode requires the network to smooth delay jitter by adding artificial delay, which in turn forces the total delay seen by the video to be the worst case network delay. For interactive video applications such as video conferencing and games, excess delay destroys lip synchronization unless the audio is similarly delayed, which in turn disrupts normal conversation. Thus, we believe synchronous video and integrated service packet networking are fundamentally a mismatch. A new model for video coding, one which matches well to the characteristics of packet networking, is delay-cognizant video coding (DCVC) [1][2].

Delay and transport efficiency is but one aspect of the system design addressed in [3], but the only one addressed in this paper. The reader is reminded that tradeoffs between delay, reliability and bandwidth usage in the context of joint source-channel coding also interact with system level issues of scalability, modularity, security and multicast. A variable Quality-of-Service (QoS) transport flow architecture was proposed in [4] to lay out the basic building blocks and interfaces of the next generation networks. In this architecture, traffic sources generate multiple flows (the term "substreams" was used in [3][4]) with different QoS requirements. Networks maximize their operating efficiency by provisioning no better QoS to each flow than requested. Packets in a flow receive the same QoS, while different flows used by the same application have different QoS attributes. Although QoS guaranteed network connections are today only experimental, the next generation Internet Protocol (IP) has incorporated a 28-bit flow label into its packet header structure to support future deployment [5].

In DCVC, a video source stream is segmented into flows with different delay properties and is reconstructed asynchronously at the receiver. Although the number of flows is arbitrary, in the following we will assume that there are only two flows, the low- and high-delay flows. The most visually significant information is carried by the low-delay flow. Less visually significant information, carried by the high-delay flow, attempts to improve the quality without introducing temporal-discontinuity artifacts. DCVC can be thought of as progressive transmission of images adapted to video.

The term DCVC is adopted since the video coder is cognizant of the delay properties of the flows. It is analogous to "error resilient video coding". The advantages of DCVC over synchronous video include an improvement in the tradeoff between traffic efficiency and perceptual delay. Given the same delay bound for the most visually significant information, a flow for the synchronous model has limited flexibility to shape its traffic and therefore the burstiness remains high. The low-delay flow of DCVC is also bursty but has a reduced bandwidth. On the other hand, the packets in the high-delay flow of DCVC can be scheduled and transmitted at the most opportune moment because of their relaxed delay bound, increasing traffic capacity through traffic smoothing. Similar arguments apply to the context of fixed network resource usage, as DCVC can minimize perceptual delay by giving priority to the low-delay flow, which carries the most visually significant information. Finally, the video coder can choose to negotiate with the network to achieve the optimal balance between delay and efficiency (cost).

This paper describes the design of a DCVC algorithm that achieves the above goals. Segmentation and compression of the coded-video flows are our primary focus. While our earlier work

The authors can be reached at http://www.eecs.berkeley.edu/~messer or messer@eecs.berkeley.edu. This research was supported by the University of California MICRO program, grants from Harris, Rockwell, LG Electronics, National, Plessey, Philips, and an IBM Fellowship.

claimed success in segmenting the video with minimal degradation, its compression performance was less competitive. The algorithm described in this paper takes a different approach to segmentation in order to accommodate aggressive compression. We report the segmentation criteria and the codec architecture. Lastly we discuss its compression performance and the evaluation of output video quality.

2. VIDEO SEGMENTATION FOR DIFFERENTIAL DELAY FLOWS

The segmentation and compression stages of the DCVC coder divides the video-coded information into low- and high-delay flows while attempting to achieve two objectives simultaneously:

- Minimize total traffic, while maximizing that portion in the high-delay flow and minimizing that portion in the low-delay flow.
- Maximize the allowable delay offset (the difference between the maximum allowed delay of the high-delay flow and that of the low-delay flow).

In our earlier work, a number of different segmentation criteria were tried [1][2], but they often failed to achieve acceptable compression ratios. If the compressed traffic in the low-delay flow is larger than the compressed traffic in a conventional single-flow video stream, the benefits of DCVC are diminished. One approach we tried was pixel-based segmentation by conditional replenishment. In segmenting head-and-shoulder scenes, less than 5% of the total pixels are carried by the low-delay flow, and the delay offset can be as great as 330 msec without incurring noticeable quality degradation in our experiments. These results demonstrate that the human visual system (HVS) has no difficulty accepting asynchronously reconstructed video. However, there is additional coding overhead in communicating the addresses of the low-delay pixels to the decoder, and the traffic generated by this information has proved to be great . To reduce this overhead, the segmentation granularity is enlarged from pixels to blocks in this paper.

The decision criterion we use is conditional block replenishment. The block diagram of the segmentation stage is shown in Figure 1. Each video source frame is first divided into 8 by 8 pixel blocks . For each block, its discrete cosine transform (DCT) is computed to obtain the frequency coefficients. Each coefficient is tested against the following two conditions:

$$|P_{i,j,n,t} - P_{i,j,n,t-1}| < V_{i,j} \text{ and } |P_{i,j,n,t} - P_{i,j,n,update}| < V_{i,j}$$

where $P_{i,j,n,t}$ is the (i, j)th coefficient for block n in frame t; $V_{i,j}$ is a fixed preset threshold for the (i, j)th coefficient; $P_{i,j,n,update}$ is the value from the last update block. If not all coefficients satisfy both conditions, which means the block has changed significantly, this block is declared a low-delay block, causing it to be fed into the low-delay loop. The segmentation stage extracts the low-delay blocks and puts them in an image plane. These frequency domain blocks are then inverse transformed back to the pixel domain. The high-delay information is the difference between the original image and the image plane formed by low-delay blocks. An example in Figure 2 is given to show the original image, the low-delay image plane and the high-delay image plane.

In the above test conditions, the percentage of blocks sent in the low-delay flow depends on the temporal activities of the video as well as the threshold preset. Since HVS is less sensitive to high spatial frequency distortion, the corresponding threshold should be set higher. Following this guideline, we designed the threshold matrix by examining blocks in the low-delay image plane and by looking for visual artifacts after the asynchronous reconstruction. The numbers in the matrix are empirical rather than theoretically proven optimal values, since a good subjective quality metric is unavailable.

3. CODEC ARCHITECTURE

3.1 Encoder

The encoder, with its block diagram shown in Figure 1, is divided into two stages: segmentation and compression. The segmentation stage, which was described in the previous section, outputs an image plane for the low-delay flow. This image plane typically exhibits significant temporal redundancy and is thus differentially coded to reduce its bandwidth. Motion estimation (ME) with block DCT [6] is used to remove the redundancy in the low-delay image plane. The high-delay image plane is obtained by subtracting the anticipated decoded low-delay image from the original video. This allows quantization errors in the low-delay ME loop (lower loop in the figure) to be passed as a part of the input to the high-delay ME loop (upper loop). If there were no quantization errors in the upper loop, adding decompressed images from both flows would regenerate the original video.

Due to the asynchronous reconstruction requirement, the encoder has to ensure there is no data dependency from the low-delay flow to the high-delay flow. Should the dependency occur, data carried by the low-delay flow, which arrives early, would have to wait for data from the high-delay flow in order to be decompressed. This would return to the synchronous reconstruction model and abandon all the benefits of DCVC. To avoid cross-flow data dependency, the encoding of the low-delay image plane cannot use the previous video source frame as the reference frame in the ME loop.

Applying the sequence of low/high-delay image planes as shown in Figure 2 directly to the ME-DCT loops results in an even higher bit rate than the compressed original video generates. The sharp, artificial boundaries between blocks with and without pixel values make ME-DCT very inefficient. However, the blank area does not have to assign zero-valued pixels. The encoder can fill the area with any values as long as it does not incur data dependency and allows the decoder to track filled values. The most compression-efficient scheme we found is to copy the pixels in the same region of the reference frame in the ME loop. These values are not necessarily uniform or smooth. However, because of motion estimation, blocks in the blank area are not coded for both motion vectors and the residual image are zero. While the only blocks encoded are still those low/high-delay blocks, the artificial boundaries are much smoother and benign to compression. Applied to both loops, this pixel-filling strategy can save as much as 70% in compressed traffic. Note that this complication is not indicated in Figure 1 for clarity. Clearly, this function should be inserted at the front of each ME-DCT loop.

3.2 Decoder

The DCVC decoder follows a set of rules to display received blocks. Compressed bit streams from both flows are tagged with frame numbers as temporal references. The decoder maintains one table for each flow, in which each entry stores the temporal reference of the received block at the coordinates. The tables are initiated to zero and replace blocks from earlier frames with those from later frames. By comparing $TR_{n,L}$, temporal reference of the *n*th block from the low-delay flow, and $TR_{n,H}$, temporal reference of the *n*th block from the high-delay flow, the decoder makes the following decision:

- $TR_{n,L} > TR_{n,H}$, display the block from the low-delay flow;
- $TR_{n,L} = TR_{n,H}$, display the sum of two blocks;
- $TR_{n,L} < TR_{n,H}$, display the block from the high-delay flow.

4. RESULTS AND EVALUATION

The performance of the DCVC codec design can be evaluated by its ability to achieve competitive compression, the percentage of traffic in the low-delay flow, and subjective video quality with a delay offset. We started by encoding four test sequences of head-and-shoulder scenes, all of which except one have static backgrounds. As listed in Table 1, compressed traffic in the low-delay flows is approximately 20% to 40% of the total output and its compression ratio reaches over 100. This is comparable to the performance of most compression algorithms used in videoconferencing.

Table 1 Average number of bits to encode a pixel

Video sequence	Low-delay flow	High-delay flow
Suzie	0.060	0.124
Salesman	0.031	0.070
Carphone	0.128	0.192
Miss America	0.026	0.085

The estimation of video subjective quality with delay offsets was performed through informal viewing by graduate students. A two-alternative forced choice (2AFC) procedure was performed. Each time the test subject was shown two short video sequences with different delay offsets applied. The subject had to choose the one that he or she thought had a better quality. The result indicated, as expected, that the quality is nonincreasing as the delay offset increases. The accuracy of pointing out the correct order increases with larger delay offsets. The most common artifacts observed are blocking artifacts and incoherent movements of objects, both of which are due to time discontinuity caused by asynchronous reconstruction. As an example, in Figure 3 the reconstructed frame has several areas marked for comparison with the original frame in Figure 2. This example was generated with the delay offset set to 8 frames (264 msec). For the sake of comparison, it was not compressed. The artifacts are caused solely by the time discontinuity.

Interested readers are welcome to visit our DCVC homepage on the World Wide Web to see more examples. Test sequences and their descriptions can be found at the web site http://ptolemy.eecs.berkeley.edu/~yuanchi/DCVC.html. We are currently collaborating with a group of vision scientists in the School of Optometry at Berkeley to conduct more formal subjective tests.

Even though we are not aware of any prior research in vision science that directly addresses the issues of asynchronous reconstruction of video, recent studies on visual masking at temporal edges such as scene cuts demonstrated our new approach to be promising [7][8]. In [7], the authors reported significant visual masking occurred for 1-2 frames after a scene cut by applying coarsely quantized MPEG II video. In [8], the authors found decreased sensitivity to artifacts for up to 50 msec before and after the presentation of a luminance edge. Although both studies reported a fairly short (1-2 frames) duration of decreased sensitivity as opposed to our observation of a much longer (8-10 frames) period, their observation might have been affected by the broadband stimuli applied. While in our case, blocks are selected based on their temporal variations in spatial frequencies, the blocks delayed thus correspond to a narrowband stimuli that prolongs the duration of decreased sensitivity.

5. SUMMARY

One important performance requirement of interactive applications is low perceptual delay, which is lower bounded by the speed-of-light propagation delay. To make applications cost effective while provisioning good quality, video coding algorithms need to adapt to the modern packet networks by minimizing its encoding/decoding delay and indicating the delay tolerance of the output data to facilitate priority transmission.

We have reported the design and subjective evaluation of such a delay cognizant coding algorithm. As simple subjective tests indicated the approach of asynchronous reconstruction to be promising, we are currently conducting formal visual tests to quantify its quality. To design a better algorithm, we also recognize the need of further understanding of spatiotemporal visual masking and better modeling of the HVS.

An interesting issue we have encountered is the tradeoff between delay cognizance and compression. As explained earlier, segmenting at the granularity of pixels reduces artifacts but renders compression ineffectual. The optimal size and shape of the unit of segmentation remain to be explored.

6. REFERENCES

[1] J. Reason, L. C. Yun, A. Lao, D. G. Messerschmitt, "Asynchronous video: coordinated video coding and transport for heterogeneous networks with wireless access," *Mobile Wireless Information Systems*, Kluwer Academic Press, 1995.

[2] Y. C. Chang and D. G. Messerschmitt, "Delay cognizant video coding," *Proceedings of International Conference on Networking and Multimedia*, Kaohsiung, Taiwan, 1996, pp. 110-17.

[3] L. C. Yun and D. G. Messerschmitt, "Digital video in a fading interference wireless environment," *Proceedings of IEEE ICASSP-96*, Atlanta, GA, 1996, pp.1069-1072.

[4] P. Haskell and D. G. Messerschmitt, "In favor of an enhanced network interface for multimedia services," *IEEE Multimedia*, to appear.
[5] S. Bradner and A. Mankin, "The recommendation for IP next generation protocol," *IETF RFC 1752*, 1995.
[6] R. J. Clarke, "Digital compression of still images and video," *Academic Press*, 1995.
[7] W. J. Tam, L. Stelmach, et al. "Visual masking at video scene cuts," *Proceedings of the SPIE, Human Vision, Visual Processing and Digital Display*, San Jose, CA, 1995, pp. 111-19.
[8] T. Carney, S. A. Klein, and Q. Hu, "Visual masking near spatiotemporal edges," *Proceedings of the SPIE, Human Vision and Electronic Imaging*, San Jose, CA, 1996, pp. 393-402.

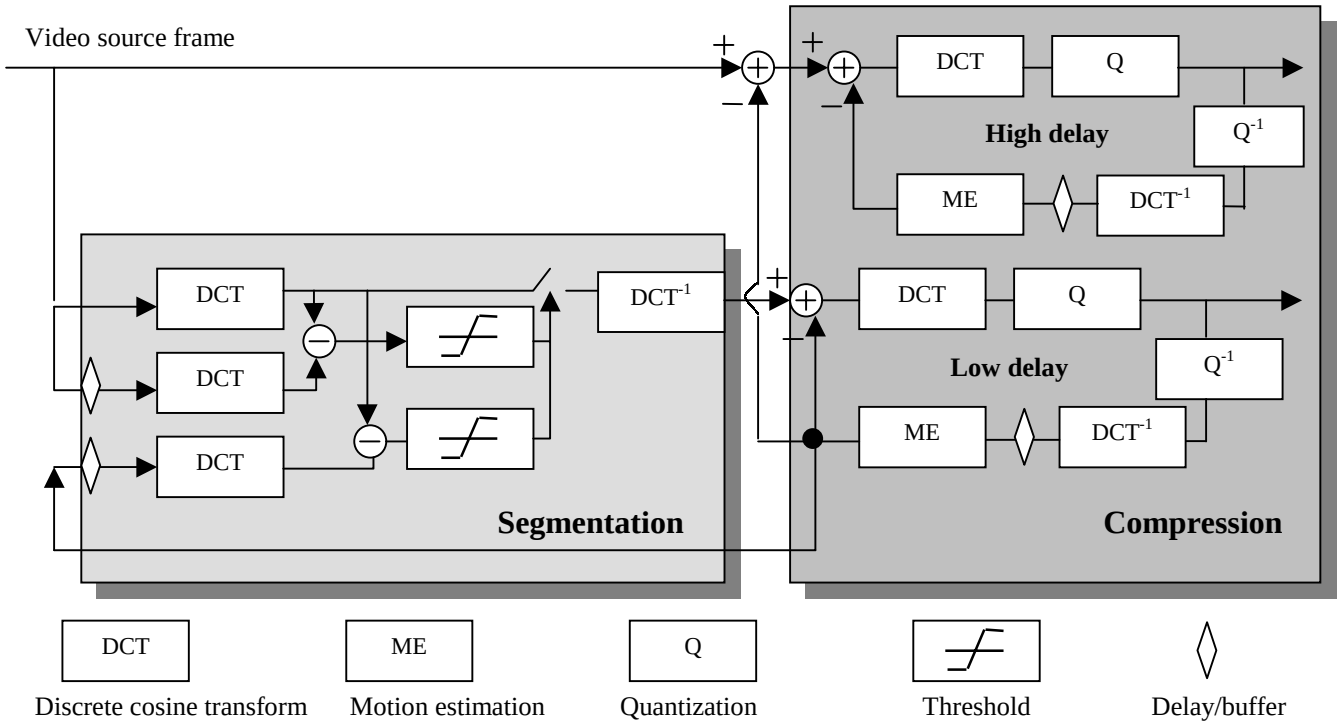

Figure 1 Encoder block diagram.

Figure 2 Left: original video frame; center: low-delay image plane; right: high-delay image plane.

Figure 3 Example showing commonly seen artifacts (marked by dotted circles) due to time discontinuity in the asynchronous reconstruction model.