Proceedings

International Conference on Image Processing

October 23 – 26, 1995

Washington, D.C.

Sponsored by

The IEEE Signal Processing Society

Volume I of III

IEEE Computer Society Press
Los Alamitos, California

Washington ● Brussels ● Tokyo

RPX Exhibit 1123
RPX v. DAE

IEEE Computer Society Press
10662 Los Vaqueros Circle
P.O. Box 3014
Los Alamitos, CA 90720-1264

Copyright and Reprint Permissions: Abstracting is permitted with credit to the source. Libraries may photocopy beyond the limits of US copyright law, for private use of patrons, those articles in this volume that carry a code at the bottom of the first page, provided that the per-copy fee indicated in the code is paid through the Copyright Clearance Center, 222 Rosewood Drive, Danvers, MA 01923.

Other copying, reprint, or republication requests should be addressed to: IEEE Copyrights Manager, IEEE Service Center, 445 Hoes Lane, P.O. Box 1331, Piscataway, NJ 08855-1331.

The papers in this book comprise the proceedings of the meeting mentioned on the cover and title page. They reflect the authors' opinions and, in the interests of timely dissemination, are published as presented and without change. Their inclusion in this publication does not necessarily constitute endorsement by the editors, the IEEE Computer Society Press, or the Institute of Electrical and Electronics Engineers, Inc.

IEEE Catalog Number 95CB35819
ISBN 0-8186-7310-9 (softbound)
ISBN 0-7803-3122-2 (casebound)
ISBN 0-7803-3123-0 (microfiche)
ISBN 0-7803-2749-7 (CD-Rom)
Library of Congress Number 95-77540

IEEE Computer Society Press
Customer Service Center
10662 Los Vaqueros Circle
P.O. Box 3014
Los Alamitos, CA 90720-1264
Tel: +1-714-821-8380
Fax: +1-714-821-4641
Email: cs.books@computer.org

IEEE Service Center
445 Hoes Lane
P.O. Box 1331
Piscataway, NJ 08855-1331
Tel: +1-908-981-1393
Fax: +1-908-981-9667
mis.custserv@computer.org

IEEE Computer Society
13, Avenue de l'Aquilon
B-1200 Brussels
BELGIUM
Tel: +32-2-770-2198
Fax: +32-2-770-8505
euro.ofc@computer.org

IEEE Computer Society
Ooshima Building
2-19-1 Minami-Aoyama
Minato-ku, Tokyo 107
JAPAN
Tel: +81-3-3408-3118
Fax: +81-3-3408-3553
tokyo.ofc@computer.org

Editorial production by Bob Werner
Cover design by Alex Torres
Cover art production by Joe Daigle/Studio Productions
Printed in the United States of America by VictorGraphics, Inc.

The Institute of Electrical and Electronics Engineers, Inc.

JOINT SOURCE/CHANNEL CODING FOR MULTICAST PACKET VIDEO

Steven McCanne

Lawrence Berkeley Laboratory
and University of California
Berkeley, CA 94720
mccanne@ee.lbl.gov

Martin Vetterli

Department of Electrical Engineering
University of California
Berkeley, CA 94720
martin@eecs.berkeley.edu

ABSTRACT

Current schemes for multicast packet video cope with congestion by adapting the transmission rate of a variable rate codec at the source. We propose a new approach, based on receiver- rather than source-adaptation, where a video source is encoded hierarchically with each layer of hierarchy distributed on a separate network channel. Each receiver can then dynamically adapt to local network capacity by adjusting the number of layers it receives. In order to deploy such a system, we must at the same time develop a layered codec tailored for this model. We present a prototype coder that has been designed specifically for our receiver-based congestion avoidance scheme. In order to evaluate the efficacy of our approach, we have implemented it in an existing Internet remote conferencing application and constrained the complexity of our design to run in real-time on standard workstations. Even with this constraint, our codec can generate a flexible range of layers while exhibiting reasonable compression performance.

1. INTRODUCTION

A recent extension to the Internet Protocol (IP) has provided an efficient mechanism for multipoint packet delivery, making possible large scale multimedia "broadcasts" over the Internet. This extension, known as IP Multicast [2], has been incrementally deployed into the Internet through research efforts and is beginning to be offered as an experimental service by commercial service providers. The demand for an IP Multicast service is driven by the utility of several "remote conferencing" applications that exploit multicast. These audio/video/whiteboard applications [7, 9, 8, 5, 13] have been used regularly over the past few years to broadcast seminars, conferences and other events over the Internet to hundreds of receivers.

A core component of these broadcasts is the transmission of video. To carry out these transmissions, one must design a "multicast video transport protocol", and in the Internet, two approaches to this transport protocol have been deployed. The first approach [6] is to simply run the transmission open-loop and to assume that there is enough spare capacity in the network to absorb the video stream. Cooperative users have established a crude bandwidth allocation scheme by coordinating the use of the network via public mailing lists. The second approach is to build congestion avoidance mechanisms into the video application [1]. Here, a video source reacts to congestion in the network by adjusting its transmission rate.

A fundamental problem with this source-based congestion control scheme is that a uniform quality of video is delivered to all the receivers in the network. If the broadcast is to be delivered free of congestion to all receivers, then the transmission must run at the most constrained rate. In the Internet, where heterogeneity is prevalent, this constraint greatly limits the usefulness of the approach.

To overcome this problem, researchers have proposed schemes based on hierarchical representations of the video signal [11, 4]. In these schemes, the layers of a hierarchically coded signal are partitioned across a set of network level abstractions, e.g., connections. Each user receives the maximum number of layers that the network can support along the distribution path from the source to the receiver. In the worst case, only the highest priority (lowest quality) layer is received. This model is well suited for the heterogeneity of the Internet, since each receiver individually adapts to local spare network capacity.

Unfortunately, these proposed schemes rely on signalling mechanisms in the network to establish the subset of layers to distribute along each path in the network. Such signalling mechanisms do not yet exist in the Internet, and deploying them would require radical changes that cannot be incrementally deployed. Because of the Internet's scale and heterogeneity, such nonincremental change is logistically impossible.

2. RECEIVER-BASED CONGESTION AVOIDANCE

One way to deploy layered transmission in the existing Internet was first suggested by Deering [3] and later described by Turletti and Bolot in [14]. The idea is to transmit each coded layer on a different multicast address and allow receivers to individually adapt to network conditions by joining and leaving multicast groups. The multicast routing protocols *prune* away portions of the distribution tree that have no downstream receivers. We call this approach *receiver-based congestion avoidance*, since receivers rather than senders adapt to congestion by dropping or adding layers.

Layering provides a successive approximation to the source R-D characteristic, while pruning provides a mechanism for receivers to specify their local operating point on the R-D curve. This pruning process is fundamentally an interaction between the source and channel coding algorithms, i.e., a method of joint source/channel coding.

In its simplest form, a video source transmits the layers on their respective multicast addresses without any explicit feedback from the receivers. If no receivers tune in to the transmission, the multicast routing protocol prunes the transmission all the way back to the source's sub-network. When a receiver subscribes to one or more layers, the network allows the corresponding multicast groups to flow along the branches of the distribution tree to that receiver. No explicit signalling is required to tell the network where or how to filter flows.

This simple mechanism, which is already deployed (to some extent) in the Internet, provides the necessary primitives to implement our receiver-based congestion avoidance scheme. Assume a receiver has subscribed to N layers of a transmission. The adaptation algorithm is as follows:

- On congestion, drop layer N.
- When capacity is available, add layer $N+1$.

Congestion can be inferred through packet loss (e.g., using sequence numbers) or conveyed directly from routers to end-systems through explicit congestion notification. On the other hand, inferring spare capacity is more difficult. One approach is to probe for available bandwidth by spontaneously adding a layer. If congestion results, the layer will be quickly dropped. By employing conservative probing and aggressive backoff strategies and by using relatively long time constants, we believe we can produce a stable system.

As an optimization, receivers generate low rate feedback messages to the source that indicate the bandwidth utilized to each destination. This receiver-feedback is carried out in a scalable fashion using RTP [10]. If the source utilizes an embedded compression algorithm, then it can use the receiver feedback to adjust the allocation of rate to the different layers of the multicast distribution. For example, a source might discover that some layers are not in use and thereby avoid coding and transmitting them, or it might discover that all receivers are connected at high bandwidth and can therefore collapse the layers into a single, high quality stream.

The success of our proposed multicast video transport system depends strongly on the efficacy of the underlying video codec, which must be designed to match the constraints of the layered transport system and the delay and loss behavior of the Internet. Since our design is motivated heavily by the goal of timely deployment in the Internet, it must be easily distributed and operate within the constraints of the existing technology base. Thus, it must be feasible to operate the codec in software on a general purpose workstation in real-time.

3. CURRENT INTERNET VIDEO CODECS

In designing our layered codec, we leveraged off current experience in the design of video transmission systems for the Internet. The two predominant applications are the INRIA Video Conferencing system (*ivs*) [13] and the Network Video (*nv*) [5] tool from Xerox PARC. The former is based on a software-only H.261 codec, while the latter utilizes a custom compression scheme.

One early outcome of the deployment of these video tools was that the user community preferred the custom *nv* compression scheme, even though the compression performance of H.261 is much higher. The reason is twofold. First, the *nv* compression algorithm does not code image differences, while the H.261 codec in *ivs* codes the majority of the block updates as differences. Consequently, the *nv* coder is more robust to packet loss since the resulting errors are relatively short-lived. Second, the *nv* coder runs fast. It utilizes a simple compression scheme based on run-length encoded Haar transform coefficients. This low complexity results in higher perceived quality because the system rarely bottlenecks in the compression process.

More recently, the advantages of standards compliance and interoperability of H.261 were combined with *nv*'s robustness in an H.261 coder implemented (by one of us) in the UCB/LBL Video Conferencing tool, *vic* [8]. The result is a scheme in which blocks are conditionally replenished, as in *nv*, but the blocks are coded using H.261. By employing a very restricted subset of H.261 (intramode block updates only), we implemented the robust coding style of *nv* using an H.261 compliant syntax. Moreover, the computational requirements were substantially reduced by eliminating motion estimation. Even with this sacrifice, the Intra-H.261 coder typically outperforms the *nv* coder by 6 to 7dB.

4. A LAYERED CODEC

We now turn to a discussion of our prototype layered coding scheme. While high-quality layered compression schemes have been proposed, these systems focus on optimizing compression performance without placing tight constraints on complexity. Our goal is not to design a codec that outperforms all existing schemes, but rather one that can be implemented to run in real-time on standard workstations, and at the same time, perform "adequately". In the short term, we must make tradeoffs in complexity for viable operation, while in the long term, we can develop more sophisticated algorithms to track processor performance advances since the codec will be software-based and easily re-deployed.

The four key design constraints for the codec are: (1) robustness to packet loss, (2) low computational complexity, (3) compute scalability, and (4) a layered representation.

Robustness. One technique for providing high robustness to loss is to avoid the interframe prediction loop by using intraframe coding as in "Motion JPEG". However, this results in poor compression performance. Our hybrid approach is to employ the block-based conditional replenishment scheme used in the existing video tools. Here, block updates are intracoded, while blocks whose difference signal is below some threshold are suppressed entirely. This results in a packet stream where data is independent of the past. In particular, it is independent of past losses.

To avoid persistent errors, a background process scans the entire image refreshing blocks at some configurable (low) rate. For the video sources with stationary backgrounds (which is common in current Internet transmissions), persistent errors are fairly uncommon. Since loss (usually) is associated with a non-stationary area of the image, it is

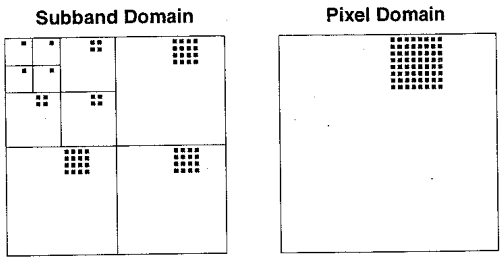

Figure 1: Relationship between subband coefficients and pixel blocks.

likely restored via conditional replenishment (before the refresh process updates it). Thus the artifact is (usually) absolved immediately.

Low computational complexity. Because the codec must operate in software, it must exhibit low computational complexity. Fortunately, the robustness constraints imposed above reduce the computational requirements substantially. Conditional replenishment can be carried out in the pixel domain, so very early in the coding process, computational load is shed by deciding not to code a block. Additionally, the elimination of interframe prediction means that expensive motion searches are not performed. A similar approach has been taken in [15], where blocks are intracoded using an efficient table driven approach based on hierarchical vector quantization.

Compute Scalability. In the heterogeneous environment like the Internet, end systems will have diverse computational resources. Users with less capable end systems should still be able to decode a transmission. With the layered scheme presented here, an end system can adapt to local computational resource availability in exactly the same way it adapts to network resource availability. When the decoder becomes computation bound, it can drop layers to reduce load. Likewise, an encoder can code fewer layers under load (albeit at the expense of all the receivers).

Layered Representation. While a block-based conditional replenishment scheme is attractive because it reduces computational overhead and results in a robust transmission, it is at odds with a layered subband coding scheme. Traditional subband coding algorithms operate on entire images rather than individual blocks. To resolve this mismatch, we can treat each conditionally replenished block as an independent image and apply an existing 2D subband coding scheme to each block. However, this produces blocking artifacts which are amplified by the signal extension techniques used to attain critically subsampled decompositions.

Instead, we take an alternate approach where we replenish subband coefficients instead of pixel blocks. Each subband coefficient represents the scale and location of a wavelet basis vector in the original image. Thus, we can relate each coefficient with its spatial location in the image, as illustrated in Figure 1. We still employ pixel-based conditional replenishment, but instead of coding the block directly, we use it to identify the subband coefficients that need to be replenished.

Unfortunately, recursive iteration of the analysis filters results in coarse scale wavelet basis functions with large regions of support. In other words, spatially local scene changes can affect a large number of wavelet coefficients. Moreover, we observed a phase inversion effect where large numbers of coefficients would change sign from frame to frame with very little scene activity. Even with relatively short filters (e.g., four taps), these effects were prohibitive. On the other hand, the two tap Haar basis does not suffer from these effects at all, since the basis function of one image block do not overlap with those of another (and the Haar is the only such basis). That is, an $N \times N$ pixel block is in exact correspondence with N^2 coefficients from the subband decomposition (as shown in the figure).

However, a marked disadvantage of the Haar basis is that quantization of its square-wave prototype functions produces blocking artifacts. Our (partial) solution to this problem is to use a hybrid approach, where the first stage (or two) of the subband analysis is carried out using a longer filter while the remaining stages are carried out using the Haar filter. As a result, the blocking artifacts of the Haar decomposition are smoothed out by the low-pass reconstruction filters. Finally, while the longer first stage filter produces some amount of block overlap, the effects are minimized by not allowing the filter to iterate.

The Prototype. Our prototype coder can be decomposed into four stages:

1) conditional replenishment
2) subband analysis
3) scanning set determination
4) bit-plane entropy coding

First, we determine the blocks to be replenished by comparing the current frame with a reference frame of transmitted blocks. Then, each block is analyzed with a subband decomposition. The first stage is the biorthogonal filter pair, $H_0(z) = -0.25 + 0.75z + 0.75z^2 - 0.25z^3$ and $H_1(z) = 0.25 + 0.75z - 0.75z^2 - 0.25z^3$, while the remaining stages use Haar filters. Note that both the Haar and biorthogonal filters can be carried out very efficiently using fixed point shifts, sums, and differences. Because the filters are cheap, the bottleneck becomes memory accesses. Hence, we minimize memory traffic by storing each coefficient in a single byte by retaining only the 8 most significant bits.

We then compute coefficient scanning sets using the well-known quadtree interpretation of the subband coefficients [12]. For each coefficient, we determine the set of bit positions such that at least one successor in the quadtree has a non-zero value in the given bit position. These sets can be computed efficiently in a single bottom-up traversal of the subbands, using only simple bit-wise operations.

Finally, we entropy code the bit-planes using a depth-first traversal of the coefficient quadtrees. Groups of bit-planes are allocated to layers, and all of the layers are encoded in parallel. During the scan, we retain a bit vector of active bit positions which is updated using the scanning sets from step 3. When we find that a bit position has no children, we remove it from the active set, effectively terminating the scan at that position. For each layer of each coefficient, we produce a Huffman code using a table lookup

DOCKET ALARM

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS

Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS

Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS

Sync your system to PACER to automate legal marketing.

WHAT WILL YOU BUILD? | sales@docketalarm.com | 1-866-77-FASTCASE

fastcase®
Smarter legal research.