system.[17] Layered content is multicasted using multiple multicast groups, and receivers are responsible for adapting to network bandwidth variations by dynamically joining and leaving multicast groups.

7. CONCLUSIONS AND FUTURE WORK

We have presented SCP, an effective flow and congestion control scheme for Internet media streaming applications. SCP uses the same congestion avoidance mechanism as TCP. However, when the network is not congested, SCP's hybrid rate- and window-based flow control policy ensures stable, smooth streaming, and low latency. To adapt quickly to drastic changes in network bandwidth following a pause in the stream, or a switch of network interfaces, the state of SCP can be reset to begin a slow-start policy that uses a reduced initial congestion window size. Finally, SCP does not retransmit lost packets and hence eliminates that source of unpredictability in latency.

The experiments described in this paper test SCP over typical network configurations, including PPP, single subnet, single router (LAN), and across the long-haul Internet (WAN). The experiments demonstrate that SCP is able to stream data packets smoothly in all the cases. The buffering delay of SCP streams, even over PPP links with deep buffers, remains suitably low. Multiple SCP streams are able to share the network bandwidth in a stable, fair manner. Although our experiments show that residual errors in base RTT measurement can have a significant impact on network bandwidth partitioning among SCP sessions. Ensuring fair bandwidth partitioning without direct cooperation from network routers and switches, while not impossible, is not easy.

Our experiments also showed that SCP streams are able to share network bandwidth with TCP streams. SCP seems somewhat less aggressive than TCP, especially in the case of the PPP experiment. This behavior is a result of SCP's steady-state policy that limits packet buildup in router buffers, unlike TCP which tends to fill those buffers. This observation raises the question of whether TCP is too aggressive or SCP is too conservative. Filling up deep buffers in PPP servers merely results in more latency, but does not help to increase throughput. In interactive streaming applications, multiple seconds of latency is intolerable, and hence TCP's behavior over PPP links is clearly detrimental.

The properties described above make SCP a promising protocol for use with best-effort multimedia streaming across the Internet. We have incorporated SCP into our adaptive streaming video player.[6] Based on the available network bandwidth discovered by SCP, the player adapts the video presentation quality in multiple dimensions, including spatial resolution, frame rate, end-end latency, etc.

In the future, we plan to fine tune SCP's parameters and policies, study its execution overhead, optimize the number of ACK packets it generates, and evaluate SCP in more complex network scenarios. We also plan to compare SCP with other congestion control schemes - especially those used in commercial streaming media players - and further study, through analysis and experiments, the interaction between multiple SCP sessions and other congestion control schemes.

ACKNOWLEDGEMENTS

Discussions with the Quasar group at OGI motivated this work. Jon Inouye, Dylan McNamee and Ashvin Goel provided thoughtful comments on early drafts of this paper. Special thanks to Jon Inouye for his insights into TCP and other network protocols, and for discussions on the ideas behind this work. We are grateful to Prof. Karsten Schwan (Georgia Tech) for providing the account used to run the Internet experiments. This project was supported in part by DARPA grants N00014-94-1-0845, N66001-97-C-8522 and N66001-97-C-8523, and by Tektronix, Inc. and Intel Corporation.

REFERENCES

1. L. A. Rowe, K. D. Patel, B. C. Smith, and K. Liu, "MPEG video in software: Representation, transmission and playback," in *Symposium on Electronic Imaging Science and Technology*, pp. 15–26, (San Jose, California), February 1994.
2. S. Cen, C. Pu, R. Staehli, C. Cowan, and J. Walpole, "A distributed real-time MPEG video audio player," *NOSSDAV'95, Lecture Notes in Computer Science* **1018**, pp. 151–162. Springer-Verlag, 1995.

Google Exhibit No. 1063
Page 1169

3. S. McCanne and V. Jacobson, "*vic:* a Flexible Framework for Packet Video," in *Proceedings of the Third ACM Conference on Multimedia*, pp. 511–522, (San Francisco, California), November 1995.
4. V. Jacobson, "Congestion avoidance and control," in *SIGCOMM'88*, pp. 79–88, August 1988.
5. J. Inouye, S. Cen, C. Pu, and J. Walpole, "System support for mobile multimedia applications," in *NOSSDAV'97*, pp. 143–154, May 19-21 1997.
6. S. Cen, *A Software Feedback Toolkit and Its Applications in Adaptive Multimedia Systems.* PhD thesis, Department of Computer Science and Engineering, Oregon Graduate Institute of Science and Technology, October 1997.
7. T. Berners-Lee, R. Fielding, and H. Nielsen, "Hypertext transfer protocol – HTTP/1.0." Internet RFC 1945, `http://ds.internic.net/ds/rfc-index.html`, May 1996, [September 11, 1997].
8. L. S. Brakmo *et al.*, "TCP Vegas: New techniques for congestion detection and avoidance," in *SIGCOMM'94*, pp. 24–35, August 1994.
9. S. Jacobs and A. Eleftheriadis, "Adaptive video applications for non-QoS networks," in *International Workshop on Quality of Service'97*, pp. 161–166, (Columbia University, New York), May 1997.
10. B. C. Smith, "Cyclic-UDP: a priority-driven best-effort protocol." Computer Science Department, Cornell University, `http://www2.cs.cornell.edu/Zeno/papers/cyclicudp.pdf`, May 1994, [October 20, 1997].
11. H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson, "RTP: A transport protocol for real-time applications." Internet RFC 1889, `http://ds.internic.net/ds/rfc-index.html`, January 1996, [September 11, 1997].
12. I. Busse, B. Deffner, and H. Schulzrinne, "Dynamic QoS control of multimedia applications based on RTP," *Computer Communications,* **19**, pp. 49–58, January 1996.
13. S. Keshav, "A control-theoretic approach to flow control," in *Proceedings of SIGCOMM'91*, pp. 3–16, (Zurich, Switzerland), Sept. 1991.
14. J. Mahdavi and S. Floyd, "TCP-friendly unicast rate-based flow control." In end2end mailing list, `ftp://ftp.isi.edu/end2end`, January 1997, [September 11, 1997].
15. K. Jeffay, M. Parris, F. D. Smith, and T. M. Talley, "A router-based congestion control scheme for real-time continuous media," in *Proceedings of NOSSDAV'96*, (Zushi, Japan), April 1996.
16. S. Floyd and K. Fall, "Router mechanisms to support end-to-end congestion control." Network Research Group, Lawrence Berkeley National Laboratory. `ftp://ftp.ee.lbl.gov/papers/collapse.ps`, February 1997, [September 11, 1997].
17. S. McCanne and V. Jacobson, "Receiver-driven layered multicast," in *SIGCOMM'96*, (Stanford, California), August 1996.

Google Exhibit No. 1063
Page 1170

PROCEEDINGS OF SPIE

SPIE—The International Society for Optical Engineering

Multimedia Computing and Networking 1998

Kevin Jeffay
Dilip D. Kandlur
Timothy Roscoe
Chairs/Editors

26–28 January 1998
San Jose, California

Sponsored by
SPIE—The International Society for Optical Engineering
IS&T—The Society for Imaging Science and Technology

Cosponsored by
ACM SIGMultimedia

SPIE P PROCEEDINGS SERIES

Volume 3310

Google Exhibit No. 1063
Page 1171

PROCEEDINGS OF SPIE
SPIE—The International Society for Optical Engineering

Multimedia Computing and Networking 1998

Kevin Jeffay
Dilip D. Kandlur
Timothy Roscoe
Chairs/Editors

26–28 January 1998
San Jose, California

Sponsored by
SPIE—The International Society for Optical Engineering
IS&T—The Society for Imaging Science and Technology

Cosponsored by
ACM SIGMultimedia

Published by
SPIE—The International Society for Optical Engineering



Volume 3310

SPIE is an international technical society dedicated to advancing engineering and scientific applications of optical, photonic, imaging, electronic, and optoelectronic technologies.

TK5105.5
M84
1998



The papers appearing in this book comprise the proceedings of the meeting mentioned on the cover and title page. They reflect the authors' opinions and are published as presented and without change, in the interests of timely dissemination. Their inclusion in this publication does not necessarily constitute endorsement by the editors or by SPIE.

Please use the following format to cite material from this book:
Author(s), "Title of paper," in *Multimedia Computing and Networking 1998*, Kevin Jeffay, Dilip D. Kandlur, Timothy Roscoe, Editors, Proceedings of SPIE Vol. 3310, page numbers (1997).

ISSN 0277-786X
ISBN 0-8194-2750-0

Published by
SPIE—The International Society for Optical Engineering
P.O. Box 10, Bellingham, Washington 98227-0010 USA
Telephone 360/676-3290 (Pacific Time) • Fax 360/647-1445

Copyright ©1997, The Society of Photo-Optical Instrumentation Engineers.

Copying of material in this book for internal or personal use, or for the internal or personal use of specific clients, beyond the fair use provisions granted by the U.S. Copyright Law is authorized by SPIE subject to payment of copying fees. The Transactional Reporting Service base fee for this volume is $10.00 per article (or portion thereof), which should be paid directly to the Copyright Clearance Center (CCC), 222 Rosewood Drive, Danvers, MA 01923. Payment may also be made electronically through CCC Online at http://www.directory.net/copyright/. Other copying for republication, resale, advertising or promotion, or any form of systematic or multiple reproduction of any material in this book is prohibited except with permission in writing from the publisher. The CCC fee code is 0277-786X/97/$10.00.

Printed in the United States of America.

Google Exhibit No. 1063
Page 1173

TA69 .S5 1-8-1

Contents

v *Conference Committee*

SESSION 1 MULTIMEDIA SYSTEM DEVELOPMENT TOOLS

2 **Middleware support for distributed multimedia and collaborative computing** [3310-01]
K. P. Birman, R. Friedman, M. Hayden, Cornell Univ.; I. Rhee, North Carolina State Univ.

14 **Multiplatform simulation of video playout performance** [3310-02]
L. Gharai, R. Gerber, Univ. of Maryland/College Park

28 **Software-only video production switcher for the Internet MBone** [3310-03]
T. H. Wong, K. D. Mayer-Patel, D. Simpson, L. A. Rowe, Univ. of California/Berkeley

SESSION 2 OPERATING SYSTEMS I

42 **Applying statistical process control to the adaptive rate control problem** [3310-04]
N. R. Manohar, M. H. Willebeek-LeMair, IBM Thomas J. Watson Research Ctr.; A. Prakash, Univ. of Michigan

57 **Integrated input/output system for kernel data streaming** [3310-05]
F. W. Miller, Univ. of Maryland/College Park; S. K. Tripathi, Univ. of California/Riverside

69 **Measurement-based resource allocation for multimedia applications** [3310-06]
P. Barham, S. Crosby, T. Granger, N. Stratford, Univ. of Cambridge (UK); M. Huggard, F. Toomey, Dublin Institute for Advanced Studies (UK)

SESSION 3 VIDEO-ON-DEMAND

84 **Supporting interactive scanning operations in VOD systems** [3310-07]
G. Apostolopoulos, Univ. of Maryland/College Park; M. Krunz, Univ. of Arizona; S. K. Tripathi, Univ. of California/Riverside

96 **Movie approximation technique for the implementation of fast bandwidth-smoothing algorithms** [3310-08]
W. Feng, C. C. Lam, M. Liu, The Ohio State Univ.

111 **Implementation of dynamic service aggregation for interactive video delivery** [3310-09]
P. Basu, A. Narayanan, R. Krishnan, T. D. C. Little, Boston Univ.

SESSION 4 OPERATING SYSTEMS II

124 **Symphony: an integrated multimedia file system** [3310-10]
P. J. Shenoy, P. Goyal, S. S. Rao, H. M. Vin, Univ. of Texas at Austin

139 **Adaptive prefetching for device-independent file I/O** [3310-11]
D. Revel, D. McNamee, D. Steere, J. Walpole, Oregon Graduate Institute of Science and Technology

LINDA HALL LIBRARY
KANSAS CITY, MISSOURI

Google Exhibit No. 1063
Page 1174

150 **Resource kernels: a resource-centric approach to real-time and multimedia systems** [3310-12]
R. Rajkumar, K. Juvva, A. Molano, S. Oikawa, Carnegie Mellon Univ.

SESSION 5 THE WORLD WIDE WEB

166 **Experiment to characterize videos stored on the Web** [3310-13]
S. Acharya, B. C. Smith, Cornell Univ.

179 **Static caching of Web servers** [3310-14]
Z. Liu, P. Nain, N. Niclausse, INRIA Ctr. Sophia Antipolis (France); D. Towsley, Univ. of Massachusetts/Amherst

191 **Resource-based caching for Web servers** [3310-15]
R. Tewari, H. M. Vin, Univ. of Texas at Austin; A. Dan, D. Sitaram, IBM Thomas J. Watson Research Ctr.

SESSION 6 MULTIMEDIA APPLICATIONS

206 **Integrated audiovisual processing for object localization and tracking** [3310-16]
G. S. Pingali, Lucent Technologies/Bell Labs.

214 **Accelerating M-JPEG compression with temporal information** [3310-17]
H. Bönisch, K. Froitzheim, P. Schulthess, Univ. Ulm (FRG)

226 **Cross-modal retrieval of scripted speech audio** [3310-18]
C. B. Owen, F. Makedon, Dartmouth College

SESSION 7 FLOW AND CONGESTION CONTROL

238 **Adaptive source rate control for wireless video conferencing** [3310-19]
H. Liu, M. El Zarki, Univ. of Pennsylvania

250 **Flow and congestion control for Internet media streaming applications** [3310-20]
S. Cen, Tektronix, Inc.; J. Walpole, C. Pu, Oregon Graduate Institute of Science and Technology

266 *Author Index*

Google Exhibit No. 1063
Page 1175

Ex. 1032B



Top
Send to
Get It
Details
Virtual Browse
Links

CONFERENCE PROCEEDING

Multimedia computing and networking 1998 : 26-28 January, 1998, San Jose, California / Kevin Jeffay, Dilip D. Kandlur, Timothy Roscoe, chairs/editors ; sponsored by IS&T--the Society for Imaging Science and Technology, SPIE--the International Society for Optical Engineering ; cosponsored by, ACM SIG Multimedia ; published by SPIE--the International Society for Optical Engineering

Jeffay, Kevin; Kandlur, Dilip D.; Roscoe, Timothy; IS & T--the Society for Imaging Science and Technology; ACM Special Interest Group on Multimedia; Society of Photo-optical Instrumentation Engineers
c1997

Available at Linda Hall Library Books - 2nd Floor (TK5105.5 .M84 1998) >

Send to

Email | QR | Permalink | Citation

EasyBib | Export RIS | Export to Excel

Get It


LOCATION ITEMS
Linda Hall Library
Available , Books - 2nd Floor ; TK5105.5 .M84 1998
(1 copy, 1 available, 0 requests)
Item in place
Loanable

Details

Title	Multimedia computing and networking 1998 : 26-28 January, 1998, San Jose, California / Kevin Jeffay, Dilip D. Kandlur, Timothy Roscoe, chairs/editors ; sponsored by IS&T--the Society for Imaging Science and Technology, SPIE--the International Society for Optical Engineering ; cosponsored by, ACM SIG Multimedia ; published by SPIE--the International Society for Optical Engineering
Other title	Proceedings of SPIE--the International Society for Optical Engineering. v. 3310.
Creator/Contributor	Jeffay, Kevin Kandlur, Dilip D. Roscoe, Timothy IS & T--the Society for Imaging Science and Technology ACM Special Interest Group on Multimedia Society of Photo-optical Instrumentation Engineers
Subjects	Multimedia systems Computer networks
Identifier	ISBN : 9780819427502 ISBN : 0819427500 OCLC : (OCoLC)38175077
Series	Proceedings / SPIE--the International Society for Optical Engineering ; v. 3310 Proceedings of SPIE--the International Society for Optical Engineering. v. 3310.
Publisher	Bellingham, Washington : SPIE
Creation date	c1997
Format	v, 266 p. : ill. ; 28 cm.
General notes	Includes bibliographic references and author index.

Virtual Browse



vork ning, urement,

Multimedia computing and networking

Multimedia computing and networking

Multim compu and networ



Links

Multimedia computing and networking 1998 : $b26-28 January, 1998, San Jose, California (Proceedings of SPIE--the International Society for Optical Engineering. v. 3310).
Linda Hall Library MARC record copied from
https://catalog.lindahall.org/discovery/sourceRecord?vid=01LINDAHALL_INST:LHL&docId=alma992821293405961&recordOwner=01LINDAHALL_INST

```
leader  01501nam a2200325Ia 4500
001     992821293405961
005     19980109083600.0
008     980109s1997    waua     b    101 0 eng d
020     ##$a9780819427502
020     ##$a0819427500
035     ##$a(OCoLC)38175077
035     ##$9391049
035     ##$a(MoKL)282129-lhalldb
035     ##$a(lhalldb)282129-lhalldb
040     ##$aLHL $cLHL
049     ##$aLHLA
090     ##$aTK5105.5 $b.M84 1998
245     00$aMultimedia computing and networking 1998 : $b26-28 January, 1998,
San Jose, California / $cKevin Jeffay, Dilip D. Kandlur, Timothy Roscoe,
chairs/editors ; sponsored by IS&T--the Society for Imaging Science and
Technology, SPIE--the International Society for Optical Engineering ;
cosponsored by, ACM SIG Multimedia ; published by SPIE--the International
Society for Optical Engineering.
260     ##$aBellingham, Washington : $bSPIE, $cc1997.
300     ##$av, 266 p. : $bill. ; $c28 cm.
490     1#$aProceedings / SPIE--the International Society for Optical
Engineering ; $vv. 3310
504     ##$aIncludes bibliographic references and author index.
650     #0$aMultimedia systems $vCongresses
650     #0$aComputer networks $vCongresses
700     1#$aJeffay, Kevin
700     1#$aKandlur, Dilip D.
700     1#$aRoscoe, Timothy
710     2#$aIS & T--the Society for Imaging Science and Technology
710     2#$aACM Special Interest Group on Multimedia
710     2#$aSociety of Photo-optical Instrumentation Engineers
830     #0$aProceedings of SPIE--the International Society for Optical
Engineering. $vv. 3310.
948     ##$aLTI 09/01/2008
```

Proceedings of SPIE Image and Video Communications and Processing, January 2000

Signal Processing for Internet Video Streaming: A Review

Jian Lu*
Apple Computer, Inc.

ABSTRACT

Despite the commercial success, video streaming remains a black art owing to its root in proprietary commercial development. As such, many challenging technological issues that need to be addressed are not even well understood. The purpose of this paper is to review several important signal processing issues related to video streaming, and put them in the context of a client-server based media streaming architecture on the Internet. Such a context is critical, as we shall see that a number of solutions proposed by signal processing researchers are simply unrealistic for real-world video streaming on the Internet. We identify a family of viable solutions and evaluate their pros and cons. We further identify areas of research that have received less attention and point to the problems to which a better solution is eagerly sought by the industry.

Keywords: video streaming, video compression, Internet video, rate adaptation, packet loss, error control, error recovery.

1. INTRODUCTION

Video streaming is one of the most exciting applications on the Internet. It already created a new business known as Internet broadcast, or Intercast/Webcast. Although Internet broadcast is still in its infancy, we get a glimpse of the future of broadcasting where people can enjoy not only scheduled live or pre-recorded broadcast but also on-demand and personalized programs. Stirring more excitement is the idea that anyone will be able to set up an Internet TV station and broadcast his or her video programs on the Internet just like we publish our "homepage" on the Web today. While video streaming as a core technology behind a new business is a success story, it remains very much a black art owing to its root in proprietary commercial development. Indeed, to the author's knowledge, there has been no published research literature to date that offers a comprehensive review of video streaming architecture and system, and related design issues.

Even the definition of streaming is sketchy; it is often described as a process. Generally speaking, streaming involves sending media (e.g., audio and video) from a server to a client over a packet-based network such as the Internet. The server breaks the media into packets that are routed and delivered over the network. At the receiving end, the packets are reassembled by the client and the media is played as it comes in. A series of time-stamped packets is called a *stream*. From a user's perspective, streaming differs from simple file transfer in that the client plays the media as it comes in over the network, rather than waiting for the entire media to be downloaded before it can be played. In fact, a client may never actually download a streaming video; it may simply reassemble the video's packets and play the video as the packets come in, and then discard them. To a user the essence of streaming video may lie in an expectation that the video stream the user has requested should come in continuously without interruption. This is indeed one of the most important goals for which a video streaming system is designed.

Over the last decade, IP multicast and particularly the MBone have received considerable attention from the research community. IP multicast provides a simple and elegant architecture and service model that are particularly suitable for real-time multimedia communications. The emergence of the MBone, an experimental virtual multicast backbone, has offered a testbed for researchers around the world to develop and test a suite of applications known as MBone tools for real-time multimedia communications and collaborations. See, for example, a recent article by McCanne[1] for the evolution of the MBone and its applications. Many of these applications have been designed for audio and video conferencing and broadcasting using IP multicast. On November 18, 1994, the MBone reached a festive milestone amid great fanfare when Rolling Stones broadcast the first major rock band concert on the MBone.

Despite the promise demonstrated by the MBone and a logical reasoning that IP multicast would serve as a foundation for video streaming on the Internet, commercial deployment of multicast applications and services on the public Internet has been painfully slow by the Internet standard. The bulk of the streaming media traffic generated by three major commercial

* Correspondence: One Infinite Loop, MS: 302-3MT, Cupertino, CA 95014, USA; Email: jian@computer.org.

Google Exhibit No. 1033
Page 001

Google Exhibit No. 1063
Page 1182

streaming systems (i.e., QuickTime, RealSystem, and Windows Media) on today's Internet consists of largely unicast streams. It is out of this paper's scope to discuss the unresolved issues that have slowed down the deployment of multicast applications and services; interested readers are referred to a recent white paper published by the IP Multicast Initiatives.[2] However, we want to note that even when multicast is fully deployed, a very large volume of streaming media traffic will remain to be unicast streams. This is because a large portion of the streaming media content is archived and played on-demand by users on the Internet. Multicast is of little help to on-demand streaming unless a large number of users from the same neighborhood of the Internet would request the same content simultaneously.

The purpose of this paper is to review several important issues in Internet video streaming from a signal processing perspective. For the reasons we have just mentioned above, our review will focus on unicast streaming while considering multicast as a possible extension. Drawing from the author's experience in developing one of the major commercial media streaming systems, we attempt to identify the most challenging problems in video streaming and evaluate a number of signal processing solutions that have been proposed; some of these solutions have already been implemented in experimental and/or commercial video streaming systems. To make the review useful to both researchers and application developers, we would like to clearly identify among the proposed solutions: what works in real world, what doesn't, and why? We want to note that although to a Web surfer streaming video commonly implies synchronized audio as well, video and audio are actually delivered in independent streams in a streaming session. We will not discuss streaming audio in this paper.

The rest of this paper is organized as follows. Section 2 introduces streaming system and architecture and describes what we believe to be the most challenging problems in video streaming. We also examine some important characteristics of the streaming system and the constraints that are imposed on any prospective signal processing solutions. Section 3 reviews a number of signal processing solutions to the video streaming problems in great detail. We first comment on some proposed solutions that are simply unrealistic for real-world video streaming on the Internet. Then, we evaluate a family of viable solutions, identify areas of research that have received less attention, and point to the problems to which a better solution is eagerly sought by the industry. Section 4 concludes the paper.

2. VIDEO STREAMING SYSTEM AND ARCHITECTURE

2.1. Video Streaming Model and Architecture

A schematic system diagram of Internet video streaming is shown in Figure 1 for both unicast and multicast scenarios. Figure 1(a) represents the unicast model for two types of video streaming services, *on-demand* or *live*. In on-demand streaming, pre-compressed, streamable video content is archived on a storage system such as a hard disk drive and served to a client on demand by the streaming server. In live streaming, a video stream coming out of a live source such as a video camera is compressed in real-time by the encoder. The compressed bitstream is packetized by the packetizer, and sent over the Internet by the streaming server. On the client side, the video bitstream is reassembled from the received packets by the reassembler, then decoded by the decoder, and finally rendered on a display. Noting that each client connecting to the server receives an individual stream, the disadvantages of the unicast streaming model can be explained. Clearly, the server load will increase proportionally to the number of clients connected to and served by the server. When a large number of clients try to connect to a server at the same time, the server gets overloaded, creating a "hot spot" on the Internet. Additionally, sending copies of the same data in individual streams to a large number of clients is obviously inefficient; it causes congestion on the network and deteriorates quality of services.

The multicast model of Figure 1(b) has many advantages over unicast in live video streaming (broadcasting) where a large audience could tune in for the broadcast at the same time. Instead of sending a stream to each client individually, the server sends out only one stream that is routed to one or more "group addresses". A client tunes in to a multicast group in its neighborhood to receive the stream. Since the server is decoupled from connecting to each client directly, server load does not increase with the number of receiving clients. In the case of delivering a stream to multiple groups, the stream on its route gets replicated and branched only at the fan-out points within the network. When combined with layered coding that will be discussed in Section 3, multicast provides a decentralized and highly efficient video streaming architecture.

Despite the clear advantages of multicast-based video streaming, it is employed in only a small fraction of today's Internet, primarily in Intranets within an institution or corporation. There are complex reasons for the slow deployment of multicast-based services in general.[2] Beyond deployment hurdles we note that a large volume of streaming video content is archived and streamed to users on demand. Multicast does not help very much in on-demand streaming where the service requests from users are random and highly desynchronized in terms of requested content and timing of these requests. Interestingly,

Google Exhibit No. 1033
Page 002

Google Exhibit No. 1063
Page 1183

recent development in the so-called "content delivery network" appears to have achieved some improvements in unicast-based on-demand streaming that are reminiscent of those we have seen with multicast-based live streaming. A content delivery network such as the one developed by Akamai Technologies[3] is essentially a load-balanced network of servers. The servers are deployed on the "edge" of the Internet, making them close to the end users. These "edge servers" communicate with a "source server" to update their content. A client requesting a streaming video is redirected dynamically to the closest edge server that is not overloaded. Like multicast-based video streaming, this decentralized service model can reduce "hot spots". In principle, a content delivery network can also help unicast-based live streaming with a possibly increased latency.

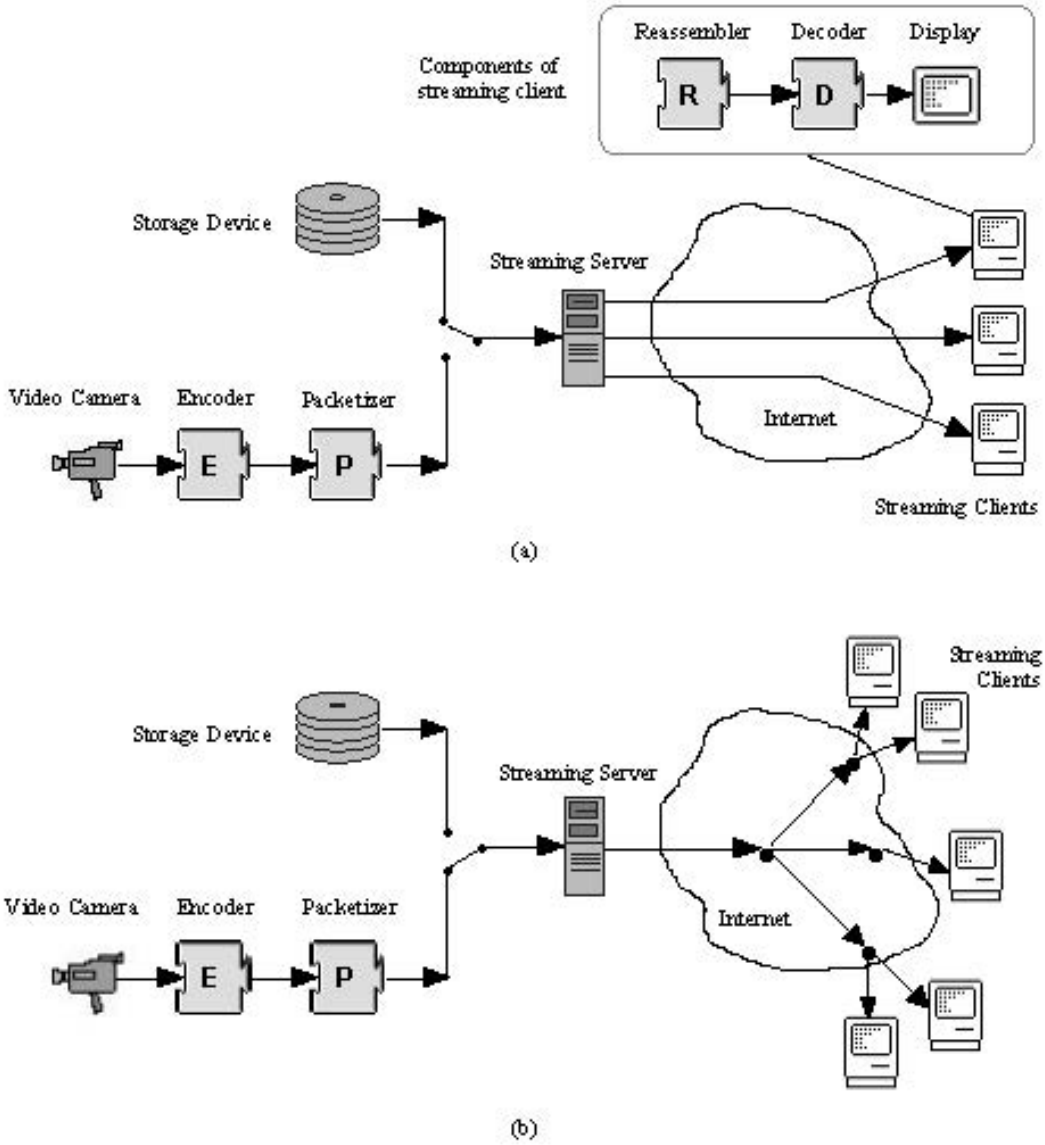


Figure 1. Video streaming system models. (a) Unicast video streaming, and (b) Multicast video streaming.

Google Exhibit No. 1033
Page 003

Google Exhibit No. 1063
Page 1184

2.2. The Internet Challenge to Video Streaming

The Internet provides a great challenge to video streaming. From a communication point of view the Internet is a shared channel with a massive number of heterogeneous components. This channel is extremely volatile over space and time in terms of communication capacity and quality. To see what it means to video streaming let us look at some data that we have collected from a real-life video streaming session.

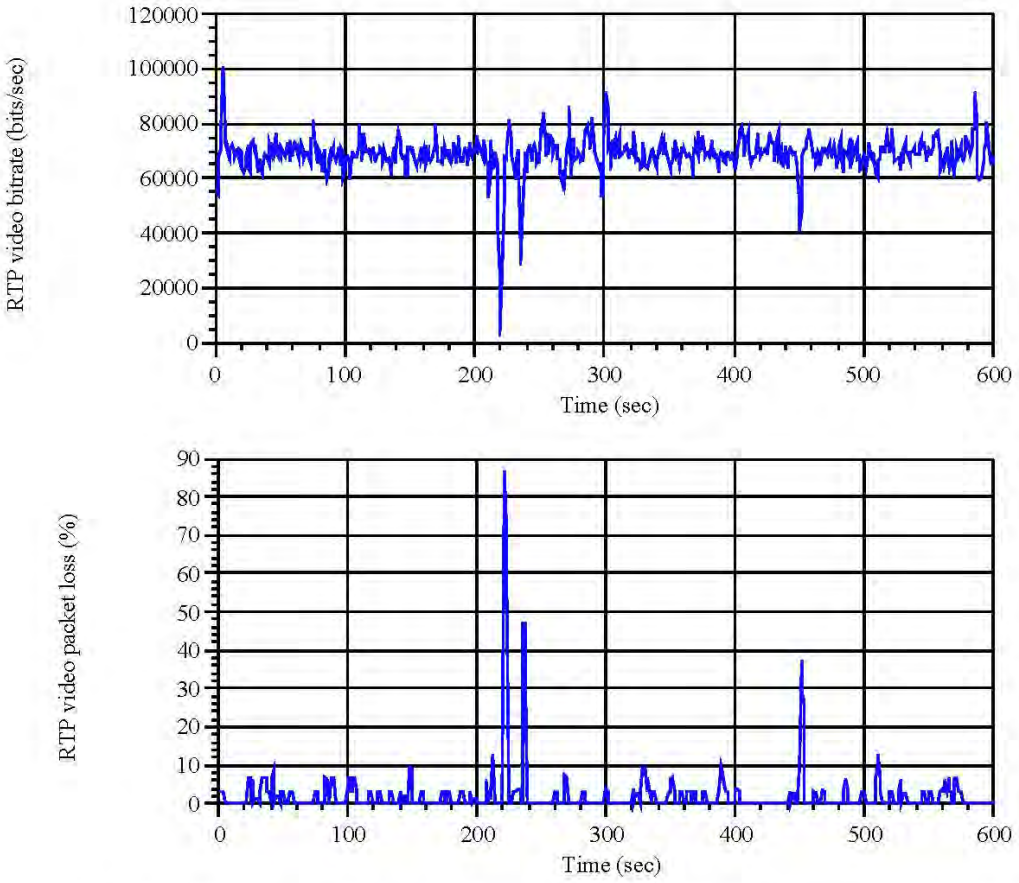


Figure 2. RTP video bitrate (top) and packet loss rate (bottom). The data was recorded at 10:08-10:18am on December 23, 1999, while watching BBC World live broadcast on QuickTimeTV from Sunnyvale, CA.

Figure 2 shows a 10-minute data log by a streaming client in Sunnyvale, California, during a session watching the BBC World channel on QuickTimeTV on December 23, 1999. The data shows the variations in bitrate and percentage of packet loss of the incoming RTP (Real-time Transport Protocol)[4] video stream. The client has signaled to the server that it had a dual ISDN connection with up to 112kbps in bandwidth. Knowing that the client's bandwidth was also shared by streaming audio, we observed that the bitrate of the incoming video fluctuated between 60 and 80kbps in general and the packet loss ranged from 0 to 10%. There are, however, a few exceptional instances where the bitrate dropped to as low as 2kbps and packet loss rate shot up as high as 85%. It is also interesting to observe from Figure 2 that there is a strong correlation between a sudden drop in bitrate and a burst of packet loss. This appears to agree with a common-sense logic about the relationship between congestion and packet loss.

Google Exhibit No. 1033
Page 004

Google Exhibit No. 1063
Page 1185

Figure 2 shows us all but a typical video streaming session when everything runs normally. There are also extreme cases. Many of us may have experience of watching streaming video that stalled completely for several times during a streaming session. In such cases, the bitrate of the incoming video is nil and the packet loss rate is 100%, making it totally worthless to watch. Generally speaking, the experience of streaming video isn't always as bad (or good) as that in our memory or in Figure 2. A person who was watching the same streaming video from another location might have had a totally different experience. The point is that the Internet channel variations are so volatile and space- and time-dependent that the observation can be different by every client on the network at every second. The data shown in Figure 2 and our experience about video streaming challenge us with two fundamental signal processing problems whose solutions can improve the performance of Internet video streaming dramatically. The first problem is concerned with how to adapt to the constantly changing bandwidth and congestion condition on the Internet. The second calls for solutions to coping with packet loss due to the best-effort service model of the current Internet. We will, for short, refer to these two problems as "rate adaptation", and "error control and recovery", respectively. Before we attempt to address these problems and evaluate a number of proposed solutions, we need to examine the constraints that have been imposed on any prospective solutions. We go back to the video streaming model and architecture of Section 2.1 and make a few important observations.

2.3. Video Streaming System Characteristics and Constraints

Many signal processing researchers are familiar with the video conferencing model thanks to the popularity of ITU H.261 and H.263 standards. In the following we review the characteristics of a video streaming system and point out its difference from video conferencing. For the comparison purpose we provide in Figure 3 a functional diagram of an IP-based peer-to-peer video conferencing system. As we have mentioned, we will focus on analyzing the unicast video streaming system.

(a) *Latency*. One difference between streaming and conferencing is that the former does not require two-way user interaction. As such the requirement on latency is more relaxed in streaming than in conferencing. Latency requirement commonly translates into a decoder/client buffer of certain size; which is often simulated by the encoder/server for rate control. When a video streaming session is initiated, the client waits for a few seconds for its buffer to be filled by the server. This process is called pre-roll. After the pre-roll the client starts playing the video. Obviously, the client's buffer has to be kept from underflow in order for the video to continue to play without interruption. Generally speaking, streaming can afford to use a much larger buffer than that can be considered in conferencing to absorb network jitters. However, latency and smooth playback must be balanced to create a good user experience, which becomes more important in streaming live video. It is common in commercial video streaming systems to pre-roll a few seconds of data into the decoder/client buffer.

(b) *Client-Server Model*. This makes a major distinction between the video streaming system of Figure 1(a) and the peer-to-peer conferencing system of Figure 3. In the video conferencing system of Figure 3, there is no server involved. Additionally, most components including encoder, decoder, packetizer, and reassembler at either end often reside on one piece of hardware, which is a desktop computer in the so-called desktop conferencing environment. The video streaming system in Figure 1(a), however, operates in a client-server environment. As such, a streaming server may be required to serve thousands of clients simultaneously. It is therefore necessary for the encoder to be separate from the server hardware so that the server hardware can be dedicated to serving the clients. For archived, on-demand video streaming, encoder and server are separate by default. In live streaming, encoder/server separation enables a more efficient "encode once, serve multiple times" service model. This model is almost always adopted in practice. A common implementation for live video streaming is to combine the encoder and packetizer to reside on one piece of hardware that broadcasts a stream to the server. The server simply "reflects" the stream to its clients. The client-server model and the nature of encoder/server separation make it unsuitable for many signal processing techniques that have been developed for video conferencing to be used in video streaming. We will come back to this observation later.

(c) *Server Load Restrictions*. As we have pointed out that a streaming video server may be required to serve thousands of clients simultaneously. When the number of clients connected to a server increases, the server experiences an elevated load due to increased processing and I/O demands. Eventually the server can get overloaded and the quality of service drops. While it is relatively well understood that a good signal processing solution should not require the server to do much processing, the requirement for I/O is often overlooked. In archived, on-demand video streaming, when the number of clients increases, heavy I/O activity, particularly disk access, often becomes the bottleneck in the streaming system. This ought to be kept in mind in designing a signal processing solution for video streaming.

Google Exhibit No. 1033
Page 005

Google Exhibit No. 1063
Page 1186

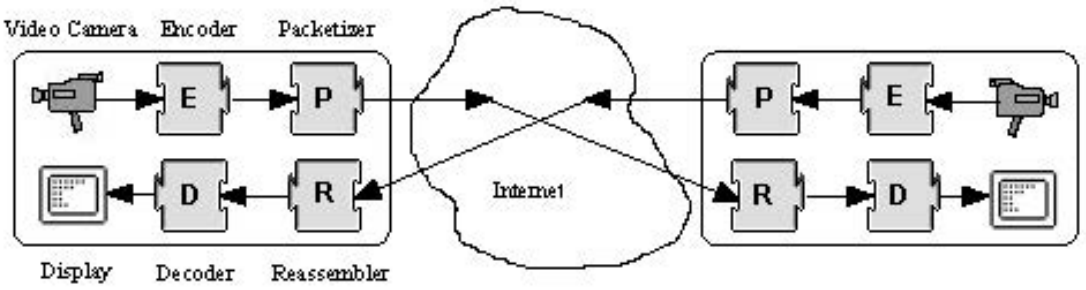


Figure 3. An IP-based peer-to-peer video conferencing system.

(d) *The Best-Effort Service Model.* The Internet is a best-effort service network. As such, each request to send a packet is honored by the Internet at best it can. Congestion on the Internet causes packet delay and loss. Normally, a packet is routed through multiple hops on its way to the destination. At each switching point linking the hops, if there are other packets that have arrived earlier, the incoming packet is queued before it can be processed by the router, resulting a queuing delay. If the router's queue is full, the incoming packet will be dropped, resulting a packet loss. Statistics have shown that packet delay and loss often occur in bursts because an overloaded router and the affected route are likely to stay congested for some time. In video streaming, each packet is scheduled to be played at certain time; a packet that has arrived later than the scheduled time is useless and will be discarded. For this reason, we don't distinguish packets that are dropped by a router due to congestion or by the streaming client because they come too late to be used. More generally, taking a signal processing approach, we can view the Internet as a "black box". A streaming client from outside the black box can observe an effective bandwidth and packet loss rate without having to know what is going on inside the black box.

(e) *Packetization.* A packetizer takes an encoded video bitstream and packetizes it for transmission over a transport protocol such as RTP. One of the most important aspects in designing a packetizer is to determine how to fragment the video bitstream. A well-designed fragmentation scheme with associated header information can help contain error in case of a packet loss. To date many video payload formats have been specified or proposed for RTP including those for the popular H.261, H.263 and the MPEG family. These payload formats offer fragmentation schemes of different granularities, such as Group-Of-Block(GOB) based or Macro-Block (MB) based. Besides fragmentation scheme it is also important to choose the size of a packet. On the one hand, small packets could control the adverse effect of a packet loss; on the other hand, it is desirable to send large packet to reduce packet header overhead. When streaming video over RTP, it is common to packetize the data for the maximum transmission unit (MTU) of a local area network (LAN). For the Ethernet, the MTU is about 1500 bytes, or 12000 bits. However, other restrictions may apply. For example, a RTP video packet is required to contain at most a single video frame. In other words, a new frame cannot start in the middle of a packet, though a frame can be split into multiple packets.

(f) *Back-channel.* A back-channel in a video streaming system is a communication channel established for a streaming client to report its state to and/or request services from the server. In RTP streaming, implementation of back-channels is facilitated by the Real-time Transport Control Protocol (RTCP) that is part of the RTP specifications.[4] The RTCP defines a type of packet for receiver report (RR). A RR packet includes such information as fraction of packets lost since the previous RR packet, cumulative number of packets lost, and interarrival jitter. It also supports profile- or application-specific extensions for including more information from the receiver. In addition to using the RR packets, it is also possible to define customized RTCP packet to carry feedback information from a streaming client. While the information about effective bandwidth and the state of packet loss can be fed back and useful to the streaming server, it is a common mistake to assume that the video encoder can always take advantage of the back-channel. We will discuss more on this in next Section.

Google Exhibit No. 1033
Page 006

Google Exhibit No. 1063
Page 1187

3. SIGNAL PROCESSING SOLUTIONS FOR VIDEO STREAMING

3.1. Solutions That Do Not Fit

After reviewing the architecture and characteristics of a video streaming system in the previous Section, it becomes clear that some of the methods that have been developed for video conferencing and other applications such as wireless video are not quite applicable to video streaming. We comment on these "unfit" solutions before moving onto evaluating a family of viable solutions.

3.1.1. Feedback-Based Encoder Adaptation

Many signal processing researchers have explored the idea of having a video encoder adapt to the changing conditions in bandwidth and packet loss. It is assumed that a back-channel exists and will feed the necessary information back to the encoder which in turn adjusts its behavior in response to the bandwidth and packet loss conditions. This idea has motivated the design of the RTCP channel along with RTP.[4-6] To date, many ways of encoder adaptation have been proposed and studied. Bolot, Turletti, and Wakeman[5] proposed an adaptive rate control mechanism by which a video encoder adjusts its output bitrate based on the feedback information from the receivers. There has been continued work along this direction.[7-10] In the same spirit as feedback-based rate control, Steinbach, Färber, and Girod[11] proposed to adapt the coding mode selection at the encoder to the error conditions at the receiver for wireless video. The idea is that the error damaged MBs may be tracked by the sender through a receiver's feedback, and a rapid error recovery can be achieved by INTRA refreshing corresponding MBs in the next frame. More recently, the scheme of adaptive coding mode selection was proposed for Internet video based on a packet loss probability model that must be estimated at the receiver and fed back to the sender[10]. Similar to adaptive INTRA MB refreshing and placement, it was proposed that a damaged reference picture could be dynamically replaced at the sender according to the back-channel signaling.[12] This mechanism is now supported in ITU H.263+ standard.[13] There are still more ways of encoder adaptation. For example, unequal error protection can be applied according to packet loss rate,[14] provided that a current estimate of the packet loss rate at a specific receiver is available.

Although the feedback-based encoder adaptation techniques that we have just reviewed can be effective in peer-to-peer video conferencing and wireless video, they don't fit in a real-world video streaming system. This is due to the encoder/server separation that we discussed in the previous Section. As we mentioned there (see Figure 1(a)), in live video streaming, for efficiency purpose the encoder will encode once, producing one bitstream for the server. Of course, the system can be configured to enable the encoder to generate a bitstream adapted to each client. However, such a design will not scale up to an industry-strength video streaming system. It is well-known that real-time video compression is computationally expensive; this is one reason why the encoder is separate from the server. Considering the server may be serving thousands of clients simultaneously, it is simply unrealistic to expect the encoder to compress and output thousands of different bitstreams that are adapted to each client. In on-demand video streaming, it is obvious that encoder adaptation can not be applied because the video content is compressed off-line.

3.1.2. Complex Server Involvement

Now that we know the encoder in a video streaming system cannot afford to adapt to each client's condition. One may wonder if the server can adapt. Generally speaking, the server does have some flexibility and capability to manipulate the video bitstream to adapt to a client's needs. In fact, all viable solutions to rate adaptation to be reviewed in Section 3.2 rely on the server to do some kind of processing. The exact requirements for the server involvement depend on a specific adaptation technique, but for the obvious reason the server should not be tasked with complex processing on a per client basis.

An example of tasks that are too complex is for the server to convert the bitrate of a pre-encoded video bitstream dynamically by transcoding. Although a transcoder is often much less complex than a full encoder, a considerable amount of computing such as bit re-allocation, and motion vector search or refinement may be required.[15-17] A server is unlikely to scale up to serving a large number of clients if it performs transcoding for each client. A simplified version of transcoding, called dynamic rate shaping (DRS),[18,19] requires even less processing because it achieves bitrate reduction by selectively dropping quantized transform coefficients or blocks, or by re-quantization of the blocks.[20] Despite the significant reduction in computational complexity at the expense of introducing a quality problem known as "drift", we still feel that DRS is too complex for a server to do on a per client basis. This is because DRS requires the server to have knowledge about a specific codec to process the bitstream.

Google Exhibit No. 1033
Page 007

Google Exhibit No. 1063
Page 1188

3.1.3. Intraframe Error Control and Concealment at Low Bitrate

There has been much work on designing error resilient bitstream for error control. For example, the emerging MPEG-4 video standard supports such mechanisms as resynchronization marker, data partition, and reversible VLC for intraframe error control. When these features are implemented in a compressed bitstream, the decoder can regain synchronization in case of bit errors, and continue to decode a video frame even if a portion of the frame is damaged or lost. Furthermore, error concealment techniques can be used to recover a damaged or lost region within a video frame to certain degree.

Obviously, any intraframe error control and concealment technique would become useless if a whole video frame is lost completely. Unfortunately, this is often the case at low bitrate when we attempt to packetize a video bitstream for the MTU of LAN. As we mentioned in Section 2.3, the MTU for the Ethernet is 12000 bits. Consider that we compress video for a target bitrate of 36kbps at a frame rate of 4fps. This would be suitable for users with a 56kbps modem connection because we need to leave some room for audio and bandwidth fluctuation. At this target bitrate and frame rate, the average size of a video frame is 9000 bits, which is less than the MTU of 12000 bits. In practice, a motion compensated frame is often much smaller in size than 9000 bits in low bitrate video streaming. This implies that when a packet is lost, it is most likely that a whole video frame is lost. Therefore, no matter how well intraframe error control and concealment techniques are implemented in the video bitstream and at the decoder, they are of little help in low bitrate video streaming.

3.2. Solutions for Rate Adaptation

As we discussed in Section 3.1, the video encoder cannot afford to adapt to individual clients, and the streaming server is constrained to doing only light processing on a per client basis. The solutions to be reviewed below for rate adaptation fit these requirements. Additionally, these solutions do not require the server to understand a compressed video bitstream. From a server design point of view, it is advantageous to make the server codec-independent, or even media-independent. A good example of media-independent RTP streaming server is Apple's Darwin QuickTime Streaming Server (QTSS). The Darwin QTSS is open-sourced and available from Apple.[21]

3.2.1. Dynamic Stream Switching

Video streaming using multiple bitstreams encoded at different bitrates has been supported in commercial streaming systems such as Apple's QuickTime, RealNetworks' RealSystem G2, and Microsoft Windows Media. RealSystem G2 was the first to support dynamic stream switching under the name of SureStream.[22,23] The idea is that the server can switch among multiple streams to serve a client with one that best matches the client's available bandwidth. The switching is dynamic, depending on the estimated available bandwidth reported by a client and the server's decision on whether it will respond. We want to note that in some research literature, video streaming using dynamic stream switching is called simulcast.[24] We prefer not to use this term because what we are describing here is different from the same word originated and used in the broadcasting industry.

In the SureStream implementation, three streams are prepared for a single target bitrate. It is worth noting that at the low end is a keyframe only stream encoded at approximately 50% of the target bitrate.[22] The bitrate of this keyframe only stream can be further reduced by dropping additional keyframes since there is no dependency among the keyframes. Furthermore, the keyframe only stream is a good fallback choice when the packet loss rate is high and there is a need to resynchronize.

Although being effective and requiring only a little server involvement, dynamic stream switching makes the encoding process more complex. The mechanism of making a switchable SureStream is not published, but one can think of some feasible approaches. For example, we can build some synchronization points to facilitate switching among the multiple streams. A keyframe shared by multiple streams would provide a convenient synchronization point, provided that the buffers of these streams can be synchronized. Besides making encoding process complicated, we note that dynamic stream switching requires redundant data storage but still has limited rate adaptability because the server can switch among only a small number of available streams.

3.2.2. Scalable Video Coding

Research in scalable video coding dates back to almost a decade ago.[25] A similar term, namely layered coding, is also used frequently to refer scalability, and has been popular in multicast community.[26,27] To date, support for scalable video coding

Google Exhibit No. 1033
Page 008

Google Exhibit No. 1063
Page 1189

can be found in modern video compression standards such as H.263+ and MPEG-4. There has also been continued research in this direction. See, for example, a few representative papers.[28-30] There are three types of scalability that can be built into a compressed video bitstream. They are *temporal scalability*, *spatial* (resolution) *scalability*, and (spatial) *quality scalability*. The last one is also called *SNR scalability* because the quality of a compressed picture is often measured in signal-to-noise ratio (SNR). Let's now look at what these different types of scalability can do for video streaming.

Temporal scalability is commonly achieved by including B-frames in a compressed video bitstream. B-frames are bidirectionally predicted from the forward and backward I- or P-frames, and not used as reference pictures for motion compensation in reconstructing other B- or P-frames. For this reason, a B-frame can be dropped without affecting the picture quality of other frames. However, dropping a B-frame will lower the frame rate or temporal quality of a video playback, giving the meaning of temporal scalability. Temporal scalability through B-frame placement is supported in all MPEG family video coding standards and H.263+. It should be noted that P-frames can also be dropped for rate reduction, but doing so will cause a decoder drift unless all P-frames are dropped. Use of temporal scalability for rate adaptation by dropping B- and P-frames at the server has been experimented in video streaming[23,31]. Although effective to some degree, it is found that temporal scalability alone is often inadequate to achieve desired rate reduction.

Spatial scalability is obtained by creating a multiresolution representation of each frame in a video bitstream. In spatial scalable coding, a base layer at a reduced resolution is first encoded, and an enhancement layer representing the difference between the original picture and the base layer is also encoded. It is worth noting that an enhancement layer can be split into multiple enhancement layers so that each one is built upon the previous ones at a lower resolution. Spatial scalability is supported in MPEG-4 and H.263+. There is a limited application of spatial scalability in video streaming. This is because that for good user experience the displayed frame resolution will not be changed during a video playback. So, even if the server could drop an enhancement layer and send only the base layer to a client in rate adaptation, the client would have to interpolate the resolution-reduced base layer to the full frame size for display. This makes spatial scalability a special case of SNR scalability that we are going to discuss below.

SNR scalability is similar to spatial scalability in that a base layer and one or more enhancement layers are encoded. The difference is that in SNR scalable coding the base layer is not subsampled to a lower resolution. Instead, the base layer is coarsely quantized and the difference between the base layer and the original picture is encoded into one or more enhancement layers. Spatial scalability can be viewed as a special case of SNR scalability if we consider an interpolated base layer in spatial scalable coding as the base layer in SNR scalable coding. The SNR or spatial scalability that we have just described assumes that an enhancement layer is either available and decoded as a whole or not available at all. It is not very effective for rate adaptation in video streaming for several reasons. First, if only one or two enhancement layers are used, we have very limited rate adaptability that is similar to the situation in dynamic stream switching. Second, if more enhancement layers are used, overhead will be up and coding efficiency will drop significantly. In fact, a recent study[32] on H.263+ scalability performance suggested that even with only two enhancement layers, the loss in coding efficiency of SNR scalable coding in comparison to multiple bitstream coding becomes unacceptable (about 1-1.5dB loss per layer) at low bitrate. This explains why the industry chose to use dynamic stream switching in commercial video streaming systems instead of scalable video coding options that are available in the modern video compression standards.

In recent years, research in scalable coding has centered on developing coders that can generate an "embedded bitstream". Much of the work was inspired by the design and excellent coding efficiency of the EZW[33] and SPIHT[34] coders for still images. An embedded bitstream is such that bits are packed in the order of "significance", a measure that is often tied to the magnitude of image coefficients in the transform domain. An intuitive way of embedding would be to order image coefficients by their magnitude and pack them in a bitstream in descending order of bit-planes. An embedded bitstream can be truncated from the end to match a reduced target bitrate. In EZW[33] and SPIHT,[34] the embedding is so fine-granular that one can scale the size of a bitstream in byte accuracy. For this reason, an embedded bitstream is said to have *fine-granularity scalability* (FGS). Embedding a video bitstream can be a straightforward extension to embedded image coding if a 3-D video transform is used.[35,36] It will be more complicated if a motion compensated residual coding is employed. This is because that any truncation or scaling performed on a reference picture will result in a decoder drift. A common solution to making a drift-free embedded video bitstream is to encode two layers as we discussed before in spatial and SNR scalable coding. The base layer, which is not embedded, will be used as a reference for motion compensation, while the enhancement layer will be embedded in fine granularity. This is a compromise, for the resulting bitstream is not fully embedded, but it still offers a higher rate adaptability than non-embedded coding using multiple layers.

Google Exhibit No. 1033
Page 009

Google Exhibit No. 1063
Page 1190

More recently, applications of FGS to video streaming have been reported.[35,37,38] In October 1998, MPEG started a new work item on fine granular video scalability.[39] The work has since moved forward under the name of MPEG-4 FGS and converged to a solution based on bit-plane coding of DCT coefficients for enhancement layers.[40,41] There is also work on defining an MPEG-4 streaming video application profile based on MPEG-4 FGS.[42,43] The advantages of FGS for rate adaptive video streaming is obvious. It has excellent rate adaptability yet the requirement for server involvement is reasonably small. There remain issues to be solved, such as how to minimize overhead in layered FGS coding, but we feel that FGS, particularly when combined with temporal scalability, provides a promising solution to rate adaptation.

3.3. Solutions for Error Control and Recovery

Error resilience can be built into streaming video to control the spread of error in case of a packet loss. A large number of error resilience tools have been developed in the last decade and now included in the latest video coding standards, such as H.263+ and MPEG-4. However, many of these tools were developed with mobile and wireless video communications in mind. See, for example, a few recent reviews on this topic.[44-46] In the following we evaluate the effectiveness of these tools for error control in video streaming. We look at solutions for error control in source and channel coding on the encoder/server side, and error recovery and concealment on the decoder/client side.

3.3.1. Error Control in Source Coding

One approach to error resilient video coding is to add some error control mechanisms during source coding and make them part of a compression codec design. Most modern image and video compression standards provide one or more error resilient modes that are supported by a special bitstream syntax and some coding tools. For example, H.263+ has several error resilience modes at source coding level, including Slice-Structured Mode, Independent Segment Decoding Mode, and Reference Picture Selection (RPS) Mode.[13] MPEG-4 on the other hand provides several error resilience tools, including resynchronization markers, data partitioning, and reversible variable length codes (RVLC)[47]. The error resilience performance of H.263+ and MPEG-4 has been reported in several papers.[13,48,49] Here we want to note that all of these tools except the RPS mode in H.263+ are designed for intraframe error control. They are most suitable for controlling errors that result from loss of small random data blocks within a video frame. On the other hand, they are not very effective in controlling errors that result from a packet loss in video streaming. This is especially true for low bitrate RTP video streaming as we discussed in Section 3.1.3.

An interesting exception in the above error resilience tools is the RPS mode supported in H.263+.[13] The RPS mode may be implemented in different ways and they don't necessarily provide a good solution to error control for video streaming. For example, one possible implementation of the RPS mode is reviewed in Section 3.1.1 and concluded to be unfit to the video streaming architecture. Nevertheless, a technique known as "video redundancy coding" (VRC)[13] can be implemented in RPS mode and provides a good interframe error control. In VRC, multiple threads of predicted video frames are synchronized at a common keyframe, but otherwise the threads are independent of each other. For example, we can choose all odd-numbered frames to form one thread and even-numbered frames to form another thread. Both threads start at a common keyframe, say frame #0, and ends at another keyframe, say #20. During video streaming, if one packet is lost and a video frame is completely lost, we would have one damaged thread. The decoder might choose to discard the damaged thread and the video playback could continue at half of the original frame rate. On the other hand, if all packets are received error-free at the decoder, the two threads of frames can be combined to give the original sequence.

Similar in spirit to VRC, multiple description coding (MDC) has received some attention recently in image coding research community.[50-55] The idea behind MDC is that if an image is a superposition of several independent descriptions, losing one or more descriptions would only degrade the quality of the reconstructed image at the decoder, but not destroy the decoding process. A simple, intuitive example of MDC is to regroup the pixels in an image into four sub-images by subsampling, and compress them independently. Clearly, if one, two, or even three sub-images are lost, we would still be able to reconstruct an image, albeit in lower quality. Although conceptually appealing, MDC has not been demonstrated to be practically useful to video streaming. There remain issues in complexity and coding efficiency that need to be addressed for MDC.

3.3.2. Error Control in Channel Coding

To enhance the error resilience of an already compressed video bitstream, error control codes can be added to the bitstream. This is commonly referred to as error control coding or channel coding. In video streaming, packetization can also be

Google Exhibit No. 1033
Page 010

Google Exhibit No. 1063
Page 1191

considered part of channel coding because a well designed packetization scheme can isolate errors caused by a packet loss so that the video bitstream can be reassembled from an incomplete set of packets.

Forward Error Correction (FEC) is a classic channel coding technique that has been widely used. It can be either media independent or dependent, and implemented at different levels. In media independent FEC for packet networks, the protection is exercised at the packet level. The implementation and operation of FEC do not depend on the content of the packets, and the error recovery means an exact replacement of a lost packet. Media dependent FEC is often designed for a compressed video bitstream before packetization. For example, H.263+ specifies a FEC mode for its own bitstream.[13] Traditional FEC, either media independent or dependent, is not very effective for video streaming for two reasons. First, FEC adds redundant information to achieve protection. Recalling that our observation from the data in Figure 2 suggests a strong correlation between congestion and packet loss, it seems to be counter-productive for us to send more data to combat packet loss. Of course, we could maintain the same output bitrate as without FEC by reducing the bandwidth for the video payload; the result would be lower video quality. A more important reason for the ineffectiveness of FEC is due to its open-looped nature. It is difficult to determine *a priori* the right amount of protection for a constantly changing packet loss pattern. In practice, the amount of protection exercised is often not enough for occasions of high packet loss rate and excessive for other time of low packet loss rate.

It is possible to assign unequal amounts of FEC to different segments of video data by a technique called Priority Encoding Transmission (PET).[56] The PET algorithm requires a user to specify the priorities of video segments which are also user defined. In Section 3.1.1, we commented on one method[14] that assigns unequal FEC according to an estimated packet loss rate. Another way of specifying unequal FEC is by the importance of data segments.[57] Unequal FEC based on the content of data can be considered a case of joint source-channel coding which has been identified as an important area for continued research.[58] We feel that this is a good approach that can be combined with the FGS coding described in Section 3.2.2.

3.3.3. Error Concealment

Error concealment by postprocessing can be performed at the decoder to restore or conceal the portion of the data that is damaged or lost. Many error concealment techniques have been developed, ranging from simple operations that just copy data from the previous frame to complex processes that compute regularized image restoration. A good review on these techniques is presented in a recent paper.[44]

Most error concealment techniques are based on an assumption that a video image and its motion field are smooth. For block-based coders, this assumption implies that the pixel values and the motion vector associated each coding block vary smoothly across the image. It has been demonstrated that when data corruption occurs in randomly distributed blocks or macroblocks, error concealment based on the smoothness assumption works well. Unfortunately, errors resulting from a packet loss rarely appear in randomly distributed blocks unless the blocks are interleaved randomly. It is more likely that many rows of blocks will be lost when a packet loss happens. In low bitrate video streaming, this is almost always the case, and very often all blocks in a frame are lost as we discussed in Section 3.1.3. When a large number of consecutive rows of blocks are lost, spatial or intraframe error concealment techniques do not work well. This is because the smoothness assumption does not hold for the block loss pattern we have. There is only a weak correlation between blocks that are far apart. Indeed, our experience has been that the simplest error concealment technique (i.e., copying data from the previous frame) provides the best tradeoff between enhanced visual quality and decoder complexity.

4. CONCLUSION

Research in scalable video coding appears to be maturing by the evidence that scalability options are now supported in standards such as H.263+ and MPEG-4. Nevertheless, the industry opt to take a brute-force approach to rate adaptation by using multiple bitstreams encoded at different bitrates. One wonders why. We tried in this paper to explain the reasons behind this kind of observations. The approach we take is to first identify the most challenging problems in Internet video streaming from a signal processing perspective. We see on top of our list are two types of problems, namely rate adaptation and error control and recovery, which demand good signal processing solutions. There are already a large number of signal processing techniques developed for video conferencing, an application that many signal processing researchers are familiar with. We have an opportunity to extend some of the work in video conferencing to video streaming.

However, although video streaming and conferencing share some core technologies such as video compression, the difference between the two systems is often overlooked. We reviewed carefully the characteristics of a video streaming

Google Exhibit No. 1033
Page 011

Google Exhibit No. 1063
Page 1192

system and architecture, and the constraints they impose on prospective signal processing solutions. Such a review is helpful because it makes us realize that some of the existing and proposed solutions, such as those employing feedback-based encoder adaptation or requiring complex server involvement, are simply unfit in the video streaming system and architecture. Among the viable solutions, we commented on the pros and cons of different approaches, and concluded that FGS combined with temporal scalability coding could provide a promising solution for rate adaptation. As for error control and recovery, we found that the existing solutions are inadequate. Particularly, we found that there exist few good solutions that are designed for error control and concealment in such circumstances that a large number of, or all, blocks in a video frame are lost. Unfortunately, this is often the case in video streaming when a packet loss happens, especially at low bitrate.

It must be pointed out that we have looked at only signal processing solutions for video streaming. There are other solutions that do not fall into the scope of signal processing. One example is the deployment of content delivery networks such as Akamai's. Even within the scope of signal processing, the coverage of our review is almost certain to be incomplete. Nonetheless, we hope this paper serves two purposes: for application developers to learn what signal processing solutions are available for video streaming, and for signal processing researchers to better understand video streaming systems and architecture, and their associated constraints on signal processing solutions.

REFERENCES

1. S. McCanne, "Scalable multimedia communication using IP multicast and lightweight sessions," *IEEE Internet Computing*, Vol. 2, No.2, pp. 33-45, Mar. 1999.
2. Stardust.com, Inc, "The evolution of multicast," IP Multicast Initiatives White Paper, December 1999. http://www.ipmi.org/
3. Akamai Technologies, Inc. http://www.akamai.com/
4. H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson, "RTP: A transport protocol for real-time applications," RFC 1889, Internet Engineering Task Force (IETF), Jan. 1996.
5. J.-C. Bolot, T. Turletti, and I. Wakeman, "Scalable feedback control for multicast video distribution in the internet," in *SIGCOMM Symposium on Communications Architectures and Protocols*, (London, England), pp. 58-67, Aug. 1994.
6. I. Busse, B. Deffner, and H. Schulzrinne, "Dynamic QoS control of multimedia applications based on RTP," *Computer Communications*, Jan. 1996.
7. J-C. Bolot, T. Turletti, "Experience with rate control mechanisms for packet video in the Internet," *Computer Communication Review*, Vol. 28, No. 1, Jan. 1998.
8. Y.J. Chung, J.W. Kim, and C.-C.J. Kuo, "Network friendly video streaming via adaptive LMS bandwidth control," in *Applications of Digital Image Processing XXI, Proc. SPIE*, vol. 3460, pp. 448-456, July 1998.
9. Y.J. Chung, J. Kim, and C.-C.J. Kuo, "Real-time streaming video with adaptive bandwidth control and DCT-based error concealment," *IEEE Trans. CAS II: Analog and Digital Signal Processing*, Vol. 46, No. 7, pp. 951-956, July 1999.
10. F. Le Léannec, F. Toutain, C. Guillemot, "Packet loss resilient MPEG-4 compliant video coding for the Internet," *Signal Processing: Image Communications*, Vol. 15, pp. 35-56, Sept. 1999.
11. E. Steinbach, N. Färber, and B. Girod, "Standard compatible extension to H.263 for robust video transmission in mobile environments," *IEEE Trans. CAS for Video Technol.*, Vol. 7, No. 6, pp. 872-881, Dec. 1997.
12. S. Fukunaga, Y. Matsumura, and T. Nakai, "Error resilient video coding controlled by backward channel signaling," *Signal Processing: Image Communications*, Vol. 14, pp. 531-540, May 1999.
13. S. Wenger, G. Knorr, J. Ott, and F. Kossentini, "Error resilience support in H.263+," *IEEE Trans. Circuits and Sys. for Video Tech.*, Vol. 8, No. 7, pp. 867-877, Nov. 1998.
14. U. Horn, K. Stuhlmüller, M. Link, and B. Girod, "Robust Internet video transmission based on scalable coding and unequal error protection," *Signal Processing: Image Communications*, Vol. 15, pp. 77-94, Sept., 1999.
15. H. Sun, W. Kowk, and J.W. Zdepski, "Architectures for MPEG compressed bitstream scaling," *IEEE Trans. CAS for Video Technol.*, Vol. 6, No. 2, pp. 191-199, April 1996.
16. G. Keesman, R. Hellinghuizen, F. Hoeksema, and G. Heideman, "Transcoding of MPEG bitstreams," *Signal Processing: Image Commun.*, Vol. 8, pp. 481-500, Sept. 1996.
17. P.A.A. Assuncao and M. Ghanbari, "A frequency-domain video transcoder for dynamic bit-rate reduction of MPEG-2 bit streams," *IEEE Trans. CAS for Video Technol.*, Vol. 8, No. 8, pp. 953-967, Dec. 1998.
18. S., Jacobs and A. Eleftheriadis, "Streaming video on the Web using dynamic rate shaping," *J. Visual Commun. Image Representation*, Vol. 9, No. 3, pp. 211-222, Sept. 1998.
19. W. Zeng and B. Liu, "Rate shaping by block dropping for transmission of MPEG-precoded video over channels of dynamic bandwidth," in *Proc. ACM Multimedia'96*, pp. 385-393, Nov. 1996.

Google Exhibit No. 1033
Page 012

Google Exhibit No. 1063
Page 1193

20. Y. Nakajima, H. Hori, and T. Kanoh, "Rate conversion of MPEG coded video by re-quantization process," in *Proceedings of IEEE Intl. Conf. Image Proc.*, Washington, DC, Oct. 1995, pp. 408-411.
21. Darwin Streaming Server Project, http://www.publicsource.apple.com/projects/streaming/
22. A. Lippman, "Video coding for multiple target audiences," in *Visual Communications and Image Processing'99, Proc. SPIE*, Vol. 3653, pp. 780-782, Jan. 1999.
23. "SureStream™–Delivering superior quality and reliability," RealNetworks White Papers, http://www.realnetworks.com/devzone/library/whitepapers/surestrm.html
24. B. Girod, N. Färber, and K. Stuhlmüller, "Internet video streaming," in: F. De Natale and S. Pupolin (eds.), *Multimedia Communications*, Springer, 1999, pp. 547-556.
25. K.M. Uz, M. Vetterli, and D. LeGall, "Interpolative multiresolution coding of advanced television with compatible subchannels," *IEEE Trans. CAS for Video Technol.*, Vol. 1, pp. 86-99, Mar. 1991.
26. S. McCanne, V. Jacobson, and M. Vetterli, "Receiver-driven layered multicast," *ACM SIGCOMM'96*, Stanford, CA, Aug. 1996.
27. S. McCanne, M. Vetterli, and V. Jacobson, "Low complexity video coding for receiver driven layered multicast," *IEEE J. Selected Areas in Communications*, Vol. 15, No. 6, pp. 983-1001, Aug. 1997.
28. D. Taubman and A. Zakhor, "Multi-rate 3-dimensional subband coding of video," *IEEE Trans. Image Proc.*, Vol. 3, pp. 572-588, Sept. 1994.
29. J.Y. Tham, S. Ranganath, and A.A. Kassim, "Highly scalable wavelet-based video codec for very low bit-rate environment," *IEEE J. Selected Areas in Communications*, vol. 16, No. 1, pp. 12-27, Jan. 1998.
30. K. Chen and E.J. Delp, "Wavelet based rate scalable video compression," *IEEE Trans. CAS for Video Technol.*, Vol. 9, No. 1, pp. 109-22, Feb. 1999.
31. K. Sripanidkulchai and T. Chen, "Network-adaptive video coding and transmission," in *Visual Communications and Image Processing'99, Proc. SPIE*, Vol. 3653, pp. 854-861, Jan. 1999.
32. L. Yang, F.C.M. Marins, and T.R. Gardos, "Improving H.263+ scalability performance for very low bit rate applications," in *Visual Communications and Image Processing'99, Proc. SPIE*, Vol. 3653, pp. 768-779, Jan. 1999.
33. J.M. Shapiro, "Embedded image coding using zerotrees of wavelet coefficients," *IEEE Trans. Signal Processing*, Vol. 41, No. 12, pp. 3445-3462, Dec. 1993.
34. A. Said and W.A. Pearlman, "A new, fast and efficient image codec based on set partitioning in hierarchical trees," *IEEE Trans. CAS for Video Technol.*, Vol. 6, pp. 243-250, June 1996.
35. W.-T. Tan and A. Zakhor, "Real-time Internet video using error resilient scalable compression and TCP-friendly transport protocol," *IEEE Trans. Multimedia*, Vol. 1, No. 2, pp. 172-186, June 1999.
36. B.-J. Kim, Z. Xiong, W.A. Pearlman, and Y.S. Kim, "Progressive video coding for noisy channels," Journal of Visual Communication and Image Representation, Vol.10, No. 2, pp.173-185, June 1999.
37. W. Li, "Bit-plane coding of DCT coefficients for fine granularity scalability," MPEG contributions, m3989, Atlantic City, NJ, Oct. 1998.
38. H. Radha, Y. Chen, K. Parthasarathy, and R. Cohen, "Scalable Internet video using MPEG-4," *Signal Processing: Image Communications*, Vol. 15, No. 1-2, pp. 95-126, Sept. 1999.
39. Y. Chen, C. Dufour, J.-F. Vial, A. Zakhor, W. Li, J. Liang, and B. Schuster, "Description of mini-experiment of fine granular video scalability," MPEG contributions, m3881, Atlantic City, NJ, Oct. 1998.
40. ISO/IEC JTC1/SC29/WG11, "Text of ISO/IEC 14496-2 FGS Amendment Working Draft 3.0," MPEG Document N3095, Maui, USA, Dec. 1999.
41. W. Li, F. Ling, and X. Chen, "Fine granularity scalability in MPEG-4 for streaming video," to appear in *Proceedings of IEEE International Symposium on Circuits and Systems (ISCAS'2000)*, Geneva, Switzerland, May 28-31, 2000.
42. W. Li, et al, "Streaming video profile," MPEG Contributions, m4990, Melbourne, Australia, Oct. 1999.
43. W. Li, el al, "Streaming video profiles based on fine-granularity-scalability," MPEG Contributions, m5582, Maui, USA, Dec. 1999.
44. Y. Wang, and Q. Zhu, "Error control and concealment for video communications: a review," *Proc. IEEE*, Vol. 86, No. 5, pp. 974-997, May 1998.
45. J. Villasenor, Y.-Q. Zhang, and J. When, "Robust video coding algorithms and systems," *Proc. IEEE*, Vol. 87, No. 10, pp. 1724-1733, Oct. 1999.
46. B. Girod and N. Färber, "Feedback-Based Error Control for Mobile Video Transmission", *Proc. IEEE*, Vol. 87, No. 10, pp. 1707-1723, Oct. 1999.
47. R. Talluri, "Error resilient video coding in the MPEG-4 standard," *IEEE Commun. Mag.*, Vol. 36, pp. 112-119, June 1998.
48. R. Talluri, I. Moccagatta, Y. Nag, and G. Cheung, "Error concealment by data partitioning," *Signal Processing: Image Communication*, vol.14, No.6-8, pp. 505-518, May 1999.

Google Exhibit No. 1033
Page 013

Google Exhibit No. 1063
Page 1194

49. L. Ducla-Soares, and F. Pereira, "Error resilience and concealment performance for MPEG-4 frame-based video coding," *Signal Processing: Image Communication*, vol.14, No.6-8, pp. 447-472, May 1999.
50. Y. Wang, M.T. Orchard, and A.R. Reibman, "Multiple description image coding for noisy channels by pairing transform coefficients," in *Proc. First IEEE Workshop on Multimedia Signal Processing*, New York, NY, June 1997. Pp. 419-424.
51. D.M. Chung and Y. Wang, "Multiple description image coding using signal decomposition and reconstruction based on lapped orthogonal transforms," *IEEE Trans. CAS for Video Technol.*, vol. 9, No. 6, pp. 895-908, Sept. 1999.
52. S.D. Servetto, K. Ramchandran, V.A. Vaishampayan, and K. Nahrstedt, "Multiple description wavelet based image coding," submitted to *IEEE Trans. Image Proc.*, Nov. 1998.
53. V.K. Goyal, J. Kovacevic, R. Arean, and M. Vetterli, "Multiple, description transform coding of images," in *Proceedings of IEEE Conference on Image Processing*, Chicago, IL, Oct. 1998, pp. 674-678.
54. X. Yang and K. Ramchandran, "Optimal multiple description subband coding," in *Proceedings of IEEE Conference on Image Processing*, Chicago, IL, Oct. 1998, pp. 654-658.
55. R. Chellappa, and M. Srinivasan, "Multiple description subband coding," in *Proceedings of IEEE Conference on Image Processing*, Chicago, IL, Oct. 1998, pp. 684-688.
56. A. Albanese, J. Bloemer, J. Edmonds, M. Luby, and M. Sudan, "Priority encoding transmission, " *IEEE Trans. Inform. Theory*, Vol. 42, No. 6, pp. 1737-1747, Nov. 1996.
57. A.E. Mohr, E.A. Riskin, and R.E. Ladner, "Graceful degradation over packet erasure channels through forward error correction," in *Proc. Data Compression Conference*, Snowbird, UT, Mar. 1999, pp. 92-101.
58. Workshop Report: 1999 NSF Sponsored Workshop on Joint Source-Channel Research, Oct. 1999.

Google Exhibit No. 1033
Page 014

Google Exhibit No. 1063
Page 1195

Ex. 1033A

Signal Processing for Internet Video Streaming: A Review

Jian Lu*
Apple Computer, Inc.

ABSTRACT

Despite the commercial success, video streaming remains a black art owing to its root in proprietary commercial development. As such, many challenging technological issues that need to be addressed are not even well understood. The purpose of this paper is to review several important signal processing issues related to video streaming, and put them in the context of a client-server based media streaming architecture on the Internet. Such a context is critical, as we shall see that a number of solutions proposed by signal processing researchers are simply unrealistic for real-world video streaming on the Internet. We identify a family of viable solutions and evaluate their pros and cons. We further identify areas of research that have received less attention and point to the problems to which a better solution is eagerly sought by the industry.

Keywords: video streaming, video compression, Internet video, rate adaptation, packet loss, error control, error recovery.

1. INTRODUCTION

Video streaming is one of the most exciting applications on the Internet. It already created a new business known as Internet broadcast, or Intercast/Webcast. Although Internet broadcast is still in its infancy, we get a glimpse of the future of broadcasting where people can enjoy not only scheduled live or pre-recorded broadcast but also on-demand and personalized programs. Stirring more excitement is the idea that anyone will be able to set up an Internet TV station and broadcast his or her video programs on the Internet just like we publish our "homepage" on the Web today. While video streaming as a core technology behind a new business is a success story, it remains very much a black art owing to its root in proprietary commercial development. Indeed, to the author's knowledge, there has been no published research literature to date that offers a comprehensive review of video streaming architecture and system, and related design issues.

Even the definition of streaming is sketchy; it is often described as a process. Generally speaking, streaming involves sending media (e.g., audio and video) from a server to a client over a packet-based network such as the Internet. The server breaks the media into packets that are routed and delivered over the network. At the receiving end, the packets are reassembled by the client and the media is played as it comes in. A series of time-stamped packets is called a *stream*. From a user's perspective, streaming differs from simple file transfer in that the client plays the media as it comes in over the network, rather than waiting for the entire media to be downloaded before it can be played. In fact, a client may never actually download a streaming video; it may simply reassemble the video's packets and play the video as the packets come in, and then discard them. To a user the essence of streaming video may lie in an expectation that the video stream the user has requested should come in continuously without interruption. This is indeed one of the most important goals for which a video streaming system is designed.

Over the last decade, IP multicast and particularly the MBone have received considerable attention from the research community. IP multicast provides a simple and elegant architecture and service model that are particularly suitable for real-time multimedia communications. The emergence of the MBone, an experimental virtual multicast backbone, has offered a testbed for researchers around the world to develop and test a suite of applications known as MBone tools for real-time multimedia communications and collaborations. See, for example, a recent article by McCanne[1] for the evolution of the MBone and its applications. Many of these applications have been designed for audio and video conferencing and broadcasting using IP multicast. On November 18, 1994, the MBone reached a festive milestone amid great fanfare when Rolling Stones broadcast the first major rock band concert on the MBone.

Despite the promise demonstrated by the MBone and a logical reasoning that IP multicast would serve as a foundation for video streaming on the Internet, commercial deployment of multicast applications and services on the public Internet has been painfully slow by the Internet standard. The bulk of the streaming media traffic generated by three major commercial

* Correspondence: One Infinite Loop, MS: 302-3MT, Cupertino, CA 95014, USA; Email: jian@computer.org.

In *Image and Video Communications and Processing 2000*, Bhaskaran Vasudev, T. Russell Hsing, Andrew G. Tescher, Robert L. Stevenson, Editors, Proceedings of SPIE Vol. 3974 (2000) • 0277-786X/00/$15.00

Google Exhibit No. 1063
Page 1197

streaming systems (i.e., QuickTime, RealSystem, and Windows Media) on today's Internet consists of largely unicast streams. It is out of this paper's scope to discuss the unresolved issues that have slowed down the deployment of multicast applications and services; interested readers are referred to a recent white paper published by the IP Multicast Initiatives.[2] However, we want to note that even when multicast is fully deployed, a very large volume of streaming media traffic will remain to be unicast streams. This is because a large portion of the streaming media content is archived and played on-demand by users on the Internet. Multicast is of little help to on-demand streaming unless a large number of users from the same neighborhood of the Internet would request the same content simultaneously.

The purpose of this paper is to review several important issues in Internet video streaming from a signal processing perspective. For the reasons we have just mentioned above, our review will focus on unicast streaming while considering multicast as a possible extension. Drawing from the author's experience in developing one of the major commercial media streaming systems, we attempt to identify the most challenging problems in video streaming and evaluate a number of signal processing solutions that have been proposed; some of these solutions have already been implemented in experimental and/or commercial video streaming systems. To make the review useful to both researchers and application developers, we would like to clearly identify among the proposed solutions: what works in real world, what doesn't, and why? We want to note that although to a Web surfer streaming video commonly implies synchronized audio as well, video and audio are actually delivered in independent streams in a streaming session. We will not discuss streaming audio in this paper.

The rest of this paper is organized as follows. Section 2 introduces streaming system and architecture and describes what we believe to be the most challenging problems in video streaming. We also examine some important characteristics of the streaming system and the constraints that are imposed on any prospective signal processing solutions. Section 3 reviews a number of signal processing solutions to the video streaming problems in great detail. We first comment on some proposed solutions that are simply unrealistic for real-world video streaming on the Internet. Then, we evaluate a family of viable solutions, identify areas of research that have received less attention, and point to the problems to which a better solution is eagerly sought by the industry. Section 4 concludes the paper.

2. VIDEO STREAMING SYSTEM AND ARCHITECTURE

2.1. Video Streaming Model and Architecture

A schematic system diagram of Internet video streaming is shown in Figure 1 for both unicast and multicast scenarios. Figure 1(a) represents the unicast model for two types of video streaming services, *on-demand* or *live*. In on-demand streaming, pre-compressed, streamable video content is archived on a storage system such as a hard disk drive and served to a client on demand by the streaming server. In live streaming, a video stream coming out of a live source such as a video camera is compressed in real-time by the encoder. The compressed bitstream is packetized by the packetizer, and sent over the Internet by the streaming server. On the client side, the video bitstream is reassembled from the received packets by the reassembler, then decoded by the decoder, and finally rendered on a display. Noting that each client connecting to the server receives an individual stream, the disadvantages of the unicast streaming model can be explained. Clearly, the server load will increase proportionally to the number of clients connected to and served by the server. When a large number of clients try to connect to a server at the same time, the server gets overloaded, creating a "hot spot" on the Internet. Additionally, sending copies of the same data in individual streams to a large number of clients is obviously inefficient; it causes congestion on the network and deteriorates quality of services.

The multicast model of Figure 1(b) has many advantages over unicast in live video streaming (broadcasting) where a large audience could tune in for the broadcast at the same time. Instead of sending a stream to each client individually, the server sends out only one stream that is routed to one or more "group addresses". A client tunes in to a multicast group in its neighborhood to receive the stream. Since the server is decoupled from connecting to each client directly, server load does not increase with the number of receiving clients. In the case of delivering a stream to multiple groups, the stream on its route gets replicated and branched only at the fan-out points within the network. When combined with layered coding that will be discussed in Section 3, multicast provides a decentralized and highly efficient video streaming architecture.

Despite the clear advantages of multicast-based video streaming, it is employed in only a small fraction of today's Internet, primarily in Intranets within an institution or corporation. There are complex reasons for the slow deployment of multicast-based services in general.[2] Beyond deployment hurdles we note that a large volume of streaming video content is archived and streamed to users on demand. Multicast does not help very much in on-demand streaming where the service requests from users are random and highly desynchronized in terms of requested content and timing of these requests. Interestingly,

Google Exhibit No. 1063
Page 1198

recent development in the so-called "content delivery network" appears to have achieved some improvements in unicast-based on-demand streaming that are reminiscent of those we have seen with multicast-based live streaming. A content delivery network such as the one developed by Akamai Technologies[3] is essentially a load-balanced network of servers. The servers are deployed on the "edge" of the Internet, making them close to the end users. These "edge servers" communicate with a "source server" to update their content. A client requesting a streaming video is redirected dynamically to the closest edge server that is not overloaded. Like multicast-based video streaming, this decentralized service model can reduce "hot spots". In principle, a content delivery network can also help unicast-based live streaming with a possibly increased latency.

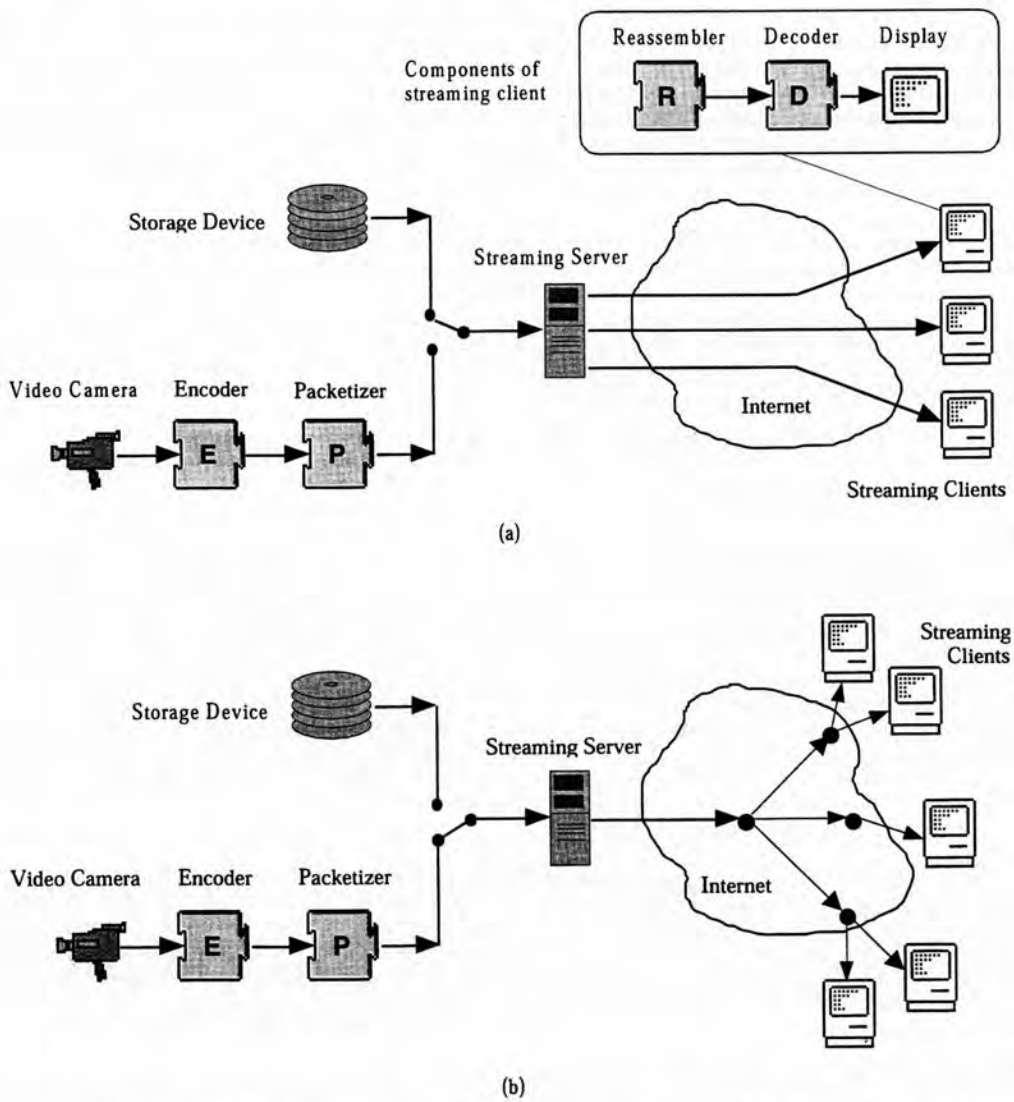


Figure 1. Video streaming system models. (a) Unicast video streaming, and (b) Multicast video streaming.

Google Exhibit No. 1063
Page 1199

2.2. The Internet Challenge to Video Streaming

The Internet provides a great challenge to video streaming. From a communication point of view the Internet is a shared channel with a massive number of heterogeneous components. This channel is extremely volatile over space and time in terms of communication capacity and quality. To see what it means to video streaming let us look at some data that we have collected from a real-life video streaming session.

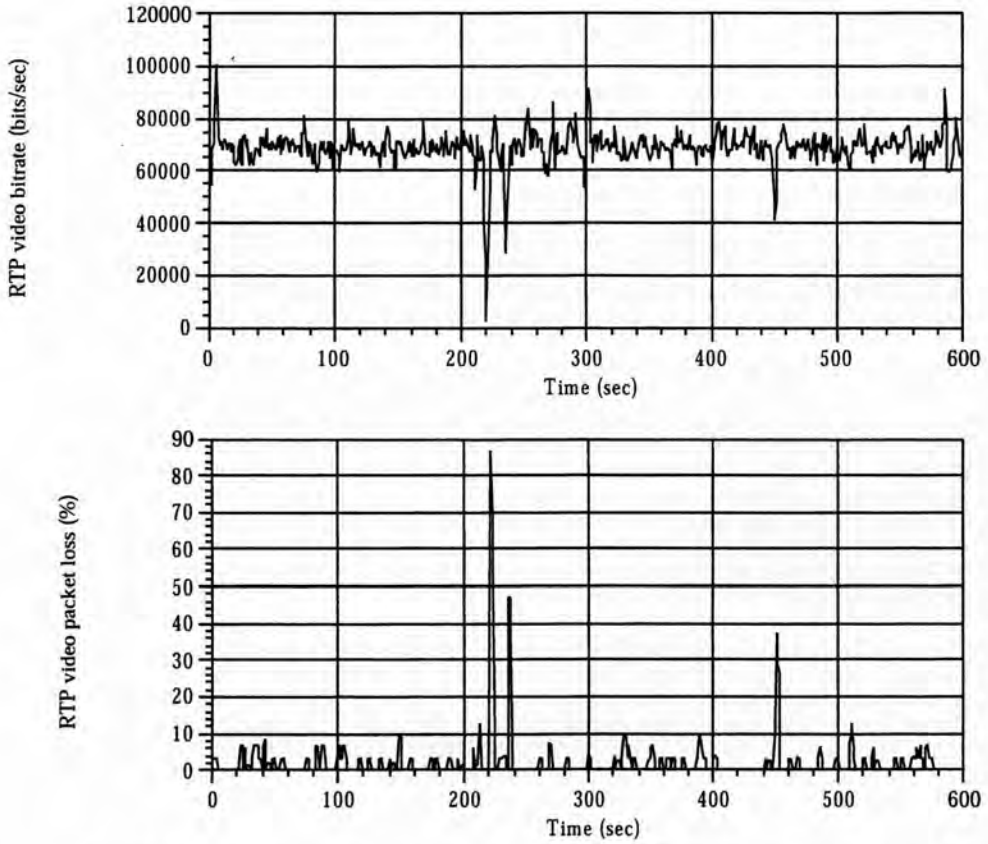


Figure 2. RTP video bitrate (top) and packet loss rate (bottom). The data was recorded at 10:08-10:18am on December 23, 1999, while watching BBC World live broadcast on QuickTimeTV from Sunnyvale, CA.

Figure 2 shows a 10-minute data log by a streaming client in Sunnyvale, California, during a session watching the BBC World channel on QuickTimeTV on December 23, 1999. The data shows the variations in bitrate and percentage of packet loss of the incoming RTP (Real-time Transport Protocol)[4] video stream. The client has signaled to the server that it had a dual ISDN connection with up to 112kbps in bandwidth. Knowing that the client's bandwidth was also shared by streaming audio, we observed that the bitrate of the incoming video fluctuated between 60 and 80kbps in general and the packet loss ranged from 0 to 10%. There are, however, a few exceptional instances where the bitrate dropped to as low as 2kbps and packet loss rate shot up as high as 85%. It is also interesting to observe from Figure 2 that there is a strong correlation between a sudden drop in bitrate and a burst of packet loss. This appears to agree with a common-sense logic about the relationship between congestion and packet loss.

Google Exhibit No. 1063
Page 1200

Figure 2 shows us all but a typical video streaming session when everything runs normally. There are also extreme cases. Many of us may have experience of watching streaming video that stalled completely for several times during a streaming session. In such cases, the bitrate of the incoming video is nil and the packet loss rate is 100%, making it totally worthless to watch. Generally speaking, the experience of streaming video isn't always as bad (or good) as that in our memory or in Figure 2. A person who was watching the same streaming video from another location might have had a totally different experience. The point is that the Internet channel variations are so volatile and space- and time-dependent that the observation can be different by every client on the network at every second. The data shown in Figure 2 and our experience about video streaming challenge us with two fundamental signal processing problems whose solutions can improve the performance of Internet video streaming dramatically. The first problem is concerned with how to adapt to the constantly changing bandwidth and congestion condition on the Internet. The second calls for solutions to coping with packet loss due to the best-effort service model of the current Internet. We will, for short, refer to these two problems as "rate adaptation", and "error control and recovery", respectively. Before we attempt to address these problems and evaluate a number of proposed solutions, we need to examine the constraints that have been imposed on any prospective solutions. We go back to the video streaming model and architecture of Section 2.1 and make a few important observations.

2.3. Video Streaming System Characteristics and Constraints

Many signal processing researchers are familiar with the video conferencing model thanks to the popularity of ITU H.261 and H.263 standards. In the following we review the characteristics of a video streaming system and point out its difference from video conferencing. For the comparison purpose we provide in Figure 3 a functional diagram of an IP-based peer-to-peer video conferencing system. As we have mentioned, we will focus on analyzing the unicast video streaming system.

(a) *Latency.* One difference between streaming and conferencing is that the former does not require two-way user interaction. As such the requirement on latency is more relaxed in streaming than in conferencing. Latency requirement commonly translates into a decoder/client buffer of certain size; which is often simulated by the encoder/server for rate control. When a video streaming session is initiated, the client waits for a few seconds for its buffer to be filled by the server. This process is called pre-roll. After the pre-roll the client starts playing the video. Obviously, the client's buffer has to be kept from underflow in order for the video to continue to play without interruption. Generally speaking, streaming can afford to use a much larger buffer than that can be considered in conferencing to absorb network jitters. However, latency and smooth playback must be balanced to create a good user experience, which becomes more important in streaming live video. It is common in commercial video streaming systems to pre-roll a few seconds of data into the decoder/client buffer.

(b) *Client-Server Model.* This makes a major distinction between the video streaming system of Figure 1(a) and the peer-to-peer conferencing system of Figure 3. In the video conferencing system of Figure 3, there is no server involved. Additionally, most components including encoder, decoder, packetizer, and reassembler at either end often reside on one piece of hardware, which is a desktop computer in the so-called desktop conferencing environment. The video streaming system in Figure 1(a), however, operates in a client-server environment. As such, a streaming server may be required to serve thousands of clients simultaneously. It is therefore necessary for the encoder to be separate from the server hardware so that the server hardware can be dedicated to serving the clients. For archived, on-demand video streaming, encoder and server are separate by default. In live streaming, encoder/server separation enables a more efficient "encode once, serve multiple times" service model. This model is almost always adopted in practice. A common implementation for live video streaming is to combine the encoder and packetizer to reside on one piece of hardware that broadcasts a stream to the server. The server simply "reflects" the stream to its clients. The client-server model and the nature of encoder/server separation make it unsuitable for many signal processing techniques that have been developed for video conferencing to be used in video streaming. We will come back to this observation later.

(c) *Server Load Restrictions.* As we have pointed out that a streaming video server may be required to serve thousands of clients simultaneously. When the number of clients connected to a server increases, the server experiences an elevated load duo to increased processing and I/O demands. Eventually the server can get overloaded and the quality of service drops. While it is relatively well understood that a good signal processing solution should not require the server to do much processing, the requirement for I/O is often overlooked. In archived, on-demand video streaming, when the number of clients increases, heavy I/O activity, particularly disk access, often becomes the bottleneck in the streaming system. This ought to be kept in mind in designing a signal processing solution for video streaming.

Google Exhibit No. 1063
Page 1201

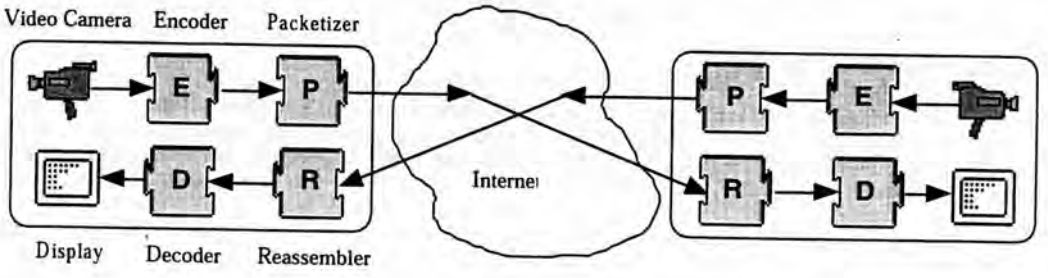


Figure 3. An IP-based peer-to-peer video conferencing system.

(d) *The Best-Effort Service Model.* The Internet is a best-effort service network. As such, each request to send a packet is honored by the Internet at best it can. Congestion on the Internet causes packet delay and loss. Normally, a packet is routed through multiple hops on its way to the destination. At each switching point linking the hops, if there are other packets that have arrived earlier, the incoming packet is queued before it can be processed by the router, resulting a queuing delay. If the router's queue is full, the incoming packet will be dropped, resulting a packet loss. Statistics have shown that packet delay and loss often occur in bursts because an overloaded router and the affected route are likely to stay congested for some time. In video streaming, each packet is scheduled to be played at certain time; a packet that has arrived later than the scheduled time is useless and will be discarded. For this reason, we don't distinguish packets that are dropped by a router due to congestion or by the streaming client because they come too late to be used. More generally, taking a signal processing approach, we can view the Internet as a "black box". A streaming client from outside the black box can observe an effective bandwidth and packet loss rate without having to know what is going on inside the black box.

(e) *Packetization.* A packetizer takes an encoded video bitstream and packetizes it for transmission over a transport protocol such as RTP. One of the most important aspects in designing a packetizer is to determine how to fragment the video bitstream. A well-designed fragmentation scheme with associated header information can help contain error in case of a packet loss. To date many video payload formats have been specified or proposed for RTP including those for the popular H.261, H.263 and the MPEG family. These payload formats offer fragmentation schemes of different granularities, such as Group-Of-Block(GOB) based or Macro-Block (MB) based. Besides fragmentation scheme it is also important to choose the size of a packet. On the one hand, small packets could control the adverse effect of a packet loss; on the other hand, it is desirable to send large packet to reduce packet header overhead. When streaming video over RTP, it is common to packetize the data for the maximum transmission unit (MTU) of a local area network (LAN). For the Ethernet, the MTU is about 1500 bytes, or 12000 bits. However, other restrictions may apply. For example, a RTP video packet is required to contain at most a single video frame. In other words, a new frame cannot start in the middle of a packet, though a frame can be split into multiple packets.

(f) *Back-channel.* A back-channel in a video streaming system is a communication channel established for a streaming client to report its state to and/or request services from the server. In RTP streaming, implementation of back-channels is facilitated by the Real-time Transport Control Protocol (RTCP) that is part of the RTP specifications.[4] The RTCP defines a type of packet for receiver report (RR). A RR packet includes such information as fraction of packets lost since the previous RR packet, cumulative number of packets lost, and interarrival jitter. It also supports profile- or application-specific extensions for including more information from the receiver. In addition to using the RR packets, it is also possible to define customized RTCP packet to carry feedback information from a streaming client. While the information about effective bandwidth and the state of packet loss can be fed back and useful to the streaming server, it is a common mistake to assume that the video encoder can always take advantage of the back-channel. We will discuss more on this in next Section.

Google Exhibit No. 1063
Page 1202

3. SIGNAL PROCESSING SOLUTIONS FOR VIDEO STREAMING

3.1. Solutions That Do Not Fit

After reviewing the architecture and characteristics of a video streaming system in the previous Section, it becomes clear that some of the methods that have been developed for video conferencing and other applications such as wireless video are not quite applicable to video streaming. We comment on these "unfit" solutions before moving onto evaluating a family of viable solutions.

3.1.1. Feedback-Based Encoder Adaptation

Many signal processing researchers have explored the idea of having a video encoder adapt to the changing conditions in bandwidth and packet loss. It is assumed that a back-channel exists and will feed the necessary information back to the encoder which in turn adjusts its behavior in response to the bandwidth and packet loss conditions. This idea has motivated the design of the RTCP channel along with RTP.[4-6] To date, many ways of encoder adaptation have been proposed and studied. Bolot, Turletti, and Wakeman[5] proposed an adaptive rate control mechanism by which a video encoder adjusts its output bitrate based on the feedback information from the receivers. There has been continued work along this direction.[7-10] In the same spirit as feedback-based rate control, Steinbach, Färber, and Girod[11] proposed to adapt the coding mode selection at the encoder to the error conditions at the receiver for wireless video. The idea is that the error damaged MBs may be tracked by the sender through a receiver's feedback, and a rapid error recovery can be achieved by INTRA refreshing corresponding MBs in the next frame. More recently, the scheme of adaptive coding mode selection was proposed for Internet video based on a packet loss probability model that must be estimated at the receiver and fed back to the sender[10]. Similar to adaptive INTRA MB refreshing and placement, it was proposed that a damaged reference picture could be dynamically replaced at the sender according to the back-channel signaling.[12] This mechanism is now supported in ITU H.263+ standard.[13] There are still more ways of encoder adaptation. For example, unequal error protection can be applied according to packet loss rate,[14] provided that a current estimate of the packet loss rate at a specific receiver is available.

Although the feedback-based encoder adaptation techniques that we have just reviewed can be effective in peer-to-peer video conferencing and wireless video, they don't fit in a real-world video streaming system. This is due to the encoder/server separation that we discussed in the previous Section. As we mentioned there (see Figure 1(a)), in live video streaming, for efficiency purpose the encoder will encode once, producing one bitstream for the server. Of course, the system can be configured to enable the encoder to generate a bitstream adapted to each client. However, such a design will not scale up to an industry-strength video streaming system. It is well-known that real-time video compression is computationally expensive; this is one reason why the encoder is separate from the server. Considering the server may be serving thousands of clients simultaneously, it is simply unrealistic to expect the encoder to compress and output thousands of different bitstreams that are adapted to each client. In on-demand video streaming, it is obvious that encoder adaptation can not be applied because the video content is compressed off-line.

3.1.2. Complex Server Involvement

Now that we know the encoder in a video streaming system cannot afford to adapt to each client's condition. One may wonder if the server can adapt. Generally speaking, the server does have some flexibility and capability to manipulate the video bitstream to adapt to a client's needs. In fact, all viable solutions to rate adaptation to be reviewed in Section 3.2 rely on the server to do some kind of processing. The exact requirements for the server involvement depend on a specific adaptation technique, but for the obvious reason the server should not be tasked with complex processing on a per client basis.

An example of tasks that are too complex is for the server to convert the bitrate of a pre-encoded video bitstream dynamically by transcoding. Although a transcoder is often much less complex than a full encoder, a considerable amount of computing such as bit re-allocation, and motion vector search or refinement may be required.[15-17] A server is unlikely to scale up to serving a large number of clients if it performs transcoding for each client. A simplified version of transcoding, called dynamic rate shaping (DRS),[18,19] requires even less processing because it achieves bitrate reduction by selectively dropping quantized transform coefficients or blocks, or by re-quantization of the blocks.[20] Despite the significant reduction in computational complexity at the expense of introducing a quality problem known as "drift", we still feel that DRS is too complex for a server to do on a per client basis. This is because DRS requires the server to have knowledge about a specific codec to process the bitstream.

Google Exhibit No. 1063
Page 1203

3.1.3. Intraframe Error Control and Concealment at Low Bitrate

There has been much work on designing error resilient bitstream for error control. For example, the emerging MPEG-4 video standard supports such mechanisms as resynchronization marker, data partition, and reversible VLC for intraframe error control. When these features are implemented in a compressed bitstream, the decoder can regain synchronization in case of bit errors, and continue to decode a video frame even if a portion of the frame is damaged or lost. Furthermore, error concealment techniques can be used to recover a damaged or lost region within a video frame to certain degree.

Obviously, any intraframe error control and concealment technique would become useless if a whole video frame is lost completely. Unfortunately, this is often the case at low bitrate when we attempt to packetize a video bitstream for the MTU of LAN. As we mentioned in Section 2.3, the MTU for the Ethernet is 12000 bits. Consider that we compress video for a target bitrate of 36kbps at a frame rate of 4fps. This would be suitable for users with a 56kbps modem connection because we need to leave some room for audio and bandwidth fluctuation. At this target bitrate and frame rate, the average size of a video frame is 9000 bits, which is less than the MTU of 12000 bits. In practice, a motion compensated frame is often much smaller in size than 9000 bits in low bitrate video streaming. This implies that when a packet is lost, it is most likely that a whole video frame is lost. Therefore, no matter how well intraframe error control and concealment techniques are implemented in the video bitstream and at the decoder, they are of little help in low bitrate video streaming.

3.2. Solutions for Rate Adaptation

As we discussed in Section 3.1, the video encoder cannot afford to adapt to individual clients, and the streaming server is constrained to doing only light processing on a per client basis. The solutions to be reviewed below for rate adaptation fit these requirements. Additionally, these solutions do not require the server to understand a compressed video bitstream. From a server design point of view, it is advantageous to make the server codec-independent, or even media-independent. A good example of media-independent RTP streaming server is Apple's Darwin QuickTime Streaming Server (QTSS). The Darwin QTSS is open-sourced and available from Apple.[21]

3.2.1. Dynamic Stream Switching

Video streaming using multiple bitstreams encoded at different bitrates has been supported in commercial streaming systems such as Apple's QuickTime, RealNetworks' RealSystem G2, and Microsoft Windows Media. RealSystem G2 was the first to support dynamic stream switching under the name of SureStream.[22,23] The idea is that the server can switch among multiple streams to serve a client with one that best matches the client's available bandwidth. The switching is dynamic, depending on the estimated available bandwidth reported by a client and the server's decision on whether it will respond. We want to note that in some research literature, video streaming using dynamic stream switching is called simulcast.[24] We prefer not to use this term because what we are describing here is different from the same word originated and used in the broadcasting industry.

In the SureStream implementation, three streams are prepared for a single target bitrate. It is worth noting that at the low end is a keyframe only stream encoded at approximately 50% of the target bitrate.[22] The bitrate of this keyframe only stream can be further reduced by dropping additional keyframes since there is no dependency among the keyframes. Furthermore, the keyframe only stream is a good fallback choice when the packet loss rate is high and there is a need to resynchronize.

Although being effective and requiring only a little server involvement, dynamic stream switching makes the encoding process more complex. The mechanism of making a switchable SureStream is not published, but one can think of some feasible approaches. For example, we can build some synchronization points to facilitate switching among the multiple streams. A keyframe shared by multiple streams would provide a convenient synchronization point, provided that the buffers of these streams can be synchronized. Besides making encoding process complicated, we note that dynamic stream switching requires redundant data storage but still has limited rate adaptability because the server can switch among only a small number of available streams.

3.2.2. Scalable Video Coding

Research in scalable video coding dates back to almost a decade ago.[25] A similar term, namely layered coding, is also used frequently to refer scalability, and has been popular in multicast community.[26,27] To date, support for scalable video coding

Google Exhibit No. 1063
Page 1204

can be found in modern video compression standards such as H.263+ and MPEG-4. There has also been continued research in this direction. See, for example, a few representative papers.[28-30] There are three types of scalability that can be built into a compressed video bitstream. They are *temporal scalability*, *spatial* (resolution) *scalability*, and (spatial) *quality scalability*. The last one is also called *SNR scalability* because the quality of a compressed picture is often measured in signal-to-noise ratio (SNR). Let's now look at what these different types of scalability can do for video streaming.

Temporal scalability is commonly achieved by including B-frames in a compressed video bitstream. B-frames are bidirectionally predicted from the forward and backward I- or P-frames, and not used as reference pictures for motion compensation in reconstructing other B- or P-frames. For this reason, a B-frame can be dropped without affecting the picture quality of other frames. However, dropping a B-frame will lower the frame rate or temporal quality of a video playback, giving the meaning of temporal scalability. Temporal scalability through B-frame placement is supported in all MPEG family video coding standards and H.263+. It should be noted that P-frames can also be dropped for rate reduction, but doing so will cause a decoder drift unless all P-frames are dropped. Use of temporal scalability for rate adaptation by dropping B- and P-frames at the server has been experimented in video streaming[23,31]. Although effective to some degree, it is found that temporal scalability alone is often inadequate to achieve desired rate reduction.

Spatial scalability is obtained by creating a multiresolution representation of each frame in a video bitstream. In spatial scalable coding, a base layer at a reduced resolution is first encoded, and an enhancement layer representing the difference between the original picture and the base layer is also encoded. It is worth noting that an enhancement layer can be split into multiple enhancement layers so that each one is built upon the previous ones at a lower resolution. Spatial scalability is supported in MPEG-4 and H.263+. There is a limited application of spatial scalability in video streaming. This is because that for good user experience the displayed frame resolution will not be changed during a video playback. So, even if the server could drop an enhancement layer and send only the base layer to a client in rate adaptation, the client would have to interpolate the resolution-reduced base layer to the full frame size for display. This makes spatial scalability a special case of SNR scalability that we are going to discuss below.

SNR scalability is similar to spatial scalability in that a base layer and one or more enhancement layers are encoded. The difference is that in SNR scalable coding the base layer is not subsampled to a lower resolution. Instead, the base layer is coarsely quantized and the difference between the base layer and the original picture is encoded into one or more enhancement layers. Spatial scalability can be viewed as a special case of SNR scalability if we consider an interpolated base layer in spatial scalable coding as the base layer in SNR scalable coding. The SNR or spatial scalability that we have just described assumes that an enhancement layer is either available and decoded as a whole or not available at all. It is not very effective for rate adaptation in video streaming for several reasons. First, if only one or two enhancement layers are used, we have very limited rate adaptability that is similar to the situation in dynamic stream switching. Second, if more enhancement layers are used, overhead will be up and coding efficiency will drop significantly. In fact, a recent study[32] on H.263+ scalability performance suggested that even with only two enhancement layers, the loss in coding efficiency of SNR scalable coding in comparison to multiple bitstream coding becomes unacceptable (about 1-1.5dB loss per layer) at low bitrate. This explains why the industry chose to use dynamic stream switching in commercial video streaming systems instead of scalable video coding options that are available in the modern video compression standards.

In recent years, research in scalable coding has centered on developing coders that can generate an "embedded bitstream". Much of the work was inspired by the design and excellent coding efficiency of the EZW[33] and SPIHT[34] coders for still images. An embedded bitstream is such that bits are packed in the order of "significance", a measure that is often tied to the magnitude of image coefficients in the transform domain. An intuitive way of embedding would be to order image coefficients by their magnitude and pack them in a bitstream in descending order of bit-planes. An embedded bitstream can be truncated from the end to match a reduced target bitrate. In EZW[33] and SPIHT,[34] the embedding is so fine-granular that one can scale the size of a bitstream in byte accuracy. For this reason, an embedded bitstream is said to have *fine-granularity scalability* (FGS). Embedding a video bitstream can be a straightforward extension to embedded image coding if a 3-D video transform is used.[35,36] It will be more complicated if a motion compensated residual coding is employed. This is because that any truncation or scaling performed on a reference picture will result in a decoder drift. A common solution to making a drift-free embedded video bitstream is to encode two layers as we discussed before in spatial and SNR scalable coding. The base layer, which is not embedded, will be used as a reference for motion compensation, while the enhancement layer will be embedded in fine granularity. This is a compromise, for the resulting bitstream is not fully embedded, but it still offers a higher rate adaptability than non-embedded coding using multiple layers.

Google Exhibit No. 1063
Page 1205

More recently, applications of FGS to video streaming have been reported.[35,37,38] In October 1998, MPEG started a new work item on fine granular video scalability.[39] The work has since moved forward under the name of MPEG-4 FGS and converged to a solution based on bit-plane coding of DCT coefficients for enhancement layers.[40,41] There is also work on defining an MPEG-4 streaming video application profile based on MPEG-4 FGS.[42,43] The advantages of FGS for rate adaptive video streaming is obvious. It has excellent rate adaptability yet the requirement for server involvement is reasonably small. There remain issues to be solved, such as how to minimize overhead in layered FGS coding, but we feel that FGS, particularly when combined with temporal scalability, provides a promising solution to rate adaptation.

3.3. Solutions for Error Control and Recovery

Error resilience can be built into streaming video to control the spread of error in case of a packet loss. A large number of error resilience tools have been developed in the last decade and now included in the latest video coding standards, such as H.263+ and MPEG-4. However, many of these tools were developed with mobile and wireless video communications in mind. See, for example, a few recent reviews on this topic.[44-46] In the following we evaluate the effectiveness of these tools for error control in video streaming. We look at solutions for error control in source and channel coding on the encoder/server side, and error recovery and concealment on the decoder/client side.

3.3.1. Error Control in Source Coding

One approach to error resilient video coding is to add some error control mechanisms during source coding and make them part of a compression codec design. Most modern image and video compression standards provide one or more error resilient modes that are supported by a special bitstream syntax and some coding tools. For example, H.263+ has several error resilience modes at source coding level, including Slice-Structured Mode, Independent Segment Decoding Mode, and Reference Picture Selection (RPS) Mode.[13] MPEG-4 on the other hand provides several error resilience tools, including resynchronization markers, data partitioning, and reversible variable length codes (RVLC)[47]. The error resilience performance of H.263+ and MPEG-4 has been reported in several papers.[13,48,49] Here we want to note that all of these tools except the RPS mode in H.263+ are designed for intraframe error control. They are most suitable for controlling errors that result from loss of small random data blocks within a video frame. On the other hand, they are not very effective in controlling errors that result from a packet loss in video streaming. This is especially true for low bitrate RTP video streaming as we discussed in Section 3.1.3.

An interesting exception in the above error resilience tools is the RPS mode supported in H.263+.[13] The RPS mode may be implemented in different ways and they don't necessarily provide a good solution to error control for video streaming. For example, one possible implementation of the RPS mode is reviewed in Section 3.1.1 and concluded to be unfit to the video streaming architecture. Nevertheless, a technique known as "video redundancy coding" (VRC)[13] can be implemented in RPS mode and provides a good interframe error control. In VRC, multiple threads of predicted video frames are synchronized at a common keyframe, but otherwise the threads are independent of each other. For example, we can choose all odd-numbered frames to form one thread and even-numbered frames to form another thread. Both threads start at a common keyframe, say frame #0, and ends at another keyframe, say #20. During video streaming, if one packet is lost and a video frame is completely lost, we would have one damaged thread. The decoder might choose to discard the damaged thread and the video playback could continue at half of the original frame rate. On the other hand, if all packets are received error-free at the decoder, the two threads of frames can be combined to give the original sequence.

Similar in spirit to VRC, multiple description coding (MDC) has received some attention recently in image coding research community.[50-55] The idea behind MDC is that if an image is a superposition of several independent descriptions, losing one or more descriptions would only degrade the quality of the reconstructed image at the decoder, but not destroy the decoding process. A simple, intuitive example of MDC is to regroup the pixels in an image into four sub-images by subsampling, and compress them independently. Clearly, if one, two, or even three sub-images are lost, we would still be able to reconstruct an image, albeit in lower quality. Although conceptually appealing, MDC has not been demonstrated to be practically useful to video streaming. There remain issues in complexity and coding efficiency that need to be addressed for MDC.

3.3.2. Error Control in Channel Coding

To enhance the error resilience of an already compressed video bitstream, error control codes can be added to the bitstream. This is commonly referred to as error control coding or channel coding. In video streaming, packetization can also be

Google Exhibit No. 1063
Page 1206

considered part of channel coding because a well designed packetization scheme can isolate errors caused by a packet loss so that the video bitstream can be reassembled from an incomplete set of packets.

Forward Error Correction (FEC) is a classic channel coding technique that has been widely used. It can be either media independent or dependent, and implemented at different levels. In media independent FEC for packet networks, the protection is exercised at the packet level. The implementation and operation of FEC do not depend on the content of the packets, and the error recovery means an exact replacement of a lost packet. Media dependent FEC is often designed for a compressed video bitstream before packetization. For example, H.263+ specifies a FEC mode for its own bitstream.[13] Traditional FEC, either media independent or dependent, is not very effective for video streaming for two reasons. First, FEC adds redundant information to achieve protection. Recalling that our observation from the data in Figure 2 suggests a strong correlation between congestion and packet loss, it seems to be counter-productive for us to send more data to combat packet loss. Of course, we could maintain the same output bitrate as without FEC by reducing the bandwidth for the video payload; the result would be lower video quality. A more important reason for the ineffectiveness of FEC is due to its open-looped nature. It is difficult to determine *a priori* the right amount of protection for a constantly changing packet loss pattern. In practice, the amount of protection exercised is often not enough for occasions of high packet loss rate and excessive for other time of low packet loss rate.

It is possible to assign unequal amounts of FEC to different segments of video data by a technique called Priority Encoding Transmission (PET).[56] The PET algorithm requires a user to specify the priorities of video segments which are also user defined. In Section 3.1.1, we commented on one method[14] that assigns unequal FEC according to an estimated packet loss rate. Another way of specifying unequal FEC is by the importance of data segments.[57] Unequal FEC based on the content of data can be considered a case of joint source-channel coding which has been identified as an important area for continued research.[58] We feel that this is a good approach that can be combined with the FGS coding described in Section 3.2.2.

3.3.3. Error Concealment

Error concealment by postprocessing can be performed at the decoder to restore or conceal the portion of the data that is damaged or lost. Many error concealment techniques have been developed, ranging from simple operations that just copy data from the previous frame to complex processes that compute regularized image restoration. A good review on these techniques is presented in a recent paper.[44]

Most error concealment techniques are based on an assumption that a video image and its motion field are smooth. For block-based coders, this assumption implies that the pixel values and the motion vector associated each coding block vary smoothly across the image. It has been demonstrated that when data corruption occurs in randomly distributed blocks or macroblocks, error concealment based on the smoothness assumption works well. Unfortunately, errors resulting from a packet loss rarely appear in randomly distributed blocks unless the blocks are interleaved randomly. It is more likely that many rows of blocks will be lost when a packet loss happens. In low bitrate video streaming, this is almost always the case, and very often all blocks in a frame are lost as we discussed in Section 3.1.3. When a large number of consecutive rows of blocks are lost, spatial or intraframe error concealment techniques do not work well. This is because the smoothness assumption does not hold for the block loss pattern we have. There is only a weak correlation between blocks that are far apart. Indeed, our experience has been that the simplest error concealment technique (i.e., copying data from the previous frame) provides the best tradeoff between enhanced visual quality and decoder complexity.

4. CONCLUSION

Research in scalable video coding appears to be maturing by the evidence that scalability options are now supported in standards such as H.263+ and MPEG-4. Nevertheless, the industry opt to take a brute-force approach to rate adaptation by using multiple bitstreams encoded at different bitrates. One wonders why. We tried in this paper to explain the reasons behind this kind of observations. The approach we take is to first identify the most challenging problems in Internet video streaming from a signal processing perspective. We see on top of our list are two types of problems, namely rate adaptation and error control and recovery, which demand good signal processing solutions. There are already a large number of signal processing techniques developed for video conferencing, an application that many signal processing researchers are familiar with. We have an opportunity to extend some of the work in video conferencing to video streaming.

However, although video streaming and conferencing share some core technologies such as video compression, the difference between the two systems is often overlooked. We reviewed carefully the characteristics of a video streaming

Google Exhibit No. 1063
Page 1207

system and architecture, and the constraints they impose on prospective signal processing solutions. Such a review is helpful because it makes us realize that some of the existing and proposed solutions, such as those employing feedback-based encoder adaptation or requiring complex server involvement, are simply unfit in the video streaming system and architecture. Among the viable solutions, we commented on the pros and cons of different approaches, and concluded that FGS combined with temporal scalability coding could provide a promising solution for rate adaptation. As for error control and recovery, we found that the existing solutions are inadequate. Particularly, we found that there exist few good solutions that are designed for error control and concealment in such circumstances that a large number of, or all, blocks in a video frame are lost. Unfortunately, this is often the case in video streaming when a packet loss happens, especially at low bitrate.

It must be pointed out that we have looked at only signal processing solutions for video streaming. There are other solutions that do not fall into the scope of signal processing. One example is the deployment of content delivery networks such as Akamai's. Even within the scope of signal processing, the coverage of our review is almost certain to be incomplete. Nonetheless, we hope this paper serves two purposes: for application developers to learn what signal processing solutions are available for video streaming, and for signal processing researchers to better understand video streaming systems and architecture, and their associated constraints on signal processing solutions.

REFERENCES

1. S. McCanne, "Scalable multimedia communication using IP multicast and lightweight sessions," *IEEE Internet Computing*, Vol. 2, No.2, pp. 33-45, Mar. 1999.
2. Stardust.com, Inc, "The evolution of multicast," IP Multicast Initiatives White Paper, December 1999. http://www.ipmi.org/
3. Akamai Technologies, Inc. http://www.akamai.com/
4. H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson, "RTP: A transport protocol for real-time applications," RFC 1889, Internet Engineering Task Force (IETF), Jan. 1996.
5. J.-C. Bolot, T. Turletti, and I. Wakeman, "Scalable feedback control for multicast video distribution in the internet," in *SIGCOMM Symposium on Communications Architectures and Protocols*, (London, England), pp. 58-67, Aug. 1994.
6. I. Busse, B. Deffner, and H. Schulzrinne, "Dynamic QoS control of multimedia applications based on RTP," *Computer Communications*, Jan. 1996.
7. J-C. Bolot, T. Turletti, "Experience with rate control mechanisms for packet video in the Internet," *Computer Communication Review*, Vol. 28, No. 1, Jan. 1998.
8. Y.J. Chung, J.W. Kim, and C.-C.J. Kuo, "Network friendly video streaming via adaptive LMS bandwidth control," in *Applications of Digital Image Processing XXI, Proc. SPIE*, vol. 3460, pp. 448-456, July 1998.
9. Y.J. Chung, J. Kim, and C.-C.J. Kuo, "Real-time streaming video with adaptive bandwidth control and DCT-based error concealment," *IEEE Trans. CAS II: Analog and Digital Signal Processing*, Vol. 46, No. 7, pp. 951-956, July 1999.
10. F. Le Léannec, F. Toutain, C. Guillemot, "Packet loss resilient MPEG-4 compliant video coding for the Internet," *Signal Processing: Image Communications*, Vol. 15, pp. 35-56, Sept. 1999.
11. E. Steinbach, N. Färber, and B. Girod, "Standard compatible extension to H.263 for robust video transmission in mobile environments," *IEEE Trans. CAS for Video Technol.*, Vol. 7, No. 6, pp. 872-881, Dec. 1997.
12. S. Fukunaga, Y. Matsumura, and T. Nakai, "Error resilient video coding controlled by backward channel signaling," *Signal Processing: Image Communications*, Vol. 14, pp. 531-540, May 1999.
13. S. Wenger, G. Knorr, J. Ott, and F. Kossentini, "Error resilience support in H.263+," *IEEE Trans. Circuits and Sys. for Video Tech.*, Vol. 8, No. 7, pp. 867-877, Nov. 1998.
14. U. Horn, K. Stuhlmüller, M. Link, and B. Girod, "Robust Internet video transmission based on scalable coding and unequal error protection," *Signal Processing: Image Communications*, Vol. 15, pp. 77-94, Sept., 1999.
15. H. Sun, W. Kowk, and J.W. Zdepski, "Architectures for MPEG compressed bitstream scaling," *IEEE Trans. CAS for Video Technol.*, Vol. 6, No. 2, pp. 191-199, April 1996.
16. G. Keesman, R. Hellinghuizen, F. Hoeksema, and G. Heideman, "Transcoding of MPEG bitstreams," *Signal Processing: Image Commun.*, Vol. 8, pp. 481-500, Sept. 1996.
17. P.A.A. Assuncao and M. Ghanbari, "A frequency-domain video transcoder for dynamic bit-rate reduction of MPEG-2 bit streams," *IEEE Trans. CAS for Video Technol.*, Vol. 8, No. 8, pp. 953-967, Dec. 1998.
18. S., Jacobs and A. Eleftheriadis, "Streaming video on the Web using dynamic rate shaping," *J. Visual Commun. Image Representation*, Vol. 9, No. 3, pp. 211-222, Sept. 1998.
19. W. Zeng and B. Liu, "Rate shaping by block dropping for transmission of MPEG-precoded video over channels of dynamic bandwidth," in *Proc. ACM Multimedia '96*, pp. 385-393, Nov. 1996.

Google Exhibit No. 1063
Page 1208

20. Y. Nakajima, H. Hori, and T. Kanoh, "Rate conversion of MPEG coded video by re-quantization process," in *Proceedings of IEEE Intl. Conf. Image Proc.*, Washington, DC, Oct. 1995, pp. 408-411.
21. Darwin Streaming Server Project, http://www.publicsource.apple.com/projects/streaming/
22. A. Lippman, "Video coding for multiple target audiences," in *Visual Communications and Image Processing'99, Proc. SPIE*, Vol. 3653, pp. 780-782, Jan. 1999.
23. "SureStream™–Delivering superior quality and reliability," RealNetworks White Papers, http://www.realnetworks.com/devzone/library/whitepapers/surestrm.html
24. B. Girod, N. Färber, and K. Stuhlmüller, "Internet video streaming," in: F. De Natale and S. Pupolin (eds.), *Multimedia Communications*, Springer, 1999, pp. 547-556.
25. K.M. Uz, M. Vetterli, and D. LeGall, "Interpolative multiresolution coding of advanced television with compatible subchannels," *IEEE Trans. CAS for Video Technol.*, Vol. 1, pp. 86-99, Mar. 1991.
26. S. McCanne, V. Jacobson, and M. Vetterli, "Receiver-driven layered multicast," *ACM SIGCOMM'96*, Stanford, CA, Aug. 1996.
27. S. McCanne, M. Vetterli, and V. Jacobson, "Low complexity video coding for receiver driven layered multicast," *IEEE J. Selected Areas in Communications*, Vol. 15, No. 6, pp. 983-1001, Aug. 1997.
28. D. Taubman and A. Zakhor, "Multi-rate 3-dimensional subband coding of video," *IEEE Trans. Image Proc.*, Vol. 3, pp. 572-588, Sept. 1994.
29. J.Y. Tham, S. Ranganath, and A.A. Kassim, "Highly scalable wavelet-based video codec for very low bit-rate environment," *IEEE J. Selected Areas in Communications*, vol. 16, No. 1, pp. 12-27, Jan. 1998.
30. K. Chen and E.J. Delp, "Wavelet based rate scalable video compression," *IEEE Trans. CAS for Video Technol.*, Vol. 9, No. 1, pp. 109-22, Feb. 1999.
31. K. Sripanidkulchai and T. Chen, "Network-adaptive video coding and transmission," in *Visual Communications and Image Processing'99, Proc. SPIE*, Vol. 3653, pp. 854-861, Jan. 1999.
32. L. Yang, F.C.M. Marins, and T.R. Gardos, "Improving H.263+ scalability performance for very low bit rate applications," in *Visual Communications and Image Processing'99, Proc. SPIE*, Vol. 3653, pp. 768-779, Jan. 1999.
33. J.M. Shapiro, "Embedded image coding using zerotrees of wavelet coefficients," *IEEE Trans. Signal Processing*, Vol. 41, No. 12, pp. 3445-3462, Dec. 1993.
34. A. Said and W.A. Pearlman, "A new, fast and efficient image codec based on set partitioning in hierarchical trees," *IEEE Trans. CAS for Video Technol.*, Vol. 6, pp. 243-250, June 1996.
35. W.-T. Tan and A. Zakhor, "Real-time Internet video using error resilient scalable compression and TCP-friendly transport protocol," *IEEE Trans. Multimedia*, Vol. 1, No. 2, pp. 172-186, June 1999.
36. B.-J. Kim, Z. Xiong, W.A. Pearlman, and Y.S. Kim, "Progressive video coding for noisy channels," Journal of Visual Communication and Image Representation, Vol.10, No. 2, pp.173-185, June 1999.
37. W. Li, "Bit-plane coding of DCT coefficients for fine granularity scalability," MPEG contributions, m3989, Atlantic City, NJ, Oct. 1998.
38. H. Radha, Y. Chen, K. Parthasarathy, and R. Cohen, "Scalable Internet video using MPEG-4," *Signal Processing: Image Communications*, Vol. 15, No. 1-2, pp. 95-126, Sept. 1999.
39. Y. Chen, C. Dufour, J.-F. Vial, A. Zakhor, W. Li, J. Liang, and B. Schuster, "Description of mini-experiment of fine granular video scalability," MPEG contributions, m3881, Atlantic City, NJ, Oct. 1998.
40. ISO/IEC JTC1/SC29/WG11, "Text of ISO/IEC 14496-2 FGS Amendment Working Draft 3.0," MPEG Document N3095, Maui, USA, Dec. 1999.
41. W. Li, F. Ling, and X. Chen, "Fine granularity scalability in MPEG-4 for streaming video," to appear in *Proceedings of IEEE International Symposium on Circuits and Systems (ISCAS'2000)*, Geneva, Switzerland, May 28-31, 2000.
42. W. Li, et al, "Streaming video profile," MPEG Contributions, m4990, Melbourne, Australia, Oct. 1999.
43. W. Li, el al, "Streaming video profiles based on fine-granularity-scalability," MPEG Contributions, m5582, Maui, USA, Dec. 1999.
44. Y. Wang, and Q. Zhu, "Error control and concealment for video communications: a review," *Proc. IEEE*, Vol. 86, No. 5, pp. 974-997, May 1998.
45. J. Villasenor, Y.-Q. Zhang, and J. When, "Robust video coding algorithms and systems," *Proc. IEEE*, Vol. 87, No. 10, pp. 1724-1733, Oct. 1999.
46. B. Girod and N. Färber, "Feedback-Based Error Control for Mobile Video Transmission", *Proc. IEEE*, Vol. 87, No. 10, pp. 1707-1723, Oct. 1999.
47. R. Talluri, "Error resilient video coding in the MPEG-4 standard," *IEEE Commun. Mag.*, Vol. 36, pp. 112-119, June 1998.
48. R. Talluri, I. Moccagatta, Y. Nag, and G. Cheung, "Error concealment by data partitioning," *Signal Processing: Image Communication*, vol.14, No.6-8, pp. 505-518, May 1999.

Google Exhibit No. 1063
Page 1209

49. L. Ducla-Soares, and F. Pereira, "Error resilience and concealment performance for MPEG-4 frame-based video coding," *Signal Processing: Image Communication*, vol.14, No.6-8, pp. 447-472, May 1999.
50. Y. Wang, M.T. Orchard, and A.R. Reibman, "Multiple description image coding for noisy channels by pairing transform coefficients," in *Proc. First IEEE Workshop on Multimedia Signal Processing*, New York, NY, June 1997. Pp. 419-424.
51. D.M. Chung and Y. Wang, "Multiple description image coding using signal decomposition and reconstruction based on lapped orthogonal transforms," *IEEE Trans. CAS for Video Technol.*, vol. 9, No. 6, pp. 895-908, Sept. 1999.
52. S.D. Servetto, K. Ramchandran, V.A. Vaishampayan, and K. Nahrstedt, "Multiple description wavelet based image coding," submitted to *IEEE Trans. Image Proc.*, Nov. 1998.
53. V.K. Goyal, J. Kovacevic, R. Arean, and M. Vetterli, "Multiple, description transform coding of images," in *Proceedings of IEEE Conference on Image Processing*, Chicago, IL, Oct. 1998, pp. 674-678.
54. X. Yang and K. Ramchandran, "Optimal multiple description subband coding," in *Proceedings of IEEE Conference on Image Processing*, Chicago, IL, Oct. 1998, pp. 654-658.
55. R. Chellappa, and M. Srinivasan, "Multiple description subband coding," in *Proceedings of IEEE Conference on Image Processing*, Chicago, IL, Oct. 1998, pp. 684-688.
56. A. Albanese, J. Bloemer, J. Edmonds, M. Luby, and M. Sudan, "Priority encoding transmission, " *IEEE Trans. Inform. Theory*, Vol. 42, No. 6, pp. 1737-1747, Nov. 1996.
57. A.E. Mohr, E.A. Riskin, and R.E. Ladner, "Graceful degradation over packet erasure channels through forward error correction," in *Proc. Data Compression Conference*, Snowbird, UT, Mar. 1999, pp. 92-101.
58. Workshop Report: 1999 NSF Sponsored Workshop on Joint Source-Channel Research, Oct. 1999.

Google Exhibit No. 1063
Page 1210

PROCEEDINGS OF SPIE

SPIE—The International Society for Optical Engineering

Image and Video Communcations and Processing 2000

Bhaskaran Vasudev
T. Russell Hsing
Andrew G. Tescher
Robert L. Stevenson
Chairs/Editors

25-28 January 2000
San Jose, California

Sponsored by
IS&T–The Society for Imaging Science and Technology
SPIE–The International Society for Optical Engineering

SPIE P PROCEEDINGS SERIES

Volume 3974

Google Exhibit No. 1063
Page 1211

PROCEEDINGS OF SPIE
SPIE—The International Society for Optical Engineering

Image and Video Communications and Processing 2000

Bhaskaran Vasudev
T. Russell Hsing
Andrew G. Tescher
Robert L. Stevenson
Chairs/Editors

25–28 January 2000
San Jose, California

Sponsored by
IS&T—The Society for Imaging Science and Technology
SPIE—The International Society for Optical Engineering

Published by
SPIE—The International Society for Optical Engineering

Volume 3974

SPIE is an international technical society dedicated to advancing engineering and scientific applications of optical , photonic, imaging, electronic, and optoelectronic technologies.

Google Exhibit No. 1063
Page 1212

TA1632
.I41123
2000



The papers appearing in this book compose the proceedings of the technical conference cited on the cover and title page of this volume. They reflect the authors' opinions and are published as presented, in the interests of timely dissemination. Their inclusion in this publication does not necessarily constitute endorsement by the editors or by SPIE. Papers were selected by the conference program committee to be presented in oral or poster format, and were subject to review by volume editors or program committees.

Please use the following format to cite material from this book:

Author(s), "Title of paper," in *Image and Video Communications and Processing 2000*, Bhaskaran Vasudev, T. Russell Hsing, Andrew G. Tescher, Robert L. Stevenson, Editors, Proceedings of SPIE Vol. 3974, page numbers (2000).

ISSN 0277-786X
ISBN 0-8194-3592-9

Published by
SPIE—The International Society for Optical Engineering
P.O. Box 10, Bellingham, Washington 98227-0010 USA
Telephone 360/676-3290 (Pacific Time) • Fax 360/647-1445

Copyright ©2000, The Society of Photo-Optical Instrumentation Engineers.

Copying of material in this book for internal or personal use, or for the internal or personal use of specific clients, beyond the fair use provisions granted by the U.S. Copyright Law is authorized by SPIE subject to payment of copying fees. The Transactional Reporting Service base fee for this volume is $15.00 per article (or portion thereof), which should be paid directly to the Copyright Clearance Center (CCC), 222 Rosewood Drive, Danvers, MA 01923. Payment may also be made electronically through CCC Online at http://www.directory.net/copyright/. Other copying for republication, resale, advertising or promotion, or any form of systematic or multiple reproduction of any material in this book is prohibited except with permission in writing from the publisher. The CCC fee code is 0277-786X/00/$15.00.

Printed in the United States of America.

Google Exhibit No. 1063
Page 1213

Contents

xi *Conference Committee*

SESSION 1 IMAGE CODING I

2 **Fast-lapped transform for image coding** [3974-01]
R. L. de Queiroz, Xerox Corp.; T. D. Tran, Johns Hopkins Univ.

11 **Online rate control in digital cameras for near-constant distortion based on minimum/maximum criterion** [3974-03]
S.-Y. Lee, A. Ortega, Univ. of Southern California

SESSION 2 IMAGE CODING II

24 **Method for variable quantization in JPEG for improved perceptual quality** [3974-04]
N. D. Memon, Polytechnic Univ.; D. R. Tretter, Hewlett-Packard Labs.

35 **Reconstruction of bilevel images from a low-quality JPEG** [3974-05]
M. Sipitca, Georgia Institute of Technology; D. W. Gillman, L. Hurd, Iterated Systems, Inc.

45 **Fast JPEG encoding for color fax using HVQ** [3974-06]
R. L. de Queiroz, Xerox Corp.; P. Fleckenstein, Rochester Institute of Technology

53 **Optimal bit allocation for channel-adaptive multiple description coding** [3974-07]
P. Sagetong, A. Ortega, Univ. of Southern California

SESSION 3 IMAGE CODING III

66 **Visual masking in wavelet compression for JPEG-2000** [3974-08]
S. J. Daly, W. Zeng, Sharp Labs. of America, Inc.; J. Li, Microsoft Research China; S. Lei, Sharp Labs. of America, Inc.

81 **Selective splitting approach to entropy-constrained single/multistage vector quantizer design** [3974-12]
H. Khalil, K. Rose, Univ. of California/Santa Barbara

89 **Compression of concentric mosaic scenery with alignment and 3D wavelet transform** [3974-10]
L. Luo, Y. Wu, Univ. of Science and Technology of China; J. Li, Y.-Q. Zhang, Microsoft Research China

101 **Mesh-based scalable image coding with rate-distortion optimization** [3974-11]
G. Marquant, S. Pateux, C. Labit, IRISA/INRIA (France)

111 **Error-resilient EBCOT image coding with content classification** [3974-09]
T.-C. Yang, S. Kumar, C.-C. J. Kuo, Univ. of Southern California

LINDA HALE LIBRARY
KANSAR CITY, MO.

Google Exhibit No. 1063
Page 1214

SESSION 4 VIDEO CODING

124 **DSP-based real-time video encoding** [3974-13]
M. Zhou, R. Talluri, Texas Instruments Inc.

133 **Constrained variable-bit-rate control algorithm for MPEG-2 encoder** [3974-14]
D. Zhang, C-Cube Microsystems; C. Y. Choo, San Jose State Univ.

144 **Conditional entropy coding of DCT coefficients for video compression** [3974-15]
M. Sipitca, Georgia Tech Research Institute; D. W. Gillman, Iterated Systems, Inc.

153 **Optimized bit allocation for scalable wavelet video coding** [3974-16]
M. Wien, M. Brünig, Rheinisch-Westfälische Technische Hochschule Aachen (Germany)

161 **Embedded wavelet video coding with error concealment** [3974-17]
P.-C. Chang, H.-C. Chen, T.-T. Lu, National Central Univ. (Taiwan)

171 **Design of wavelet-based scalable video codec** [3974-18]
Y. Bao, Media Fair, Inc.; C.-C. J. Kuo, Univ. of Southern California

SESSION 5 MPEG-4

184 **MPEG-4 playback of multimedia objects searchable by their description** [3974-19]
A. Puri, R. L. Schmidt, Q. Huang, B. Paul, R. K. Rajendran, AT&T Labs.

191 **Analysis of object segmentation methods for VOP generation in MPEG-4** [3974-20]
K. Vaithianathan, S. Panchanathan, Arizona State Univ.

204 **New video object segmentation technique based on flow-thread features for MPEG-4 and multimedia systems** [3974-21]
H.-C. Huang, Institute of Information Science (Taiwan); Y.-C. Lin, National Taiwan Univ.; Y.-P. Hung, Institute of Information Science (Taiwan) and National Taiwan Univ.; C.-S. Fuh, National Taiwan Univ.

SESSION 6 MOTION ESTIMATION

214 **Fast motion estimation for frame-rate conversion** [3974-22]
J. W. C. Wong, O. C. Au, P. H. W. Wong, Hong Kong Univ. of Science and Technology

223 **Global/local motion-compensated frame interpolation for low-bit-rate video** [3974-23]
S. Liu, Z. Yan, J. Kim, C.-C. J. Kuo, Univ. of Southern California

235 **Fast full-search block matching using subblocks and successive approximation of the error measure** [3974-25]
M. Brünig, B. Menser, Rheinisch-Westfälische Technische Hochschule Aachen (Germany)

SESSION 7 IMAGE AND VIDEO TRANSMISSION I

246 **Signal processing for Internet video streaming: a review** [3974-26]
J. Lu, Apple Computer, Inc.

Google Exhibit No. 1063
Page 1215

260 **Simple video format for mobile applications** [3974-27]
J. R. Smith, Z. Miao, C.-S. Li, IBM Thomas J. Watson Research Ctr.

270 **Robust optimization solution to the data hiding problem using distributed source coding principles** [3974-29]
J. C. Chou, S. S. Pradhan, L. El Ghaoui, K. Ramchandran, Univ. of California/Berkeley

SESSION 8 IMAGE AND VIDEO TRANSMISSION II

282 **Context-based multiple bit-stream image transmission over noisy channels** [3974-31]
L. Cao, C. W. Chen, Univ. of Missouri/Columbia

291 **Video delivery over wireless channel with dynamic QoS control** [3974-32]
S. Wang, H. Zheng, J. A. Copeland, Georgia Institute of Technology

300 **Wireless image transmission using multiple-description-based concatenated codes** [3974-33]
D. G. Sachs, Univ. of Illinois/Urbana-Champaign; A. Raghavan, K. Ramchandran, Univ. of California/Berkeley

312 **Joint source-channel coding with allpass filtering source shaping for image transmission over noisy channels** [3974-34]
J. Cai, C. W. Chen, Univ. of Missouri/Columbia

324 **Joint source-channel coding for scalable video** [3974-35]
L. P. Kondi, Northwestern Univ.; F. Ishtiaq, Motorola; A. K. Katsaggelos, Northwestern Univ.

336 **Reliable video transmission over fading channels via channel state estimation** [3974-36]
W. Kumwilaisak, J. Kim, C.-C. J. Kuo, Univ. of Southern California

SESSION 9 RENDERING

348 **Progressive rendering with multiresolution 3D meshes** [3974-37]
C. Wang, J. Li, C.-C. J. Kuo, W. Proskurowski, Univ. of Southern California

360 **Error-resilient coding technique for 3D graphic models** [3974-38]
Z. Yan, S. Kumar, J. Li, C.-C. J. Kuo, Univ. of Southern California

372 **Spatially adaptive regularized pel-recursive motion estimation based on the EM algorithm** [3974-39]
V. V. Estrela, N. P. Galatsanos, Illinois Institute of Technology

SESSION 10 VIDEO PROCESSING

386 **Majority-selection de-interlacing: an advanced motion-compensated spatiotemporal interpolation technique for interlaced video** [3974-40]
E. B. Bellers, G. de Haan, Philips Research Labs. (Netherlands)

396 **Postprocessing of interframe coded images based on convex projection and regularization** [3974-41]
S.-C. Joung, S.-J. Kim, J.-K. Paik, Chung-Ang Univ. (Korea)

Google Exhibit No. 1063
Page 1216

405 **Removing blocking effects and dropouts in DCT-based video sequences** [3974-42]
J. Jung, M. Antonini, M. Barlaud, Univ. de Nice-Sophia Antipolis (France)

415 **Object-based postprocessing of block motion fields for video applications** [3974-43]
A. Amer, E. Dubois, INRS-Télécommunications (Canada)

SESSION 11 IMAGE AND VIDEO SEGMENTATION AND RETRIEVAL I

426 **CBIR: from low-level features to high-level semantics** [3974-44]
X. S. Zhou, T. S. Huang, Univ. of Illinois/Urbana-Champaign

432 **Model-based video segmentation for vision-augmented interactive games** [3974-45]
L.-K. Liu, IBM Thomas J. Watson Research Ctr.

440 **Fast and adaptive semantic object extraction from video** [3974-46]
J. Guo, J. Kim, C.-C. J. Kuo, Univ. of Southern California

452 **Image-object extraction using a genetic-programming-based object model** [3974-47]
A. Shimizu, Sanyo Electric Co., Ltd. (Japan); S.-F. Chang, Columbia Univ.

462 **Kernel-based multiple-cue algorithm for object segmentation** [3974-48]
J. Wang, Z.-N. Li, Simon Fraser Univ. (Canada)

SESSION 12 IMAGE AND VIDEO SEGMENTATION AND RETRIEVAL II

476 **Automatic segmentation for very low bit-rate video coding** [3974-49]
S. Desmet, A. A. DeKnuydt, L. Van Eycken, Katholieke Univ. Leuven (Belgium)

487 **Object-oriented hybrid segmentation using stereo images** [3974-50]
W. Woo, Y. Iwadate, Advanced Telecommunications Research (Japan)

496 **Video structuring for multimedia applications** [3974-52]
A. Chakraborty, Siemens Corporate Research, Inc.

508 **Object tracking under compressed domain** [3974-53]
X. Li, I. K. Sethi, Wayne State Univ.

515 **Minimum description length region tracking with level sets** [3974-54]
A.-R. Mansouri, J. Konrad, INRS-Télécommunications (Canada)

SESSION 13 IMAGE PROCESSING

528 **Multiscale scheme for image magnification** [3974-55]
F. Nicolier, O. Laligant, F. Truchetet, IUT Le Creusot/Univ. de Bourgogne (France)

537 **Wavelet-domain edge modeling with applications to image interpolation** [3974-56]
B. Tao, M. T. Orchard, Princeton Univ.

Google Exhibit No. 1063
Page 1217

549 **Stochastic wavelet-based image modeling using factor graphs and its application to denoising** [3974-57]
S. Xiao, I. V. Kozintsev, Univ. of Illinois/Urbana-Champaign; K. Ramchandran, Univ. of California/Berkeley

SESSION 14 MOTION ESTIMATION AND VIDEO CODING

560 **Three-dimensional motion and deformation estimation of deformable mesh** [3974-59]
A. A. DeKnuydt, S. Desmet, K. Cox, L. Van Eycken, Katholieke Univ. Leuven (Belgium)

567 **Generation and tracking of mesh objects in image sequences** [3974-60]
J.-T. Ko, B.-W. Shyr, S.-J. Wang, National Chiao Tung Univ. (Taiwan)

576 **Matching algorithm based on Gödel coding scheme** [3974-61]
N. Dicle, V. Atalay, Middle East Technical Univ. (Turkey)

584 **Multiple motion segmentation with level sets** [3974-62]
A.-R. Mansouri, B. Sirivong, J. Konrad, INRS-Télécommunications (Canada)

SESSION 15 POSTER SESSION

598 **Efficient algorithm for identifying dependency regions for fast fractal image decoding** [3974-64]
K. W. Kang, Pukyung National Univ. (Korea); G. S. Jung, CUNY/Lehman College; K. S. Moon, Pukyung National Univ. (Korea)

606 **Modified symmetrical reversible variable length code and its theoretical bounds** [3974-65]
C.-W. Tsai, Lunghwa Institute of Technology (Taiwan); J.-L. Wu, S.-W. Liu, National Taiwan Univ.

617 **Scalable wavelet video coding using long-term memory motion-compensated prediction** [3974-66]
M. Brünig, M. Wien, Rheinisch-Westfälische Technische Hochschule Aachen (Germany)

625 **New class of VQ codebook design algorithms using adjacency maps** [3974-67]
A. D. Constantinou, D. R. Bull, C. N. Canagarajah, Univ. of Bristol (UK)

635 **Wavelet image coding using intercontext arithmetic adaptation** [3974-68]
N. V. Boulgouris, D. Vyzovitis, M. G. Strintzis, Aristotle Univ. of Thessaloniki (Greece)

643 **Embedded DCT and wavelet methods for fine granular scalable video: analysis and comparison** [3974-69]
M. van der Schaar, Y. Chen, H. Radha, Philips Research Labs. (Netherlands)

654 **Lossless image compression by adaptive contextual encoding** [3974-70]
B. Aiazzi, IROE-CNR (Italy); L. Alparone, Univ. of Florence (Italy); S. Baronti, IROE-CNR (Italy)

661 **Semi-optimized padding method for arbitrarily shaped image coding** [3974-71]
K. Takagi, A. Koike, S. Matsumoto, Kokusai Denshin Denwa R&D Labs. Inc. (Japan)

Google Exhibit No. 1063
Page 1218

673 **Exploiting the third dimension in the lossless and near-lossless compression of medical volume images and video** [3974-72]
D. De Rycke, S. Van Assche, W. R. Philips, I. L. Lemahieu, Univ. of Ghent (Belgium)

685 **Achieving idempotence in near-lossless JPEG-LS** [3974-73]
L. P. Hurd, Iterated Systems, Inc.; J. G. Rosiles, Georgia Institute of Technology

691 **Adaptive segmentation of wavelet transform coefficients for video compression** [3974-74]
P. Wasilewski, Technical Univ. of Lodz (Poland)

700 **Component ratio preserving compression for remote sensing applications** [3974-75]
İ. Avcıbaş, B. Sankur, Boğaziçi Univ. (Turkey); K. Sayood, Univ. of Nebraska/Lincoln; N. D. Memon, Polytechnic Univ.

709 **Efficient shape coding algorithm by quadtree decomposition for MPEG-4** [3974-76]
C.-C. Ma, M.-J. Chen, National Dong-Hwa Univ. (Taiwan)

720 **Using a model of the human visual system to identify and enhance object contours in natural images** [3974-78]
J. A. Black, Jr., S. Panchanathan, Arizona State Univ.

730 **MRF-based texture segmentation using wavelet decomposed images** [3974-79]
H. Noda, Kyushu Institute of Technology (Japan); M. N. Shirazi, Communications Research Lab. (Japan); E. Kawaguchi, Kyushu Institute of Technology (Japan)

741 **Using density and spatial cues for clustering image pixels in real time** [3974-80]
V. Kravtchenko, J. J. Little, Univ. of British Columbia (Canada)

751 **Feasibility of using a stabilizer in constraining the motion of epicardial tissues during MIDCAB surgery: analysis using image processing techniques** [3974-81]
V. K. Subramaniam, S. S. Rao, Villanova Univ.; K. Pallegadda, Univ. of Pennsylvania

759 **Implementation and analysis of an optimized rainfalling watershed algorithm** [3974-82]
P. De Smet, R. L. V. P. M. Pires, Univ. of Ghent (Belgium)

767 **Blocking artifact removal based on blockiness estimation** [3974-83]
Q. Zhou, C. Basoglu, W. Lee, Equator Technologies, Inc.

776 **Improved method for digital image manipulation: storage, transmission, and display** [3974-84]
C. A. P. da Silva Martins, Pontifícia Univ. Católica de Minas Gerais (Brazil); S. T. Kofuji, Univ. de São Paulo (Brazil)

788 **Multiframe combination and blur deconvolution of video data** [3974-85]
T. F. Gee, T. P. Karnowski, K. W. Tobin, Oak Ridge National Lab.

796 **2D wavelet feature detection for defining curved boundaries in Landsat images** [3974-86]
K. J. Jones, Rice Univ.

807 **Facial motion parameter estimation and error criteria in model-based image coding** [3974-88]
Y. Liu, L. Yu, Q. Yao, Zhejiang Univ. (China)

Google Exhibit No. 1063
Page 1219

816 **Triangle mesh-based motion compensation scheme with shape-adaptive wavelet transform** [3974-89]
M. Eckert, D. Ruiz, J. I. Ronda, N. N. García, Univ. Politécnica de Madrid (Spain)

826 **Fast four-step search algorithm using UESA and quadrant selection approach for motion estimation** [3974-90]
H. Nisar, T.-S. Choi, Kwangju Institute of Science and Technology (Korea)

835 **Fast block motion estimation algorithm based on combined subsamplings on pixels and search candidates** [3974-91]
Y. Yu, Univ. of Missouri/Columbia; J. Zhou, Univ. of Washington; C. W. Chen, Univ. of Missouri/Columbia

844 **Fast block-matching algorithm using threshold-based half stop, cross search, and partial distortion elimination** [3974-92]
J.-N. Kim, T.-S. Choi, Kwangju Institute of Science and Technology (Korea)

853 **Motion-based segmentation for object-based video coding and indexing** [3974-93]
B. Chupeau, E. François, Thomson Multimedia (France)

861 **Fast block search using Haar decomposition** [3974-94]
J. Kari, Univ. of Iowa; G. Liang, Intel Corp.; S. Dube, STMicroelectronics, Inc.; K. Kenemer, Nortel Networks; D. W. Gillman, Iterated Systems, Inc.

865 **Efficient motion estimation algorithm for video transcoding** [3974-96]
M.-J. Chen, M.-C. Chu, C.-W. Pan, National Dong-Hwa Univ. (Taiwan)

873 **Efficient image denoising using side information** [3974-97]
S. S. Pradhan, K. Ramchandran, Univ. of California/Berkeley

882 **Automatic face detection and tracking for H.263-compatible region-of-interest coding** [3974-98]
B. Menser, M. Wien, Rheinisch-Westfälische Technische Hochschule Aachen (Germany)

892 **Efficient multimedia distribution framework for Internet video** [3974-99]
J. Yao, J. Vass, Univ. of Missouri/Columbia; Y. Huang, Lucent Technologies/Bell Labs.; X. Zhuang, Univ. of Missouri/Columbia

900 **Video communications over wireless ATM networks** [3974-100]
J. Vass, Univ. of Missouri/Columbia; Y. Huang, Lucent Technologies/Bell Labs; X. Zhuang, Univ. of Missouri/Columbia

908 **Performance analysis of image transmission over wireless communication channels** [3974-101]
P. P. Dang, B. L. Ochoa, P. M. Chau, Univ. of California/San Diego

917 **Dynamic quality-of-service management in wireless multimedia networks** [3974-102]
L. Huang, S. Kumar, C.-C. J. Kuo, Univ. of Southern California

929 **Comparison of multiple-description coding and layered coding based on network simulations** [3974-103]
R. Singh, A. Ortega, L. Perret, W. Jiang, Univ. of Southern California

Google Exhibit No. 1063
Page 1220

940 **Differentiated QoS-aware priority handoff in cell-based multimedia wireless network** [3974-104]
H. Chen, S. Kumar, C.-C. J. Kuo, Univ. of Southern California

949 **Performance of syntax-based error detection in H.263 video coding: a quantitative analysis** [3974-105]
M. Barni, Univ. of Siena (Italy); F. Bartolini, P. Bianco, Univ. of Florence (Italy)

957 **Real-time object tracking and human face detection in cluttered scenes** [3974-106]
S. L. Dockstader, A. M. Tekalp, Univ. of Rochester

969 **Rotational invariant similarity measurement for content-based image indexing** [3974-109]
Y. M. Ro, K. W. Yoo, Information and Communications Univ. (Korea)

977 **Video indexing using object motion map** [3974-110]
S.-Y. Kim, Y. M. Ro, Information and Communications Univ. (Korea)

986 **Facial animation reconstruction from FAP** [3974-115]
L. Yu, J. Zhang, Zhejiang Univ. (China)

994 **Compression issues in multiview autostereo displays** [3974-116]
D. Shah, N. A. Dodgson, Univ. of Cambridge (UK)

1006 **DVD-RAM-based network storage system** [3974-117]
T. Ura, T. Tanabe, M. Yamamoto, NTT Cyber Space Labs. (Japan)

1016 **Non-uniform shape-preserving subdivision scheme for surface interpolation** [3974-118]
W. Dong, J. Li, C.-C. J. Kuo, Univ. of Southern California

1028 **Facial expression recognition on a people-dependent personal facial expression space (PFES)** [3974-119]
N. P. Chandrasiri, M. C. Park, T. Naemura, H. Harashima, Univ. of Tokyo (Japan)

1040 **Image-fusion-based adaptive regularization for image expansion** [3974-120]
J. H. Shin, J. S. Yoon, J. K. Paik, Chung-Ang Univ. (Korea)

1052 *Addendum*
1054 *Author Index*

Google Exhibit No. 1063
Page 1221



CONFERENCE PROCEEDING

Image and video communications and processing 2000 : 25-28 January, 2000, San Jose, California / Bhaskaran Vasudev ... [et al.], chairs/editors ; sponsored by IS&T--the Society for Imaging Science and Technology, SPIE--the International Society for Optical Engineering ; published by SPIE--the International Society for Optical Engineering

Bhaskaran, Vasudev, 1956-; IS & T--the Society for Imaging Science and Technology; Society of Photo-optical Instrumentation Engineers

c2000

Top
Send to
Get It
Details
Virtual Browse
Links

Available at Linda Hall Library Books - 2nd Floor (TA1632 .I41123 2000) >

Send to

Email | QR | Permalink | Citation

EasyBib | Export RIS | Export to Excel

Get It

Back to Locations

LOCATION ITEMS

Linda Hall Library
Available , Books - 2nd Floor ; TA1632 .I41123 2000
(1 copy, 1 available, 0 requests)

Item in place
Loanable

Details

Title	Image and video communications and processing 2000 : 25-28 January, 2000, San Jose, California / Bhaskaran Vasudev ... [et al.], chairs/editors ; sponsored by IS&T--the Society for Imaging Science and Technology, SPIE--the International Society for Optical Engineering ; published by SPIE--the International Society for Optical Engineering
Other title	Proceedings of SPIE--the International Society for Optical Engineering. v. 3974.
Creator/Contributor	Bhaskaran, Vasudev, 1956- > IS & T--the Society for Imaging Science and Technology > Society of Photo-optical Instrumentation Engineers >
Subjects	Computer graphics > Video compression > Image compression > Image processing >
Identifier	ISBN : 9780819435927 ISBN : 0819435929 OCLC : (OCoLC)43987418
Series	Proceedings / SPIE--the International Society for Optical Engineering ; v. 3974 > Proceedings of SPIE--the International Society for Optical Engineering, v. 3974 >
Publisher	Bellingham, Washington : SPIE
Creation date	c2000
Format	xii, 1052 p. : ill. ; 28 cm.
General notes	Includes bibliographic references and author index.

Virtual Browse



ge algebra

phologica
age
essing IV : ...



Image algebra and morphologica l image processing V : ...



Image analysis : principles and practice : a technical ...



Image analysi applica

Links

Display Source Record

Image and video communications and processing 2000 : 25-28 January, 2000, San Jose, California (Proceedings of SPIE--the International Society for Optical Engineering. v. 3974).

Linda Hall Library MARC record copied from

https://catalog.lindahall.org/discovery/sourceRecord?vid=01LINDAHALL_INST:LHL&docId=alma993356813405961&recordOwner=01LINDAHALL_INST

```
leader  01463nam a2200325Ia 4500
001     993356813405961
005     20020117201000.0
008     000505s2000    waua     b    101 0 eng d
020     ##$a9780819435927
020     ##$a0819435929
035     ##$a(OCoLC)43987418
035     ##$9459483
035     ##$a(MoKL)335681-lhalldb
035     ##$a(lhalldb)335681-lhalldb
040     ##$aLHL $cLHL
049     ##$aLHLA
090     ##$aTA1632 $b.I41123
245     00$aImage and video communications and processing 2000 : $b25-28
January, 2000, San Jose, California / $cBhaskaran Vasudev ... [et al.],
chairs/editors ; sponsored by IS&T--the Society for Imaging Science and
Technology, SPIE--the International Society for Optical Engineering ;
published by SPIE--the International Society for Optical Engineering.
260     ##$aBellingham, Washington : $bSPIE, $cc2000.
300     ##$axii, 1052 p. : $bill. ; $c28 cm.
490     1#$aProceedings / SPIE--the International Society for Optical
Engineering ; $vv. 3974
504     ##$aIncludes bibliographic references and author index.
650     #0$aComputer graphics $vCongresses
650     #0$aVideo compression. $vCongresses.
650     #0$aImage compression $vCongresses
650     #0$aImage processing $vCongresses
700     1#$aBhaskaran, Vasudev, $d1956-
710     2#$aIS & T--the Society for Imaging Science and Technology
710     2#$aSociety of Photo-optical Instrumentation Engineers
830     #0$aProceedings of SPIE--the International Society for Optical
Engineering. $vv. 3974.
948     ##$aLTI 09/01/2008
994     ##$aE0 $bLHL
```

CONSUMER APPLICATIONS OF THE IEEE 1394 SERIAL BUS, AND A 1394/DV VIDEO EDITING SYSTEM

Alan T. Wetzel, Michael R. Schell*

Texas Instruments, Inc., Dallas, Tx.
*Interactive Images, Inc., Colorado Springs, Co.

Copyright 1996 by The Institute of Electrical and Electronic Engineers, Inc.

Reprinted by permission of the IEEE.

This paper was prepared for and presented at the International Conference on Consumer Electronics held in Chicago, Ill., June 1996, conference session WPM-6, Intelligent Home Networks.

Google Exhibit No. 1038
Page 001

Google Exhibit No. 1063
Page 1226

CONSUMER APPLICATIONS OF THE IEEE 1394 SERIAL BUS, AND A 1394/DV VIDEO EDITING SYSTEM

Alan T. Wetzel, Michael R. Schell*

Texas Instruments, Inc., Dallas, Tx.
*Interactive Images, Inc., Colorado Springs, Co.

ABSTRACT

The IEEE 1394-1995 High Performance Serial Bus [1] has been created with consumer use in mind. Some of the more important features of the base standard are reviewed. Enhancements and additions to the standard to support new consumer applications, such as MPEG-2 transport and the new Digital Video (DV) tape format, are described. Consumer DV products incorporating 1394 are now on the market, more have been announced, and related developments portend even more in the future. The HD-DVC "DV" [2] system has features which have significant advantages for video editing systems. The 1394 interface used by DV equipment brings many benefits to a video editing system. A PC based nonlinear video editing system is described, which takes advantage of the combination of DV and 1394 features in a high performance, low cost implementation.

WHY CONSUMERS WILL BE WINNERS WITH 1394

Superior digital quality and performance at low cost

The most obvious benefit seen by consumers in the shift to digital technologies is the improvement in quality. This has been true for some time with audio CDs, as well as the newer DBS systems. This is now becoming true again with the new HD-DVC "DV" (digital video) tape systems. The digital recording and playback, combined with the digital 1394 interface, completely eliminates the noise pick-up and signal losses that occur in analog systems and interconnects. The digital record/playback avoids the audio flutter and video picture jitter that is common in present analog systems. The 1394/DV combination also provides for virtually lossless audio/video editing and dubbing. This recognition of the superior quality of the all-digital systems has resulted in an unprecedented demand for products.

The 1394 serial bus interface is designed to support these digital devices, both from the user experience and feature/cost points of view. The 1394 interface has been completely implemented in generic digital ASIC CMOS silicon processes. Multiple integration possibilities exist, from today's first generation, multi-chip architectures, to single-chips, to full integration (zero additional chips) into other ASIC chips. Full ASIC integration is a key benefit to reducing circuit board space requirements in space critical applications. As the silicon processes evolve to smaller feature sizes, the area required to implement the 1394 logic will also decrease.

Hassle-free hook-up

The first, and perhaps only, 1394 item that a consumer is likely to encounter is the plug and cable. A user-friendly connector and cabling system is fundamental to 1394. The cable

Google Exhibit No. 1038
Page 002

Google Exhibit No. 1063
Page 1227

assemblies feature small and durable connectors. The plugs and sockets are designed for ease of use. The shape of the connector nose and socket opening provide unambiguous tactile feedback, which facilitates frustration-free blind insertion. A very flexible cable, less than ¼ inch in diameter, is standard. This system has been evolved from the child-proof connectors used by the Ninetendo Game Boy.

The second thing that a consumer is likely to notice about the 1394 interconnect system is the freedom from connection constraints. Because 1394 is a bus, with all devices sharing the same transmission domain, it is irrelevant as to which way the different pieces of equipment are connected together. Because all the digital signals for a particular piece of equipment can be carried on the same 1394 cable, only one connection is required for any unit. There is no in-out or up-down direction sense in 1394, so the connectors at each end of the cable will generally be the same. And it makes no difference which socket is used on a particular device, since they are all equivalent.

One of the many things the consumer will probably not notice is the automatic configuration and management that is built in to 1394. The user does not have to set any configuration switches because the node IDs are assigned automatically during 1394 bus initialization. And no special device needs to be provided or designated as a central controller or bus manager.

CONSUMER ORIENTED FEATURES OF THE IEEE 1394 SYSTEM

The 1394 connector and cabling system

Two connector and cable systems are presently defined. The original 6-pin/6-conductor style contains two shielded twisted pairs for data, plus another pair for optional power and ground. The 6-pin system uses a detent or optional latching retention system. A newer and smaller 4-pin/4-conductor system was developed for use on digital camcorders and other size sensitive applications. It has the two twisted pairs for data signaling, but eliminates the optional power and ground pair. Both cables have overall shielding. The connectors for the two cable systems are shown in Fig 1.



Fig 1: The 1394 connectors

Unconstrained interconnect

The topology of a 1394 bus may be a simple daisy-chain, a tree, a star, or a combination of these. The 1394 standard indicates that the most widely separated devices should have no more than 16 cable hops between them, but in reality the generous margins designed in to the 1394 system ensures that nothing breaks even if this limit is exceeded.

Automatic configuration

During initialization, each device on the 1394 bus takes a turn to announce its configuration and capabilities, and acquires a unique node ID. This process of bus enumeration and device ID assignments is handled completely automatically, and provides support for Plug-and-Play compatibility.

All 1394 devices have the capability of being the root, which is a simple function to resolve

bus arbitration requests and grants. This means that no special device must be included to initialize or manage the operation of any 1394 bus. This is a significant difference from some other busses which require a central processor running dedicated real-time processes for management and control. In a similar fashion, any isochronous-capable consumer device is able to function as the cycle master, which regulates the isochronous cycles.

The distributed intelligence that handles the initial configuration and root arbitration functions is contained entirely within the 1394 physical layer. This allows a device to be an active member of a bus even if its higher layer (link and above) functions are not active. By eliminating the need for higher layer activity the process is much faster, and power saving strategies can be realized. The implementation of these physical layer functions is purposefully simple enough to allow it to be built as a small state machine. Hardware implementation also contributes to the speed of configuration. Because most of the 1394 functionality is implemented in silicon, the cost for 1394 will continue to drop as die size scales down with smaller process sizes.

Isochronous "real-time" channels

Besides scaling the 1394 implementation costs with each successive generation of silicon processes, the 1394 system provides a special feature to minimize the system complexity and costs. This feature is isochronous data transport.

The 1394 Serial Bus provides two distinct modes of operation, asynchronous and isochronous. The bus alternates between these two modes of operation, with isochronous operation taking precedence over asynchronous communication. Isochronous operation occurs at a cycle rate of 8,000 times per second.

Under normal circumstances the isochronous operation may take up to 80% (100 μsec) of the available cycle time, reserving at least 20% for asynchronous traffic. The asynchronous data delivery and verification is assured by an acknowledge and retry protocol, and is normally used for control, status, and accuracy-critical data.

Isochronous data delivery is a "just in time" type of service. Data is broadcast on assigned channels with guaranteed bandwidth or time allocations. Delivery of preallocated real time data packets, at a uniform rate, is built into the 1394 protocol.

Up to 64 isochronous channels may be allocated on the 1394 Serial Bus. All allocated channels can send their data during each isochronous cycle. Only one device may transmit data on a given channel, but any number of devices may listen to a particular channel.

A major advantage of isochronous transport is that a much smaller FIFO is required for buffering the video and audio data before and after transmission across the 1394 bus. Because the bandwidth and latency are guaranteed, the 1394 interface silicon can use a minimally-sized FIFO memory, which reduces the die size and the ultimate product cost to the consumer. Non-isochronous transport systems require significantly larger FIFO memories to accommodate the wide swings in data-rates or transport delays.

ENHANCEMENTS AND EXTENSIONS TO IEEE 1394

MPEG-2 transport

The MPEG-2 System standard, ISO/IEC 13818-1, Part 1 [3] assumes that the transmission system for transporting packets has a constant delay characteristic. The maximum deviation or jitter allowed is 500 nanoseconds. This limit is a significant

challenge to any shared access system, including 1394.

Two primary factors may contribute to variations in delivery delay of 1394 isochronous packets: 1) Delays in starting an isochronous cycle, and, 2) Changes in the order in which channels (packets) are transmitted. Cycle start delays can be caused by an asynchronous transaction being active at the time when cycle start should occur. With a maximum sized asynchronous packet this delay can be approximately 75 microseconds. The ordering of isochronous channels in any particular cycle is unlikely to change unless there is a change in bus configuration, or in bus utilization. The worst case would be for a channel's packet to move between the beginning and end of the 100 microsecond transmission window of the isochronous cycle.

In order to meet this MPEG-2 jitter requirement, a technique was developed that allows cancellation of almost all of the 1394 induced packet delivery jitter. This technique takes advantage of the fact that all nodes participating in isochronous traffic share a common time reference. The cycle start packets that signal the beginning of an isochronous cycle also include the current value of the master clock. This clock is based on the basic 1394 reference frequency of 24.576 MHz.

When an MPEG-2 transport packet is received by a 1394 device, the current 1394 system time value is prefixed to the transport packet. This time stamp is one quadlet, or four bytes of data. At the receiving end of the 1394 transport system, the 1394 time stamp is removed, and is compared to the current system time. By passing the transport packets to the destination device at a constant offset from system time, the original temporal relationship between the packets will be restored. This eliminates 1394 transport jitter to the resolution of two 1394 clock periods, or 2 x 1/(24.576 x 10^6) = 81.41 nanoseconds.

A proposed addition to the MPEG2 standards, Extensions for real time interfaces for system decoders [4] has clarified transport stream jitter measurements and suggested a more generous limit. The new transport jitter limit is 25 microseconds, measured peak-to-average (or 50 μsec peak-peak). The 1394 time stamping system provides jitter cancellation capabilities that account for less than 1/6 of 1% of this proposed limit, so 1394-induced jitter effectively can be ignored.

The CIP header system

As part of the development of the 1394 time stamping and jitter canceling functions, a standardized formatting of the data was developed. This has been named the Common Isochronous Packet, or CIP, header. The CIP header is an extensible format that can be adapted for future applications as they arise. Today there are sets defined for the SD, HD and SDL formats of the DV tape systems, along with MPEG-2 transport packets as used by the DVB and ATV digital television systems. These formats are defined in the HD DVC "Blue Book."

In addition to the previously described time stamping function, the CIP header system also supports a number of other system functions. The two most notable of these are the Plug Control Register, or PCR, and the Function Control Protocol, or FCP. The specifications for both of these, together with the associated Connection Management Protocol (CMP), also are covered in the HD-DVC "Blue Book."

The purpose of PCR/CMP is to provide a method for controlling the virtual connection between transmitting and receiving devices on a 1394 bus. Two primary functions are: 1) To prevent active connections from being accidentally broken, and 2) To permit broadcasting devices with no listeners to be turned off. In the first case, a listener can register with the sender, and the sender knows not to stop as long as the connection is

active. In the second case, a broadcasting device is able to tell when there are no listeners, so it can stop sending and release its allocated bandwidth and channel.

The Function Control Protocol (FCP) and its related AV/C Command and Transaction Set provide for a standardized register location where information can be obtained as to what control languages are understood by a particular device. At present two values are defined, one for the AV/C language used by DV tape systems, and the other for the Common Application Language (CAL) of CEBus.

The relation between the 1394 protocol stack and the HDDV-DVC layers is shown in Fig 2. The areas delineated by the dashed lines designate the sections of the "Blue Book" that deal with the 1394 interface.

Longer and faster

The specifications for the cables used in 1394 are based on electrical performance. An example given as a reference in the standard is suitable for use over the full range of signaling speeds (100 Mbit/s base rate, plus the 200 and 400 Mbit/sec options) at a length of up to 4.5 meters. A slightly larger and lower loss cable has been developed which may be used to increase the distance per hop into the 15-20 meter range.

Restricting the maximum signaling frequency also allows a given cable to be used at a greater length. For consumer use, the 200 Mbit/sec optional speed appears to provide more than adequate performance and has

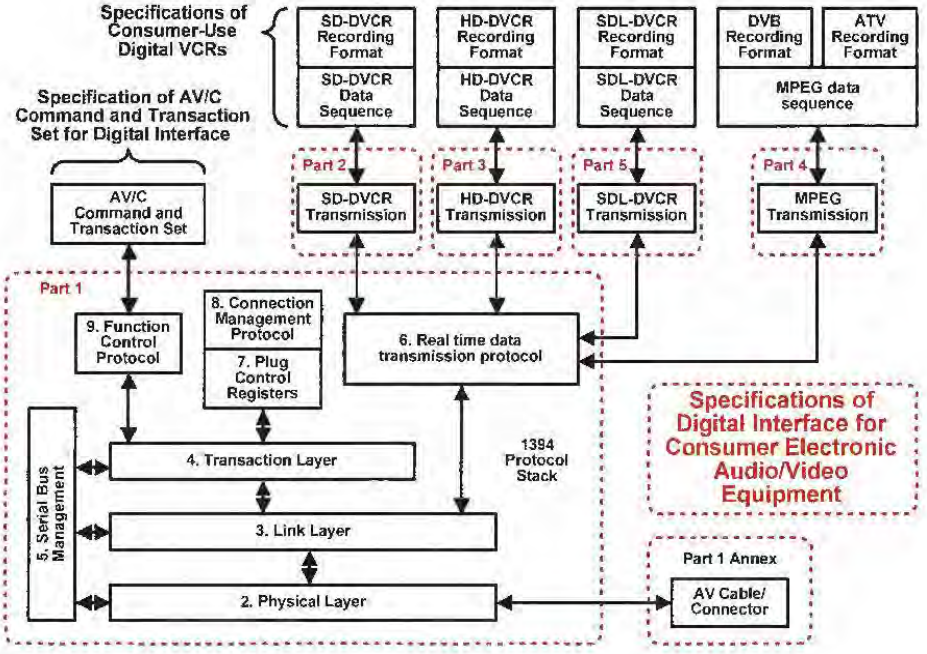


Fig 2: Relation of 1394 Protocol Stack to HD-DVC Blue Book

been specified as the maximum signaling rate supported by the HD-DVC specification. This also allows for smaller cables to be used for any particular distance.

Even with these types of cable enhancements and signaling rate compromises it is still difficult to extend the reach of a single 1394 cable hop into the 25-50 meter range needed for a home or small office network. The use of active signal repeaters is a simple solution for certain situations. A repeater is nothing more than a two port physical layer device, with no link or higher layers, that is a fully compliant node in the network. However, the use of repeaters in permanently installed and inaccessible wiring is problematic.

A number of alternative physical layers and configurations have been examined for longer distances. One very important parameter that should be considered when increasing the distances within a single 1394 bus is the likely inclusion of a larger number of devices than would be found in a typical local equipment cluster. A larger number of devices on the bus raises the prospects for running out of isochronous bandwidth to support all the desired applications. Another consideration is the arbitration signaling and gap (idle) time. The longer the end-to-end distance, the longer one must wait for arbitration and acknowledgment round trip delays. Longer wait times translates to more timing overhead or lower efficiency.

For these reasons it is believed that most longer distance 1394 networks will benefit from being arranged as separate (local cluster) busses connected by bridges or routers. Traffic belonging to one bus is not seen by any others unless specifically addressed to another bus. The performance of each separate local cluster bus may also be independently optimized.

The addressing scheme used in 1394 allocates 10 bits for bus_ID, 6 bits for node_ID, and 48 bits for internal device addresses. This translates to a maximum 1023 busses, each having up to 63 nodes (the all-1's addresses are reserved for broadcast purposes). The protocols for managing bridges was purposefully not included in the base IEEE standard in order to avoid unnecessary delays. Work to define bridging/routing started in 1995 as a working group project within the 1394 Trade Association [5]. It is expected to become an authorized project of the IEEE mid-year 1996 [6].

One of the proposed features of the bridge definition work is the concept of an extended bridge. In its simplest implementation this could be a pair of "half bridges" separated by some longer distance point-to-point link. Possible links for the extended bridge include unshielded twisted pair (UTP), plastic optical fiber (POF), coax, or wireless. Distance capabilities are expected to be at least 50 meters.

In addition to this work to define bridges for 1394, work has also started to define a new gigabit rate physical layer and protocol. This work also started as a working group within the 1394 Trade Association, and is also expected to become an authorized project of the IEEE mid-year 1996[7]. The gigabit physical layer and protocol will provide a backbone supporting multiple standard 1394 busses through bridges. The goals are a minimum distance of 25 meters, and that the interface will look and behave the same to the user as the present 1394 bus.

PRODUCTS USING 1394

DV camcorders

In the fall of 1995 Sony introduced two DV camcorders, the DCR-VX1000 and DCR-VX700. Both of these units meet the HD-DVC requirements for SD (525-60 and 625-50 systems) format digital camcorders. Both models include a 1394 based DV digital interface. This allows for virtually lossless audio/video dubbing and editing. Many other manufacturers also have introduced DV

camcorders, and several have announced that future units will include the 1394/DV interface.

DV VCR tape decks

In April 1996 Sony introduced its DHR-1000 digital VCR deck companion to its DV camcorders. This unit, currently available in Europe, includes a single front-panel 1394/DV connector.

D-VHS VCR

JVC has announced the finalization of the specifications for D-VHS, Standard Mode [8]. 1394 has been adopted as the digital interface for this system. This technology offers bit stream recording of compressed digital data in a variety of formats, such as those used by Digital Broadcast Satellites (DBS).

Desktop camera

Sony has introduced the CCM-DS250 desktop video camera [9] which utilizes a single standard (6-pin) 1394 cable to provide the complete camera connection to a PC adapter card. The camera is powered through the 1394 cable. Camera command and status messages are carried as asynchronous packets interleaved with the isochronous packets of uncompressed video data. This camera supports a wide range of operating modes with options for resolution, frame rate, and color depth. A 100 Mbit/sec 1394 interface easily supports this camera operating in a 320 x 240, 30 frames/sec, 16 bit/pixel (YUV 4:2:2) mode. The command and register format specifications have been documented by a working group of the 1394 Trade Association.

Digital video editing system

miro Computer Products AG has announced [10] a licensing agreement with Sony Corporation to develop 1394 based desk-top video editing products for the PC. This action is "designed to bridge the gap between the highly successful DV format camcorders and the non-linear editing world of desktop video."

Truevision has announced a DVCPRO version of TARGA 2000 RTX digital video engine [11] that includes a 1394 interface. DVCPRO is Panasonic Broadcast & Television Systems Company's implementation of DV with a different recording (but not data) format. The new adapter card provides non-linear editing and real time 2D effects processing of DV-encoded data on a desktop system.

Digital still image capture board

Sony has introduced a digital still image video capture board, the DVBK-1000 [12], as a companion to its DV camcorders. This ISA adapter card features a 1394 interface to Sony's consumer DV camcorders.

Set top boxes and digital television

The Digital Audio Visual Interface Council (DAVIC) has been working to define a common set top box architecture. A technical subcommittee of DAVIC is considering the addition of 1394 as the "external A0" interface for the DAVIC 1.1 specification The Digital Video Broadcasting (DVB) Project is defining standards for digital television broadcasts, and a subcommittee is considering 1394 for the digital interface to digital televisions and peripherals. The EIA R-4.1 subcommittee of the R-4 Television systems committee has selected 1394 as the basis for the digital interface for US-ATV digital television systems. All of the above items are drafts or work in progress, so should not be assumed to be finalized. A recent specification from TELE-TV for a set top box includes a reference to the DAVIC architecture, and includes a requirement for a 1394 interface.

PCs and motherboards

Several major manufacturers of computers and related hardware and software have announced support of 1394 [13]: Adaptec,

Cirrus Logic Inc., IBM PC Company, Microsoft Corp., Sun Microsystems, and Texas Instruments Mobile Computing Business.

At the April 1, 1996, WinHEC (Windows Hardware Engineering Conference) Microsoft Corp. announced the Simply Interactive PC (SIPC) [14] which will allow the PC platform to be the "center of entertainment, communications and productivity in both home and office." One of the keys technologies to SIPC is 1394, which will provide users with low-cost, easy-to-install peripherals. Endorsement for SIPC [15] has been given by Adaptec Inc., Compaq Computer Corp., Gateway 2000 Inc., Hewlett-Packard Co., Intel Corp., and Toshiba Corp.

Apple Computer has announced [16] that its Pippin 1997 reference platform will include 1394 for consumer product connectivity. Apple also plans to include 1394 on selected computer systems in 1997, and across the entire product line in 1998. The Mac OS is expected to include 1394 support in the first half of 1997.

1394/DV VIDEO EDITING

The advantages of 1394/DV in video editing

1394 and the DV (Digital Video) standard promise to revolutionize video editing over the next 3-5 years. Compared to the current analog/JPEG video editing systems now widely used, 1394/DV offers superior video quality at a lower, constant data-rate, smaller tape size, fewer tape drop-outs, built-in data archiving, guaranteed audio-video synchronization, virtual elimination of dropped frames, and file interchangeability. These advantages hold the promise of bringing professional digital video editing to the high-volume consumer market.

DV, a new video compression standard, does not contain the bi-directional and predictive frames of MPEG-2, so editing boundaries may be at any frame. DV produces a fixed data-rate of approximately 3.6 MBytes/sec, utilizing a fixed 5:1 compression based on 4:1:1 YUV video sampling. DV compression relies on discrete cosine transform like JPEG and MPEG, but adds the enhancement of field interpolation on low-motion scenes. This interpolation, as well as other enhancements, produces video which many observers say is equivalent to Sony Betacam SP which has been JPEG compressed at 2.5:1. Therefore, DV produces equivalent video quality to JPEG, but at about a 30% lower data-rate.

The DV data is stored on a new, smaller evaporated-metal tape specifically designed for digital data. This new tape uses a 10 micron track (versus VHS at 50 microns) and can store 60 minutes of video and audio (over 15 GBytes of data) on a tape which is about 75% the size of an 8mm cassette. Prior to writing to this DV tape, error detection and correction bits are added to the data stream. Additionally, the digital data is interleaved across the tape within a single frame. These techniques greatly reduced dropouts and other tape artifacts commonly found in 8mm and other analog formats.

In addition to excellent video and audio quality, the 1394/DV system provides built-in data archiving. After creating a video tape using a typical analog/JPEG system the editor may wish to save the video clips on a digital tape for long-term archiving. This often requires the added cost of a high capacity tape drive (e.g. Exabyte) and tape. With a 1394/DV system the user can simply use the built-in digital storage capability of the DV cassette for archiving.

The constant digital data-rate (3.6 MBytes/sec) of the 1394/DV system eliminates the problems associated with JPEG data-rate bursts. JPEG is a "constant quality, variable data-rate" algorithm. The compressed data-rate from a JPEG system can easily vary by a factor of 4:1, depending on the complexity of the image. These data-rate bursts cause

dropped video frames if the data-rate exceeds the capability of the disk I/O subsystem. On the other hand, the constant 3.6 MByte data-rate of a 1394/DV system is well within the capabilities of a modern SCSI disk system, virtually eliminating dropped frames.

Finally, 1394/DV systems will support file interchangeability. In most analog/JPEG systems the JPEG data stream parameters are proprietary to a given manufacturer. However, the DV data stream is well defined and a widely accepted standard within all DV camcorder products. This includes Panasonic's DVCPRO and Sony's DVCam professional products. This means that video footage shot on a high end professional camcorder can be played or edited on a consumer DVCR tape deck or on a consumer 1394/DV editing system.

Early drafts of the DV "Blue Book" specification defined the tape format and the video compression algorithm, but not a digital interface or data transport protocols. While several digital video transport protocols such as parallel and serial D1 were well known, these protocols were soon recognized as too complex and too expensive for a consumer application. 1394 with its bandwidth of 100 Mbits/sec or greater, guaranteed isochronous delivery of data, low-cost interconnect and relatively simple silicon implementation was quickly identified as the best overall choice for the digital transport of DV data. Accordingly, 1394 was added to the final 1996 "Blue Book" specification and has been widely accepted as the digital standard for DV transport.

A proposed 1394/DV editing card

A software based 1394/DV editing system can be easily created on a Pentium class PC with the addition of a 1394 adapter card and a fast disk subsystem. A fast disk subsystem, such as SCSI, is required to meet the 3.6 Mbytes/sec DV data-rate. While functional, such a system would be extremely slow as each video effect would be rendered entirely in software.



Fig 3: A PC based 1394/DV video editing system

The creation of a video effect, such as titling, picture-in-a-picture, dissolves or wipes, requires the decompression, and ultimately the re-compression of each video frame involved in the video effect. Each frame of rendered video typically requires the decompressions of two video source frames, and the re-compression of the final video frame. This decompression and re-compression can be performed in hardware or software.

In a software based solution, the time required for a decompression or compression of a frame of video is approximately one second. Each second of video is made up of 30 frames of video. Therefore, producing a one second video effect will take about 90 seconds of CPU time just for the decompression and compression: Decompression of two source frames plus compression of the rendered frame. A hardware codec (compressor-decompressor) card, on the other hand can perform each required compression or decompression in approximately 33 milliseconds - a 30X improvement over software.

The diagram in Fig 4 shows a proposed hardware based 1394/DV adapter-codec. The product uses a 1394 link layer controller and physical layer interface to connect to the camcorder via a 1394 cable. The DV data is moved across the CPU bus via a PCI controller. A DV codec (DVCPRO or equivalent) chip set is used to compress or

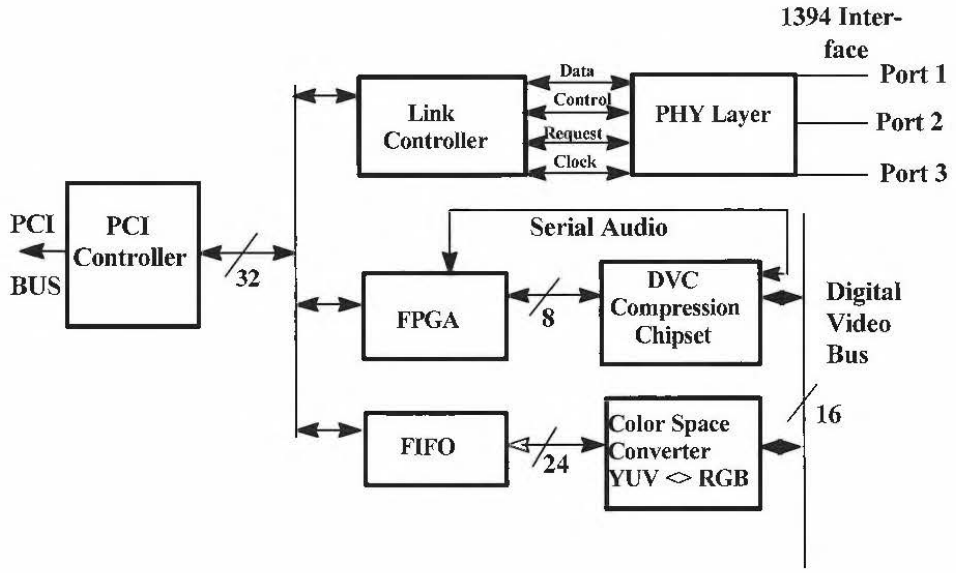


Fig4: 1394/DV adapter-codec block diagram

decompress the DV data at full video rates of approximately 33 milliseconds per frame for rendering or display operations. The color space converter transforms between the uncompressed YUV video data from the color space of the camcorder to the RGB color space of the computer for rendering operations. The FIFO is used to buffer this RGB data prior to transport across the PCI bus.

This design uses an FPGA (Field Programmable Gate Array) to control the flow of video and audio data within the card. Additionally, the PCI controller must be bus master capable to maximize PCI bus bandwidth and to minimize dependencies on the interrupt latency of the CPU. Ring zero software drivers are mandatory to enhance overall system performance. Actual render times using this hardware design is expected to be 30 to 40 times faster than a software only solution.

The 1394/DV system is expected to become the de facto consumer video editing standard in the next 3-5 years. While software based editing solutions hold the promise of extremely low-cost entry solutions, the time required to create video effects will motivate most users to move up to a hardware based adapter-codec card.

CONCLUSION

1394 is rapidly gaining acceptance as the interconnect standard for digital consumer (as well as professional) video and audio equipment. 1394's simple, low cost cabling, interface chip integration and high speed data transport make it a natural for MPEG-2 based set-top boxes, camcorders and video editing. 1394/DV provides numerous compelling advantages over the current analog/JPEG method, including superior video and audio quality, built-in archiving and guaranteed audio/video synchronization. Undoubtedly,

1394 will find applications in many diverse consumer applications.

BIOGRAPHIES



Alan Wetzel is a Senior Member Technical Staff at Texas Instruments, Inc., Dallas Texas. He received a BSEE from Northwestern University, and an MSEE from Southern Methodist University. He has worked as an IC designer, design manager, and product specialist in numerous fields at TI including consumer, automotive, data storage and data transmission, and communication systems. Several of his products have been in production for over 25 years. His present assignment is with the 1394 Product Development Team, a part of the Computer and Computer Peripheral Products organization.



Michael Schell is President and founder of Interactive Images of Colorado Springs, CO. He has an extensive background in both analog and digital circuit design and was the hardware architect for the company's first digital video editing product, the Plum. Michael holds a BSEE from SMU and was a IC designer for Motorola and a Static Ram Device engineer for INMOS. He has authored over 20 technical articles in digital design. He is an ardent supporter of DV and 1394 technology.

REFERENCES

1 IEEE 1394-1995: Standard for a High Performance Serial Bus.

2 HD Digital VCR Conference "Blue Book" Specification, 1996.

3 ISO/IEC 13818-1, Information Technology - Generic coding of moving pictures and associated audio information, Part 1: Systems.

4 ISO/IEC DIS 13818-9, Draft International Standard for Information Technology - Generic coding of moving pictures and associated audio information, Part 9: Extensions for real time interface for systems decoders.

5 1394 Trade Association, c/o Skipstone, Inc., http://www.firewire.org.

6 IEEE Project Authorization Request: High Performance Serial Bus Bridges.

7 IEEE Project Authorization Request: A Scaleable Gigabit Extension to 1394 Serial Bus.

8 http://www.jvc-victor.co.jp/dvhssd-e.html.

9 http://www.sony.co.jp/ProductsPark/JAPAN/Professional/Camera/CCM-DS250-j.html

10 Business Wire, April 15, 1996: "miro Computer Products AG is Granted License for Sony DV Codec Source Code ..."

11 NAB, Las Vegas, April 16, 1996 release, see http://www.truevision.com/PRs/DVCPRO.html.

12 http://www.sony.co.jp/Exhibition/DVCap/index.html.

13 Microsoft, et al. press releases dated circa Jan. 8, 1996.

14 PR Newswire, April 1, 1996: "Microsoft unveils Simply Interactive PC framework."

15 Infoworld, April 1, 1996, p 1: "Developers rally behind new PC design."

16 PR Newswire, May 13, 1996: "Apple Plans to Put 1394/Firewire Inside Products, Aggressive Plans for Breakthrough Technology Revealed.".

Ex. 1038A

From the companion CD-ROM to the IEEE CS Press book,
"The Anatomy of a Microprocessor: A Systems Perspective,"
by Shriver & Smith

CONSUMER APPLICATIONS OF THE IEEE 1394 SERIAL BUS, AND A 1394/DV VIDEO EDITING SYSTEM

Alan T. Wetzel, Michael R. Schell*

Texas Instruments, Inc., Dallas, Tx.
***Interactive Images, Inc., Colorado Springs, Co.**

Copyright 1996 by The Institute of Electrical and Electronic Engineers, Inc.

Reprinted by permission of the IEEE.

This paper was prepared for and presented at the International Conference on Consumer Electronics held in Chicago, Ill., June 1996, conference session WPM-6, Intelligent Home Networks.

CONSUMER APPLICATIONS OF THE IEEE 1394 SERIAL BUS, AND A 1394/DV VIDEO EDITING SYSTEM

Alan T. Wetzel, Michael R. Schell*

Texas Instruments, Inc., Dallas, Tx.
*Interactive Images, Inc., Colorado Springs, Co.

ABSTRACT

The IEEE 1394-1995 High Performance Serial Bus [1] has been created with consumer use in mind. Some of the more important features of the base standard are reviewed. Enhancements and additions to the standard to support new consumer applications, such as MPEG-2 transport and the new Digital Video (DV) tape format, are described. Consumer DV products incorporating 1394 are now on the market, more have been announced, and related developments portend even more in the future. The HD-DVC "DV" [2] system has features which have significant advantages for video editing systems. The 1394 interface used by DV equipment brings many benefits to a video editing system. A PC based nonlinear video editing system is described, which takes advantage of the combination of DV and 1394 features in a high performance, low cost implementation.

WHY CONSUMERS WILL BE WINNERS WITH 1394

Superior digital quality and performance at low cost

The most obvious benefit seen by consumers in the shift to digital technologies is the improvement in quality. This has been true for some time with audio CDs, as well as the newer DBS systems. This is now becoming true again with the new HD-DVC "DV" (digital video) tape systems. The digital recording and playback, combined with the digital 1394 interface, completely eliminates the noise pick-up and signal losses that occur in analog systems and interconnects. The digital record/playback avoids the audio flutter and video picture jitter that is common in present analog systems. The 1394/DV combination also provides for virtually lossless audio/video editing and dubbing. This recognition of the superior quality of the all-digital systems has resulted in an unprecedented demand for products.

The 1394 serial bus interface is designed to support these digital devices, both from the user experience and feature/cost points of view. The 1394 interface has been completely implemented in generic digital ASIC CMOS silicon processes. Multiple integration possibilities exist, from today's first generation, multi-chip architectures, to single-chips, to full integration (zero additional chips) into other ASIC chips. Full ASIC integration is a key benefit to reducing circuit board space requirements in space critical applications. As the silicon processes evolve to smaller feature sizes, the area required to implement the 1394 logic will also decrease.

Hassle-free hook-up

The first, and perhaps only, 1394 item that a consumer is likely to encounter is the plug and cable. A user-friendly connector and cabling system is fundamental to 1394. The cable

assemblies feature small and durable connectors. The plugs and sockets are designed for ease of use. The shape of the connector nose and socket opening provide unambiguous tactile feedback, which facilitates frustration-free blind insertion. A very flexible cable, less than ¼ inch in diameter, is standard. This system has been evolved from the child-proof connectors used by the Nintendo Game Boy.

The second thing that a consumer is likely to notice about the 1394 interconnect system is the freedom from connection constraints. Because 1394 is a bus, with all devices sharing the same transmission domain, it is irrelevant as to which way the different pieces of equipment are connected together. Because all the digital signals for a particular piece of equipment can be carried on the same 1394 cable, only one connection is required for any unit. There is no in-out or up-down direction sense in 1394, so the connectors at each end of the cable will generally be the same. And it makes no difference which socket is used on a particular device, since they are all equivalent.

One of the many things the consumer will probably not notice is the automatic configuration and management that is built in to 1394. The user does not have to set any configuration switches because the node IDs are assigned automatically during 1394 bus initialization. And no special device needs to be provided or designated as a central controller or bus manager.

CONSUMER ORIENTED FEATURES OF THE IEEE 1394 SYSTEM

The 1394 connector and cabling system

Two connector and cable systems are presently defined. The original 6-pin/6-conductor style contains two shielded twisted pairs for data, plus another pair for optional power and ground. The 6-pin system uses a detent or optional latching retention system. A newer and smaller 4-pin/4-conductor system was developed for use on digital camcorders and other size sensitive applications. It has the two twisted pairs for data signaling, but eliminates the optional power and ground pair. Both cables have overall shielding. The connectors for the two cable systems are shown in Fig 1.

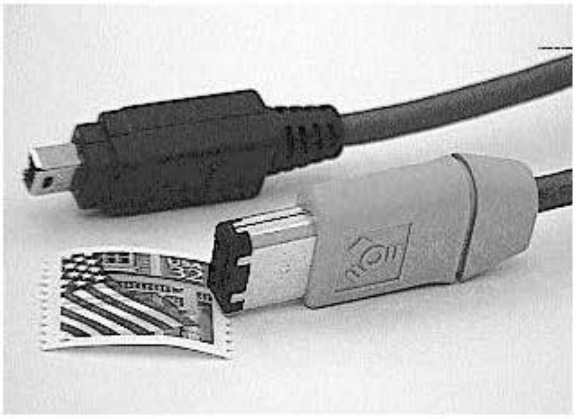


Fig 1: The 1394 connectors

Unconstrained interconnect

The topology of a 1394 bus may be a simple daisy-chain, a tree, a star, or a combination of these. The 1394 standard indicates that the most widely separated devices should have no more than 16 cable hops between them, but in reality the generous margins designed in to the 1394 system ensures that nothing breaks even if this limit is exceeded.

Automatic configuration

During initialization, each device on the 1394 bus takes a turn to announce its configuration and capabilities, and acquires a unique node ID. This process of bus enumeration and device ID assignments is handled completely automatically, and provides support for Plug-and-Play compatibility.

All 1394 devices have the capability of being the root, which is a simple function to resolve

bus arbitration requests and grants. This means that no special device must be included to initialize or manage the operation of any 1394 bus. This is a significant difference from some other busses which require a central processor running dedicated real-time processes for management and control. In a similar fashion, any isochronous-capable consumer device is able to function as the cycle master, which regulates the isochronous cycles.

The distributed intelligence that handles the initial configuration and root arbitration functions is contained entirely within the 1394 physical layer. This allows a device to be an active member of a bus even if its higher layer (link and above) functions are not active. By eliminating the need for higher layer activity the process is much faster, and power saving strategies can be realized. The implementation of these physical layer functions is purposefully simple enough to allow it to be built as a small state machine. Hardware implementation also contributes to the speed of configuration. Because most of the 1394 functionality is implemented in silicon, the cost for 1394 will continue to drop as die size scales down with smaller process sizes.

Isochronous "real-time" channels

Besides scaling the 1394 implementation costs with each successive generation of silicon processes, the 1394 system provides a special feature to minimize the system complexity and costs. This feature is isochronous data transport.

The 1394 Serial Bus provides two distinct modes of operation, asynchronous and isochronous. The bus alternates between these two modes of operation, with isochronous operation taking precedence over asynchronous communication. Isochronous operation occurs at a cycle rate of 8,000 times per second.

Under normal circumstances the isochronous operation may take up to 80% (100 μsec) of the available cycle time, reserving at least 20% for asynchronous traffic. The asynchronous data delivery and verification is assured by an acknowledge and retry protocol, and is normally used for control, status, and accuracy-critical data.

Isochronous data delivery is a "just in time" type of service. Data is broadcast on assigned channels with guaranteed bandwidth or time allocations. Delivery of preallocated real time data packets, at a uniform rate, is built into the 1394 protocol.

Up to 64 isochronous channels may be allocated on the 1394 Serial Bus. All allocated channels can send their data during each isochronous cycle. Only one device may transmit data on a given channel, but any number of devices may listen to a particular channel.

A major advantage of isochronous transport is that a much smaller FIFO is required for buffering the video and audio data before and after transmission across the 1394 bus. Because the bandwidth and latency are guaranteed, the 1394 interface silicon can use a minimally-sized FIFO memory, which reduces the die size and the ultimate product cost to the consumer. Non-isochronous transport systems require significantly larger FIFO memories to accommodate the wide swings in data-rates or transport delays.

ENHANCEMENTS AND EXTENSIONS TO IEEE 1394

MPEG-2 transport

The MPEG-2 System standard, ISO/IEC 13818-1, Part 1 [3] assumes that the transmission system for transporting packets has a constant delay characteristic. The maximum deviation or jitter allowed is 500 nanoseconds. This limit is a significant

challenge to any shared access system, including 1394.

Two primary factors may contribute to variations in delivery delay of 1394 isochronous packets: 1) Delays in starting an isochronous cycle, and, 2) Changes in the order in which channels (packets) are transmitted. Cycle start delays can be caused by an asynchronous transaction being active at the time when cycle start should occur. With a maximum sized asynchronous packet this delay can be approximately 75 microseconds. The ordering of isochronous channels in any particular cycle is unlikely to change unless there is a change in bus configuration, or in bus utilization. The worst case would be for a channel's packet to move between the beginning and end of the 100 microsecond transmission window of the isochronous cycle.

In order to meet this MPEG-2 jitter requirement, a technique was developed that allows cancellation of almost all of the 1394 induced packet delivery jitter. This technique takes advantage of the fact that all nodes participating in isochronous traffic share a common time reference. The cycle start packets that signal the beginning of an isochronous cycle also include the current value of the master clock. This clock is based on the basic 1394 reference frequency of 24.576 MHz.

When an MPEG-2 transport packet is received by a 1394 device, the current 1394 system time value is prefixed to the transport packet. This time stamp is one quadlet, or four bytes of data. At the receiving end of the 1394 transport system, the 1394 time stamp is removed, and is compared to the current system time. By passing the transport packets to the destination device at a constant offset from system time, the original temporal relationship between the packets will be restored. This eliminates 1394 transport jitter to the resolution of two 1394 clock periods, or 2 x 1/(24.576 x 10^6) = 81.41 nanoseconds.

A proposed addition to the MPEG2 standards, Extensions for real time interfaces for system decoders [4] has clarified transport stream jitter measurements and suggested a more generous limit. The new transport jitter limit is 25 microseconds, measured peak-to-average (or 50 μsec peak-peak). The 1394 time stamping system provides jitter cancellation capabilities that account for less than 1/6 of 1% of this proposed limit, so 1394-induced jitter effectively can be ignored.

The CIP header system

As part of the development of the 1394 time stamping and jitter canceling functions, a standardized formatting of the data was developed. This has been named the Common Isochronous Packet, or CIP, header. The CIP header is an extensible format that can be adapted for future applications as they arise. Today there are sets defined for the SD, HD and SDL formats of the DV tape systems, along with MPEG-2 transport packets as used by the DVB and ATV digital television systems. These formats are defined in the HD DVC "Blue Book."

In addition to the previously described time stamping function, the CIP header system also supports a number of other system functions. The two most notable of these are the Plug Control Register, or PCR, and the Function Control Protocol, or FCP. The specifications for both of these, together with the associated Connection Management Protocol (CMP), also are covered in the HD-DVC "Blue Book."

The purpose of PCR/CMP is to provide a method for controlling the virtual connection between transmitting and receiving devices on a 1394 bus. Two primary functions are: 1) To prevent active connections from being accidentally broken, and 2) To permit broadcasting devices with no listeners to be turned off. In the first case, a listener can register with the sender, and the sender knows not to stop as long as the connection is

active. In the second case, a broadcasting device is able to tell when there are no listeners, so it can stop sending and release its allocated bandwidth and channel.

The Function Control Protocol (FCP) and its related AV/C Command and Transaction Set provide for a standardized register location where information can be obtained as to what control languages are understood by a particular device. At present two values are defined, one for the AV/C language used by DV tape systems, and the other for the Common Application Language (CAL) of CEBus.

The relation between the 1394 protocol stack and the HDDV-DVC layers is shown in Fig 2. The areas delineated by the dashed lines designate the sections of the "Blue Book" that deal with the 1394 interface.

Longer and faster

The specifications for the cables used in 1394 are based on electrical performance. An example given as a reference in the standard is suitable for use over the full range of signaling speeds (100 Mbit/s base rate, plus the 200 and 400 Mbit/sec options) at a length of up to 4.5 meters. A slightly larger and lower loss cable has been developed which may be used to increase the distance per hop into the 15-20 meter range.

Restricting the maximum signaling frequency also allows a given cable to be used at a greater length. For consumer use, the 200 Mbit/sec optional speed appears to provide more than adequate performance and has

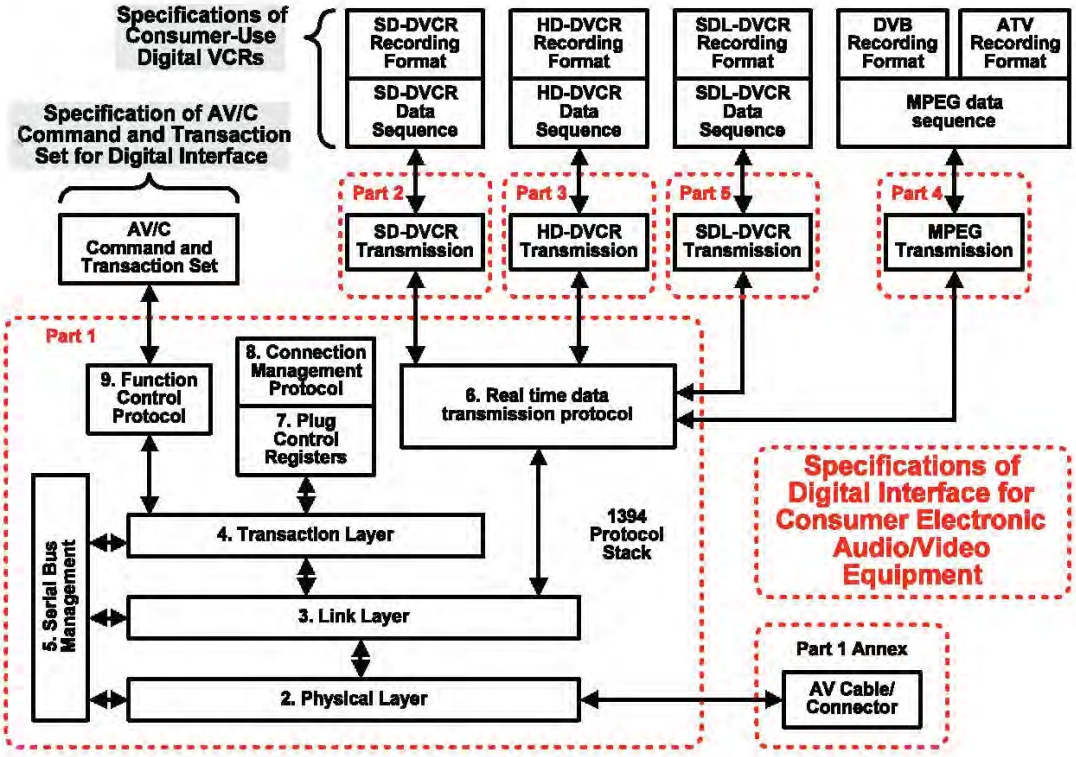


Fig 2: Relation of 1394 Protocol Stack to HD-DVC Blue Book

been specified as the maximum signaling rate supported by the HD-DVC specification. This also allows for smaller cables to be used for any particular distance.

Even with these types of cable enhancements and signaling rate compromises it is still difficult to extend the reach of a single 1394 cable hop into the 25-50 meter range needed for a home or small office network. The use of active signal repeaters is a simple solution for certain situations. A repeater is nothing more than a two port physical layer device, with no link or higher layers, that is a fully compliant node in the network. However, the use of repeaters in permanently installed and inaccessible wiring is problematic.

A number of alternative physical layers and configurations have been examined for longer distances. One very important parameter that should be considered when increasing the distances within a single 1394 bus is the likely inclusion of a larger number of devices than would be found in a typical local equipment cluster. A larger number of devices on the bus raises the prospects for running out of isochronous bandwidth to support all the desired applications. Another consideration is the arbitration signaling and gap (idle) time. The longer the end-to-end distance, the longer one must wait for arbitration and acknowledgment round trip delays. Longer wait times translates to more timing overhead or lower efficiency.

For these reasons it is believed that most longer distance 1394 networks will benefit from being arranged as separate (local cluster) busses connected by bridges or routers. Traffic belonging to one bus is not seen by any others unless specifically addressed to another bus. The performance of each separate local cluster bus may also be independently optimized.

The addressing scheme used in 1394 allocates 10 bits for bus_ID, 6 bits for node_ID, and 48 bits for internal device addresses. This translates to a maximum 1023 busses, each having up to 63 nodes (the all-1's addresses are reserved for broadcast purposes). The protocols for managing bridges was purposefully not included in the base IEEE standard in order to avoid unnecessary delays. Work to define bridging/routing started in 1995 as a working group project within the 1394 Trade Association [5]. It is expected to become an authorized project of the IEEE mid-year 1996 [6].

One of the proposed features of the bridge definition work is the concept of an extended bridge. In its simplest implementation this could be a pair of "half bridges" separated by some longer distance point-to-point link. Possible links for the extended bridge include unshielded twisted pair (UTP), plastic optical fiber (POF), coax, or wireless. Distance capabilities are expected to be at least 50 meters.

In addition to this work to define bridges for 1394, work has also started to define a new gigabit rate physical layer and protocol. This work also started as a working group within the 1394 Trade Association, and is also expected to become an authorized project of the IEEE mid-year 1996[7]. The gigabit physical layer and protocol will provide a backbone supporting multiple standard 1394 busses through bridges. The goals are a minimum distance of 25 meters, and that the interface will look and behave the same to the user as the present 1394 bus.

PRODUCTS USING 1394

DV camcorders

In the fall of 1995 Sony introduced two DV camcorders, the DCR-VX1000 and DCR-VX700. Both of these units meet the HD-DVC requirements for SD (525-60 and 625-50 systems) format digital camcorders. Both models include a 1394 based DV digital interface. This allows for virtually lossless audio/video dubbing and editing. Many other manufacturers also have introduced DV

camcorders, and several have announced that future units will include the 1394/DV interface.

DV VCR tape decks

In April 1996 Sony introduced its DHR-1000 digital VCR deck companion to its DV camcorders. This unit, currently available in Europe, includes a single front-panel 1394/DV connector.

D-VHS VCR

JVC has announced the finalization of the specifications for D-VHS, Standard Mode [8]. 1394 has been adopted as the digital interface for this system. This technology offers bit stream recording of compressed digital data in a variety of formats, such as those used by Digital Broadcast Satellites (DBS).

Desktop camera

Sony has introduced the CCM-DS250 desktop video camera [9] which utilizes a single standard (6-pin) 1394 cable to provide the complete camera connection to a PC adapter card. The camera is powered through the 1394 cable. Camera command and status messages are carried as asynchronous packets interleaved with the isochronous packets of uncompressed video data. This camera supports a wide range of operating modes with options for resolution, frame rate, and color depth. A 100 Mbit/sec 1394 interface easily supports this camera operating in a 320 x 240, 30 frames/sec, 16 bit/pixel (YUV 4:2:2) mode. The command and register format specifications have been documented by a working group of the 1394 Trade Association.

Digital video editing system

miro Computer Products AG has announced [10] a licensing agreement with Sony Corporation to develop 1394 based desk-top video editing products for the PC. This action is "designed to bridge the gap between the highly successful DV format camcorders and the non-linear editing world of desktop video."

Truevision has announced a DVCPRO version of TARGA 2000 RTX digital video engine [11] that includes a 1394 interface. DVCPRO is Panasonic Broadcast & Television Systems Company's implementation of DV with a different recording (but not data) format. The new adapter card provides non-linear editing and real time 2D effects processing of DV-encoded data on a desktop system.

Digital still image capture board

Sony has introduced a digital still image video capture board, the DVBK-1000 [12], as a companion to its DV camcorders. This ISA adapter card features a 1394 interface to Sony's consumer DV camcorders.

Set top boxes and digital television

The Digital Audio Visual Interface Council (DAVIC) has been working to define a common set top box architecture. A technical subcommittee of DAVIC is considering the addition of 1394 as the "external A0" interface for the DAVIC 1.1 specification The Digital Video Broadcasting (DVB) Project is defining standards for digital television broadcasts, and a subcommittee is considering 1394 for the digital interface to digital televisions and peripherals. The EIA R-4.1 subcommittee of the R-4 Television systems committee has selected 1394 as the basis for the digital interface for US-ATV digital television systems. All of the above items are drafts or work in progress, so should not be assumed to be finalized. A recent specification from TELE-TV for a set top box includes a reference to the DAVIC architecture, and includes a requirement for a 1394 interface.

PCs and motherboards

Several major manufacturers of computers and related hardware and software have announced support of 1394 [13]: Adaptec,

Cirrus Logic Inc., IBM PC Company, Microsoft Corp., Sun Microsystems, and Texas Instruments Mobile Computing Business.

At the April 1, 1996, WinHEC (Windows Hardware Engineering Conference) Microsoft Corp. announced the Simply Interactive PC (SIPC) [14] which will allow the PC platform to be the "center of entertainment, communications and productivity in both home and office." One of the keys technologies to SIPC is 1394, which will provide users with low-cost, easy-to-install peripherals. Endorsement for SIPC [15] has been given by Adaptec Inc., Compaq Computer Corp., Gateway 2000 Inc., Hewlett-Packard Co., Intel Corp., and Toshiba Corp.

Apple Computer has announced [16] that its Pippin 1997 reference platform will include 1394 for consumer product connectivity. Apple also plans to include 1394 on selected computer systems in 1997, and across the entire product line in 1998. The Mac OS is expected to include 1394 support in the first half of 1997.

1394/DV VIDEO EDITING

The advantages of 1394/DV in video editing

1394 and the DV (Digital Video) standard promise to revolutionize video editing over the next 3-5 years. Compared to the current analog/JPEG video editing systems now widely used, 1394/DV offers superior video quality at a lower, constant data-rate, smaller tape size, fewer tape drop-outs, built-in data archiving, guaranteed audio-video synchronization, virtual elimination of dropped frames, and file interchangeability. These advantages hold the promise of bringing professional digital video editing to the high-volume consumer market.

DV, a new video compression standard, does not contain the bi-directional and predictive frames of MPEG-2, so editing boundaries may be at any frame. DV produces a fixed data-rate of approximately 3.6 MBytes/sec, utilizing a fixed 5:1 compression based on 4:1:1 YUV video sampling. DV compression relies on discrete cosine transform like JPEG and MPEG, but adds the enhancement of field interpolation on low-motion scenes. This interpolation, as well as other enhancements, produces video which many observers say is equivalent to Sony Betacam SP which has been JPEG compressed at 2.5:1. Therefore, DV produces equivalent video quality to JPEG, but at about a 30% lower data-rate.

The DV data is stored on a new, smaller evaporated-metal tape specifically designed for digital data. This new tape uses a 10 micron track (versus VHS at 50 microns) and can store 60 minutes of video and audio (over 15 GBytes of data) on a tape which is about 75% the size of an 8mm cassette. Prior to writing to this DV tape, error detection and correction bits are added to the data stream. Additionally, the digital data is interleaved across the tape within a single frame. These techniques greatly reduced dropouts and other tape artifacts commonly found in 8mm and other analog formats.

In addition to excellent video and audio quality, the 1394/DV system provides built-in data archiving. After creating a video tape using a typical analog/JPEG system the editor may wish to save the video clips on a digital tape for long-term archiving. This often requires the added cost of a high capacity tape drive (e.g. Exabyte) and tape. With a 1394/DV system the user can simply use the built-in digital storage capability of the DV cassette for archiving.

The constant digital data-rate (3.6 MBytes/sec) of the 1394/DV system eliminates the problems associated with JPEG data-rate bursts. JPEG is a "constant quality, variable data-rate" algorithm. The compressed data-rate from a JPEG system can easily vary by a factor of 4:1, depending on the complexity of the image. These data-rate bursts cause

dropped video frames if the data-rate exceeds the capability of the disk I/O subsystem. On the other hand, the constant 3.6 MByte data-rate of a 1394/DV system is well within the capabilities of a modern SCSI disk system, virtually eliminating dropped frames.

Finally, 1394/DV systems will support file interchangeability. In most analog/JPEG systems the JPEG data stream parameters are proprietary to a given manufacturer. However, the DV data stream is well defined and a widely accepted standard within all DV camcorder products. This includes Panasonic's DVCPRO and Sony's DVCam professional products. This means that video footage shot on a high end professional camcorder can be played or edited on a consumer DVCR tape deck or on a consumer 1394/DV editing system.

Early drafts of the DV "Blue Book" specification defined the tape format and the video compression algorithm, but not a digital interface or data transport protocols. While several digital video transport protocols such as parallel and serial D1 were well known, these protocols were soon recognized as too complex and too expensive for a consumer application. 1394 with its bandwidth of 100 Mbits/sec or greater, guaranteed isochronous delivery of data, low-cost interconnect and relatively simple silicon implementation was quickly identified as the best overall choice for the digital transport of DV data. Accordingly, 1394 was added to the final 1996 "Blue Book" specification and has been widely accepted as the digital standard for DV transport.

A proposed 1394/DV editing card

A software based 1394/DV editing system can be easily created on a Pentium class PC with the addition of a 1394 adapter card and a fast disk subsystem. A fast disk subsystem, such as SCSI, is required to meet the 3.6 Mbytes/sec DV data-rate. While functional, such a system would be extremely slow as each video effect would be rendered entirely in software.

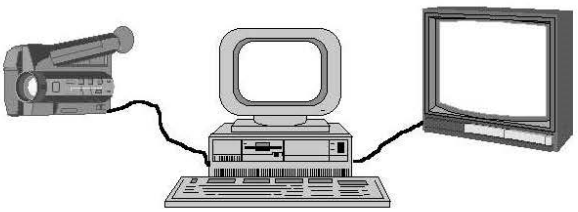


Fig 3: A PC based 1394/DV video editing system

The creation of a video effect, such as titling, picture-in-a-picture, dissolves or wipes, requires the decompression, and ultimately the re-compression of each video frame involved in the video effect. Each frame of rendered video typically requires the decompressions of two video source frames, and the re-compression of the final video frame. This decompression and re-compression can be performed in hardware or software.

In a software based solution, the time required for a decompression or compression of a frame of video is approximately one second. Each second of video is made up of 30 frames of video. Therefore, producing a one second video effect will take about 90 seconds of CPU time just for the decompression and compression: Decompression of two source frames plus compression of the rendered frame. A hardware codec (compressor-decompressor) card, on the other hand can perform each required compression or decompression in approximately 33 milliseconds - a 30X improvement over software.

The diagram in Fig 4 shows a proposed hardware based 1394/DV adapter-codec. The product uses a 1394 link layer controller and physical layer interface to connect to the camcorder via a 1394 cable. The DV data is moved across the CPU bus via a PCI controller. A DV codec (DVCPRO or equivalent) chip set is used to compress or

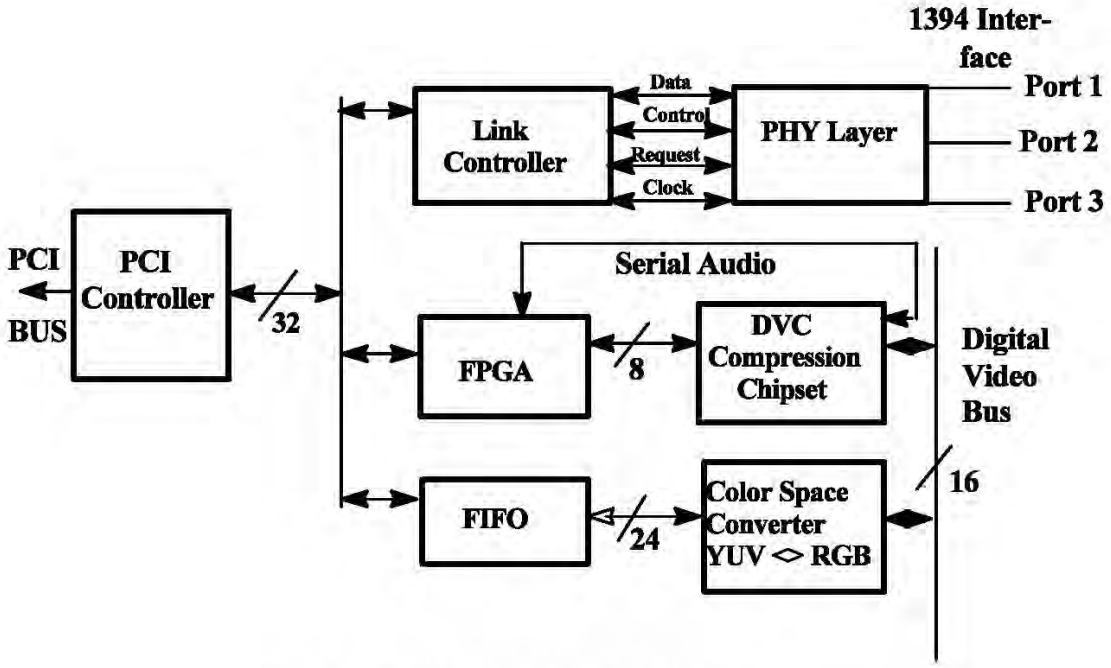


Fig4: 1394/DV adapter-codec block diagram

decompress the DV data at full video rates of approximately 33 milliseconds per frame for rendering or display operations. The color space converter transforms between the uncompressed YUV video data from the color space of the camcorder to the RGB color space of the computer for rendering operations. The FIFO is used to buffer this RGB data prior to transport across the PCI bus.

This design uses an FPGA (Field Programmable Gate Array) to control the flow of video and audio data within the card. Additionally, the PCI controller must be bus master capable to maximize PCI bus bandwidth and to minimize dependencies on the interrupt latency of the CPU. Ring zero software drivers are mandatory to enhance overall system performance. Actual render times using this hardware design is expected to be 30 to 40 times faster than a software only solution.

The 1394/DV system is expected to become the de facto consumer video editing standard in the next 3-5 years. While software based editing solutions hold the promise of extremely low-cost entry solutions, the time required to create video effects will motivate most users to move up to a hardware based adapter-codec card.

CONCLUSION

1394 is rapidly gaining acceptance as the interconnect standard for digital consumer (as well as professional) video and audio equipment. 1394's simple, low cost cabling, interface chip integration and high speed data transport make it a natural for MPEG-2 based set-top boxes, camcorders and video editing. 1394/DV provides numerous compelling advantages over the current analog/JPEG method, including superior video and audio quality, built-in archiving and guaranteed audio/video synchronization. Undoubtedly,

1394 will find applications in many diverse consumer applications.

BIOGRAPHIES



Alan Weisel is a Senior Member Technical Staff at Texas Instruments, Inc., Dallas Texas. He received a BSEE from Northwestern University, and an MSEE from Southern Methodist University. He has worked as an IC designer, design manager, and product specialist in numerous fields at TI including consumer, automotive, data storage and data transmission, and communication systems. Several of his products have been in production for over 25 years. His present assignment is with the 1394 Product Development Team, a part of the Computer and Computer Peripheral Products organization.



Michael Schell is President and founder of Interactive Images of Colorado Springs, CO. He has an extensive background in both analog and digital circuit design and was the hardware architect for the company's first digital video editing product, the Pluto. Michael holds a BSEE from SMU and was a IC designer for Motorola and a Static Ram Device engineer for INMOS. He has authored over 20 technical articles in digital design. He is an ardent supporter of DV and 1394 technology.

REFERENCES

1 IEEE 1394-1995: Standard for a High Performance Serial Bus.

2 HD Digital VCR Conference "Blue Book" Specification, 1996.

3 ISO/IEC 13818-1, Information Technology - Generic coding of moving pictures and associated audio information, Part 1: Systems.

4 ISO/IEC DIS 13818-9, Draft International Standard for Information Technology - Generic coding of moving pictures and associated audio information, Part 9: Extensions for real time interface for systems decoders.

5 1394 Trade Association, c/o Shipstone, Inc., http://www.firewire.org.

6 IEEE Project Authorization Request: High Performance Serial Bus Bridges.

7 IEEE Project Authorization Request: A Scaleable Gigabit Extension to 1394 Serial Bus.

8 http://www.jvc-victor.co.jp/dvhead-a.html.

9 http://www.sony.co.jp/ProductsParis/JAPAN/Professional/Camera/CCM-DS250-j.html

10 Business Wire, April 15, 1996: "intro Computer Products AG is Granted License for Sony DV Codec Source Code ..."

11 NAB, Las Vegas, April 16, 1996 release, see http://www.truevision.com/PRs/DVCPRO.html.

12 http://www.sony.co.jp/Exhibition/DVCap/index.html.

13 Microsoft, et al. press releases dated circa Jan. 8, 1996.

14 PR Newswire, April 1, 1996: "Microsoft unveils Simply Interactive PC framework."

15 Infoworld, April 1, 1996, p 1: "Developers rally behind new PC design."

16 PR Newswire, May 13, 1996: "Apple Plans to Put 1394/Firewire Inside Products, Aggressive Plans for Breakthrough Technology Revealed.".

Ex. 1038B

The Anatomy of a High-Performance Microprocessor A Systems Perspective

Bruce Shriver
Bennett Smith

IEEE COMPUTER SOCIETY

IEEE Computer Society Press
Los Alamitos, California

Washington • Brussels • Tokyo

ISBN Number

Google Exhibit No. 1063

Library of Congress Cataloging-in-Publication Data

Shriver, Bruce D., 1940–
The anatomy of a high-performance microprocessor: a systems perspective /
Bruce Shriver, Bennett Smith
p. cm.
Includes index.
ISBN 0-8186-8400-3
1. Microprocessors. 2. High performance computing.
3. Computer architecture.I. Smith, Bennett. II. Title.
QA76.5.S499 1998
004.16—dc21

98-17345
CIP

Copyright © 1998 by The Institute of Electrical and Electronics Engineers, Inc. All rights reserved.

Copyright and Reprint Permissions: See the *Copyright and Legal Notices* section which follows the Glossary/Index in this book. 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.

IEEE Computer Society Press Order Number BP08400
Library of Congress Number 98-17345
ISBN 0-8186-8400-3

Additional copies may be ordered from:

IEEE Computer Society Press
Customer Service Center
10662 Los Vaqueros Circle
P.O. Box 3014
Los Alamitos, CA 90720-1314
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: +1-732-981-1393
Fax: +1-732-981-9667
mis.custserv@computer.org

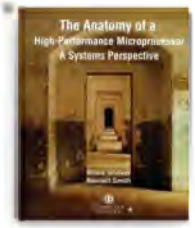
IEEE Computer Society
13, Avenue de l'Aquilon
B-1200 Brussels
BELGIUM
Tel: +32-2-770-2198
Fax: +32-2-770-8505
euro.ofc@computer.org

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

Publisher: Matt Loeb
Project Editor: Cheryl Baltes
Advertising/Promotions: Tom Fink
Production Editor: Lisa O'Conner
Printed in the United States of America



Ex. 1038C



BOOK

The anatomy of a high-performance microprocessor : a systems perspective / Bruce Shriver, Bennett Smith

Shriver, Bruce D., 1940-; Smith, Bennett

1998

Available at Linda Hall Library Books - 2nd Floor (QA76.5 .S49 1998) and other locations

Send to

Email | QR | Permalink | Citation | EasyBib | Export RIS | Export to Excel

Get It

LOCATIONS

Linda Hall Library
Available , Books - 2nd Floor ; QA76.5 .S49 1998
Go to Location Items

Linda Hall Library
Available , Closed Stacks - Books Digital Media ; CDRM 67
Go to Location Items

Details

Title	The anatomy of a high-performance microprocessor : a systems perspective / Bruce Shriver, Bennett Smith
Creator/Contributor	Shriver, Bruce D., 1940- > Smith, Bennett >
Subjects	Microprocessors > High performance computing > Computer architecture >
Identifier	ISBN : 9780818684005 ISBN : 0818684003 OCLC : (OCoLC)43318831
Publisher	Los Alamitos, Calif. : IEEE Computer Society
Creation date	1998
Format	xxxvii, 545 p. ill. ; 26 cm. + 1 computer laser optical disc (4 3/4 in.)
General notes	Includes index.

Virtual Browse



The Ray Dream handbook ...



Secure electronic transactions : introduction and technical ...



Guidelines for pressure relief and effluent handling systems ...

Links

Display Source Record >

Google Exhibit No. 1063
Page 1257

```
leader  00994cam a22003258a 4500
001     992984703405961
005     20100715021300.0
008     980311s1998    cau           001 0 eng
010     ##$a   98017345
019     ##$a38739047
020     ##$a9780818684005
020     ##$a0818684003
035     ##$a(OCoLC)43318831
035     ##$9411620
035     ##$a(MoKL)298470-lhalldb
035     ##$a(lhalldb)298470-lhalldb
035     ##$z(OCoLC)38739047
040     ##$aDLC $cDLC $dLHL
049     ##$aLHLA
050     00$aQA76.5 $b.S49 1998
100     1#$aShriver, Bruce D., $d1940-
245     14$aThe anatomy of a high-performance microprocessor : $ba systems perspective / $cBruce Shriver, Bennett Smith
260     ##$aLos Alamitos, Calif. : $bIEEE Computer Society, $c1998.
263     ##$a9807
300     ##$axxxvii, 545 p. $bill. ; $c26 cm. + $e1 computer laser optical disc (4 3/4 in.)
500     ##$aIncludes index.
650     #0$aMicroprocessors
650     #0$aHigh performance computing
650     #0$aComputer architecture
700     1#$aSmith, Bennett
948     ##$aLTI 09/01/2008
948     ##$aLTI 07/14/2010
948     ##$aLTI 10/19/2010
```

Ex. 1038D

LIBRARY | CATALOG

1 of 1

BOOK

The anatomy of a high-performance microprocessor : a systems perspective

Full Record MARC Tags

Personal name

Shriver, Bruce D., 1940-

Main title

The anatomy of a high-performance microprocessor : a systems perspective / Bruce Shriver, Bennett Smith

Published/Created

Los Alamitos, Calif. : IEEE Computer Society, c1998.

Request this Item LC Find It

Links

Links

Contributor biographical information http://www.loc.gov/catdir/bios/wiley046/98017345.html
Table of contents http://www.loc.gov/catdir/toc/wiley031/98017345.html
Publisher description http://www.loc.gov/catdir/enhancements/fy0706/98017345-d.html

More Information

Google Exhibit No. 1063
Page 1260

LCCN Permalink	https://lccn.loc.gov/98017345
Description	xxxvii, 545 p. : ill. ; 26 cm. + 1 computer laser disc (4 3/4 in.)
ISBN	0818684003
LC classification	QA76.5 .S499 1998
Related names	Smith, Bennett.
LC Subjects	Microprocessors. High performance computing. Computer architecture.
Browse by shelf order	QA76.5
Notes	Includes bibliographical references and index.
LCCN	98017345
Dewey class no.	004.16
Type of material	Book

Item Availability

CALL NUMBER	QA76.5 .S499 1998 Copy 1
Request in	See Reference Staff. By Appt in Jefferson Main RR (MRC)
Status	Not Charged

Google Exhibit No. 1063
Page 1261

LIBRARY CATALOG

1 of 1

BOOK

The anatomy of a high-performance microprocessor : a systems perspective

Full Record MARC Tags

000		01422cam a2200313 a 4500
001		1197741
005		20190118093512.0
008		980311s1998 caua b 001 0 eng
906	__	|a 7 |b cbc |c orignew |d 1 |e ocip |f 19 |g y-gencatlg
925	0_	|a acquire |b 1 shelf copy |x policy default
955	__	|a sc20 to ja00 03-13-98; jf06 to subj 03-13-98; jf11 to sl 03-13-98;jf12 03-13-98 to ddc; CIP ver. jf03 01-18-00; jf12 to BCCD 01-20-00; yj07 to BCCD 05-30-00
010	__	|a 98017345
020	__	|a 0818684003
035	__	|9 (DLC) 98017345
040	__	|a DLC |c DLC |d DLC
050	00	|a QA76.5 |b .S4991998
082	00	|a 004.16 |2 21
100	1_	|a Shriver, Bruce D., |d 1940-
245	14	|a The anatomy of a high-performance microprocessor : |b a systems perspective / |c Bruce Shriver, Bennett Smith
260	__	|a Los Alamitos, Calif. : |b IEEE Computer Society, |c c1998.
300	__	|a xxxvii, 545 p. : |b ill. ; |c 26 cm. + |e 1 computer laser disc (4 3/4 in.)
504	__	|a Includes bibliographical references and index.
650	_0	|a Microprocessors.
650	_0	|a High performance computing.

Google Exhibit No. 1063
Page 1262

650	_0	|a Computer architecture.
700	1_	|a Smith, Bennett.
856	42	|3 Contributor biographical information |u http://www.loc.gov/catdir/bios/wiley046/98017345.html
856	41	|3 Table of contents |u http://www.loc.gov/catdir/toc/wiley031/98017345.html
856	42	|3 Publisher description |u http://www.loc.gov/catdir/enhancements/fy0706/98017345-d.html

Request this Item | LC Find It

Item Availability

CALL NUMBER	QA76.5 .S499 1998 Copy 1
Request in	See Reference Staff. By Appt in Jefferson Main RR (MRC)
Status	Not Charged

Google Exhibit No. 1063
Page 1263

Ex. 1038E

Copyright
United States Copyright Office

Try the Copyright Public Records System (CPRS) pilot with enhanced search features and filters.

Help | Search | History | Titles | Start Over

Public Catalog

Copyright Catalog (1978 to present) at DC4
Search Request: Left Anchored Title = anatomy of a high-performance microprocessor
Search Results: Displaying 1 of 1 entries

previous next

Labeled View

The anatomy of a high-performance microprocessor : a systems perspective /...

Type of Work: Text
Registration Number / Date: TX0004938030 / 1998-10-15
Title: The anatomy of a high-performance microprocessor : a systems perspective / Bruce Shriver, Bennett Smith.
Imprint: Los Alamitos, CA : IEEE Computer Society Press, c1998.
Description: 545 p.
Notes: Includes CD-ROM.
Copyright Claimant: Institute of Electrical and Electronics Engineers (employer for hire)
Date of Creation: 1998
Date of Publication: 1998-07-01
Previous Registration: Some prev. pub. material.
Basis of Claim: New Matter: additional text.
Copyright Note: C.O. correspondence.
Names: Smith, Bennett
Shriver, Bruce D., 1940-
Institute of Electrical and Electronics Engineers

previous next

Save, Print and Email (Help Page)		
Select Download Format	Full Record	Format for Print/Save
Enter your email address:		Email

Help Search History Titles Start Over

Contact Us | Request Copies | Get a Search Estimate | Frequently Asked Questions (FAQs) about Copyright | Copyright Office Home Page | Library of Congress Home Page

Google Exhibit No. 1063
Page 1265

DESIGN OF A NOVEL SYNTHESIS FILTER FOR REAL-TIME MPEG-2 AUDIO DECODER IMPLEMENTATION ON A DSP CHIP

Won-Kyu Paik and Sun-Young Hwang
CAD & Computer Systems Lab, Dept. of Electronic Engineering
Sogang University, C.P.O. Box 1142, Seoul, 100-611, Korea
E-mail: hwang@ccs.sogang.ac.kr

Abstract – **This paper describes the design of a novel synthesis filter for real-time implementation of MPEG-2 audio decoder on a single, programmable DSP chip. In the first phase of our design we propose a fast subband synthesis filter bank employing a decomposed IDCT block with reduced multiplication stages, shared by both MPEG-1 and MPEG-2 decoders. In the second phase of the design, a parameter analyzer block is adopted to take advantage of the multichannel computing structure in MPEG-2 thereby further reducing the number of MAC operations. In the final phase, we propose a low-cost decoder employing unorthodox methods of boosting performance by gracefully degrading quality of service.**

The overall runtime of the decoder has been reduced by 60% when compared to the modified ISO decoder employing a synthesis filter bank with fast IDCT block. Real-time performance validations have been made by re-tuning time-critical code sections for both native VLIW and DSP architectures. The final implementation successfully passes all ISO/IEC MPEG-2 audio compliance tests and achieves near-transparent audio decoding and signal reconstruction on the fly.

I. Introduction

The demand for reductions in data rate for wideband digital audio signal transmission and storage in multimedia applications substantiates the need for efficient compression techniques. A serial bitstream of CD quality audio material consists of 44,100 16-bit PCM samples per channel which corresponds to 1.4 Mbps in stereo [1]. Audio at that bitrate contains a lot of redundancies and this led to the development of perceptually coded, lossy compression techniques to eliminate information that is irrelevant from a human sound-perception viewpoint. Studies have shown that by masking inaudible components of audio signals via psychoacoustic models [1][2], perceptual audio systems are able to achieve significant compression while maintaining perceived audio quality roughly equal to the original.

The International Organization for Standardization (ISO) has established a working group known as the Motion Picture Experts Group (MPEG), to develop standards for coding audio-visual presentation materials aimed at digital storage media such as CD-ROMs, not exceeding a combined continuous transfer rate of 1.5 Mbps. The audio portion specifies the coded representation of high-quality audio signals intended for stand-alone or multiplexed-with-video applications [3][4]. The MPEG audio system features a wide mix of channel configurations, multiple sampling and data rates, layer compatibility, decoder version scalability, dynamic range control, Dolby Pro Logic compatibility, and multi-lingual support, thereby addressing to a variety of consumer and professional applications. Application areas run a wide gamut from DVD players, multichannel High Definition Television (HDTV), Integrated Service Digital Network (ISDN) transmission codecs, cable and satellite-based TV (e. g. DVB, USSB) services, Digital Audio Broadcasting (DAB), studio recording and editing, to multimedia/Internet audio.

Contributed Paper
Manuscript received July 7, 1999
0099 3063/99 $10.00 © 1999 IEEE

Authorized licensed use limited to: Reprints Desk. Downloaded on February 03,2025 at 18:39:53 UTC from IEEE Xplore. Restrictions apply.

Google Exhibit No. 1039
Page 001

Google Exhibit No. 1063
Page 1266

MPEG-2 was first developed with the broadcast application in mind. The concept of a "1-encoder to N-decoders," makes for non-symmetric systems with a relatively complex, high-quality encoder on one end and fast, low-complexity, low-cost decoders at the other delivering quality service on limited bandwidth. Hence, some researches leading to the real-time implementation of MPEG standards have based the decoder designs on programmable DSPs [5][6], for their merits in easy implementation of complex algorithms, reduced design cycles, and development cost. A floating-point implementation of a real-time MPEG-1 system was presented by Murphy [5]. But limitations to processing speed and memory of the host DSP chip prohibited the implementation of a fully functional MPEG-1 system on a single DSP, let alone multichannel systems. Similar work has been found implementing the MPEG-2 standard on a fixed-point, multimedia DSP [6], indicating a move toward low-cost, uniprocessor solutions. Fixed-point implementations are cost-effective, but complications may be introduced by the repetitive computational tasks typically populated in signal processing applications. Rounding-off error introduced by multiplying two fixed-point data are trivial as is, but the propagation of these smaller errors onto successive multiplication stages ultimately corrupts reconstructed signal quality to a noticeable level. The synthesis filter bank as defined by the standard, is considered by far the most computationally intensive block in the entire decoding system well populated with MAC operations, thereby efforts to reduce the complexity of the synthesis filter has been essential for real-time decoder implementations. The most common practice in reducing the number of multiplications and additions, has been by exploiting the structural properties of the IDCT block resulting in manifold decoder performance improvements [7]. There have been other fast IDCT schemes that are effective for designing speedy synthesis filter blocks [8]. However, not all IDCT schemes are suitable for fixed-point implementation, since a trade-off between the number of multiplication operations and rounding errors due to splitting the internal IDCTs exist. To achieve fast decoding of high quality audio, it is essential to keep the numerical errors to an acceptable level without running into hard real-time constraints of the decoding system. Further improvements to the synthesis filter design can be achieved by exploiting the multichannel computation structure of MPEG-2, thus contributing for a faster MPEG-2 dedicated decoder operation. This paper fully exploits the multichannel operation of the MPEG-2 decoding system to reduce the number of MAC operations.

In this paper, we present the design and implementation of a real-time MPEG-2 audio decoder with an improved synthesis filter on a single, fixed-point DSP. The proposed system employs a decomposed IDCT block with reduced multiplication stages, and a parameter analyzer block for dedicated MPEG-2 decoding. The paper is organized as follows: Section II provides an overview into the MPEG-2 multichannel audio decoding process. In section III, the we propose a versatile subband synthesis filter routine employing a decomposed IDCT block with reduced multiplication stages, followed by optimizations for dedicated multichannel computation structure. For cost-effective measures, we propose unorthodox methods of gracefully degrading audio signal quality for faster decoder speed. The enhanced decoder is realized on the DSP board, mounted on a host PC. In section IV, experimental results are reported, and a conclusion is drawn at the final section of the paper.

II. MPEG-2 Multichannel Audio Coding System

The MPEG audio standards, ISO/IEC 11172-3 (MPEG-1) [3] and ISO/IEC 138180-3 (MPEG-2) [4] define three layers of increasing complexity and subjective performance varying with compression efficiency. With successive layers, however, the coding/decoding system is subject to increased delay and sensitivity to transmission errors.

2.1 Overview of the MPEG Audio Coding System

The main decoding sub-blocks of the MPEG audio decoder are comprised of a frame unpacker, dequantizer, dematricing block (MPEG-2), and a synthesis filter bank. After frame disassembly from the input encoded bitstream, the decoder passes the bit-allocation, scalefactor side-informations and audio samples to the inverse quantizer for restoring the quantized subband values. These subband samples are fed into the synthesis filter block for reconstruction and output of PCM samples on a per channel basis.

Authorized licensed use limited to: Reprints Desk. Downloaded on February 03,2025 at 18:39:53 UTC from IEEE Xplore. Restrictions apply.

Google Exhibit No. 1039
Page 002

Google Exhibit No. 1063
Page 1267

2.2 Multichannel Extension

Restrictions to two-channel stereophony imposed by the MPEG-1 standard are lifted with multichannel sound defined by ISO/IEC 138180-3. The extension to the multichannel system provides support for 5.1 channels without simulcast intended for broadcast complying with the HDTV format. The MPEG-2 decoding system shares the same audio algorithm core as MPEG-1, hence providing forward and backward compatibility [1]. An MPEG-2 decoder will accept an MPEG-1 formatted bitstream and an MPEG-1 decoder can derive a stereo representation from an MPEG-2 bitstream. The multichannel format uses an encoder matrix that allows a two-channel decoder to decode stereo compatible signals that is a subset of a multichannel bitstream. The multiple channels are matrixed to form compatible MPEG-1 left/right channels, as well as other MPEG-2 channels, as shown in Figure 1. These are encoded into MPEG-1 frames, and additional multichannel data is placed in the extended auxiliary data field. MPEG-2 features intensity coding, crosstalk reduction, inter-channel prediction coding, and center channel phantom image coding to achieve a combined bit rate of 384 kbps, using Layer II at a 48-kHz sampling frequency.

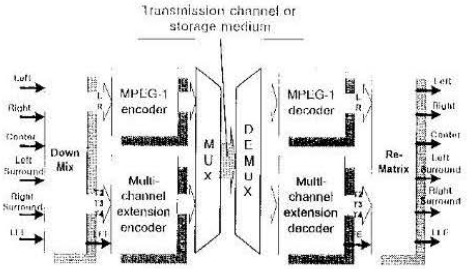


Fig. 1. Forward and backward compatible MPEG-2 audio system.

2.3 Synthesis Filter Bank

The subband synthesis filter bank is a polyphase frequency-time domain transform engine, shared by both MPEG-1 and MPEG-2 algorithms. After the dequantization process from the previous stage, 32 subband samples per channel are input to the synthesis filter bank and 32 reconstructed time-domain PCM samples are output to a D/A converter for reproduction. As shown in Figure 2, the synthesis filter bank is defined by the MPEG standard with three key operations: Initialization, Matrix transformation, and Window/overlap-add. Initialization involves shifting of 64 samples, from a queue of 1024 points at a time. The matrix transform operation receives the 32 frequency-domain samples and returns 64 intermediate elements, which are written to the input IQMF buffer. The buffer has 1024 elements for every channel and is cyclic.

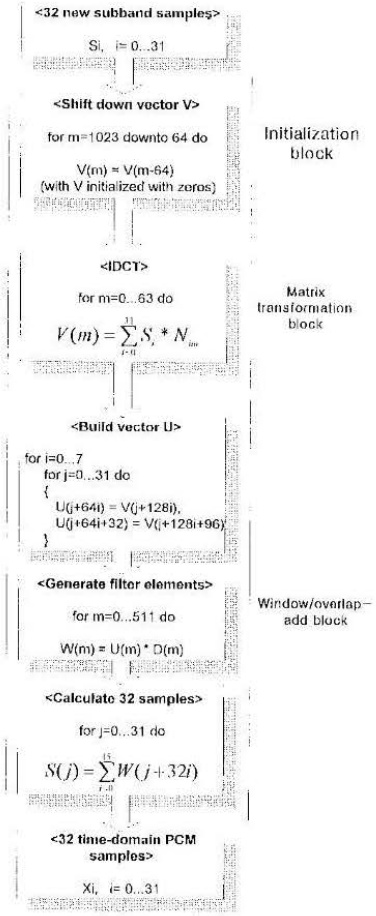


Fig. 2. Synthesis filter bank flow [3].

Authorized licensed use limited to: Reprints Desk. Downloaded on February 03,2025 at 18:39:53 UTC from IEEE Xplore. Restrictions apply.

Google Exhibit No. 1039
Page 003

Google Exhibit No. 1063
Page 1268

The window/overlap-add block performs the actual filtering operations. The filtering operation is performed on 512 elements that includes the last 64. A new vector is built from these elements and is multiplied with the tabulated windowing coefficients in the IQMF block to produce 32 time-domain samples. The 32 samples are formatted into PCM samples for output. In the next iteration, all the elements of V are shifted down 64 places in preparation for the next IDCT output. These operations are performed recursively for each represented channel.

Table 1 displays the profiling output of the ISO MPEG-2 decoder software implementation on UNIX and DSP platforms, respectively. The profile result is an average taken from 20 consecutive frames of a random mix of multichannel audio clips, including the ISO/MPEG-2 conformance test streams. This table indicates that the synthesis filter bank is the most computationally intensive block in the entire decoder, therefore any design improvements made here would yield a faster decoder. For MPEG-2 multichannel operation, the speedup is more pronounced since the synthesis filter iterations increase in direct proportion to the number of represented channels.

Table 1. Profiling of the ISO MPEG-2 decoder functions.

Decoding Stage	Percentage of total time spent in each *decoding* task	
	UNIX	DSP
Synthesis filter	90%	83%
Dequantizer	4%	7%
Stream decoder	2%	4%
Dematrix & Denormalizer	3%	4%
Others	1%	2%

The most DSP cycle-hungry block in Figure 2 of the synthesis filter bank is the matrix transformation block. As shown in Table 2, any improvements to the synthesis filter bank centers around a more efficient IDCT block, requiring optimization of nested loops and reductions in number of multiplications in the loop kernels.

Table 2. Profiling of the ISO synthesis filter sub-blocks.

Synthesis filter bank sub-blocks	# multipli -cations	# addi-tions	# DSP cycles	Percentage of total cycles consumed
IDCT	2,048	1,984	410k	91%
Window/ overlap-Add	512	480	32k	7%
Others	0	0	6k	2%

III. The Proposed Synthesis Filter

As presented in the previous section, optimizations to the synthesis filter are essential for a fast realization of the MPEG decoder. In this section, we first propose a subband synthesis filter bank employing decomposed IDCT and windowing blocks, shared by both MPEG-1 and MPEG-2 decoders. Next, we propose the addition of a parameter analyzer block for further performance improvements to MPEG-2 decoders, taking advantage of the multichannel computing structure. Finally we propose a synthesis filter employing graceful degradation techniques targeting low-cost implementations with marginally degraded audio output quality.

Figure 3 shows the proposed synthesis filter block shared by both MPEG-1 and MPEG-2 multichannel decoders. It implements a modified IDCT block with a multistage butterfly network at the input followed by decomposed IDCT and efficient windowing functions. Matched against other well-known 32-point IDCTs [8] the IDCT scheme employed here profits from minimal fixed-point arithmetic propagation errors while reducing MAC counts.

Authorized licensed use limited to: Reprints Desk. Downloaded on February 03,2025 at 18:39:53 UTC from IEEE Xplore. Restrictions apply.

Google Exhibit No. 1039
Page 004

Google Exhibit No. 1063
Page 1269

3.1 Design of a Fast MPEG-1/2 Synthesis Filter Bank

In this section we propose a fast subband synthesis filter bank employing a decomposed IDCT block with reduced multiplication stages, shared by both MPEG-1 and MPEG-2 decoders. The IDCT scheme employed in the proposed design recursively decomposes the IDCT block into smaller matrices by exploiting the symmetric properties of the cosine transform. An efficient windowing scheme has been proposed that exploits the symmetries observed in the IDCT output.

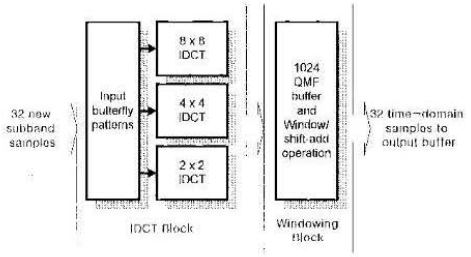


Fig. 3. Block diagram of the proposed synthesis filter.

3.1.1 IDCT Block with Decomposed Matrix Transformations

Symmetric Properties of the Cosine Transform: The IDCT transforms the input subband samples from frequency to time-domain by using Equation (1) as defined in the MPEG standard.

$$V(i) = \sum_{sb=0}^{31} N(i,sb)X(sb) = \sum_{sb=0}^{31} \cos\left[\frac{\pi}{64}(2sb+1)(i+16)\right]X(sb), \quad i = 0...63 \tag{1}$$

Vector V denotes an array of 64-intermediate values needed at the window/overlap-add block, the discrete cosine terms $N(i,sb)$, and subband coefficients $X(sb)$. To evaluate V as is, 2,048 multiplications and 1,984 additions are required as shown in Table 2. However, V can be built exploiting the inherent symmetries of $N(i,sb)$ as shown in Equation (2), thereby using only half the multiplications and additions required in Equation (1).

$$V(i) = -V(32 - i)$$
$$V(32 + i) = V(64 - i), \qquad i = 1...16 \tag{2}$$

Butterfly Stage: The 32-point IDCT derived from Equation (2) can be further decomposed into two 16 × 16 even and odd matrices by way of shuffling and grouping X(sb) into butterfly patterns, denoted in Equation (3). Thus, an N-point IDCT of Equation (2) can be decomposed into sums of two N/2-point IDCTs, ontaining a four-fold reduction in the number of MAC operations. As shown in Figure 3, the first input butterfly pattern of X(sb), can be further decomposed in a similar manner by exploiting trigonometric and structural properties, thereby reducing the complexity of the IDCT block of the synthesis filter.

$$V(i) = \sum_{sb=0}^{16} N(i,sb)\left[X(sb) + X(31 - sb)\right], \quad i = 0,2,...,30$$
$$V(i) = \sum_{sb=0}^{16} N(i,sb)\left[X(sb) - X(31 - sb)\right], \quad i = 1,3,...,31 \tag{3}$$

Successive Decompositions: By recursively breaking down the input even 16 × 16 butterfly pattern into 8 × 8, 4 × 4, and 2 × 2 parts, the IDCT can be further decomposed into smaller transform matrices. The However, decomposing any odd butterfly pattern requires performing multistage multiplication with non-unity trigonometric multipliers, which are deleterious for fixed-point arithmetic. It is imperative to keep the propagating rounding errors to a minimum, hence multiplication is performed at the end of the butterfly decomposition stage and any splitting of internal IDCT that require intermediate non-unity multiplications are omitted. The generation of butterfly patterns requires only addition and subtractions but no multiplications. Once the butterfly patterns are completely broken down, matrix transformation is performed.

Authorized licensed use limited to: Reprints Desk. Downloaded on February 03,2025 at 18:39:53 UTC from IEEE Xplore. Restrictions apply.

Google Exhibit No. 1039
Page 005

Google Exhibit No. 1063
Page 1270

3.1.2 Windowing Block

A 511th-order IQMF filter bank with a 96-dB sidelobe rejection and a steep pass-to-stop-band transition is used to perform the window/overlap-add block upon the 1024 intermediate element output from the IDCT block. A new vector is built from these elements and is multiplied with the tabulated windowing coefficients *D(m)* to generate the 512-window elements. A 16-point filtering operation is performed on the generated filter elements to produce 32 time-domain samples *S(i)*. Each *S(i)* are formatted into PCM audio samples, and output to the channel FIFO buffers for DAC.

$$\begin{aligned} U(j+64i) &= V(j+128i) \\ U(j+64i+32) &= V(j+128i+96), \\ & \qquad j=0\ldots31, \quad i=0\ldots7 \\ W(m) &= U(m)D(m), \quad m=0\ldots511 \\ S(j) &= \sum_{i=0}^{15} W(j+32i), \quad j=0\ldots31 \end{aligned} \tag{4}$$

Equation (4) holds when the inverse transform given in Equation (1) is re-defined, by exploiting the symmetrical properties of the windowing coefficients D. The odd and even input butterflies derived from Equation (3) for the calculation of *V*.

The 512 coefficients are split into 16 blocks of 32 *V* coefficient, the even numbered blocks are multiplied with the first 32 IDCT outputs, and the odd blocks are multiplied with the next 32 IDCT outputs. The windowed results of 16 blocks are then overlapped and added, resulting in half the multiplication and additions of the original.

3.2 Optimizations Exploiting MPEG-2 Computing Structure

As stated earlier, MPEG-2 provides full MPEG-1 support and features key multichannel operations such as dynamic crosstalk reduction, inter-channel prediction coding, and center channel phantom image coding. In this section, a parameter analyzer block is adopted to take advantage of the multichannel computing structure thereby achieving further performance improvements. With the availability of dynamic bit-allocation information and multichannel coding parameters after frame parsing from the encoded bitstream, the synthesis filter loop can be optimized to exploit the redundancies and zero-allocation profiles of subband samples, resulting in the elimination of wasted MAC operations.

3.2.1 Exploiting Intra, Inter-channel Redundancies

The MPEG-2 multichannel decoder exploits the use of information from neighboring channels in order to achieve optimal data reductions without resorting to simulcast operations. One of the main features supported by the MPEG-2 decoding system is dynamic crosstalk reduction, which reduces the amount of data through copying stereo-irrelevant signal components for extension channel reconstruction [4]. With the existing ISO MPEG decoders, rescaling, dequantizing, dematricing and denormalizing operations precede the synthesis filter bank operation. However, the proposed decoder performs the synthesis filtering prior to other operations, thereby if dynamic crosstalk reduction is used, the application of the synthesis filter bank to extension channels is bypassed altogether. Thus inter-channel redundancies can be exploited at the decoder level, eliminating unnecessary iterations of costly IDCT and windowing operations resulting in savings of the number of MAC operations.

In a similar context, intra-channel redundancies are exploited using a temporary buffer for storage of intermediate *V*s. If a match is found in the next iteration of the synthesis filter, the IDCT operation is omitted, permitting us to reuse older *V* coefficients for windowing. A match can be found by comparing the values of subband bit allocation and corresponding sample coefficients. The intra-channel technique exploits the temporal locality of periodic audio signals.

3.2.2 IDCT by Subband Coefficient Analysis

An examination into subband coefficient profiles for various audio streams indicate that not all subband coefficients are allocated with bits. For each zero subband coefficient in an N-point DCT, N multiplications and N-1 additions are saved. Skipping the DCT matrix operation for zero subband coefficients permits us to reuse older V vector

Authorized licensed use limited to: Reprints Desk. Downloaded on February 03,2025 at 18:39:53 UTC from IEEE Xplore. Restrictions apply.

Google Exhibit No. 1039
Page 006

Google Exhibit No. 1063
Page 1271

coefficients for windowing. Statistically then, about 50% of the 2048 multiplies and 1984 additions required in the original DCT implementation are eliminated. Empirical analysis results dictate to an average of 80% reduction in multiplies and 70% adds per 10 frame samples. These averages were taken over a wide range of diverse of audio clips, including jazz, classical, vocal pieces. A combination of DCT optimizations exploiting symmetric properties and subband coefficient analysis yields to empirical results of 90% reduction of multiplies and 80% reduction of additions compared to the DCT performance of the ISO/MPEG decoder.

Figure 4 shows the bit allocation profile across the spectral subtends of an audio frame fragment. If no bits were allocated for a particular subband index, any attempts at applying the matrix transform operation would only result in a waste of CPU cycles. For efficiency, zero bit allocation samples are dropped from the IDCT operation, thereby relieving the synthesis filter bank, with no performance compromises. Benefits are added by applying bit-allocation analysis to surround sound channels that have spares bit-allocation profiles, or to the bandwidth-limited phantom-coded center channel.

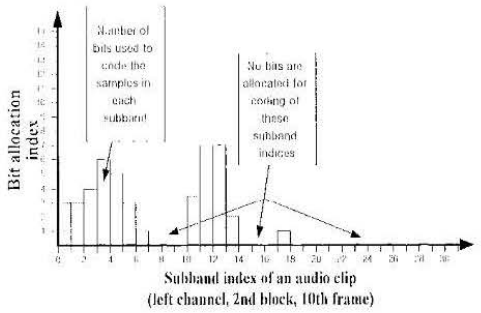


Fig. 4. Bit allocation across subbands.

Our statistical analysis of subband coefficients indicates that close to 50% of the subband coefficients are not allocated across a diverse selection of musical scores and compliance bitstreams. These are the average values for 10 frames of various bit and sampling rate configurations. Empirical results show that the upper half of the subband indices have a higher probability of zeroing out.

3.3 Graceful Degradation

Providing uninterrupted audio at lower-cost requires accurate real-time decoder system complexity measure such that efficient resource allocation can be attained. However, the measure of processing power for real-time operation of a software-driven decoder remains a difficult task. We assume that the absolute measure of decoder complexity can be determined by the total decoding time with respect to the bitstream or in more abstract measures, elementary operations such as the IDCT / windowing operations.

For each audio frame, the sub-band filter is repeated 12 or 36 times per channels depending on the Layer. Since the IDCT and windowing blocks are executed with every iteration of the synthesis filter, they form the kernels that lie on the "critical path" of the decoding process. Apart from scaling down the multichannel stream (from a 3/2 configuration to 2/1 for example) a more fine-grained approach of gracefully degrading quality of audio service is proposed in this paper.

We devised two ways of gracefully degrading the audio output: Partial zeroing of the $D(m)$ window coefficients, and eliminating the upper half subband coefficients of the spectral image. As shown in the latter half of the window/overlap-add block of Figure 2, 32 time-domain samples $S(sb)$ are calculated by Equation (4) with the generated 512 $W(m)$. However, about 40% of the values of $D(m)$ are close to being zero, hence the 16-point windowing operation can be reduced. It is also observed that most of the upper half of the subband samples are zero. Hence, IDCT can be performed for only 16 subband samples. The combined effort provides for minimal quality of service degradation for speedy decoding.

3.4 DSP Implementation

The proposed MPEG-2 decoder designs were implemented on a high-performance DSP with a VLIW architecture. To maximize real-time performance, platform dependent and independent optimizations were performed on the critical code segments. Timing of real-time communication of the I/O subsystem, between host and the DSP chip and

Authorized licensed use limited to: Reprints Desk. Downloaded on February 03,2025 at 18:39:53 UTC from IEEE Xplore. Restrictions apply.

Google Exhibit No. 1039
Page 007

Google Exhibit No. 1063
Page 1272

control of the output channel buffers, is essential to ensure continuous I/O operations.

3.4.1 DSP Architecture

The proposed MPEG-2 audio decoder has been implemented and tested on a commercial DSP development board, mounted on a host PC [9]. The processor has a 200 MHz, 16-bit fixed-point VLIW core, claimed to run at 1,600 MIPS. It implements a load-store architecture and manifests two cross-linked datapaths, each having four execution units, a split 4-Mbyte on-chip program and data memories, a uniform general-purpose register file, operating on 32/40-bit operands. The eight execution units include two multipliers, four ALUs, two shift/logical units. Hence, the processor can execute up to eight orthogonal 32-bit RISC-like instructions in a single cycle, enabling it to achieve the high level of parallelism. Salient DSP features such as saturation logic and circular addressing are supported. The development board is equipped with an array of peripherals such as 8 Mbytes of global memory, Host Port Interface (HPI), external JTAG scan path emulation support, two-buffered serial ports, 16-bit stereo audio interface supporting standard sample rates of up to 48 kHz.

Implementing real-time multimedia standards such as MPEG audio on programmable DSPs requires code adaptations to the native hardware. Despite the ease in translating the MPEG standard by a high-level language such as C, the efficient use of the processing power of the VLIW machine remains a difficult task. For maximal utilization of on-chip hardware resources, we have adopted both DSP and VLIW architectural optimizations to the MPEG algorithm. VLIW architecture dependent and independent optimization techniques such as software pipelining, loop unrolling and other DSP intrinsics [9] have been employed to resolve time-critical loop blocks of the decoder.

3.4.2 Real-time Communications

Timing of real-time operations ensures correct synchronous operation: all data is produced and consumed at exactly the rate that it is output or input. Decompression of audio signals are characterized by an amplified data output that far exceeds the consumed input of encoded data, creating a processing core that far exceeds that of the I/O subsystem. The hardware architecture is designed to connect directly to a PCI bus, as a PCI bus master, and to enable efficient use of the PCI bus for real-time communication of data between the DSP and the host. These built-in features enable efficient software to be produced for processing the streaming audio data. The software does not need to be aware of the input data format, as it will be converted to the standard 16 bit stereo format without intermediate modifications. In addition, the software need not be aware of the origin of the data. The DMA engine automatically places the input data into a circular buffer. When data has been consumed from the buffer the hardware automatically create the DMA request and queue it for filling. A similar process is used to empty out buffers to the DAC.

Once the data is available upon the completion of a synthesis filter iteration, the reconstructed audio PCM data are rounded to the DAC word width and copied to the corresponding output channel FIFO buffer. When buffers are full for all the specified channels, audio data are output through the DAC. In contrast with the input buffer stage, the decoder output processing uses very few MIPS but a relatively large amount of memory. For each output channel, the decoder requires 1152-long PCM sample buffers.

IV. Experimental Results

The performance of the proposed MPEG algorithm has been validated on a UNIX and implemented on the DSP chip, using MPEG-2 compliance test stream sequences. Tables 3 and 4 show the experimental results for the proposed decoder designs, compared with the performance of the ISO decoder and a modified ISO decoder with the synthesis filter employing a fast IDCT [7].

In Table 3, the total number of multiplications and additions required in the proposed synthesis filter has been reduced to an average of 60% when compared to the modified ISO synthesis filter sub-blocks. The average execution time information of the proposed DSP-based decoder is reported in Table 4, for MPEG-1 and MPEG-2 multichannel bitstreams. Note that the direct implementation of the ISO decoder on the DSP chip does not provide for successful real-time performance, as portrayed by the Jazz and Classical clips. The proposed decoder

Authorized licensed use limited to: Reprints Desk. Downloaded on February 03,2025 at 18:39:53 UTC from IEEE Xplore. Restrictions apply.

Google Exhibit No. 1039
Page 008

Google Exhibit No. 1063
Page 1273

achieves an average of 60% speedup over the modified ISO decoder.

The results of Table 4 have direct implication upon deciding the lower bound for DSP platforms for which decoder systems can be implemented. Albeit the modified ISO decoder meets the current real-time specifications, it is certainly not a cost-effective implementation since the hardware on which it was implemented is one of industry's high-end DSP cores available today. Consequently, a faster decoder design opens the possibility for a more cost-effective implementation. Since real-time environments put a premium on the availability of hardware resources, cost-effective implementations are imperative.

Another low-cost implementation for the MPEG algorithm lies in adopting computationally graceful degradation (CGD) techniques to computation intensive blocks of the decoder. CGD-based designs provide minimum Quality of service (QoS) when the processing needs exceed the available resources. We found those further reductions in MAC counts and decoder speedup, were possible by a combination of eliminating the upper half of the subband coefficients and utilizing a trimmed *D(m)* windowing function. The combined effort gives a guarantee towards meeting hard real-time constraints of the cost-effective decoding systems.

The proposed decoder has been implemented on high-performance fixed-point VLIW DSP. The final implementation provides for near-transparent real-time audio decoding and reconstruction of PCM samples.

Table 3. Number of operations in IDCT and window schemes.

Operations		ISO	Modified ISO	Proposed					
				For MPEG-1/2		For MPEG-2 only		Degraded MPEG-1/2	
				#	Δ%	#	Δ%	#	Δ%
IDCT block	# Multiplications	2,048	512	384	-35	288	-43	156	-70
	# Additions	1,984	496	350	-30	204	-58	168	-66
Windowing block	# Multiplications	512	512	256	-50	256	-50	198	-61
	# Additions	496	496	256	-52	256	-52	225	-54

* Experimental results are compared against the modified ISO decoder.

Table 4. Execution time profiles of real-time DSP-based MPEG decoders.

Audio Clips	Total play time (sec)	Execution times of decoders (sec)				
		ISO	Modified ISO	Proposed		
				For MPEG-1/2	For MPEG-2 only	Degraded MPEG-1/2
Jazz	32.8	36.5	18.7	11.2	9.6	7.5
Classical	58.1	59.1	32.5	22.3	18.9	14.1
Vocal (MPEG-1)	24.3	21.4	10.5	7.1	-	4.8
Conformance test	5.6	4.2	2.3	1.6	1.4	0.9

* Experimental results are compared against the modified ISO decoder.

Authorized licensed use limited to: Reprints Desk. Downloaded on February 03,2025 at 16:39:53 UTC from IEEE Xplore. Restrictions apply.

Google Exhibit No. 1039
Page 009

Google Exhibit No. 1063
Page 1274

V. Conclusion & Future Researches

In this paper, we described a real-time realization of an MPEG-2 audio decoder on a single, fixed-point, VLIW DSP chip. The proposed subband synthesis bank employs a decomposed matrix transform block for both MPEG-1 and MPEG-2 decoding, and an additional parameter analyzer block for multichannel decoding. A decoder design implementing graceful degradation techniques has been proposed as a low-cost alternative, by keeping output audio quality to a minimal level while hiking up performance. Our effort concentrated on reducing the number of MAC operations of the computationally intensive synthesis filter bank while preserving signal quality integrity.

For real-time realization on a DSP chip, both DSP and VLIW architectural optimizations were performed. Optimal performance results have been obtained by judiciously adopting platform dependent and independent optimization techniques such as software pipelining loop unrolling, and DSP intrinsics, without falling back on the usual bottom-up assembly patch-up. Despite the versatility of translating and mapping complex standards onto the native DSP hardware via high-level language, efficient compiler targets, robust and tightly integrated debug/simulation environments are still yearned for.

The final realization provides a 60% faster decoder when matched against the modified ISO decoder implementing a fast IDCT block in the synthesis filter bank. Real-time performance obtaining near-transparent audio decoding and reconstruction of PCM samples, have been achieved. The techniques described in this paper can be extended the implementation of multiplexed-with-video applications, stand-alone low-cost decoder solutions, or elevated to core-based ASIC designs with few modifications. As for future CGD-based real-time decoder implementations, a practical complexity estimation policy for programmable cores need to be developed in order to accurately predict decoding time, and thereby guaranteeing real-time performance with resource allocation constraints.

Acknowledgement

This work was supported partly by Daewoo Electronics Co. as part of Digital VDR Technology Development Project.

References

[1] K. Pohlmann, Principles of Digital Audio, 3rd edition, McGrawHill: New York, 1997.

[2] E. Zwicker and H. Fastl, Psychoacoustics: Facts and Models, Springer-Verlag: New York, 1990.

[3] ISO-IEC JTC1/SC29/WG11, 'Information Technology - Coding of Moving Pictures and Associated Audio for Digital Storage Media at up to about 1.5 Mbit/s - Part 3: Audio,' IS 11172-3, Nov. 1992.

[4] ISO-IEC JTC1/SC29/WG11, 'Information Technology - Generic Coding of Moving Pictures and Associated Audio Information - Part 3: Audio,' IS 13818-3, May 1995.

[5] C. Murphy and K. Anandakumar, 'Real-Time MPEG-1 Audio Coding and Decoding on a DSP Chip,' IEEE Trans. on Consumer Electronics, Vol. 43, No. 1, Feb. 1997, pp. 40~47.

[6] J. Chen and H. Tai, 'Real-time Implementation of the MPEG-2 Audio Codec on a DSP,' IEEE Trans. on Consumer Electronics, Vol. 44, No. 3, Aug. 1998, pp. 866~871.

[7] T. Tsai, L. Chen, and Y. Liu, 'A Novel MPEG-2 Audio Decoder with Efficient Data Arrangement and Memory Configuration,' IEEE Trans. on Consumer Electronics, Vol. 43, No. 3, Aug. 1997, pp. 598~604.

[8] K. Rao and P. Yip, Discrete Cosine Transform, Academic Press Inc.: London, 1990.

[9] Texas Instruments, TMS320C62x/C67x Programmer's Guide, 1998.

Authorized licensed use limited to: Reprints Desk. Downloaded on February 03,2025 at 18:39:53 UTC from IEEE Xplore. Restrictions apply.

Google Exhibit No. 1039
Page 010

Google Exhibit No. 1063
Page 1275

Biographies



Won-Kyu Paik received the B.S. degree in electronic engineering from Sogang University, Seoul, Korea, in 1997. He is currently working towards the M.S. degree in electrical engineering at Sogang University. His current research interests include digital video/audio compression, real-time DSP systems, and CAD systems.



Sun-Young Hwang received the B.S. degree in electronic engineering from Seoul National University, Seoul, Korea, in 1976, the M.S. degree from Korea Advanced Institute of Science in 1978, and the Ph.D. degree in electrical engineering from Stanford University, California, U.S.A., in 1986. Since 1986, he has been with the Center for Integrated Systems at Stanford University, working on design of a high-level synthesis and simulation system. In 1986 and 1987, he held a consulting position at Palo Alto Research Center of Fairchild Semiconductor Corporation. In 1989, he joined the Department of Electronic Engineering at Sogang University, where he is now a professor. His current research interests include hardware/software co-design, and DSP/VLSI systems design.

Authorized licensed use limited to: Reprints Desk. Downloaded on February 03,2025 at 18:39:53 UTC from IEEE Xplore. Restrictions apply.

Google Exhibit No. 1039
Page 011

Google Exhibit No. 1063
Page 1276

Ex. 1039A

IEEE
TRANSACTIONS
ON
CONSUMER
ELECTRONICS

45
AUG-NOV
1999

TK 6563
.I2

Google Exhibit No. 1063
Page 1279

IEEE TRANSACTIONS ON

CONSUMER ELECTRONICS

NOVEMBER 1999 VOLUME 45 NUMBER 4 ITCEDA (ISSN 0098-3063)

A Publication by the IEEE Consumer Electronics Society

Administrative Committee i
Officers and Committee Chairmen ii

An Adaptive Digital Image Watermarking Technique for Copy Right Protection ..... C.-H. Lee and Y.-K. Lee 1005
Color Consistency Between Different Printing Materials Using Dot Gain and Intensity Compensation ..... C.-H. Lee, C.-S. Lee, and Y.-H. Ha 1016
Block Coded Modulation for the Reduction of the Peak to Average Power Ratio in OFDM Systems ..... P. Fan and X.-G. Xia 1025
Method for Improving Low-Resolution Video on Higher-Resolution Displays ..... T. S. Rzeszewski 1030
Experimental Characterization of a Direct Sequence Spread Spectrum System for In-House Wireless Infrared Communications ..... R. Vento, J. Rabadán, R. Pérez-Jiménez, A. Santamaría and F. J. López-Hernández 1038
Experimental Investigation of Vibration of the Shadow Mask with Different Assembly Conditions ..... W.-C. Wang and K.H. Lai 1046
Adaptive Watermarking by using Pixel Position Shifting Technique ..... K. S. Ng, L. M. Cheng, L. L. Cheng and M. K. Wong 1057
An Autonomous I/O Module for Low-Cost Media On-Demand Servers ..... H. Batatia, D. Xu 1065
Secure Communication Mechanisms for GSM Networks ..... C.-C. Lo and Y.-J. Chen 1074
A Flexible Design for Timing Signals Generation for the Conversion of Computer Video Formats to SDTV 480P ..... A. Moreno and E. John 1081
Wide Band AC Power Line Characterization ..... D. Liu, E. Flint, B. Gaucher, and Y. Kwark 1087
A Digital Multistandard Paging Receiver ..... J. Vuori 1098
Efficient FFT and Equalizer Implementation for OFDM Receivers ..... S. A. Fechtel, A. Blaickner 1104
Delivering Object-Based Audio-Visual Services ..... H. Kalva, A. Eleftheriadis, J. Zamora 1108
Visible Light Communication for Audio Systems ..... G. Pang, K.-L. Ho, T. Kwan, E. Yang 1112
Design of a Novel Synthesis Filter for Real-Time MPEG-2 Audio Decoder Implementation on a DSP Chip ..... W.-K. Paik and S.-Y. Hwang 1119
A 4:2:2P@ML MPEG-2 Video Encoder Board using an Enhanced MP@ML Video Encoder LSI ..... T. Yoshitome, T. Minami, M. Ikeda, K. Nitta, and K. Suguri 1130
Ensuring Interoperability in a Home Networking System: A Case Study ..... D. Kaleshi, M. H. Barton 1134
A Compact and High-Resolution Video Projector Based on Silicon Light Valves ..... H. C. Huang, P. W. Cheng and H. S. Kwok 1144
General Quality Maintenance Module for Motion Picture Watermarking ..... I. Echizen, H. Yoshiura, T. Arai, H.Kimura, and T. Takeuchi 1150
Feasibility Study of Airplane Fiber Optical Communication Network ..... Y.-H. Lee 1159
Stabilized Precoder with Antenna Diversity for Wireless LANs ..... M. P. Sellars, S. D.Greaves, and I. J. Wassell 1169
Design and Implementation of a Java-Based MPEG-1 Video Decoder ..... D. Kim, J. Kim 1176
A Single Chip Digital Satellite Receiver with 6-Bit A/D Converter ..... Y.-S. Choi and S.-J. Lee 1183
Reconstruction of Lost Blocks using Codeword Estimation ..... K.-L. Hung, C.-C. Chang and T.-S. Chen 1190
MPEG-2 AAC Decoder on a Fixed-Point DSP ..... J. Chen and H.-M. Tai 1200
A Network MPEG Camera ..... H. Kami, T. Kurashige, M. Higashi, T. Imaide, T. Kinugasa 1206
Design of MPEG-2 Video Test Bitstreams ..... C.-M. Kim, B.-U. Lee, and R.-H. Park 1213
A Variable Block-Size DCT-Based Watermarking Method ..... O-H.. Kwon, Y.-S. Kim, and R.-H. Park 1221
A Video Display Processing Platform for Future TV Concepts ..... P. H. N. de With, E. G. T. Jaspers, J. L. van Meerbergen, A. H. Timmer and M. T. J. Strik 1230
A Software/Hardware Co-Design Methodology For Embedded Microprocessor Core Design ..... Y. Zhang, K. K. Ma and Q. Yao 1241
A Digital Convergence System using Cubic Splines ..... C. G. Oh, Y. Chung, C. W. Hong, S. P. Hong 1247
An Open DSP Software Architecture for Consumer Audio ..... T. Chen, J. Datta, B. Karley 1253
A Wideband DS-CDMA Modem for a Mobile Station ..... D.-H. Lee, A. Choi, J.-M. Koo, J.-I. Lee, and B. M. Kim 1259
4.7 GBYTE Re-Writable Disc System Based on DVD-R System ..... M. Yoshida, Y. Shimoda, H. Ishii, A. Inoue 1270
Software Decoding of HDTV ..... I. Defée 1277
Techniques for Reading DVD-Ram Disc with DVD-Rom Drives ..... K. Izumi, K. Ohnishi, M. Inoue, T. Shimano, Y. Sugi 1284
An Efficient Host/Co-Processor Solution for MPEG-4 Audio Composition ..... L. L. Bourhis, G. Zoia, M. Mattavelli, D. J. Mlynek 1290

Google Exhibit No. 1063
Page 1280



IEEE CONSUMER ELECTRONICS SOCIETY

The Consumer Electronics Society is an organization, within the framework of the IEEE, of members, with technical interests in all facets of consumer electronics products. All members of the IEEE may join the Society and have the option of receiving the Transactions on CONSUMER ELECTRONICS, for which the annual fee is $20.00. For more information, see address below. *Member copies of Transactions/Journals are for personal use only.*

IEEE TRANSACTIONS ON CONSUMER ELECTRONICS

WAYNE C. LUPLOW, *Publication Chairman and Editor*

Publications committee:
Dr. Jay Bao
Dr. Walter Ciciora
Dr. Zabih Ghassemlooy
John Henderson
Leander Hoke, Jr.
Dave Hostetler
C. Bailey Neal
Dr. Howard Prosser
Dr. Ted Rzeszewski

THE INSTITUTE OF ELECTRICAL AND ELECTRONICS ENGINEERS, INC.

1999 Officers

KENNETH R. LAKER, *President*
BRUCE A. EISENSTEIN, *President-Elect*
MAURICE PAPO, *Secretary*
DAVID A. CONNER, *Treasurer*

ARTHUR WINSTON, *Vice President, Educational Activities*
PAUL J. KOSTEK, *President, IEEE USA*
LLOYD A. MORLEY, *Vice President, Publication Activities*
DANIEL R. BENIGNI, *Vice President, Regional Activities*
DONALD C. LOUGHRY, *Vice President, Standards Association*
MICHAEL S. ADLER, *Vice President, Technical Activities*

WILLIAM G. GJERTSON, *Director, Division IV*

Executive Staff
DANIEL J. SENESE, *Executive Director*

DONALD CURTIS, *Human Resources*
ANTHONY DURNIAK, *Publications*
CECELIA JANKOWSKI, *Regional Activities*
PETER A. LEWIS, *Educational Activities*
JUDITH GORMAN, *Standards Activities*

RICHARD D. SCHWARTZ, *Business Administration*
W. THOMAS SUTTLE, *Professional Activities*
JOHN WITSKEN, *Information Technology*
MARY WARD-CALLAN, *Technical Activities*

IEEE TRANSACTIONS ON CONSUMER ELECTRONICS is published quarterly by The Institute of Electrical and Electronics Engineers, Inc. Responsibility for the contents rests upon the authors and not upon the IEEE, the Society, or its members. **IEEE Corporate Office,** 3 Park Avenue, 17th Floor, New York, NY 10016-5997. **IEEE Operations Center,** 445 Hoes Lane, P.O. Box 1331, Piscataway, NJ 08855-1331. **NJ Telephone:** 732-981-0060. **Price/Publication Information:** Individual copies: IEEE members $10.00 (first copy only), nonmembers $20.00 per copy. (Note: Add $4.00 postage and handling charge to any order from $1.00 to $50.00, including prepaid orders.) For single copy orders: 1-800-678-4333 in the U.S.; 732-981-0060 outside the U.S. Member and nonmember subscription prices available on request. Available in microfiche and microfilm.

Copyright and Reprint Permissions: Abstracting is permitted with credit to the source. Libraries are permitted to photocopy for private use of patrons, provided the per-copy fee indicated in the code at the bottom of the first page is paid through the Copyright Clearance Center, 222 Rosewood Drive, Danvers, MA 01923. For all other copying, reprint, or republication permission, write to: Copyrights and Permissions Department, IEEE Publications Administration, 445 Hoes Lane, P.O. Box 1331, Piscataway, NJ 08855-1331. Copyright © 1999 by The Institute of Electrical and Electronics Engineers, Inc. All rights reserved.

Periodicals postage paid at New York, NY, and at additional mailing offices. **Postmaster:** Send address changes to IEEE TRANSACTIONS ON CONSUMER ELECTRONICS, IEEE, 445 Hoes Lane, P.O. Box 1331, Piscataway, N.J. 08855-1331. GST Registration No. 125634188. Printed in U.S.A.

1999
IEEE CONSUMER ELECTRONICS SOCIETY
ADMINISTRATIVE COMMITTEE

Term Expires 1999

Jean Agostinelli
Eastman Kodak Company
232 Seneca Parkway
Rochester, NY 14613

George Hanover
Electronic Industries Association
2500 Wilson Blvd.
Arlington, VA 22201-3834

John G. N. Henderson
Hitachi America, Ltd.
307 College Road East
Princeton, NJ 08540

Roger G. Kozlowski
Motorola Imaging & Entertainment Solutions
100 East Sybelia Drive, Suite 215
Maitland, FL 32751

Richard S. Prodan
Cable Television Laboratories, Inc.
400 Centennial Pkwy.
Louisville, CO 80027-1208

Term Expires 2000

Gene Frantz
Texas Instruments
P.O. Box 1443, M/S 701
Houston, TX 77251-1443

Kees A. S. Immink
University of Essen
Ellernstr. 29
45326 Essen, Germany

Michael A. Isnardi
Sarnoff Corporation
CN 5300
Princeton, NJ 08543-5300

Gopal Miglani
SoftProse, Inc.
5701 Skylark Place
LaJolla, CA 92037

William Miller
Philips Consumer Electronics Co.
P.O. Box 14810
Knoxville, TN 37914-1810

Term Expires 2001

Michael A. Brown
Intel Corporation
5000 W. Chandler Blvd.
Chandler, AZ 85226-3699

James O. Farmer
Antec Corporation
11450 Technology Circle
Duluth, GA 30097

Wayne C. Luplow
Zenith Electronics Corporation
1000 Milwaukee Avenue
Glenview, IL 60025

Vishal Markandey
Texas Instruments
P.O. Box 660199, MS 8635
Dallas, TX 75266-0199

Terrence R. Smith
Sarnoff Corporation
CN 5300
Princeton, NJ 08543-5300

Ex-Officio Members

Stefan Mozar
Singapore Polytechnic
500 Dover Road
Singapore 139651

Almon Clegg
Fellow Evaluator/Liaison to the History Center
Consultant
6070 Lively Court, Suite 101
Cumming, GA 30130-6264

William G. Gjertson
Director, Division IV
Boeing Information Space & Defense Systems
P.O. Box 3999, MS 8H-10
Seattle, WA 98124

David M. Lewis
Nominations
Intel Corporation
5000 W. Chandler Blvd.
Chandler, AZ 85226

Stuart Lipoff
1st Past President, Standards
A. D. Little, Inc.
20 Acorn Park
Cambridge, MA 02140

C. Bailey Neal
Awards 1999 ICCE Program
Thomson R&D Laboratories
P.O. Box 6139, INH-725
Indianapolis, IN 46206-6139

Robert W. Rieder
Web Master
8716 N. Central Avenue
Skokie, IL 60076-2305

Diane Williams
CES Executive Administrator
67 Raspberry Patch Dr.
Rochester, NY 14612-2868

LIBRARY OF CONGRESS
JAN 10 2000
COPY
COPYRIGHT OFFICE

Google Exhibit No. 1063
Page 1282

DESIGN OF A NOVEL SYNTHESIS FILTER FOR REAL-TIME MPEG-2 AUDIO DECODER IMPLEMENTATION ON A DSP CHIP

Won-Kyu Paik and Sun-Young Hwang
CAD & Computer Systems Lab, Dept. of Electronic Engineering
Sogang University, C.P.O. Box 1142, Seoul, 100-611, Korea
E-mail: hwang@ccs.sogang.ac.kr

Abstract – **This paper describes the design of a novel synthesis filter for real-time implementation of MPEG-2 audio decoder on a single, programmable DSP chip. In the first phase of our design we propose a fast subband synthesis filter bank employing a decomposed IDCT block with reduced multiplication stages, shared by both MPEG-1 and MPEG-2 decoders. In the second phase of the design, a parameter analyzer block is adopted to take advantage of the multichannel computing structure in MPEG-2 thereby further reducing the number of MAC operations. In the final phase, we propose a low-cost decoder employing unorthodox methods of boosting performance by gracefully degrading quality of service.**

The overall runtime of the decoder has been reduced by 60% when compared to the modified ISO decoder employing a synthesis filter bank with fast IDCT block. Real-time performance validations have been made by re-tuning time-critical code sections for both native VLIW and DSP architectures. The final implementation successfully passes all ISO/IEC MPEG-2 audio compliance tests and achieves near-transparent audio decoding and signal reconstruction on the fly.

I. Introduction

The demand for reductions in data rate for wideband digital audio signal transmission and storage in multimedia applications substantiates the need for efficient compression techniques. A serial bitstream of CD quality audio material consists of 44,100 16-bit PCM samples per channel which corresponds to 1.4 Mbps in stereo [1]. Audio at that bitrate contains a lot of redundancies and this led to the development of perceptually coded, lossy compression techniques to eliminate information that is irrelevant from a human sound-perception viewpoint. Studies have shown that by masking inaudible components of audio signals via psychoacoustic models [1][2], perceptual audio systems are able to achieve significant compression while maintaining perceived audio quality roughly equal to the original.

The International Organization for Standardization (ISO) has established a working group known as the Motion Picture Experts Group (MPEG), to develop standards for coding audio-visual presentation materials aimed at digital storage media such as CD-ROMs, not exceeding a combined continuous transfer rate of 1.5 Mbps. The audio portion specifies the coded representation of high-quality audio signals intended for stand-alone or multiplexed-with-video applications [3][4]. The MPEG audio system features a wide mix of channel configurations, multiple sampling and data rates, layer compatibility, decoder version scalability, dynamic range control, Dolby Pro Logic compatibility, and multi-lingual support, thereby addressing to a variety of consumer and professional applications. Application areas run a wide gamut from DVD players, multichannel High Definition Television (HDTV), Integrated Service Digital Network (ISDN) transmission codecs, cable and satellite-based TV (e. g. DVB, USSB) services, Digital Audio Broadcasting (DAB), studio recording and editing, to multimedia/Internet audio.

Contributed Paper
Manuscript received July 7, 1999

0099 3063/99 $10.00 © 1999 IEEE

This material may be protected by Copyright law (Title 17 U.S. Code)

Google Exhibit No. 1063
Page 1283

MPEG-2 was first developed with the broadcast application in mind. The concept of a "1-encoder to N-decoders," makes for non-symmetric systems with a relatively complex, high-quality encoder on one end and fast, low-complexity, low-cost decoders at the other delivering quality service on limited bandwidth. Hence, some researches leading to the real-time implementation of MPEG standards have based the decoder designs on programmable DSPs [5][6], for their merits in easy implementation of complex algorithms, reduced design cycles, and development cost. A floating-point implementation of a real-time MPEG-1 system was presented by Murphy [5]. But limitations to processing speed and memory of the host DSP chip prohibited the implementation of a fully functional MPEG-1 system on a single DSP, let alone multichannel systems. Similar work has been found implementing the MPEG-2 standard on a fixed-point, multimedia DSP [6], indicating a move toward low-cost, uniprocessor solutions. Fixed-point implementations are cost-effective, but complications may be introduced by the repetitive computational tasks typically populated in signal processing applications. Rounding-off error introduced by multiplying two fixed-point data are trivial as is, but the propagation of these smaller errors onto successive multiplication stages ultimately corrupts reconstructed signal quality to a noticeable level. The synthesis filter bank as defined by the standard, is considered by far the most computationally intensive block in the entire decoding system well populated with MAC operations, thereby efforts to reduce the complexity of the synthesis filter has been essential for real-time decoder implementations. The most common practice in reducing the number of multiplications and additions, has been by exploiting the structural properties of the IDCT block resulting in manifold decoder performance improvements [7]. There have been other fast IDCT schemes that are effective for designing speedy synthesis filter blocks [8]. However, not all IDCT schemes are suitable for fixed-point implementation, since a trade-off between the number of multiplication operations and rounding errors due to splitting the internal IDCTs exist. To achieve fast decoding of high quality audio, it is essential to keep the numerical errors to an acceptable level without running into hard real-time constraints of the decoding system. Further improvements to the synthesis filter design can be achieved by exploiting the multichannel computation structure of MPEG-2, thus contributing for a faster MPEG-2 dedicated decoder operation. This paper fully exploits the multichannel operation of the MPEG-2 decoding system to reduce the number of MAC operations.

In this paper, we present the design and implementation of a real-time MPEG-2 audio decoder with an improved synthesis filter on a single, fixed-point DSP. The proposed system employs a decomposed IDCT block with reduced multiplication stages, and a parameter analyzer block for dedicated MPEG-2 decoding. The paper is organized as follows: Section II provides an overview into the MPEG-2 multichannel audio decoding process. In section III, the we propose a versatile subband synthesis filter routine employing a decomposed IDCT block with reduced multiplication stages, followed by optimizations for dedicated multichannel computation structure. For cost-effective measures, we propose unorthodox methods of gracefully degrading audio signal quality for faster decoder speed. The enhanced decoder is realized on the DSP board, mounted on a host PC. In section IV, experimental results are reported, and a conclusion is drawn at the final section of the paper.

II. MPEG-2 Multichannel Audio Coding System

The MPEG audio standards, ISO/IEC 11172-3 (MPEG-1) [3] and ISO/IEC 138180-3 (MPEG-2) [4] define three layers of increasing complexity and subjective performance varying with compression efficiency. With successive layers, however, the coding/decoding system is subject to increased delay and sensitivity to transmission errors.

2.1 Overview of the MPEG Audio Coding System

The main decoding sub-blocks of the MPEG audio decoder are comprised of a frame unpacker, dequantizer, dematricing block (MPEG-2), and a synthesis filter bank. After frame disassembly from the input encoded bitstream, the decoder passes the bit-allocation, scalefactor side-informations and audio samples to the inverse quantizer for restoring the quantized subband values. These subband samples are fed into the synthesis filter block for reconstruction and output of PCM samples on a per channel basis.

Google Exhibit No. 1063
Page 1284

2.2 Multichannel Extension

Restrictions to two-channel stereophony imposed by the MPEG-1 standard are lifted with multichannel sound defined by ISO/IEC 138180-3. The extension to the multichannel system provides support for 5.1 channels without simulcast intended for broadcast complying with the HDTV format. The MPEG-2 decoding system shares the same audio algorithm core as MPEG-1, hence providing forward and backward compatibility [1]. An MPEG-2 decoder will accept an MPEG-1 formatted bitstream and an MPEG-1 decoder can derive a stereo representation from an MPEG-2 bitstream. The multichannel format uses an encoder matrix that allows a two-channel decoder to decode stereo compatible signals that is a subset of a multichannel bitstream. The multiple channels are matrixed to form compatible MPEG-1 left/right channels, as well as other MPEG-2 channels, as shown in Figure 1. These are encoded into MPEG-1 frames, and additional multichannel data is placed in the extended auxiliary data field. MPEG-2 features intensity coding, crosstalk reduction, inter-channel prediction coding, and center channel phantom image coding to achieve a combined bit rate of 384 kbps, using Layer II at a 48-kHz sampling frequency.

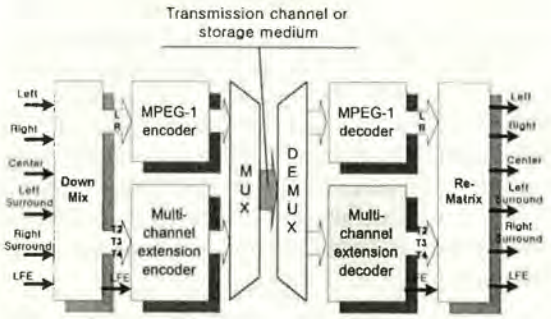


Fig. 1. Forward and backward compatible MPEG-2 audio system.

2.3 Synthesis Filter Bank

The subband synthesis filter bank is a polyphase frequency-time domain transform engine, shared by both MPEG-1 and MPEG-2 algorithms. After the dequantization process from the previous stage, 32 subband samples per channel are input to the synthesis filter bank and 32 reconstructed time-domain PCM samples are output to a D/A converter for reproduction. As shown in Figure 2, the synthesis filter bank is defined by the MPEG standard with three key operations: Initialization, Matrix transformation, and Window/overlap-add. Initialization involves shifting of 64 samples, from a queue of 1024 points at a time. The matrix transform operation receives the 32 frequency-domain samples and returns 64 intermediate elements, which are written to the input IQMF buffer. The buffer has 1024 elements for every channel and is cyclic.

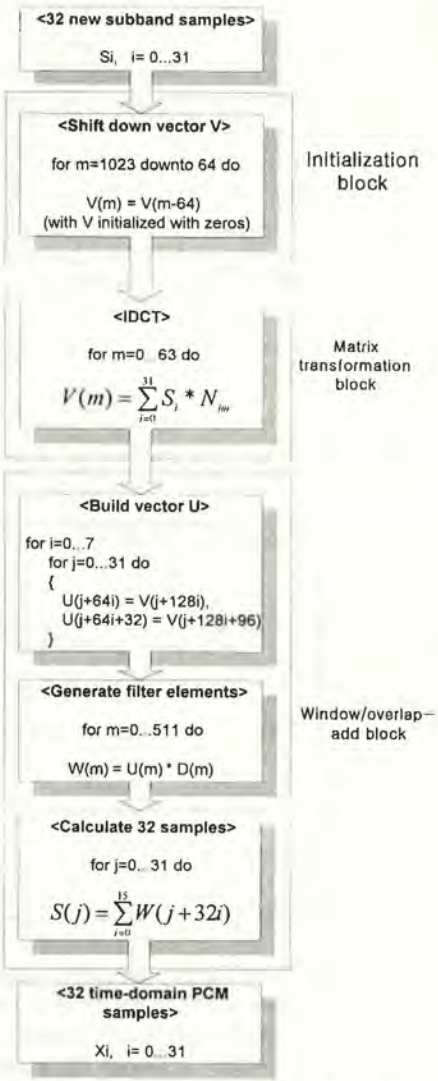


Fig. 2. Synthesis filter bank flow [3].

Google Exhibit No. 1063
Page 1285

The window/overlap-add block performs the actual filtering operations. The filtering operation is performed on 512 elements that includes the last 64. A new vector is built from these elements and is multiplied with the tabulated windowing coefficients in the IQMF block to produce 32 time-domain samples. The 32 samples are formatted into PCM samples for output. In the next iteration, all the elements of V are shifted down 64 places in preparation for the next IDCT output. These operations are performed recursively for each represented channel.

Table 1 displays the profiling output of the ISO MPEG-2 decoder software implementation on UNIX and DSP platforms, respectively. The profile result is an average taken from 20 consecutive frames of a random mix of multichannel audio clips, including the ISO/MPEG-2 conformance test streams. This table indicates that the synthesis filter bank is the most computationally intensive block in the entire decoder, therefore any design improvements made here would yield a faster decoder. For MPEG-2 multichannel operation, the speedup is more pronounced since the synthesis filter iterations increase in direct proportion to the number of represented channels.

Table 1. Profiling of the ISO MPEG-2 decoder functions.

Decoding Stage	Percentage of total time spent in each decoding task	
	UNIX	DSP
Synthesis filter	90%	83%
Dequantizer	4%	7%
Stream decoder	2%	4%
Dematrix & Denormalizer	3%	4%
Others	1%	2%

The most DSP cycle-hungry block in Figure 2 of the synthesis filter bank is the matrix transformation block. As shown in Table 2, any improvements to the synthesis filter bank centers around a more efficient IDCT block, requiring optimization of nested loops and reductions in number of multiplications in the loop kernels.

Table 2. Profiling of the ISO synthesis filter sub-blocks.

Synthesis filter bank sub-blocks	# multipli -cations	# addi-tions	# DSP cycles	Percentage of total cycles consumed
IDCT	2,048	1,984	410k	91%
Window/ overlap-Add	512	480	32k	7%
Others	0	0	6k	2%

III. The Proposed Synthesis Filter

As presented in the previous section, optimizations to the synthesis filter are essential for a fast realization of the MPEG decoder. In this section, we first propose a subband synthesis filter bank employing decomposed IDCT and windowing blocks, shared by both MPEG-1 and MPEG-2 decoders. Next, we propose the addition of a parameter analyzer block for further performance improvements to MPEG-2 decoders, taking advantage of the multichannel computing structure. Finally we propose a synthesis filter employing graceful degradation techniques targeting low-cost implementations with marginally degraded audio output quality.

Figure 3 shows the proposed synthesis filter block shared by both MPEG-1 and MPEG-2 multichannel decoders. It implements a modified IDCT block with a multistage butterfly network at the input followed by decomposed IDCT and efficient windowing functions. Matched against other well-known 32-point IDCTs [8] the IDCT scheme employed here profits from minimal fixed-point arithmetic propagation errors while reducing MAC counts.

Google Exhibit No. 1063
Page 1286

3.1 Design of a Fast MPEG-1/2 Synthesis Filter Bank

In this section we propose a fast subband synthesis filter bank employing a decomposed IDCT block with reduced multiplication stages, shared by both MPEG-1 and MPEG-2 decoders. The IDCT scheme employed in the proposed design recursively decomposes the IDCT block into smaller matrices by exploiting the symmetric properties of the cosine transform. An efficient windowing scheme has been proposed that exploits the symmetries observed in the IDCT output.

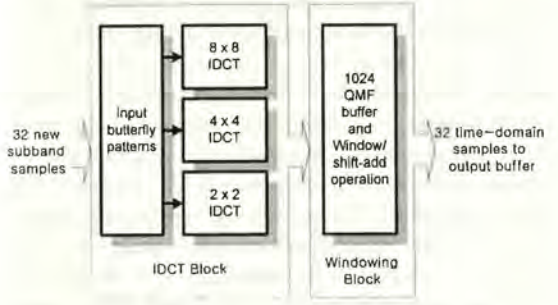


Fig. 3. Block diagram of the proposed synthesis filter.

3.1.1 IDCT Block with Decomposed Matrix Transformations

Symmetric Properties of the Cosine Transform: The IDCT transforms the input subband samples from frequency to time-domain by using Equation (1) as defined in the MPEG standard.

$$V(i) = \sum_{sb=0}^{31} N(i,sb)X(sb) = \sum_{sb=0}^{31} \cos\left[\frac{\pi}{64}(2sb+1)(i+16)\right]X(sb), \quad i = 0...63 \tag{1}$$

Vector V denotes an array of 64-intermediate values needed at the window/overlap-add block, the discrete cosine terms $N(i,sb)$, and subband coefficients $X(sb)$. To evaluate V as is, 2,048 multiplications and 1,984 additions are required as shown in Table 2. However, V can be built exploiting the inherent symmetries of $N(i,sb)$ as shown in Equation (2), thereby using only half the multiplications and additions required in Equation (1).

$$V(i) = -V(32-i)$$
$$V(32+i) = V(64-i), \qquad i = 1...16 \tag{2}$$

Butterfly Stage: The 32-point IDCT derived from Equation (2) can be further decomposed into two 16 × 16 even and odd matrices by way of shuffling and grouping X(sb) into butterfly patterns, denoted in Equation (3). Thus, an N-point IDCT of Equation (2) can be decomposed into sums of two N/2-point IDCTs, ontaining a four-fold reduction in the number of MAC operations. As shown in Figure 3, the first input butterfly pattern of X(sb), can be further decomposed in a similar manner by exploiting trigonometric and structural properties, thereby reducing the complexity of the IDCT block of the synthesis filter.

$$V(i) = \sum_{sb=0}^{16} N(i,sb)\left[X(sb) + X(31-sb)\right], \quad i = 0,2,...,30$$
$$V(i) = \sum_{sb=0}^{16} N(i,sb)\left[X(sb) - X(31-sb)\right], \quad i = 1,3,...,31 \tag{3}$$

Successive Decompositions: By recursively breaking down the input even 16 × 16 butterfly pattern into 8 × 8, 4 × 4, and 2 × 2 parts, the IDCT can be further decomposed into smaller transform matrices. The However, decomposing any odd butterfly pattern requires performing multistage multiplication with non-unity trigonometric multipliers, which are deleterious for fixed-point arithmetic. It is imperative to keep the propagating rounding errors to a minimum, hence multiplication is performed at the end of the butterfly decomposition stage and any splitting of internal IDCT that require intermediate non-unity multiplications are omitted. The generation of butterfly patterns requires only addition and subtractions but no multiplications. Once the butterfly patterns are completely broken down, matrix transformation is performed.

Google Exhibit No. 1063
Page 1287

3.1.2 Windowing Block

A 511th-order IQMF filter bank with a 96-dB sidelobe rejection and a steep pass-to-stop-band transition is used to perform the window/overlap-add block upon the 1024 intermediate element output from the IDCT block. A new vector is built from these elements and is multiplied with the tabulated windowing coefficients *D(m)* to generate the 512-window elements. A 16-point filtering operation is performed on the generated filter elements to produce 32 time-domain samples *S(i)*. Each *S(i)* are formatted into PCM audio samples, and output to the channel FIFO buffers for DAC.

$$
\begin{aligned}
&U(j+64i) = V(j+128i)\\
&U(j+64i+32) = V(j+128i+96),\\
&\qquad\qquad j = 0...31, \quad i = 0...7\\
&W(m) = U(m)D(m), \quad m = 0...511\\
&S(j) = \sum_{i=0}^{15} W(j+32i), \quad j = 0...31
\end{aligned}
\tag{4}
$$

Equation (4) holds when the inverse transform given in Equation (1) is re-defined, by exploiting the symmetrical properties of the windowing coefficients D. The odd and even input butterflies derived from Equation (3) for the calculation of *V*.

The 512 coefficients are split into 16 blocks of 32 *V* coefficient, the even numbered blocks are multiplied with the first 32 IDCT outputs, and the odd blocks are multiplied with the next 32 IDCT outputs. The windowed results of 16 blocks are then overlapped and added, resulting in half the multiplication and additions of the original.

3.2 Optimizations Exploiting MPEG-2 Computing Structure

As stated earlier, MPEG-2 provides full MPEG-1 support and features key multichannel operations such as dynamic crosstalk reduction, inter-channel prediction coding, and center channel phantom image coding. In this section, a parameter analyzer block is adopted to take advantage of the multichannel computing structure thereby achieving further performance improvements. With the availability of dynamic bit-allocation information and multichannel coding parameters after frame parsing from the encoded bitstream, the synthesis filter loop can be optimized to exploit the redundancies and zero-allocation profiles of subband samples, resulting in the elimination of wasted MAC operations.

3.2.1 Exploiting Intra, Inter-channel Redundancies

The MPEG-2 multichannel decoder exploits the use of information from neighboring channels in order to achieve optimal data reductions without resorting to simulcast operations. One of the main features supported by the MPEG-2 decoding system is dynamic crosstalk reduction, which reduces the amount of data through copying stereo-irrelevant signal components for extension channel reconstruction [4]. With the existing ISO MPEG decoders, rescaling, dequantizing, dematricing and denormalizing operations precede the synthesis filter bank operation. However, the proposed decoder performs the synthesis filtering prior to other operations, thereby if dynamic crosstalk reduction is used, the application of the synthesis filter bank to extension channels is bypassed altogether. Thus inter-channel redundancies can be exploited at the decoder level, eliminating unnecessary iterations of costly IDCT and windowing operations resulting in savings of the number of MAC operations.

In a similar context, intra-channel redundancies are exploited using a temporary buffer for storage of intermediate *V*s. If a match is found in the next iteration of the synthesis filter, the IDCT operation is omitted, permitting us to reuse older *V* coefficients for windowing. A match can be found by comparing the values of subband bit allocation and corresponding sample coefficients. The intra-channel technique exploits the temporal locality of periodic audio signals.

3.2.2 IDCT by Subband Coefficient Analysis

An examination into subband coefficient profiles for various audio streams indicate that not all subband coefficients are allocated with bits. For each zero subband coefficient in an N-point DCT, N multiplications and N-1 additions are saved. Skipping the DCT matrix operation for zero subband coefficients permits us to reuse older V vector

Google Exhibit No. 1063
Page 1288

coefficients for windowing. Statistically then, about 50% of the 2048 multiplies and 1984 additions required in the original DCT implementation are eliminated. Empirical analysis results dictate to an average of 80% reduction in multiplies and 70% adds per 10 frame samples. These averages were taken over a wide range of diverse of audio clips, including jazz, classical, vocal pieces. A combination of DCT optimizations exploiting symmetric properties and subband coefficient analysis yields to empirical results of 90% reduction of multiplies and 80% reduction of additions compared to the DCT performance of the ISO/MPEG decoder.

Figure 4 shows the bit allocation profile across the spectral subtends of an audio frame fragment. If no bits were allocated for a particular subband index, any attempts at applying the matrix transform operation would only result in a waste of CPU cycles. For efficiency, zero bit allocation samples are dropped from the IDCT operation, thereby relieving the synthesis filter bank, with no performance compromises. Benefits are added by applying bit-allocation analysis to surround sound channels that have spares bit-allocation profiles, or to the bandwidth-limited phantom-coded center channel.

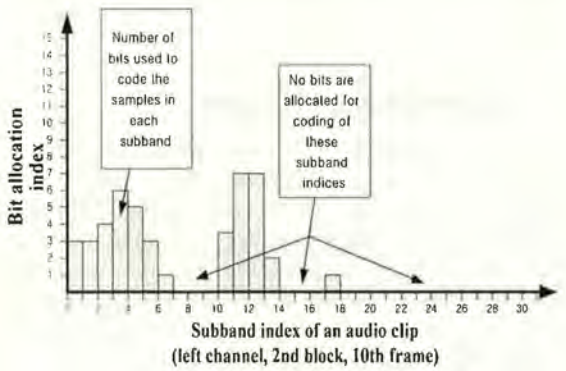


Fig. 4. Bit allocation across subbands.

Our statistical analysis of subband coefficients indicates that close to 50% of the subband coefficients are not allocated across a diverse selection of musical scores and compliance bitstreams. These are the average values for 10 frames of various bit and sampling rate configurations. Empirical results show that the upper half of the subband indices have a higher probability of zeroing out.

3.3 Graceful Degradation

Providing uninterrupted audio at lower-cost requires accurate real-time decoder system complexity measure such that efficient resource allocation can be attained. However, the measure of processing power for real-time operation of a software-driven decoder remains a difficult task. We assume that the absolute measure of decoder complexity can be determined by the total decoding time with respect to the bitstream or in more abstract measures, elementary operations such as the IDCT / windowing operations.

For each audio frame, the sub-band filter is repeated 12 or 36 times per channels depending on the Layer. Since the IDCT and windowing blocks are executed with every iteration of the synthesis filter, they form the kernels that lie on the "critical path" of the decoding process. Apart from scaling down the multichannel stream (from a 3/2 configuration to 2/1 for example) a more fine-grained approach of gracefully degrading quality of audio service is proposed in this paper.

We devised two ways of gracefully degrading the audio output: Partial zeroing of the $D(m)$ window coefficients, and eliminating the upper half subband coefficients of the spectral image. As shown in the latter half of the window/overlap-add block of Figure 2, 32 time-domain samples $S(sb)$ are calculated by Equation (4) with the generated 512 $W(m)$. However, about 40% of the values of $D(m)$ are close to being zero, hence the 16-point windowing operation can be reduced. It is also observed that most of the upper half of the subband samples are zero. Hence, IDCT can be performed for only 16 subband samples. The combined effort provides for minimal quality of service degradation for speedy decoding.

3.4 DSP Implementation

The proposed MPEG-2 decoder designs were implemented on a high-performance DSP with a VLIW architecture. To maximize real-time performance, platform dependent and independent optimizations were performed on the critical code segments. Timing of real-time communication of the I/O subsystem, between host and the DSP chip and

Google Exhibit No. 1063
Page 1289

control of the output channel buffers, is essential to ensure continuous I/O operations.

3.4.1 DSP Architecture

The proposed MPEG-2 audio decoder has been implemented and tested on a commercial DSP development board, mounted on a host PC [9]. The processor has a 200 MHz, 16-bit fixed-point VLIW core, claimed to run at 1,600 MIPS. It implements a load-store architecture and manifests two cross-linked datapaths, each having four execution units, a split 4-Mbyte on-chip program and data memories, a uniform general-purpose register file, operating on 32/40-bit operands. The eight execution units include two multipliers, four ALUs, two shift/logical units. Hence, the processor can execute up to eight orthogonal 32-bit RISC-like instructions in a single cycle, enabling it to achieve the high level of parallelism. Salient DSP features such as saturation logic and circular addressing are supported. The development board is equipped with an array of peripherals such as 8 Mbytes of global memory, Host Port Interface (HPI), external JTAG scan path emulation support, two-buffered serial ports, 16-bit stereo audio interface supporting standard sample rates of up to 48 kHz.

Implementing real-time multimedia standards such as MPEG audio on programmable DSPs requires code adaptations to the native hardware. Despite the ease in translating the MPEG standard by a high-level language such as C, the efficient use of the processing power of the VLIW machine remains a difficult task. For maximal utilization of on-chip hardware resources, we have adopted both DSP and VLIW architectural optimizations to the MPEG algorithm. VLIW architecture dependent and independent optimization techniques such as software pipelining, loop unrolling and other DSP intrinsics [9] have been employed to resolve time-critical loop blocks of the decoder.

3.4.2 Real-time Communications

Timing of real-time operations ensures correct synchronous operation: all data is produced and consumed at exactly the rate that it is output or input. Decompression of audio signals are characterized by an amplified data output that far exceeds the consumed input of encoded data, creating a processing core that far exceeds that of the I/O subsystem. The hardware architecture is designed to connect directly to a PCI bus, as a PCI bus master, and to enable efficient use of the PCI bus for real-time communication of data between the DSP and the host. These built-in features enable efficient software to be produced for processing the streaming audio data. The software does not need to be aware of the input data format, as it will be converted to the standard 16 bit stereo format without intermediate modifications. In addition, the software need not be aware of the origin of the data. The DMA engine automatically places the input data into a circular buffer. When data has been consumed from the buffer the hardware automatically create the DMA request and queue it for filling. A similar process is used to empty out buffers to the DAC.

Once the data is available upon the completion of a synthesis filter iteration, the reconstructed audio PCM data are rounded to the DAC word width and copied to the corresponding output channel FIFO buffer. When buffers are full for all the specified channels, audio data are output through the DAC. In contrast with the input buffer stage, the decoder output processing uses very few MIPS but a relatively large amount of memory. For each output channel, the decoder requires 1152-long PCM sample buffers.

IV. Experimental Results

The performance of the proposed MPEG algorithm has been validated on a UNIX and implemented on the DSP chip, using MPEG-2 compliance test stream sequences. Tables 3 and 4 show the experimental results for the proposed decoder designs, compared with the performance of the ISO decoder and a modified ISO decoder with the synthesis filter employing a fast IDCT [7].

In Table 3, the total number of multiplications and additions required in the proposed synthesis filter has been reduced to an average of 60% when compared to the modified ISO synthesis filter sub-blocks. The average execution time information of the proposed DSP-based decoder is reported in Table 4, for MPEG-1 and MPEG-2 multichannel bitstreams. Note that the direct implementation of the ISO decoder on the DSP chip does not provide for successful real-time performance, as portrayed by the Jazz and Classical clips. The proposed decoder

Google Exhibit No. 1063
Page 1290

achieves an average of 60% speedup over the modified ISO decoder.

The results of Table 4 have direct implication upon deciding the lower bound for DSP platforms for which decoder systems can be implemented. Albeit the modified ISO decoder meets the current real-time specifications, it is certainly not a cost-effective implementation since the hardware on which it was implemented is one of industry's high-end DSP cores available today. Consequently, a faster decoder design opens the possibility for a more cost-effective implementation. Since real-time environments put a premium on the availability of hardware resources, cost-effective implementations are imperative.

Another low-cost implementation for the MPEG algorithm lies in adopting computationally graceful degradation (CGD) techniques to computation intensive blocks of the decoder. CGD-based designs provide minimum Quality of service (QoS) when the processing needs exceed the available resources. We found those further reductions in MAC counts and decoder speedup, were possible by a combination of eliminating the upper half of the subband coefficients and utilizing a trimmed $D(m)$ windowing function. The combined effort gives a guarantee towards meeting hard real-time constraints of the cost-effective decoding systems.

The proposed decoder has been implemented on high-performance fixed-point VLIW DSP. The final implementation provides for near-transparent real-time audio decoding and reconstruction of PCM samples.

Table 3. Number of operations in IDCT and window schemes.

Operations		ISO	Modified ISO	Proposed					
				For MPEG-1/2		For MPEG-2 only		Degraded MPEG-1/2	
					△%		△%		△%
IDCT block	# Multiplications	2,048	512	384	-35	288	-43	156	-70
	# Additions	1,984	496	350	-30	204	-58	168	-66
Windowing block	# Multiplications	512	512	256	-50	256	-50	198	-61
	# Additions	496	496	256	-52	256	-52	225	-54

* Experimental results are compared against the modified ISO decoder.

Table 4. Execution time profiles of real-time DSP-based MPEG decoders.

Audio Clips	Total play time (sec)	Execution times of decoders (sec)				
		ISO	Modified ISO	Proposed		
				For MPEG-1/2	For MPEG-2 only	Degraded MPEG-1/2
Jazz	32.8	*36.5*	18.7	11.2	9.6	7.5
Classical	58.1	*59.1*	32.5	22.3	18.9	14.1
Vocal (MPEG-1)	24.3	21.4	10.5	7.1	-	4.8
Conformance test	5.6	4.2	2.3	1.6	1.4	0.9

* Experimental results are compared against the modified ISO decoder.

Google Exhibit No. 1063
Page 1291

V. Conclusion & Future Researches

In this paper, we described a real-time realization of an MPEG-2 audio decoder on a single, fixed-point, VLIW DSP chip. The proposed subband synthesis bank employs a decomposed matrix transform block for both MPEG-1 and MPEG-2 decoding, and an additional parameter analyzer block for multichannel decoding. A decoder design implementing graceful degradation techniques has been proposed as a low-cost alternative, by keeping output audio quality to a minimal level while hiking up performance. Our effort concentrated on reducing the number of MAC operations of the computationally intensive synthesis filter bank while preserving signal quality integrity.

For real-time realization on a DSP chip, both DSP and VLIW architectural optimizations were performed. Optimal performance results have been obtained by judiciously adopting platform dependent and independent optimization techniques such as software pipelining loop unrolling, and DSP intrinsics, without falling back on the usual bottom-up assembly patch-up. Despite the versatility of translating and mapping complex standards onto the native DSP hardware via high-level language, efficient compiler targets, robust and tightly integrated debug/simulation environments are still yearned for.

The final realization provides a 60% faster decoder when matched against the modified ISO decoder implementing a fast IDCT block in the synthesis filter bank. Real-time performance obtaining near-transparent audio decoding and reconstruction of PCM samples, have been achieved. The techniques described in this paper can be extended the implementation of multiplexed-with-video applications, stand-alone low-cost decoder solutions, or elevated to core-based ASIC designs with few modifications. As for future CGD-based real-time decoder implementations, a practical complexity estimation policy for programmable cores need to be developed in order to accurately predict decoding time, and thereby guaranteeing real-time performance with resource allocation constraints.

Acknowledgement

This work was supported partly by Daewoo Electronics Co. as part of Digital VDR Technology Development Project.

References

[1] K. Pohlmann, Principles of Digital Audio, 3rd edition, McGrawHill: New York, 1997.

[2] E. Zwicker and H. Fastl, Psychoacoustics: Facts and Models, Springer-Verlag: New York, 1990.

[3] ISO-IEC JTC1/SC29/WG11, 'Information Technology - Coding of Moving Pictures and Associated Audio for Digital Storage Media at up to about 1.5 Mbit/s - Part 3: Audio,' IS 11172-3, Nov. 1992.

[4] ISO-IEC JTC1/SC29/WG11, 'Information Technology - Generic Coding of Moving Pictures and Associated Audio Information - Part 3: Audio,' IS 13818-3, May 1995.

[5] C. Murphy and K. Anandakumar, 'Real-Time MPEG-1 Audio Coding and Decoding on a DSP Chip,' IEEE Trans. on Consumer Electronics, Vol. 43, No. 1, Feb. 1997, pp. 40～47.

[6] J. Chen and H. Tai, 'Real-time Implementation of the MPEG-2 Audio Codec on a DSP,' IEEE Trans. on Consumer Electronics, Vol. 44, No. 3, Aug. 1998, pp. 866～871.

[7] T. Tsai, L. Chen, and Y. Liu, 'A Novel MPEG-2 Audio Decoder with Efficient Data Arrangement and Memory Configuration,' IEEE Trans. on Consumer Electronics, Vol. 43, No. 3, Aug. 1997, pp. 598～604.

[8] K. Rao and P. Yip, Discrete Cosine Transform, Academic Press Inc.: London, 1990.

[9] Texas Instruments, TMS320C62x/C67x Programmer's Guide, 1998.

Google Exhibit No. 1063
Page 1292

Biographies



Won-Kyu Paik received the B.S. degree in electronic engineering from Sogang University, Seoul, Korea, in 1997. He is currently working towards the M.S. degree in electrical engineering at Sogang University. His current research interests include digital video/audio compression, real-time DSP systems, and CAD systems.



Sun-Young Hwang received the B.S. degree in electronic engineering from Seoul National University, Seoul, Korea, in 1976, the M.S. degree from Korea Advanced Institute of Science in 1978, and the Ph.D. degree in electrical engineering from Stanford University, California, U.S.A., in 1986. Since 1986, he has been with the Center for Integrated Systems at Stanford University, working on design of a high-level synthesis and simulation system. In 1986 and 1987, he held a consulting position at Palo Alto Research Center of Fairchild Semiconductor Corporation. In 1989, he joined the Department of Electronic Engineering at Sogang University, where he is now a professor. His current research interests include hardware/software co-design, and DSP/VLSI systems design.

Google Exhibit No. 1063
Page 1293

Ex. 1039B

LIBRARY | CATALOG

1 of 1

PERIODICAL OR NEWSPAPER

IEEE transactions on consumer electronics

Full Record MARC Tags

Main title

IEEE transactions on consumer electronics.

Published/Created

New York : Institute of Electrical and Electronics Engineers, ©1975-

Request this Item | LC Find It

More Information

LCCN Permalink	https://lccn.loc.gov/75643472
Publication history	Vol. CE-21, no. 1 (Feb. 1975)-
Description	volumes : illustrations ; 28 cm
Current frequency	Quarterly, Feb. 1980-
Former frequency	Quarterly, Feb. 1975-May 1979 Five times a year, July 1979-Nov. 1979
ISSN	0098-3063
Linking ISSN	0098-3063

Google Exhibit No. 1063
Page 1295

LC classification	TK6563 .I2
Continues	IEEE transactions on broadcast and television receivers 0018-9308 (DLC) 57039724 (OCoLC)1604343
Variant title	Institute of Electrical and Electronics Engineers transactions on consumer electronics
Serial key title	IEEE transactions on consumer electronics
Abbreviated title	IEEE trans. consum. electron.
Related names	IEEE Consumer Electronics Society. IEEE Broadcast, Cable, and Consumer Electronics Society. IEEE Consumer Electronics Group. IEEE International Conference on Consumer Electronics.
LC Subjects	Household electronics--Periodicals. Radio--Receivers and reception--Periodicals. Television--Receivers and reception--Periodicals. Electronic apparatus and appliances--Periodicals.
Other Subjects	Electronic apparatus and appliances. Household electronics. Radio--Receivers and reception. Television--Receivers and reception.
Form/Genre	Periodicals. Periodicals.
Browse by shelf order	TK6563
Notes	Title from cover. Some volumes have supplements. Issues for <Nov. 1982, Aug. 1985> include some papers from International Conference on Consumer Electronics. A publication of: the IEEE Consumer Electronics Group, <Feb. 1976>-May 1976; IEEE Broadcast, Cable, and Consumer Electronics Society, Aug. 1976-<Nov. 1982>; IEEE Consumer Electronics Society, <May 1984->
Additional formats	Online version: IEEE transactions on consumer electronics (Online) 1558-4127 (DLC) 2005212495 (OCoLC)60639335 Online version: IEEE transactions on consumer electronics (OCoLC)752430525

Google Exhibit No. 1063
Page 1296

LCCN	75643472
CODEN	ITCEDA
Dewey class no.	621.38
National bib agency no.	PH 24/03/94
Other system no.	(OCoLC)ocm02241216
Additional Links	http://ieeexplore.ieee.org/servlet/opac?punumber=30
Type of material	Periodical or Newspaper
Content type	text
Media type	unmediated
Carrier type	volume

Item Availability

CALL NUMBER	TK6563 .I2 Set 1
Request in	Jefferson or Adams Building Reading Rooms
Status	Not Charged
Older receipts	v.22 (1976), v.24 (1978), v.27-v.61 (1981-2015), v.62:no.2-v.62:no.4 (2016:May-2016:Nov.), v.63:no.2 (2017:May). v.63:no.4-v.64:no.4 (2017:Nov.-2018:Nov.)
Supplements	v.30:no.2 (1984:May)

CALL NUMBER	TK6563 .I2 Copy 1 Unbound issues
Request in	Newspaper & Current Periodical Reading Room (Madison LM133)

Google Exhibit No. 1063
Page 1297

Status	Not Charged
CALL NUMBER	TK6563 .I2 Copy 2 Unbound issues
Request in	Newspaper & Current Periodical Reading Room (Madison LM133)

Google Exhibit No. 1063
Page 1298

LIBRARY CATALOG

1 of 1

PERIODICAL OR NEWSPAPER

IEEE transactions on consumer electronics

Full Record MARC Tags

000		04336cas a2200877 a 4500
001		11189359
005		20240917071747.0
008		751021c19759999nyuqr p 0 a0eng
010	__	|a 75643472
016	00	|a PH |z 24/03/94
022	0_	|a 0098-3063 |l 0098-3063 |2 1
030	__	|a TCEDA
032	__	|a 256480 |b JSPS
035	__	|a (OCoLC)ocm02241216
040	__	|a DLC |b eng |c DLC |d VSD |d OCL |d NSD |d DLC |d NST |d OCL |d NSD |d AIP | d OCL |d NST |d DLC |d NST |d DLC |d OCL |d DLC |d OCL |d NST |d NSD |d NST | d HUL |d VSD |d MYG |d CUS |d MYG |d OCL |d LYU |d IEEEX |d OCLCQ |d ORE | d VSD |d _VB |d OCLCQ |d OCLCO |d OCLCF |d OCLCO |d IUL |d OCLCQ |d BRF | d OCLCO |d PUL
042	__	|a pcc |a nsdp
050	00	|a TK6563 |b J2
082	04	|a 621.38 |2 19
210	0_	|a EEE trans. consum. electron.
222	_0	|a EEE transactions on consumer electronics
245	00	|a EEE transactions on consumer electronics.
246	3_	|a nstitute of Electrical and Electronics Engineers transactions on consumer electronics
249	0_	|i ACQUIRE Title: |a IEEE transactions on consumer electronics.
260	__	|a NewYork : |b Institute of Electrical and Electronics Engineers, |c ©1975-

Google Exhibit No. 1063
Page 1299

300	__	|a volumes : |b illustrations ; |c 28 cm
310	__	|a Quarterly, |b Feb. 1980-
321	__	|a Quarterly, |b Feb. 1975-May 1979
321	__	|a Five times a year, |b July 1979-Nov. 1979
336	__	|a text |b txt |2 rdacontent
337	__	|a unmediated |b n |2 rdamedia
338	__	|a volume |b nc |2 rdacarrier
362	0_	|a Vol. CE-21, no. 1 (Feb. 1975)-
500	__	|a Title from cover.
525	__	|a Some volumes have supplements.
550	__	|a Issues for <Nov. 1982, Aug. 1985> include some papers from International Conference on Consumer Electronics.
550	__	|a A publication of: the IEEE Consumer Electronics Group, <Feb. 1976>-May 1976; IEEE Broadcast, Cable, and Consumer Electronics Society, Aug. 1976-<Nov. 1982>; IEEE Consumer Electronics Society, <May 1984->
590	__	|a SERBIB/SERLOC merged record
592	__	|a ACQN: aq 94014910
592	__	|a 5 times a year
592	__	|a FRO, title update per FAXON list dated 9-4-86
650	_0	|a Household electronics |v Periodicals.
650	_0	|a Radio |x Receivers and reception |v Periodicals.
650	_0	|a Television |x Receivers and reception |v Periodicals.
650	_0	|a Electronic apparatus and appliances |v Periodicals.
650	_6	|a Appareils électroniques domestiques |v Périodiques.
650	_6	|a Radio |x Récepteurs et réception |v Périodiques.
650	_6	|a Télévision |x Récepteurs et réception |v Périodiques.
650	_6	|a Appareils électroniques |v Périodiques.
650	_7	|a Electronic apparatus and appliances. |2 fast |0 (OCoLC)fst00906772
650	_7	|a Household electronics. |2 fast |0 (OCoLC)fst00962119
650	_7	|a Radio |x Receivers and reception. |2 fast |0 (OCoLC)fst01087124
650	_7	|a Television |x Receivers and reception. |2 fast |0 (OCoLC)fst01146597
655	_7	|a Periodicals. |2 fast |0 (OCoLC)fst01411641
655	_7	|a Periodicals. |2 lcgft |0 http://id.loc.gov/authorities/genreForms/gf2014026139 |0 (uri) http://id.loc.gov/authorities/genreForms/gf2014026139
710	2_	|a IEEE Consumer Electronics Society.
710	2_	|a IEEE Broadcast, Cable, and Consumer Electronics Society.
710	2_	|a IEEE Consumer Electronics Group.

Google Exhibit No. 1063
Page 1300

711	2_	|a IEEE International Conference on Consumer Electronics.
776	08	|i Online version: |t IEEE transactions on consumer electronics (Online) |x 1558-4127 | w (DLC) 2005212495 |w (OCoLC)60639335
776	08	|i Online version: |t IEEE transactions on consumer electronics |w (OCoLC)752430525
780	00	|t IEEE transactions on broadcast and television receivers |x 0018-9308 |w (DLC) 57039724 |w (OCoLC)1604343
850	__	|a ArU |a CaBVaS |a DLC |a FU |a ICU |a IaAS |a MH-GM |a MoU |a NN |a NcD |a NcRS | a PPi |a PU |a TxArU |a ViBlbV
859	41	|u http://ieeexplore.ieee.org/servlet/opac?punumber=30
853	20	|8 1 |a v. |b no. |u 4 |v r |i (year) |j (month) |w q |x 02
863	41	|8 1.1 |a <34> |b <1> |i <1988> |j <02>
890	__	|a IEEE transactions on consumer electronics. New York. |i 75-643472
906	__	|a 7 |b cbc |c serials |d 3 |e ncip |f 19 |g n-oclcserc
920	__	|a Keep 1
923	__	|d 20020726 |s Group Registration
963	__	|a k.m.burke@ieee.org
984	__	|a srvf
985	__	|e eserial 200406
991	__	|b c-GenColl |h TK6563 |i .I2 |w SERIALS
991	__	|b o-NLS/REF |w ACQUIRE
992	__	|b 93000437 |w ACQUIRE
992	__	|b SER |w SERLOC

Request this Item | LC Find It

Item Availability

CALL NUMBER	TK6563 .I2 Set 1
Request in	Jefferson or Adams Building Reading Rooms
Status	Not Charged
Older receipts	v.22 (1976), v.24 (1978), v.27-v.61 (1981-2015), v.62:no.2-v.62:no.4 (2016:May-2016:Nov.), v.63:no.2 (2017:May). v.63:no.4-v.64:no.4

Google Exhibit No. 1063
Page 1301

(2017:Nov.-2018:Nov.)

Supplements	v.30:no.2 (1984:May)

CALL NUMBER	TK6563 .I2 Copy 1 Unbound issues
Request in	Newspaper & Current Periodical Reading Room (Madison LM133)
Status	Not Charged

CALL NUMBER	TK6563 .I2 Copy 2 Unbound issues
Request in	Newspaper & Current Periodical Reading Room (Madison LM133)

Google Exhibit No. 1063
Page 1302

Ex. 1039C

Copyright
United States Copyright Office

Try the Copyright Public Records System (CPRS) pilot with enhanced search features and filters.

Help | Search | History | Titles | Start Over

Public Catalog

Copyright Catalog (1978 to present) at DC4
Search Request: Left Anchored Title = IEEE Transactions on Consumer Electronics
Search Results: Displaying 24 of 47 entries

previous | next

Labeled View

IEEE Transactions on consumer electronics : a publication by the IEEE...

Type of Work: Serial
Title: IEEE Transactions on consumer electronics : a publication by the IEEE Consumer Electronics Group of the BCCE Society / [editor, Chester W. Sall].
Serial Publication Year: 1999
Serial Key Title: IEEE Transactions on consumer electronics
Imprint: New York : IEEE.
ISSN: 0098-3063
Description: print material.
Frequency: Quarterly.
Continues: IEEE transactions on broadcast and television receivers.
Description based on: Vol. CE-23, no. 4, Nov. 1977.
Copyright Claimant: Institute of Electrical and Electronics Engineers, Inc. (employer for hire)
Issues Registered: v. 45, no. 1, Feb99.; Pub. 1999-03-19; Reg. 2000-01-10; TX0005053510
v. 45, no. 2, May99.; Pub. 1999-05-27; Reg. 2000-01-10; TX0005053510
v. 45, no. 3, Aug99. Created 1999; Pub. 1999-09-15; Reg. 2000-01-10; TX0005055247
v. 45, no. 4, Nov99. (CO correspondence.) Created 1999; Pub. 1999-12-17; Reg. 2000-01-10; TX0005082759
Names: Sall, Chester W.
IEEE Consumer Electronics Group of the BCCE Society
BCCE Society. IEEE Consumer Electronics Group
Institute of Electrical and Electronics Engineers, Inc.

previous | next

Save, Print and Email (Help Page)
Select Download Format: Full Record | Format for Print/Save
Enter your email address: | Email

Help Search History Titles Start Over

Contact Us | Request Copies | Get a Search Estimate | Frequently Asked Questions (FAQs) about Copyright | Copyright Office

Google Exhibit No. 1063
Page 1304

Home Page | Library of Congress Home Page

Google Exhibit No. 1063
Page 1305

SOFTWARE OPTIMIZATION OF THE MPEG-AUDIO DECODER USING A 32-BIT MCU RISC PROCESSOR

Keun-Sup Lee, Young Cheol Park[1] and Dae Hee Youn
MCSP Lab., Dept. of Electrical and Electronic Eng., Yonsei University, Seoul, Korea
[1]Dept. of Division of Information Technology, Yonsei University, Wonju, Korea

ABSTRACT

This paper proposes optimization techniques to implement MPEG audio decoding algorithms in real-time using general-purpose 32-bit MCU RISC processor. Both MP3 and AAC LC profile decoding algorithms can be partitioned into two parts: control-intensive part and computation-intensive part. Optimization techniques in this study are focused on developing methodologies that are suitable for each part. We implemented MP3/AAC decoder using ARM-based RISC MPU and its test board. The both designed decoders with proposed optimizations could achieve decoding processes in real-time within an operating frequency of 35 MHz. And ISO 13818-4 compliance test results confirmed that the proposed decoders ensured full compliance with ISO 13818-3 audio decoder. These implementation results illustrate the efficiency of the proposed design methodology in both performance and cost.

1. INTRODUCTION

Recently, portable MPEG audio devices has been actively developed and already become as popular as portable Compact Disc (CD) or Mini-Disc (MD) devices in the consumer market. Since digital audio players do not require mechanical components needed for other devices such as CD, decoding processors consume a large portion of the battery power in use. Therefore, the power consumption of the decoding processors affects the battery life of the player.

Many portable players adopt DSP because of decoding complexity. But high power consumption of the DSP architecture brings short battery life. Moreover, complete digital audio decoding system may need an additional micro-controller for the bitstream parsing and overall system control. Therefore low power architecture suitable for portable devices is required.

As processor design technology develops, the performance of micro-controller unit (MCU) reaches old DSP level. If MPEG audio decoding algorithm can be implemented using MCU without DSP, it is possible to integrate system control functionalities with audio decoding on a single MCU processor. It would be beneficial to the overall power consumption of portable MPEG player. However, since MCU has architectural disadvantages such as simple and limited hardware resource, MCU-based implementation requires a great deal of optimization compared with DSP-based case. And different optimization techniques for DSP-based case are required.

This paper proposes optimization techniques to implement MPEG audio decoding algorithms in real-time using general-purpose 32-bit MCU RISC processor. Decoding algorithms implemented in the paper are MPEG-1/2 Layer-III (MP3) and MPEG-2 AAC (Advanced Audio Coding) Low Complexity profile[1][2]. We adopted ARM7TDMI architecture as the target 32-bit MCU RISC processor. MP3/AAC decoding software was written in C codes and compiled with the ARM C-cross compiler. But the compiled codes have much redundancy. According to implementation result, the C-compiled MP3 code with no optimization required about 60–70 MIPS to decode a 128 kbps 44.1 kHz stereo bitstream. Therefore, we performed further optimization for these codes.

The organization of this paper is as follows. In Section 2, the design methodology of MPEG-audio decoders is presented. Software optimization processes and techniques based on this proposed methodology are described in Section 3. In Section 4, we will report implementation results containing complexity and output audio accuracy. Finally, we summarize our research in Section 5.

Manuscript received June 24, 2002

0098 3063/00 $10.00 © 2002 IEEE

Authorized licensed use limited to: Reprints Desk. Downloaded on January 06,2025 at 23:28:02 UTC from IEEE Xplore. Restrictions apply.

Google Exhibit No. 1040
Page 001

Google Exhibit No. 1063
Page 1306

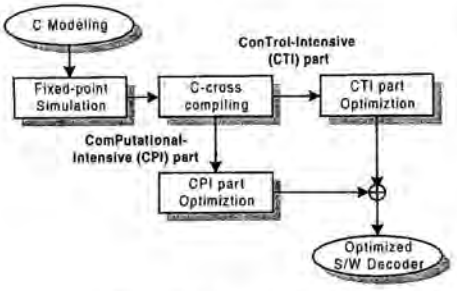


Figure 1. design methodology

2. DESIGN METHODOLOGY

The overall flow of the design methodology is illustrated in Figure 1. The flow describes implementation steps from C modeling to the optimized software decoder. In the fixed-point simulation stage, quantization errors caused by fixed-point operations are to be minimized. After the fixed-point simulation, C codes are converted into Assembly coded using a C-cross compiling tool. Since the resultant Assembly codes can contain much redundancy due to the inefficiency of the C-compiler, further optimization is required. As shown in Figure 1, there exist two distinct paths regarding algorithmic properties.

Figure 2 shows MP3 and AAC decoding flows. Both MP3 and AAC decoding algorithms consist of bit parsing, bit allocation decoding and unique synthesis filtering of their own. As shown in Figure 2, each decoding algorithm can be partitioned into two parts, which are control-intensive part and computation-intensive part. Control-intensive part consists of various operations related to control program flow. They have relatively low computational loads, but are difficult to control. On the contrary, Computation-intensive part has regular operations of high computational loads. Considering these algorithmic properties, distinct optimization technique must be performed for each part of the algorithmic. The optimized software decoder is obtained from these two distinct optimizations.

3. DESIGN OPTIMIZATION

Before the main optimization, we performed C-code level optimization to improve the performance of the code. We modified the code to be more suitable for the ARM processor and to help the compiler generate better code.

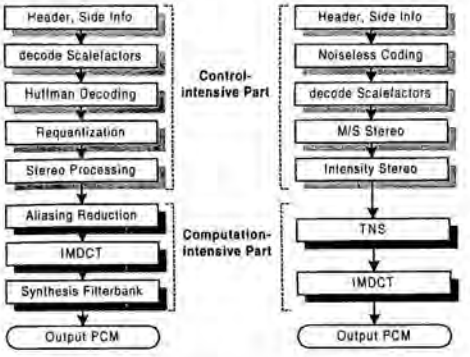


Figure 2. Partitioning MP3 and AAC decoding algorithm

3.1. C-compiler based Implementation

In general, assembly code from C-compiler has much redundancy. So, in order to get more efficient assembly code, hand assembly coding technique is required. But hand assembly coding requires so much time, and therefore, development cost increases.

Although C-compiler produces much redundancy, the redundancy from compiler varies according to processor architecture. Figure 3 shows the result of performance comparison between hand-optimized code and C-cross compiled assembly code of ARM7 Core. Target C code is Huffman decoding core routine, which belongs to control-intensive part.

According to Figure 3, two assembly codes are the same and the efficiency of C-cross compiler is very high. It is because of simple architecture and minimum resources of MCU processor that these results are obtained. Few instructions and resources due to simple architecture reduce the flexibility of assembly coding. So the optimizer of MCU compiler has more efficiency than that of DSP compiler with versatility.

Therefore, in this paper we compiled C code using C-cross compiler firstly, and then performed further optimization.

Authorized licensed use limited to: Reprints Desk. Downloaded on January 06,2025 at 23:28:02 UTC from IEEE Xplore. Restrictions apply.

Google Exhibit No. 1040
Page 002

Google Exhibit No. 1063
Page 1307

Original C-code

```
while( (x = *val++) < 0 ){
  if( Mask >= 0 )   val -= x;
  Mask <<= 1;
  num++;
}
```

Hand Assembly Code

```
    B ReadX
Loop:
    CMP Rd, #0
    SUBGE pHuff, pHuff, Re, LSL #2
    MOV Rd, Rd, LSL #1
    ADD Rb, Rb, #1
ReadX:
    LDR Re, [pHuff], #4
    CMP Re, #0
    BLT Loop
```

Cross Compiled Code

```
0xf894:B 0xf8a8
0xf898:cmp r4, #0
0xf89c:subgr r12, r12, r0, lsr r3
0xf8a0:mov r4, r4, lsl #1
0xf8a4:add r14, r14, #1
0xf8a8:ldr r0, [r12], #4
0xf8ac:cmp r0, #0
0xf8b0:blt 0xf898
```

Figure 3. *Comparison between two assembly coding methods*

3.2. Control-intensive Part Optimization

The control-intensive part of MPEG audio algorithm accompanies instructions related to the program flow control such as loop, conditional branch, and so on. Of all control-intensive routines Huffman decoding and scalefactor unpacking routines are very complex routines.

Huffman decoding consists mainly of bit unpacking and Huffman table searching routine, and these routines are iterated. Since bit unpacking and table search routines refer to bitstream buffer and ROM tables respectively, Huffman decoding routine requires to access memory frequently. These frequent memory accesses degrade the performance of overall system based on MCU, which has a weak point in memory access cycle. To minimize the number of bitstream buffer access, we use in-line hand-assembly to read maximum bits of Huffman codebook at once from bitstream buffer. This implementation reduces unpacking function call and bitstream buffer access overheads in iteration loop.

It is bit unpack routine that extracts Huffman code and scalefactor from bitstream buffer. Since bitstream buffer is finite size, to extract information from it circular addressing mode is required. But most of MCU's don't have complex addressing modes such as circular and bit-reverse addressing mode. So, to implement circular addressing with MCU additional computational loads due to if-clause are required. Whenever bit unpack routine is called, it must be checked whether bitstream read point reaches the end of bitstream buffer.

In order to remove this additional routine, we modified bitstream buffer control method as shown in Figure 4. By shifting unused bitstream to the first position at the start of each frame decoding, read point can be prevented from reaching the end of buffer. Considering the number of bit unpacking routine call during decoding process, complexity for shifting unused bitstream is very low. Furthermore, since bit unpacking routine called very often during control-intensive routines, the loads of subroutine call are somewhat heavy. So, it is advantageous to apply inline function to bit unpacking routine.

3.3. Computation-intensive Part Optimization

In MPEG audio decoding algorithm, frequency to time mapping synthesis filterbank routines come under computation-intensive part. For heavy computational loads of these routines fast algorithm commonly applies to them.

From the algorithmic point of view, most of operations are memory access, addition, and multiplication rather than branch. For example, as shown in Equation 1 one MAC operation needs one addition, one multiplication and three memory accesses that are two loads and one store.

$$y(i) = \sum_{n} x(n) \times c(n,i) \qquad (1)$$

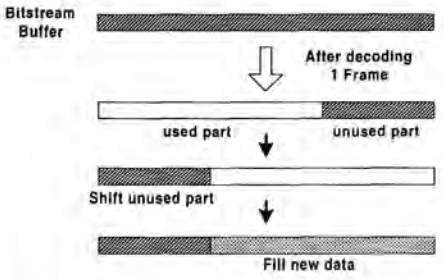


Figure 4. *Modified bitstream buffer control method*

Authorized licensed use limited to: Reprints Desk. Downloaded on January 06,2025 at 23:28:02 UTC from IEEE Xplore. Restrictions apply.

Google Exhibit No. 1040
Page 003

Google Exhibit No. 1063
Page 1308

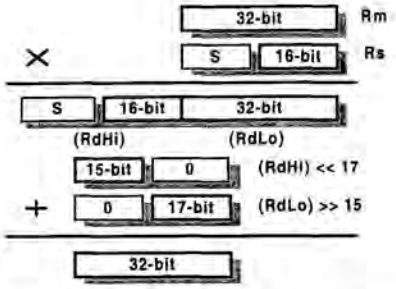


Figure 5. 32-bit x 16-bit fixed-point multiplication

Considering that MAC is the major operation of MPEG audio synthesis filterbank, computational loads for memory access are dominant. And due to low performance of multiplier in MCU multiplication also requires many clock cycles. Therefore we concentrated optimization on reducing the computational loads of memory access and multiplication.

Many MCU processors support multiple memory access operation. Multiple memory access makes it possible to load from and store to memory within less clock cycles than sequential memory access. For example, although ARM7 processor is capable of implementing a memory load in 3 cycles and store 2 cycles, by organizing data so that memory access is via consecutive addresses it is possible to achieve effective load and store operations in about single-cycle[3]. So with this multiple access the performance of synthesis filterbank can be significantly enhanced.

As mentioned above, computation-intensive routines consist of many multiplications. Since multiplication in MCU is a heavy burden, before the optimization for this we adopted fast algorithms that have few multiplications. Contrary to DSP, many MCU processors cannot operate full-length multiplication in 1 clock cycle due to simple multiplier. For that reason, many MCU processors support partial multiplication such as 32-bit x 8-bit. But, since multiplication with short data length can result in degrading data precision, it is important to determine the length of multiplicand and multiplier. Previous work showed that 32-bit x 16-bit multiplication is enough to obtain good audio output quality in MPEG audio decoder[4]. Therefore, considering not only data precision but also complexity, we adopted 32-bit x 16-bit. 16-bit data is mainly used for coefficient table. Figure 5 shows fixed-point multiplication operation we used. These operations require 5 clock cycles. But, since MAC operation don't need to execute last two instructions for extracting high product before the accumulation is completed, the computational loads for multiplication is smaller than 5 clock cycle in many cases.

4. IMPLEMENTATION

4.1. Target RISC Processor Architecture

In order to determine target MCU RISC processor for implementing MPEG audio decoder, next two requirements must be considered; the first requirement is high operational performance of processor suitable for real-time decoding process, and the second one is high accuracy of decoder's PCM output.

For real-time implementation of audio decoder, the performance of MCU RISC processor must be able to reach at least 30 DSP MIPS[4]. Therefore, considering the architecture of general MCU, target processor is required to operate in very high clock frequency. And, since MPEG audio decoding algorithms consists of many MAC (Multiply-ACcumulation) operations, target processor must have a multiplier to reduce the computational loads.

The sound quality of decoder's PCM output is affected by the accuracy of decoding operation. Especially, since fixed-point ALU produces a lot of quantization noise, it is required to reduce these errors for the cost of increasing operational complexity. To minimize the both these efforts and additional complexity, 32-bit data path was chosen for fixed-point ALU architecture. Since long data path produces high resolution and large dynamic range, 32-bit data path has higher accuracy than 16-bit or 24-bit data path.

After considering above requirements, we adopted ARM7TDMI as target MCU RISC processor. ARM7TDMI satisfies above all requirements, and besides, now it is used for digital audio codec applications. The components of this processor are as follows[3]:

- 32-bit RISC Architecture
- 65 ~ 100 MHz clock frequency
- 31 x 32-bit registers and 6 status registers
- 32-bit x 8-bit multiplier and 32-bit ALU
- 32-bit Barrel shifter

Authorized licensed use limited to: Reprints Desk. Downloaded on January 06,2025 at 23:28:02 UTC from IEEE Xplore. Restrictions apply.

Google Exhibit No. 1040
Page 004

Google Exhibit No. 1063
Page 1309

4.2. Implementation Result

In order to measure the sound quality for the output PCM of designed decoders objectively, ISO 13818-4 compliance test was performed[5]. According to Table 1 both MP3 and AAC decoder don't exceed the criteria of compliance test, and therefore, both satisfy ISO 13818-4 full compliance.

Table 2 summarizes the computational complexity of designed software decoders. For MP3 bitstream popular music signal and for AAC bitstream white noise signal are used for performance test. All two bitstream have the specification of 2 channels, 128 kbps and sampling frequency of 44.1 kHz.

In order to verify these results, we implemented MP3/AAC decoder using Samsung S3C4510B01 ARM-based RISC MPU and its test board. According to this implementation results, the both designed decoders with proposed optimizations could achieve decoding processes in real-time within a maximum operating frequency of 35 MHz. According to software libraries fully optimized by ARM Ltd., the operating frequency of MP3 decoder is 29 MHz and that of AAC decoder is 22 MHz (without TNS) and 28 MHz (with TNS)[6]. Considering that the proposed methodology is the optimization in C-cross compiling level with low development cost, the performance of designed software decoder is comparable with that of ARM's software libraries. These implementation results illustrated the efficiency of the proposed optimization techniques in both performance and cost.

5. CONCLUSIONS

In this paper, we proposed optimization techniques for implementing MPEG audio decoding algorithms in real-time using a general-purpose 32-bit MCU RISC processor. Target MPEG audio algorithms were MPEG-2 Layer-III (MP3) and MPEG-2 AAC Low Complexity profile. Considering algorithmic properties, both MP3 and AAC decoding algorithms could be partitioned into two parts, which are control-intensive part and computation-intensive part. Therefore, we proposed optimization techniques suitable for each part.

Test result confirmed that the proposed decoding algorithm ensured full compliance with ISO 13818-3 audio decoder. According to the implementation results, the both designed decoders with proposed methodology could achieve decoding processes in real-time within an operating frequency of 35 MHz. These implementation results illustrated the efficiency of the proposed optimization techniques in both performance and cost.

Table 1. Results of compliance test

	RMS of Error	Maximum Error
Full Compliance	**8.8097×10^{-6}**	**6.1035×10^{-5}**
MP3	**4.7483×10^{-6}**	**2.8729×10^{-5}**
AAC	**5.6012×10^{-6}**	**3.0518×10^{-5}**

Table 2. Maximum complexity of designed decoder

MP3	Complexity	AAC	Complexity
Bitstream	**17 %**	Bitstream	**18 %**
Decoding	**5 %**	Decoding	**6 %**
IMDCT	**16 %**	TNS	**22 %**
Filterbank	**62 %**	IMDCT	**54 %**
Total	**33.4 MHz**	Total	**32.6 MHz**

REFERENCES

[1] ISO/IEC 13818-3 "Generic Coding of Moving Pictures and Associated Audio (Part 3: MPEG-Audio)", 2nd Edition, Feb., 1997

[2] ISO/IEC JTC1/SC29/WG11 No.1650 "IS 13818-7 (MPEG-2 Advanced Audio Coding)", Apr., 1997.

[3] ARM, *ARM7TDMI Data Sheet*, 1995

[4] Keun-Sup Lee, Hyen-O Oh, Young-Cheol Park, and Dae Hee Youn, "High Quality MPEG-Audio Layer III Algorithm for a 16-BIT DSP", in Proc. IEEE ISCAS, vol. II, pp.205-208, May 6-9, 2001

[5] ISO/IEC 13818-4 "Generic coding of moving pictures and associated audio information (Part 4: Compliance testing)"

[6] ARM Ltd. Homepage (http://www.arm.com)

Authorized licensed use limited to: Reprints Desk. Downloaded on January 06,2025 at 23:28:02 UTC from IEEE Xplore. Restrictions apply.

Google Exhibit No. 1040
Page 005

Google Exhibit No. 1063
Page 1310

BIOGRAPHY



Keun-Sup Lee was born in Seoul, Korea, on 1975. He received his B.S. and M.S. degrees in Electronic Engineering from Yonsei University, Seoul, Korea in 1997 and 1999 respectively. He is currently a Ph.D. candidate in Department of Electrical and Electronic Engineering at Yonsei University, Seoul, Korea. His research interests include audio signal processing, DSP & ASIC implementation, high quality audio coding, HDL chip design, and algorithm optimization.



Young Cheol Park received his B.S., M.S., and Ph.D. degrees in Electronic Engineering from Yonsei University, Seoul, Korea in 1986, 1988, and 1993 respectively. From 1993 to 1995, he was a Post Doctoral Research Scholar of the graduate program in acoustics in Pennsylvania State University, PA, USA. From 1996 to 1998, he was with Samsung Electronics Co., Ltd., Kiheung Korea, working on the development of digital hearing aids. In 1999 he joined InTime Corp. in Seoul, Korea. His research interests include digital audio, speech coding, and adaptive filters. He is currently an assistant professor in Yonsei University, Wonju, Korea.



Dae Hee Youn received his B.S. degree in Electronic Engineering from Yonsei University, Seoul, Korea, in 1977, and the M.S. and Ph.D. degrees in Electrical Engineering from Kansas State University, Manhattan, Kansas, in 1979 and 1982, respectively. From 1982 to 1985, he was an assistant professor at the University of Iowa, Iowa City, Iowa. Since 1985, he has been with the Department of Electrical and Electronic Engineering at Yonsei University, Seoul, Korea, where he is currently a professor. His research interests include adaptive digital filter and its application, speech and audio signal processing, and real-time implementation of DSP algorithms.

Authorized licensed use limited to: Reprints Desk. Downloaded on January 06,2025 at 23:28:02 UTC from IEEE Xplore. Restrictions apply.

Google Exhibit No. 1040
Page 006

Google Exhibit No. 1063
Page 1311

Ex. 1040A

LIBRARY OF CONGRESS
0 011 575 754 4

IEEE
TRANSACTIONS
ON CONSUMER
ELECTRONICS

48

2002

TK 6563
.I2

IEEE TRANSACTIONS ON

CONSUMER ELECTRONICS

AUGUST 2002 VOLUME 48 NUMBER 3 ITCEDA (ISSN 0098-3063)

A Publication by the IEEE Consumer Electronics Society

Administrative Committee i
Officers and Committee Chairmen ii

Improvement of Low Gray Scale Linearity Using Multi-Luminance-Level Subfield Method in Plasma Display Panel
Ki-Duck Cho, Heung-Sik Tae and Sung-Il Chien 377
New Driving Scheme for White Color Balancing of Plasma Display Panel Television ..... *Sang-Hun Jang, Heung-Sik Tae and Sung-Il Chien* 382
Ultra Portable LCOS Projector with High-Performance Optical System
Satoshi Ohuchi, Tomohiro Miyoshi, Taro Imahase, Tsutomu Nakashima and Kazuo Shikita 388
A New Readout Circuit for an Ultrahigh Sensitivity CMOS Image Sensor
Toshihisa Watabe, Masahide Goto, Hiroshi Ohtake, Hirotaka Maruyama and Kenkichi Tanioka 394
JPEG2000 Coded Images Optimization Using a Content-Dependent Approach
Sebastiano Battiato, Antonio Buemi, Gaetano Impoco and Massimo Mancuso 400
Improving Image Resolution by Adaptive Back-Projection Correction Techniques
Giuseppe Messina, Sebastiano Battiato, Massimo Mancuso and Antonio Buemi 409
Digital Camera Connectivity Solutions Using the Picture Transfer Protocol (PTP) ... *Petronel Bigioi, George Susanu, Peter Corcoran and Irina Mocanu* 417
A Video Camera System with Enhanced Zoom Tracking and Auto White Balance ..... *Yoon Kim, June-Sok Lee, Aldo W. Morales and Sung-Jea Ko* 428
A New Enhancement Method for Digital Video Applications ..... *Yibin Yang and Lilla Boroczky* 435
Adaptive Motion-Compensated Interpolation for Frame Rate Up-Conversion
Sung-Hee Lee, Yoon-Cheol Shin, Seungjoon Yang, Heon-Hee Moon and Rae-Hong Park 444
Robust Transmission of JPEG2000 Images Over Noisy Channels ..... *V. S. Sanchez and M. K. Mandal* 451
A Modulation Scheme for Broadband Wireless Access in High Capacity Networks ..... *B. Fong, G. Y. Hong and A. C. M. Fong* 457
Error Localization for Robust Video Transmission ..... *G. Y. Hong, B. Fong and A. C. M. Fong* 463
NAMBA: Location-Aware Collaboration System for Shopping and Meeting ..... *Takashi Yoshino, Tomohiro Muta and Jun Munemori* 470
Device Discovery Via Residential Gateways ..... *Andrew Wils, Frank Matthijs, Yolande Berbers, Tom Holvoet and Karel De Vlaminck* 478
Framework for the Composition and Interoperation of the Home Appliances Based on
Heterogeneous Middleware in Residential Networks ..... *Song Yean Cho* 484
Home Network Infrastructure for HandHeld/Wearable Appliances ..... *Peter M. Corcoran, Joe Desbonnet, Petronel Bigioi and Ioan Lupu* 490
A Java-Based Home Network Middleware Architecture Supporting IEEE1394 and TCP/IP ... *Tai-Yeon Ku, Dong-Hwan Park and Kyeong-Deok Moon* 496
Massihn: A Multi-Agent Architecture for Intelligent Home Network Service ..... *Cheng-Fa Tsai and Hang-Chang Wu* 505
Complexity-Quality Analysis of Transcoding Architectures for Reduced Spatial Resolution
Anthony Vetro, Toshihiko Hata, Naoki Kuwahara, Hari Kalva and Shun-ichi Sekiguchi 515
A Frame-Based MPEG Characteristics Extraction Tool and Its Application in Video Transcoding
Kan-Li Huang, Yi-Shin Tung, Ja-Ling Wu, Po-Kang Hsiao and Hsien-Shuo Chen 522
Enhanced VSB System ..... *Wayne E. Bretl and Azzedine Touzni* 533
Automatic Measurements of Digital Receivers Using Sampled Signals ..... *Uwe Wiedemann, Dirk Ommer and Claus Muschallik* 539
Delivering Video, Voice and Data to Consumers Via an All-Fiber Network ..... *James O. Farmer* 548
Packet Loss Resilience for Video Stream Over the Internet ..... *Jae-Young Pyun, Jae-Jeong Shim, Sung-Jea Ko and Sang-Hyun Park* 556
Integrated System for Multimedia Delivery Over Broadband IP Networks ... *Florin Lohan, Irek Defée, Marius Vlad, Aurelian Pop and Prakash Sastry* 564
Design and Implementation of the SMIL (Synchronized Multimedia Integration Language) Player ..... *Dongkyoo Shin and Dongil Shin* 575
New Communication Style Using Television Broadcasting and the Internet ..... *Masahiro Kageyama, Tomokazu Murakami and Hisao Tanabe* 579
Comparison of Two Guided Scrambling Schemes for Optical Disks ..... *Akiomi Kunisa* 584
A Mobile Storage Device with the Magneto-Optical Disk of a 50mm Diameter and 730MB Capacity
S. Takahashi, T. Hamaguchi, T. Hisamitsu, M. Takuma and A. Yamada 589
Equalization Techniques Based on Fast RLS Algorithms to Improve Mobile Reception Performance of 8-VSB Receivers
Young Ho Oh, Sung Woo Park, SeongJoon Baek and Dae Jin Kim 596
Blind Decision Feedback Equalization for VSB-Based DTV Receivers ..... *Hyoung-Nam Kim, Yong-Tae Lee and Seung Won Kim* 602

(Continued on back cover)

"2002 International Conference on Consumer Electronics"

Google Exhibit No. 1063
Page 1314

CONTENTS *(Continued from front cover)*

Frequency Synchronization Algorithm Robust to Pilot Distortion in ATSC System ..... *Young-Jae Ryu and Dong-Seog Han* 610
Adaptive Frame Selection Algorithm for DVB-T ..... *Jung-Jin Kim and Dong-Seog Han* 617
A Timing Recovery for Return Channel of Digital Video Broadcasting ..... *Kyoung-Ha Mo, Hyoung-Kyu Song and Dae-Ki Hong* 624
Architecture for Decoding Adaptive Reed-Solomon Codes with Variable Block Length
..... *Mook Kyou Song, Eung Bae Kim, Hee Sun Won and Min Han Kong* 631
Performance Issues on the Wireless 2,4 GHz ISM Band in a Multisystem Environment . *Lauri Sydänheimo, Mikko Keskilammi and Markku Kivikoski* 638
Adaptive QOS Control of Multimedia Transmission Over Band-Limited Networks ..... *G. Y. Hong, A. C. M. Fong and B. Fong* 644
A Control Network Architecture Based on EIA-709.1 Protocol for Power Line Data Communications
..... *Woo-Seop Kim, Lok-Won Kim, Chang-Eun Lee, Kyeong-Deok Moon and Suki Kim* 650
An MPEG-2 CODEC LSI with an Audio Accelerator for Camcorders ..... *Taku Nakamura, Hiroyuki Marumori, Masaru Takahashi and Yukio Fujii* 656
A Fast Audio Bit Allocation Technique Based on a Linear R-D Model
..... *Chun-Yi Lee, Yao-Chun Fang, Hsiao-Chiang Chuang, Chung-Neng Wang, and Tihao Chiang* 662
Software Optimization of the MPEG-Audio Decoder Using a 32-Bit MCU Risc Processor ..... *Keun-Sup Lee, Young Cheol Park and Dae Hee Youn* 671
Class-D Amplification Combined with Switched-Mode Power Conditioning—The Route to Long Battery Life
..... *D. A. Grant, Y. Darroman and S. Burrow* 677
A Reconfigurable Audio Power Amplifier Delivering 150 W to 1-Ω Loudspeakers for In-Car Sound Systems ..... *F. Chelli, D. Brambilla and F. Stefani* 684
Efficient Video Coding Integrating MPEG-2 and Picture-Rate Conversion ..... *F. J. de Bruijn, W. H. A. Bruls, D. Burazerović and G. de Haan* 688
Object Based Refinement of Motion Vector Fields Applying Probabilistic Homogenization Rules .... *H. Blume, G. Herczeg, O. Erdler and T. G. Noll* 694
An Innovative, Programmable Architecture for Ultra-Low Power Motion Estimation in Reduced Memory MPEG-4 Encoder
..... *Daniele Alfonso, Fabrizio Rovati, Danilo Pau and Luca Celetto* 702
A Three-Band-Tuner for Digital Terrestrial and Multistandard Reception ..... *Sanggyu Sim, Ralph Kuhn, Bernd Pflaum and Claus Muschallik* 709
Versatile COFDM Demodulator Design Based on the Cordic-Algorithm ..... *Stephan W. Mondwurf* 718
Robust Wireless Image Transmission Based on Turbo-Coded OFDM ..... *Jie Yang, Moon Ho Lee, MingQi Jiang and Ju Yong Park* 724
A Measurement Based Feasibility Study of Space-Frequency MIMO Detection and Decoding Techniques
for Next Generation Wireless LANs ..... *Robert Piechocki, Paul Fletcher, Andrew Nix, Nishan Canagarajah and Joe McGeehan* 732
Optimal Antenna Diversity Combining for IEEE 802.11a System ..... *Xuemei Ouyang, Monisha Ghosh and Joseph P. Meehan* 738
The Application of Iterative Equalisation to High Data Rate Wireless Personal Area Networks
..... *A. G. Lillie, A. R. Nix, P. N. Fletcher and J. P. McGeehan* 743
Scenario Driven Evaluation and Interference Mitigation Proposals for Bluetooth and
High Data Rate Bluetooth Enabled Consumer Electronic Devices ..... *A. K. Arumugam, A. R. Nix, P. N. Fletcher, S. M. D. Armour and B. S. Lee* 754
A Systematic Evaluation Method for the Operation and Communication Reliabilities of the PLC Networked Home Appliances
..... *Tae-Gyu Chang, Nam-Ho Ahn and Hoon Kim* 765
Design and Development of a New POD Security Module Chip-Set Based on the OpenCable Specification
..... *Won Hee Kim, Won Jay Song, Bo Gwan Kim and Byung Ha Ahn* 770
A Hard Disk Drive Embedded Digital Satellite Receiver With Scene Change Detector for Video Indexing
..... *Yeong Kyeong Seong, Yun-Hee Choi, Jong-An Park and Tae-Sun Choi* 776
Development of a Consumer VCR Format and a High-Density Magnetic Recording Using an MR Head
..... *Keizo Tsunki, Kazuyuki Iesaka, Yuji Yanagi, Yoichiro Senshu and Yasutaka Kotani* 783
Design and VLSI Implementation of a Digital Audio-Specific DSP Core for MP3/AAC
..... *Kyoung Ho Bang, Nam Hun Jeong, Joon Seok Kim, Young Cheol Park and Dae Hee Youn* 790
Digital Image Stabilization by Adaptive Block Motion Vectors Filtering .... *Filippo Vella, Alfio Castorina, Massimo Mancuso and Giuseppe Messina* 796
HDD Characterization for A/V Streaming Applications ..... *Özcan Mesut and Niek Lambert* 802

Google Exhibit No. 1063
Page 1315



IEEE CONSUMER ELECTRONICS SOCIETY

The Consumer Electronics Society is an organization, within the framework of the IEEE, of members, with technical interests in all facets of consumer electronics products. All members of the IEEE may join the Society and have the option of receiving the Transactions on CONSUMER ELECTRONICS, for which the annual fee is $25.00. For more information, see address below. *Member copies of Transactions/Journals are for personal use only.*

IEEE TRANSACTIONS ON CONSUMER ELECTRONICS

WAYNE C. LUPLOW, *Publication Chairman and Editor*

Publications committee:
Dr. Walter Ciciora
John Henderson
Leander Hoke, Jr.
Dave Hostetler
C. Bailey Neal
Dr. Howard Prosser
Dr. Ted Rzeszewski
Dr. Anthony Vetro

THE INSTITUTE OF ELECTRICAL AND ELECTRONICS ENGINEERS, INC.

2002 Officers

RAYMOND D. FINDLEY, *President*
MICHAEL S. ADLER, *President-Elect*
HUGO M. FERNANDEZ VERSTEGEN, *Secretary*
DALE M. CASTON, *Treasurer*
LYLE D. FEISEL, *Vice President, Educational Activities*
LEEARL A. BRYANT, *President, IEEE USA*
JAMES M. TIEN, *Vice President, Publications, Products & Services*
W. CLEON ANDERSON, *Vice President, Regional Activities*
BEN C. JOHNSON, *President, Standards Association*
MICHAEL R. LIGHTNER, *Vice President, Technical Activities*

PETER W. STAECKER, *Director, Division IV*

Executive Staff
DANIEL J. SENESE, *Executive Director*

DONALD CURTIS, *Human Resources*
ANTHONY DURNIAK, *Publications*
CECELIA JANKOWSKI, *Regional Activities*
PETER A. LEWIS, *Educational Activities*
JUDITH GORMAN, *Standards Activities*
RICHARD D. SCHWARTZ, *Business Administration*
W. THOMAS SUTTLE, *Professional Activities*
JOHN WITSKEN, *Information Technology*
MARY WARD-CALLAN, *Technical Activities*

IEEE TRANSACTIONS ON CONSUMER ELECTRONICS is published quarterly by The Institute of Electrical and Electronics Engineers, Inc. Responsibility for the contents rests upon the authors and not upon the IEEE, the Society, or its members. **IEEE Corporate Office,** 3 Park Avenue, 17th Floor, New York, NY 10016-5997. **IEEE Operations Center,** 445 Hoes Lane, P.O. Box 1331, Piscataway, NJ 08855-1331. **NJ Telephone:** 732-981-0060. **Price/Publication Information:** Individual copies: IEEE members $10.00 (first copy only), nonmembers $20.00 per copy. (Note: Add $4.00 postage and handling charge to any order from $1.00 to $50.00, including prepaid orders.) For single copy orders: 1-800-678-4333 in the U.S.; 732-981-0060 outside the U.S. Member and nonmember subscription prices available on request. Available in microfiche and microfilm.

Copyright and Reprint Permissions: Abstracting is permitted with credit to the source. Libraries are permitted to photocopy for private use of patrons, provided the per-copy fee indicated in the code at the bottom of the first page is paid through the Copyright Clearance Center, 222 Rosewood Drive, Danvers, MA 01923. For all other copying, reprint, or republication permission, write to: Copyrights and Permissions Department, IEEE Publications Administration, 445 Hoes Lane, P.O. Box 1331, Piscataway, NJ 08855-1331. Copyright © 2002 by The Institute of Electrical and Electronics Engineers, Inc. All rights reserved. Periodicals postage paid at New York, NY, and at additional mailing offices.

Postmaster: Send address changes to IEEE TRANSACTIONS ON CONSUMER ELECTRONICS, IEEE, 445 Hoes Lane, P.O. Box 1331, Piscataway, N.J. 08855-1331. GST Registration No. 125634188. Printed in U.S.A.

COPYRIGHT OFFICE
OCT 31 2002
GROUP REGIS
COPY
LIBRARY OF CONGRESS

2002
IEEE CONSUMER ELECTRONICS SOCIETY
ADMINISTRATIVE COMMITTEE

Term Expires 2002

Jean Goolsby
Eastman Kodak Company
2338 Edgemere Dr.
Rochester, NY 14612

Ralph Justus
Consumer Electronics Association
2500 Wilson Blvd.
Arlington, VA 22201-3834

John Lappington
Broadcom Corporation
4835 River Green Parkway
Duluth, GA 30096

David M. Lewis
Intel Corporation
1501 S. Mopac Expy. AN1-4B-11
Austin, TX 78746

Stuart Lipoff
Applied Value Corporation
45 Hayden Avenue, Suite 2600
Lexington, MA 02421

Term Expires 2003

Hans Baumgartner
Sarnoff Corporation
933 Golden Gate Drive
San Diego, CA 92116

Patrick D. Griffis
Microsoft Corporation
109 Drakestown Road
Long Valley, NJ 07853

Andrew Hackett
Thomson Multimedia R&D France
1 Avenue Belle-Fontaine, BP 19
35511 Cesson-Sevigne Cedex, France

Sorin G. Stan
Philips Optical Storage
Bldg. SFJ-1, P.O. Box 80002
5600 JB Eindhoven, The Netherlands

Anthony Vetro
Mitsubishi Electric Research Labs
558 Central Avenue
Murray Hill, NJ 07974

Term Expires 2004

Wayne Brell
Zenith Electronics Corporation
2000 Millbrook Drive
Lincolnshire, IL 60069

Leander H. Hoke, Jr.
613 East Fox Den Drive
Knoxville, TN 37922

Wayne C. Luplow
Zenith Electronics Corporation
2000 Millbrook Drive
Lincolnshire, IL 60069

Vishal Markandey
Texas Instruments
P.O. Box 660199, MS 8635
Dallas, TX 75266-0199

Virginia Williams
Consumer Electronics Association
2500 Wilson Blvd.
Arlington, VA 22201-3834

Ex-Officio Members

Stefan Mozar
2nd Vice President – International Affairs
8 Doulton Drive
Cherrybrook
NSW 2126, Australia

C. Bailey Neal
Awards
295 Camden Drive
Zionsville, IN 46077

Terrence R. Smith
1st Past President
Sirius Satellite Radio
1221 Avenue of the Americas
New York, NY 10020

Almon Clegg
Fellow Evaluator/Liaison to the History Center
Consultant
6070 Lively Court, Suite 101
Cumming, GA 30130-6264

David Hostetler
Web Master
29221 Whitley Collins Drive
Rolling Hills Estates, CA 90275

Peter Staecker
Director, Division IV
167 Cedar Street
Lexington, MA 02421

Diane Williams
AdCom Executive Administrator
67 Raspberry Patch Drive
Rochester, NY 14612-2868

Google Exhibit No. 1063
Page 1317