System Level Modeling and Implementation of Video CODEC

Jong-il Kim

The author is with the Laboratory for Image and Video Engineering, Department of Electrical and Computer Engineering, The University of Texas at Austin, Austin, TX 78712-1084, USA. E-mail: jikim@vision.ece.utexas.edu Web: http://anchovy.ece.utexas.edu/ jikim .

Keywords

Embedded Systems, MPEG, H.263, H.263+, Multimedia, Ptolemy

Abstract

The project focuses on the system-level modeling of image and video processing systems. Based on the models given, image and video compression and decompression standards (ex. H.263, MPEG-4) will be implemented in a formal, consistent and extensible framework such as in Ptolemy with well-defined and optimized processing primitives. The models of computation and an implementation of image and video processing algorithm will also help automate partitioning of the design into hardware and software. And to automate the design of scalable, modular software and hardware for image and video processing systems, formal, mathematical models that confirm to the parallelism of image and video CODEC (encoder and decoder) will be developed.

I. Introduction

The multimedia industries are witnessing a rapid evolution toward integrating complete systems on a single chip. The most important characteristic of multimedia applications is the rapid growth of the complexity of the design. As an example, the new videophone and teleconferencing standard (H.263+) [1] is much more complex than the previous H.263 and H.261 standards [2], and the continued evolution of video coding standards from MPEG-1 and MPEG-2 to MPEG-4 [3] is accompanied by significant complexity increase and user flexible mode. The conventional approach for the system-level design and integration of video processing, combined the complexity with extremely short product design cycles, cannot illuminate the promising solution simply by implementing large parts of the system functionality in software running on application-specific instruction set processor (ASIP) cores.

To support mixed domain simulation and synthesis for video CODECs as an new design paradigm, this project focuses on the system-level modeling of video processing systems. Based on the models given, new video compression and decompression standards (ex. H.263, MPEG-4) will be implemented in a formal, consistent and extensible framework such as in Ptolemy with well-defined and optimized processing primitives. The models of computation and an implementation of image and video processing algorithm will also help automate partitioning of the design into hardware and software. The proposed design scheme will be implemented and verified in Ptolemy [4].

The following section II summarizes video compression standards and their design perspectives using embedded processors. Section III describes new design paradigm and Section IV concludes the report with future work.

II. Background

The main features of the standards for multimedia applications such as video CODECs and teleconferencing are the significantly increasing complexity and fast emerging. At the

time of MPEG-1 chips were available at the market, MPEG-2 standard was fixed, and before the commercial market for the previous H.263 standard [2] begins, new videophone and teleconferencing standard (H.263+) [1] emerges with multiple options and increased complexity. The continued evolution of video coding standards [3] from MPEG-1 and MPEG-2 to MPEG-4 is accompanied by significant complexity increase with user composable object-oriented fashion. The multimedia video CODEC standards recommended by the Motion Picture Experts Group (MPEG) and International Telecommunication Union (ITU), and their embedded design approaches are discussed in this section.

A. Overview of Video CODEC Standards

In technical point of view, video compression standards can be categorized into two groups. One group is for low-complexity, frame-based coding schemes including MPEG-2 and H.263. The other one is for high-complexity, object-based coding algorithms such as MPEG-4, H.263+ [3], [5], [2]. In MPEG-2 and H.263, the video information is assumed to be rectangular, or of fixed size, displayed at fixed interval. With the predecessors like MPEG-1 and H.261, MPEG-2 and H.263 standards have been applied for CD-ROM, Digital TV, Digital Video Disk (DVD), teleconferencing and High Definition TV (HDTV).

In MPEG-4, the concept of video object (VO) and video object plane (VOP) have been introduced [3]. VO and VOP correspond to entities in the bitstreams that a user can access and manipulate (with e.g., cut and paste operations). The VOP can have arbitrary shape. At the decoder side, together with the VOP, composition information is sent to indicate where and when each VOP is to be displayed. Also the user at decoder side may be allowed to change the composition of the scene displayed by interacting on the composition information. H.263 and H.263+ can be applied for object-oriented coding with additional information such as binary mask [2], but the functionality is not so broad as that of MPEG-4 [5].

As depicted in Figure 1, MPEG-4 core coder has similar structure as H.263 and MPEG-2,

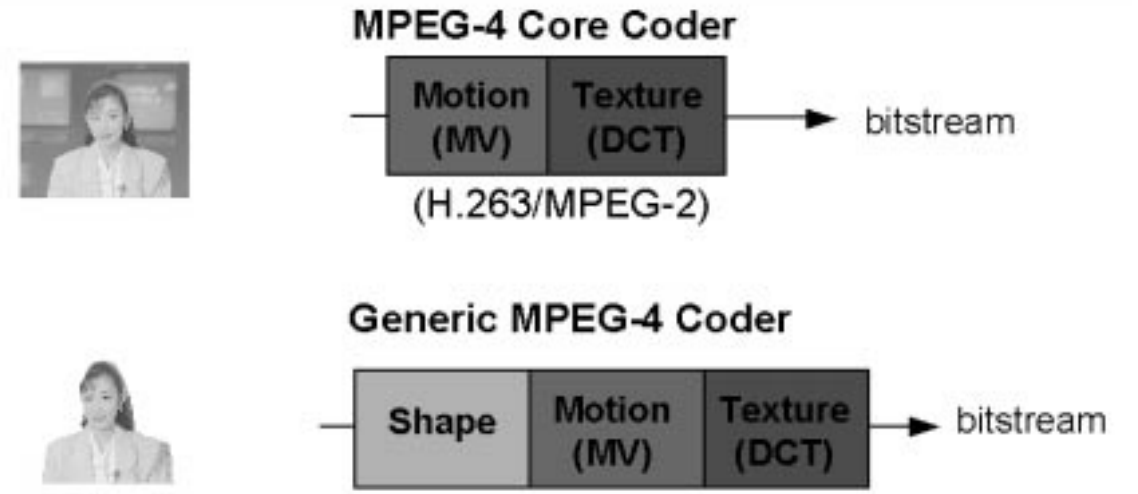

Fig. 1. MPEG-4 core and generic coding scheme

and MPEG-4 generic coder requires shape information to support arbitrary shape VOP. The overall MPEG-4 encoder blocks are described in Figure 2. Discrete Cosine Transform (DCT) and quantization are performed to compress the input VOP utilizing spatial correlation of the texture, and the inverse operations (IDCT and inverse quantization) reconstruct the input VOP to frame store block for motion estimation as in Figure 2. Next input VOP is compared with the best motion estimated block, and differential data between the input and predicted block is coded if it generates more compact code. Decoder block is embedded in encoder because of differential coding.

H.263+ is an extension of H.263 [1], [2], providing twelve new negotiable modes. These modes improve compression performance, allow the use of scalable bitstreams, and the user flexibility. The basic coding techniques are common to other standards like MPEG standards, and H.263 base coder is bitstream compatible with MPEG-4. As described in this section, the functionality of MPEG-4 coder includes other coding standards, and it will provide generic technology for multimedia communication applications and services.

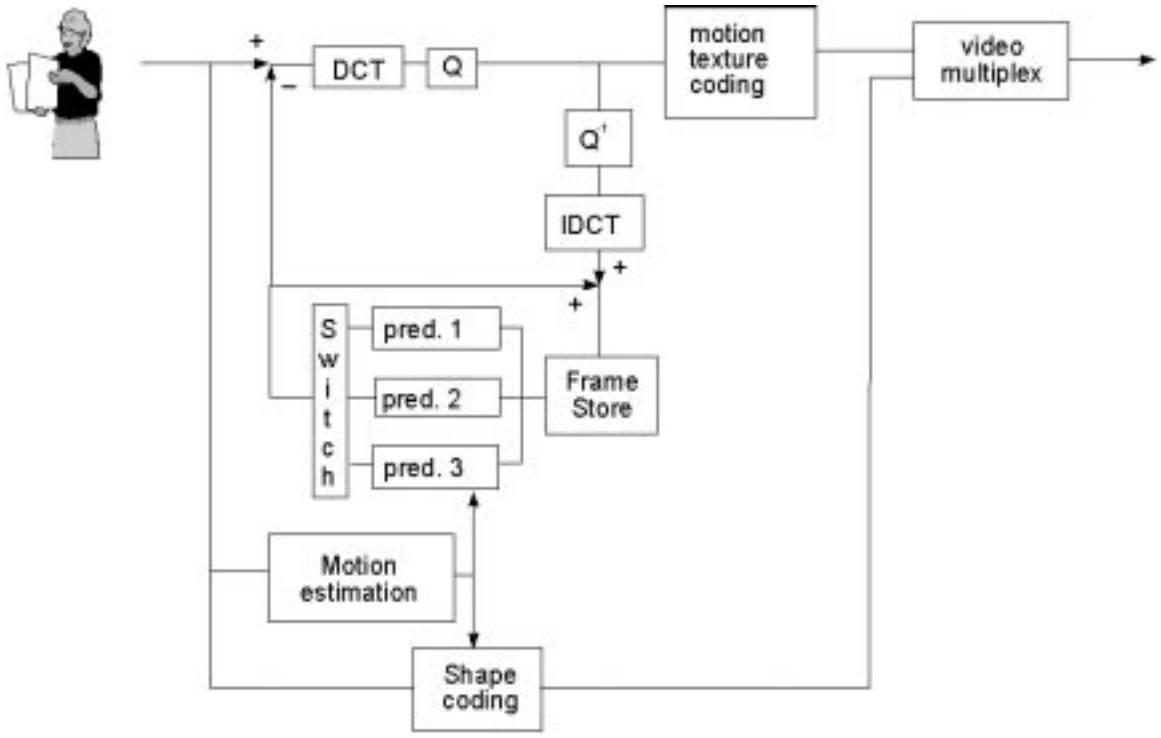

Fig. 2. Generic MPEG-4 encoder

B. Embedded Processors for video CODECs

Table I gives a list of products in the areas of MPEG and videophone multimedia applications. This category of the multimedia processors is of particular interest for this project as a modular design approach, since many common blocks like MPEG-1, MPEG-2, H.263, MPEG-4 video encoders and decoders are shared over a variety of emerging applications like set-top boxes for satellite and cable digital TV, DVB, DVD and PC-based multimedia accelerators [6].

As Table I shows, virtually all of the MPEG CODECs are based on application-specific instruction set processor (ASIP) cores and very long instruction word (VLIW), or on reduced instruction set processor (RISC) and custom hard-wired logic. The widespread decision to develop special ASIP cores highlights two important aspects of the multimedia market. First, cost is a critical factor. If a tailored architecture can deliver the same function at a low cost, then it will become a viable engineering choice. Second, none of the

TABLE I

EMBEDDED PROCESSORS FOR MULTIMEDIA

Company	*Processor*	*Applications*
Philips	VLIW DSP	MPEG-1/2 Decoder, H.261
NTT	Two RISC hard-wired ASIP	MPEG-2 Encoder
TI	MCU/DSP ASIP	MPEG-2 Decoder
TCEC	VLIW hard-wired DSP	MPEG-2 Decoder
SGC-Thomson	VLIW DSP ASIP	H261 video phone

current 32-bit microprocessor architectures is well-suited to the task performing routines necessary for video compression and decompression, e.g., motion estimation, DCT and IDCT.

Two key features of the embedded processors for multimedia applications are the increasing diversity of processor architectures, driven by low-cost consumer-oriented markets, and the wide range of architecture partitioning strategies and the diversity of building blocks. Many products performing the same function are designed using very different combinations of RISC, ASIP, and/or hard-wired co-processors.

But the significant growth of application complexity and the fast emerging new standards require new design strategies. As a results, many functions currently in hardware can be performed in software, and an ASIP will be necessary only for performance or cost reasons. However, because of the heterogeneity in video processing systems, no one design method is applicable to the entire system.

III. NEW DESIGN PARADIGM FOR VIDEO CODEC

Observing the fast emerging multimedia standards and significantly increasing complexity, new design paradigm is suggested in this report for the system level designers.

While system designers seek to shorten design cycles, improve quality, and reduce cost of the systems, no single design method is applicable to the entire systems such as multimedia in that they contain a variety of algorithms implemented by many software and hardware technologies. Video processing and conferencing systems are heterogeneous in that they contain a variety of algorithms implemented by many software and hardware technologies. Because of the heterogeneity in video processing systems, no one design method is applicable to the entire system design. Successful system design approaches integrate application-specific design methods and implementation technologies in a formal, consistent, extensible framework. The entire system can be specified, simulated, and synthesized in a formal, consistent framework, and extensibility supports new algorithms, tools, languages, and architectures to be integrated into the given framework to be rapidly retargeted to new technologies.

The design approach should be based on the use of one or more formal models to describe the behavior of the system at a high level of abstraction. During the implementation process, many different specifications and modeling techniques will be used. Some of the video CODEC building blocks can be model by synchronous dataflow (SDF) and other blocks which have data-dependent behaviors can be modeled by discrete event (DE) or by boolean dataflow (BDF) [4]. The modeling for the video CODEC is not fixed but variable for the implementation purpose. If the video CODEC combines with some network or communication protocols, DE model is better because of the notion of time. Ptolemy gives rich libraries for SDF, allows mixed model simulation and supports code generation in C and VHDL. Once SDF model is fixed, the construction of efficient loop structure from SDF graphs allows the advantages of inline code generation under stringent memory constraints. A variety of efficient design metric exist, so that during the modeling and optimizing the entire systems, different metric will be applied. Ptolemy supports multiple scheduling algorithm with minimization of code size and memory. The project will include the evaluation of Ptolemy for video CODEC implementation using two softwares which

are generated code under Ptolemy and public-domain hand written code.

The project focuses on the system-level modeling of video processing systems. Based on the models given, image and video compression and decompression standards (ex. H.263, MPEG-4) will be implemented in a formal, consistent and extensible framework such as in Ptolemy with well-defined and optimized processing primitives. The models of computation and an implementation of video processing algorithm will also help automate partitioning of the design into hardware and software. And to automate the design of scalable, modular software and hardware for multimedia video processing systems, formal, mathematical models that confirm to the parallelism of image and video CODEC will be developed. The proposed design scheme will be verified in Ptolemy [4] to automate the design of scalable (both compile-time and run-time scalability) video CODEC.

IV. Conclusion

The fast emerging standards and significantly increasing complexity require new design paradigm. Because of the heterogeneity in video processing systems, conventional design method cannot provide promising solution to the entire system design. Managing the design complexity and heterogeneity is the key problem for system-level designer. The future work will be for the models of computation for video CODEC and its implementation on Ptolemy.

References

[1] G. Côté, B. Erol, M. Gallant, and F. Kossentini, "H.263+: video coding at low bit rates," *submitted to IEEE Trans. on Circuits and Systems for Video Tech.*, http://www.ece.ubc.ca/image/papers/h263plus.pdf.

[2] J. Hartung, A. Jacquin, J. Pawlyk, J. Rosenberg, H. Okada, and P. E. Crouch, "Object-oriented H.263 compatible video coding platform for conferencing applications," *IEEE Journal on Selected Areas in Communications*, vol. 16, no. 1, pp. 42–55, Jan. 1998.

[3] T. Sikora, "MPEG digital video-coding standards," *IEEE Signal Processing Magazine*, vol. 14, no. 5, pp. 82–100, Sept. 1997.

[4] S. S. Bhattacharyya, P. K. Murthy, and E. A. Lee, "Synthesis of embedded software from synchronous dataflow specifications," *to appear in Journal of VLSI Signal Processing*, http://ptolemy.eecs.berkeley.edu/papers/publications.html/pubs1998.html.

[5] L. Chiariglione, "MPEG and multimedia communications," *IEEE Trans. on Circuits and Systems for Video Tech.*, vol. 7, no. 1, pp. 5–18, Feb. 1997.

[6] P. Paulin, C. Liem, M. Cornero, F. Nacaba, and Gert Gossens, "Embedded software in real-time signal processing systems: Application and architecture trends," *Proc. of the IEEE*, vol. 85, no. 3, pp. 419–435, Mar. 1997.