REAL-TIME SVC DECODER IN EMBEDDED SYSTEM

Srijib Narayan Maiti, Amit Gupta
STMicroelectronics Pvt. Ltd., Greater Noida, Uttar Pradesh, India

Emiliano Mario Piccinelli, Kaushik Saha
STMicroelectronics, Agrate, Italy

Keywords: Scalable Video Coding (SVC), Spatial Scalability, H.264/AVC.

Abstract: Scalable Video Coding (SVC) has been standardized as an annexure to the already existing H.264 specification to bring more scalability into the already existing video standard, keeping the compatibility with it. Naturally, the immediate support for this in embedded system will be based on the existing implementations of H.264. This paper deals with the implementation of SVC decoder in SoC built on top of existing implementation of H.264. The additional processing of various functionalities as compared to H.264 is also substantiated in terms of profiling information on a four issue VLIW processor.

1 INTRODUCTION

The purpose of SVC specification is to enable encoding of high-quality video bitstreams containing one or more subsets that can themselves be decoded with a complexity and reconstruction quality similar to that achieved using the existing H.264/MPEG-4 AVC design with the similar quantity of data as in the subset bitstream. Thus making it possible to adapt the bit rate of the transmitted stream to the network bandwidth, and/or the resolution of the transmitted stream to the resolution or rendering capability of the receiving device. The new standard (ITU-T, 2007) implements the best existing techniques for scalable video coding, as well as some new interesting algorithms. This is accomplished by maintaining a layered structure of the compressed video bit stream. The bitstream which is of minimum reconstruction quality is termed as 'base layer' and is AVC compliant. All the higher quality bitstreams are called enhancement layers. SVC standard gives the opportunity to have the scalability either temporally, spatially or in terms of quality, giving huge flexibility and choice to the end customers and broadcasters (Heiko et al, 2007). Temporal scalability is possible by using a hierarchical picture scheme and only adds syntax to high-level AVC that enables "easy" identification of temporal layers. Spatial scalability (Segall et al, 2007) consists of adding inter-layer prediction modes (prediction of texture, motion parameters and residual signal) to AVC motion-compensated prediction and intra-coding modes. The spatial layers need not be dyadic: the standard supports the use of any arbitrary ratio by ESS (Extended Spatial Scalability) tool. Medium Grain Scalability (MGS) is the quality scalability supported in the standard. Coarse grain fidelity scalability (CGS) improves the video quality and works in exactly the same way as spatial scalability, with the same size for base and enhancement layers. For the purpose of this paper, we would be restricted to spatial scalability. But the work is applicable to CGS also because CGS is a special case of spatial scalability only.

A vast published literature is already available, describing the algorithms supporting the new features and tools introduced by SVC: this paper will not analyze in details such novelties, but it will focus to substantiate the implementation of SVC decoder (to support spatial scalability) in real time embedded system which is already capable of decoding H.264/MPEG-4 AVC.

Francois and Vieron (2006), describes inter-layer motion and texture prediction processes, highlights performance comparisons with alternate solutions for Extended Spatial Scalability (ESS), which is a special tool supported only in SVC scalable high profile. F. Wu and his team proposed a

Maiti S., Gupta A., Piccinelli E. and Saha K. (2009).
REAL-TIME SVC DECODER IN EMBEDDED SYSTEM.
In *Proceedings of the International Conference on Signal Processing and Multimedia Applications*, pages 5-8
DOI: 10.5220/0002226300050008

framework for Fine Grain Scalability (FGS) in 2001, which is one of the quality scalability proposed in SVC but yet to come in force in the standard. Pelcat, Blestel and Raulet discussed about the data flow for SVC decoding built on top of H.264 decoder. But their objective of the study was to see the impact of SVC decoding on the data flow of H.264. There are many studies on the upsampling filtering introduced in SVC (Shin et al, 2008) (Frajka and Jegger, 2004) as well as on the motion estimation/compensation for faster processing applicable to SVC (Wu and Tang) (Lee et al) (Lin et al, 2007) but none of these talks about SVC decoding system as a whole, specially for real-time embedded system.

In section 2, we describe typical functional blocks for decoding a slice of base layer of SVC stream which is compatible with H.264/AVC. Specifically, the blocks which need modifications for decoding a slice of an enhanced layer of an SVC streams are highlighted. Section 3 describes the modifications of existing blocks and additional blocks to support SVC. Section 4 explains the simulation results to support the new features introduced in SVC, followed by highlighting future developments in this direction and references.

2 DECODING OF H.264/AVC STREAM

Figure 1 shows the block diagram of a typical H.264/AVC decoder. A picture or frame consists of one or more slices, which in turn consists of macroblocks. The whole processing can be described in two phases. In the first phase, processing is at slice level and the other at the macroblock level.

An H.264/AVC bitstream is coded either using CABAC (Context Adaptive Binary Arithmetic Coding) or CAVLC (Context Adaptive Variable Length Coding). So, the first step in the decoding process is to parse the slice header and find the parameters required to decode the macroblocks.

In the second phase of processing, all the operations are performed on each of the macroblock one by one which constitutes a slice. The residual signal is first inverse quantized (IQ) and inverse transformed (IDCT) to convert into time-domain signals. Then either intra-prediction or inter-prediction is performed based on the type of prediction information. Inter-prediction is performed from the reference pictures stored in the decoded picture buffer (DPB). The last step is to perform loop-filtering or deblocking. The output is a picture or frame of YUV samples.

3 MODIFICATIONS FOR SVC DECODING

Figure 2 shows the block diagram to decode an enhanced spatial layer of SVC bitstream built on top of H.264 decoder. The functional blocks which are modified and the additional functionalities are filled with dots.

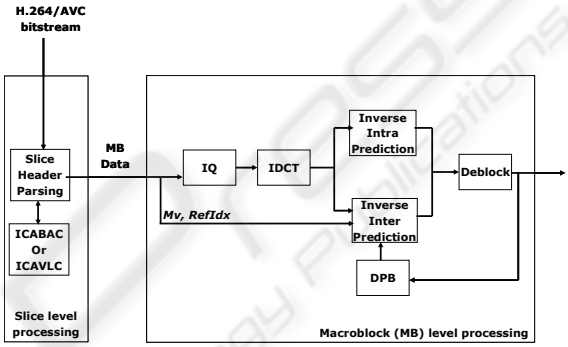

Figure 1: Block Diagram of H.264/AVC decoder.

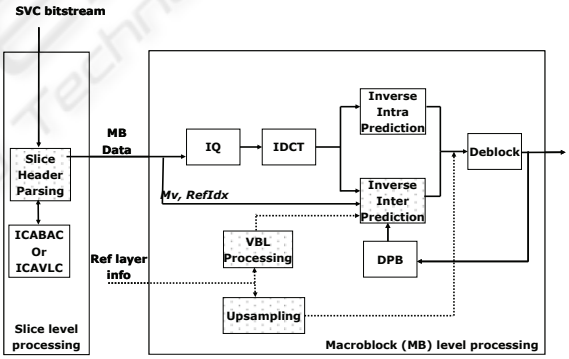

Figure 2: Block Diagram of H.264/SVC decoder.

'Slice Header Parsing' block has been modified to process the additional syntax element introduced by the SVC standard, keeping compatibility with normal H.264 stream.

As far as macroblock level processing is concerned, there is one new feature introduced in SVC as compared to H.264 which is called inter-layer-prediction. All the modifications/additions of functionalities for SVC support are basically to address this only. Inter-layer-prediction can be described in two categories, inter-layer-intra prediction and inter-layer-inter prediction. In inter-layer-intra prediction, the co-located pixels of

reference layer macroblocks are upsampled and added with that of the enhanced layer to reconstruct the enhanced layer intra macroblock. This feature is highlighted in the 'Upsampling' block alongwith the dotted arrows associated with this block. This particular macroblock mode is called I_BL as per the SVC standard. The 'Upsampling' block of figure 2 is also used to upsample the reference layer residuals to support inter-layer-inter prediction as explained in details in the following paragraphs.

Inter-layer-inter prediction can be classified in two categories, inter-layer-motion-prediction and inter-layer-residual-prediction. In inter-layer-motion-prediction, the motion information for a particular macroblock of enhanced layer is completely/partially inferred from the collocated macroblock of reference layer. These are controlled by few flags introduced in the enhanced layer macroblock syntax as specified in the SVC standard. These are base_mode_flag, motion-prediction_flag_l0, motion_prediction_flag_l1. When base_mode_flag is enabled for a particular macroblock of enhanced layer, the motion information for that macroblock are not transmitted in the bitstream, so, it is completely inferred/derived from that of the collocated reference layer macroblock. motion-prediction_flag_l0, motion_prediction_flag_l1 are both zero in this case. Processing of this is done by the 'VBL (Virtual Base Layer) Processing' block. This block is also responsible for conversion of interlaced reference layer to progressive enhanced layer or vice versa. But in this paper, we will discuss the computational load of VBL generation for progressive reference layer to progressive enhanced layer.

In inter-layer-residual-prediction, residual signal of reference collocated macroblock is upsampled and added to that of the enhanced layer macroblock. This is controlled by another flag, residual_prediction_flag, introduced in the macroblock syntax of enhanced layer as specified in the SVC standard. In theory, this upsampled residual is to be added with the residual of the enhanced layer macroblock before IDCT block, but this can be done after 'Inverse Inter Prediction' also since this a simple addition operation. This has been done to re-use the adder, both for 'Inverse Intra Prediction' and 'Inverse Inter Prediction'. So, the 'Upsampling' block of figure 2 is used both for pixel upsampling as well as residual upsampling. The 'Inverse Inter Prediction' block has been modified accordingly to accommodate these changes. We will discuss about the computational loads of these modified and added functional blocks in terms of profiling on a four-issue VLIW processor in the next section.

4 SIMULATION RESULTS

A typical implementation of H.264/AVC in real-time embedded system is a mixture of dedicated hardware accelerators and software (firmware). As an example, as shown in figure 1, the entire processing upto slice header parsing can be implemented in this processor whereas ICABAC/ICAVLC, IQ, IDCT, intra & inter-prediction, deblocking are implemented with the help of special purpose hardware. These hardware blocks are typically controlled by the processor/(s). Our experimental framework is also similar to this kind architecture. The processor is a 4-issue VLIW, assisted by hardware accelerators for operations mentioned above.

Profiling results have been obtained in terms of cycles needed by all the functionalities required to decode one particular type of macroblock this processor. This does not include the cycles needed by the hardware accelerators to complete the decoding of the particular type of macroblock.

The incremental load of the 'Slice Header Parsing' block as shown in figure 2, is not significant. It's only the additional control flow introduced because of the new syntax elements added in the slice header syntax of an enhanced layer. Depending on specific bitstream, this, at times consuming lesser cycles than that used to be for H.264/AVC only. This has been possible because of the efficient design of the standard itself. The profiling information is merely testifying this.

'VBL' block is a new functionality introduced and we have implemented this in the same processor i.e. the computation of motion vector predictors for enhanced layer macroblocks from the reference layer. Profiling results have been obtained for all kinds of macroblock types in the SVC/H.264 bitstream but for the functionalities performed by this processor. On an average this block takes about 20% of the cycles consumed to decode a particular type of macroblock mode. By applying specific optimization for the 4-issue VLIW processor this can be improved further. The computational load of the functionalities performed by the dedicated hardware accelerators is not included in this result.

The only modification in the 'Inverse Inter Prediction' hardware is the introduction of additional control information for motion_prediction_flags. When the motion-prediction_flag for a reference list of a partition is

enabled, the motion vector predictors are inferred from the reference layer, generated in the 'VBL' processing block earlier. The conventional median predictions etc. are not required in this case. As a result, the incremental computational load is not significant in this block also.

Upsampling block is probably the most computationally intensive which has been introduced. In our implementation, this has been implemented as a hardware accelerator controlled by the VLIW processor. This is about 10-15% of the total decoding time, when a full software implementation is simulated on a PC.

5 FUTURE DEVELOPMENTS

All the implementations/ modifications to decode a slice/macroblock of enhanced layer bitstream conforming to SVC which has been discussed in earlier sections are presently for spatial enhanced layer only. Although, the impact might not be so much, these will be tested/modified further to support the rest of the features introduced by the SVC standard. The 'VBL' block needs to support conversion of interlace to progressive and vice versa. Also, support for arbitrary resolution ratios between the enhanced layer and the reference layer needs to be incorporated. Some indications about the gate count and frequency of operation to support a specific profile of SVC can be obtained by simulating the 'Upsampling' block in an ASIC (Application Specific Integrated Circuit) or FPGA (Field Programmable Gate Array).

Modifications corresponding to quality scalability will be incorporated. Memory requirements corresponding to these features will also be profiled in near future.

REFERENCES

Advanced video coding for generic audiovisual services. ITU-T H.264 (11/07).

Heiko S, D Marpe, member, IEEE, and T Wiegand, member, IEEE, "Overview of the Scalable Video Coding Extension of the H.264/AVC Standard", *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 17, no. 9, 2007, pp. 1103-1120.

C. A Segall, *Member, IEEE*, and G. J. Sullivan, fellow, IEEE, "Spatial Scalability Within the H.264/AVC Scalable Video Coding Extension", *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 17, no. 9, September 2007, pp. 1121-1135.

E. Francois, J Vieron, "Extended Spatial Scalability: A Generalization of Spatial Scalability for non Dyadic Configurations", *ICIP 2006*, pp. 169-172.

F. Wu, S. Li, and Y. Q. Zhang, "A framework for efficient progressive fine granular scalable video coding," *IEEE Trans. Circuits Syst. Video Technol.* 11(3), March 2001, pp. 332–344.

M Pelcat, M Blestel, M Raulet, "From AVC Decoder to SVC: Minor Impact on a Dataflow Graph Description", Institute of Electronics and Telecommunications of Rennes (IETR), Image Processing and Remote Sensing Group, 20, Avenue des Buttes de Coësmes, 5043 RENNES Cedex, France.

Il-hong Shin and H W Park, "Efficient down-up sampling using DCT kernel for MPEG-21 SVC", Dept. of Electrical Engineering, Korea Advanced Institute of Science and Technology, 373-1 Guseong-dong, Yuseong-gu, Daejeon 305-701, Korea.

Yun-Da Wu and Chih-Wei Tang, "The Motion Attention Directed Fast Mode Decision for Spatial and CGS Scalable Video Coding", Visual Communications Laboratory, Department of Communication Engineering, National Central University, Jhongli, Taiwan.

Yu Wang, Lap-Pui Chau* and Kim-Hui Yap, "GOP-based Unequal Error Protection for Scalable Video over Packet Erasure Channel", *IEEE Broadband Multimedia 2008*.

B Lee, M Kim, S Hahm, C Park, K Park, "A Fast Mode Selection Scheme in Inter-layer Prediction of H.264 Scalable Extension Coding", Information and Communications University, Korea, Korean Broadcasting System, Technical Research Institute, Korea.

Jia-Bin Huang, Yu-Kun Lin, and Tian-Sheuan Chang, "A Display Order Oriented Scalable Video Decoder", *APCCAS 2006*, pp. 1976 -1079

IlHong Shin, Haechul Choi, Jeong Ju Yoo, and Jin Woo Hong, "Fast decoder for H.264 scalable video coding with selective up-sampling for spatial scalable video coding", *SPIE* VOL. 47(7), July 2008.

T. Frajka and K. Jegger, "Downsampling dependent upsampling of images," *Signal Process. Image Commun.* 19, March 2004, pp. 257–265.

H. C. Lin, W. H. Peng, H. M. Hang, and W. J. Ho, "Layer-adaptive mode decision and motion search for scalable video coding with the combination of coarse grain scalability(CGS) and temporal Scalability," *IEEE International Conference on Image Processing*, Sept. 2007 pp. 289-292.