A COST-EFFICIENT RESIDUAL PREDICTION VLSI ARCHITECTURE FOR H.264/AVC SCALABLE EXTENSION

Yi-Hau Chen, Tzu-Der Chuang, Chuan-Yung Tsai, Yu-Jen Chen, and Liang-Gee Chen

DSP/IC Design Lab.,
Graduate Institute of Electronics Engineering,
National Taiwan University, Taipei, Taiwan
Email: {ttchen, peterchuang, cytsai, yjchen, lgchen}@video.ee.ntu.edu.tw

ABSTRACT

In this paper, we propose a cost-efficient residual prediction hardware architecture to support inter-layer prediction in the state-of-art H.264/AVC scalable extension. Several residual prediction schemes are analyzed in hardware architecture and coding performances, and an integer motion estimation (IME)-simplified scheme is adopted. Then, the linearity of Hadamard transform is introduced to achieve data sharing for residual prediction in fractional motion estimation (FME) architecture. An Hadamard-free residual prediction FME architecture is proposed with 40% cost saving compared to direct implementation of duplicating FME module. The proposed architecture is implemented with 86K gates at 220 MHz by UMC 90nm technology for encoding HDTV720p 30fps. The proposed design concept can be also applied in other FME VLSI designs and software acceleration for supporting residual prediction.

1. INTRODUCTION

In former video coding standards, such as MPEG-2 and H.264, coding efficiency is always the main target. However, due to the prevalence of streaming multimedia applications over wired and wireless networks, more and more researchers pay attention to the functionality and scalability on video. Since late 2002, the Moving Picture Experts Group (MPEG) began to call for proposals of scalable video coding (SVC). These SVC proposals are merged into a Joint Scalable Video Model (JSVM) as the scalable extension of H.264/AVC [1][2]. In current JSVM, three main types of scalability, temporal scalability, spatial scalability, and SNR scalability are provided.

Currently, SVC adopts multi-layer coding structure to provide various video scalabilities. Figure 1 shows the SVC encoder architecture by using a multi-scale pyramid with 2 levels of spatial scalability[1][2]. Spatial scalability is achieved by using different spatial resolution layers. To improve the coding performance of SVC, it is necessary to remove all the redundant information among the multi-layer coding structure. In H.264, the inter- and intra-prediction for single-layer coding scheme have been maturely developed. To remove the rest redundancy between spatial layers, Schwarz et al. proposed an inter-layer prediction scheme in [3]. The main concept is to remove the redundancy of motion and residual between base layer (BL, smaller frame resolution) and enhancement layer (EL, higher frame resolution) as shown in Fig. 1. The coding performance can be improved about 1 dB compared to that without any inter-layer prediction tools as shown in Fig. 2.

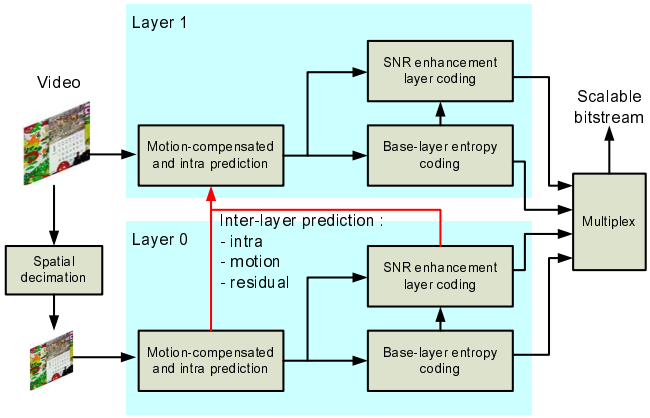


Fig. 1. SVC encoder structure with two spatial layers.

Although the inter-layer prediction can improve the coding performance, it nearly doubles the computation complexity since the current block's pixels are subtracted by upsampled residual block in residual prediction and it may lead to different prediction results. In following context, the prediction without base-layer residual is represented as "normal prediction" to distinguish from residual prediction. If the prediction engine which contains both normal prediction and residual prediction are directly implemented from previous H.264 designs [4, 5], the operating frequency will be doubled or the cost of processing elements will be largely increased. In this paper, we propose an integer motion estimation (IME)-simplified scheme and introduce linearity of

Hadamard transform into fractional motion estimation (FME) architecture. An Hadamard-free residual prediction FME architecture which can parallel process normal prediction and residual prediction is proposed. Compared to direct implementation, the proposed architecture can save 80% hardware cost and maintain the same processing capability according to referenced FME designs.

2. INTER-LAYER RESIDUAL PREDICTION

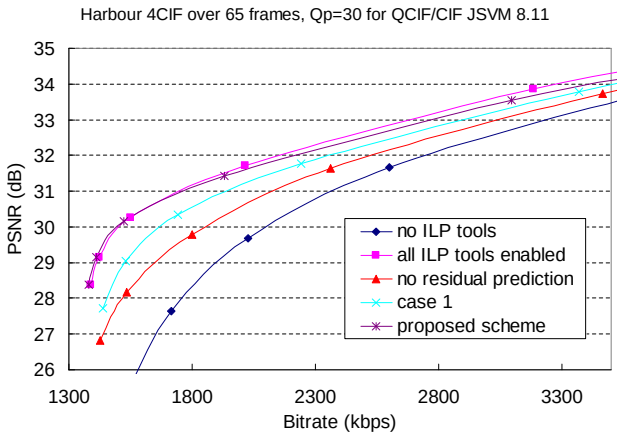
2.1. Residual Prediction in SVC

As described in [1, 3], inter-layer prediction can be regarded as three aspects, motion vector prediction, intra block prediction and residual prediction. Inter-layer residual prediction uses the upsampled BL residual information to reduce the information of EL residual. To provide best coding performance, the above inter-layer prediction tools are arbitrarily selected after comparing to the rate-distortion results of normal prediction. While inter-layer intra prediction only increases one search candidate in intra prediction, the residual prediction in JSVM performs during whole IME and FME stages to find a best matched candidate as shown in Fig. 3. It greatly increases the computation complexity of SVC encoder since the computation of IME and FME are nearly doubled. Therefore, in this paper, we focus on the cost-efficient VLSI architecture of inter-layer residual prediction.

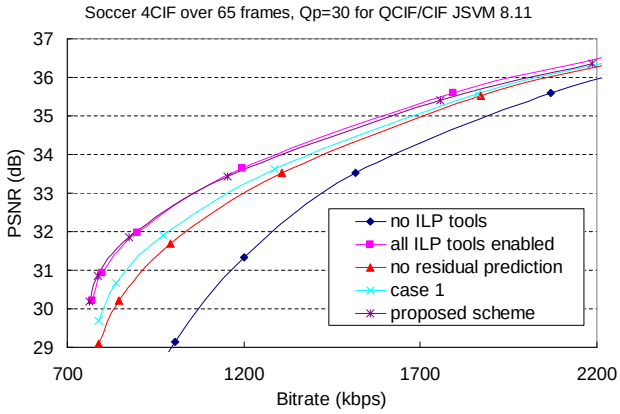
2.2. Design Challenge of Residual Prediction

In many previous H.264/AVC hardware designs[6, 7, 8], the hardware designs of IME and FME are well-investigated. However, none of these designs consider residual prediction. To achieve both normal and residual prediction as shown in Fig. 3 by above designs, the computation cycles will be doubled since they can not parallel processing normal prediction and normal prediction. It nearly doubles the required operating frequency of reference designs while supporting inter-layer prediction in SVC.

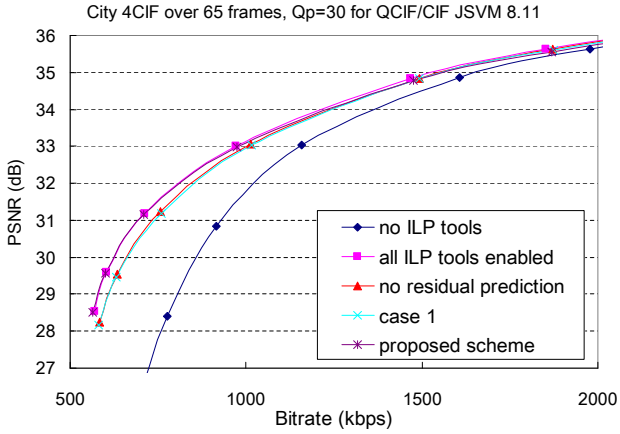
Because the processing cycles for each MB becomes more and more critical in encoding HDTV resolution and parallel processing may achieve higher data reuse to save power, parallel processing for both normal and residual prediction schemes is a better design strategy. However, to maintain the same processing capability, the processing elements (PE) of above designs should be doubled and it leads to a large hardware cost overhead from residual prediction. For example, the hardware cost of sum-of-absolute-difference (SAD) tree in IME [6, 7] will be doubled, and the number of 4×4 processing unit (PU) and interpolation unit adopted in FME [4, 8] will be doubled, too. How to efficiently reduce the hardware overhead of residual prediction becomes an necessity for future SVC designs.



(a)Harbour



(b)Soccer



(c)City

Fig. 2. PSNR comparison of inter-layer prediction(ILP) for the luminance part of (a)“Harbour” (b)“Soccer” (c)“City” 4CIF 30fps over 65 frames with GOP=16 by JSVM 8.11 hierarchical B-frame coding; ILP : inter-layer prediction; “case 1” is to check residual prediction only on normal prediction’s results

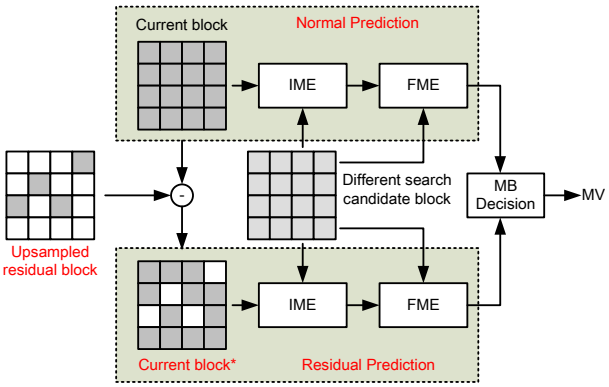


Fig. 3. Computation procedure of inter-layer residual prediction in JSVM encoder.

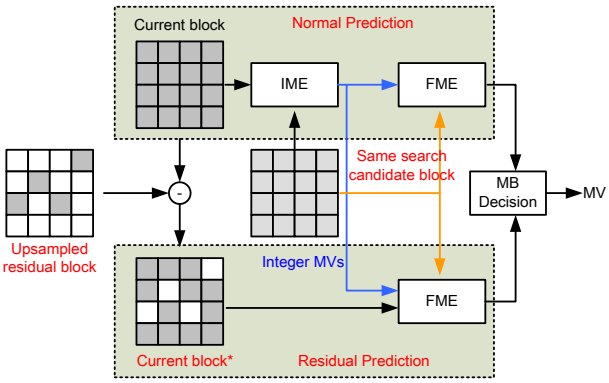


Fig. 4. Computation procedure of Proposed IME-simplified inter-layer residual prediction.

2.3. Simplified Residual Prediction Scheme

Figure 2 demonstrates the coding performance comparison of inter-layer prediction under different residual prediction constraints for sequence Harbour, Soccer and City at 4CIF 30fps, GOP = 16 by 4-level hierarchical B-frame decomposition. FGS is not involved into our analysis and QPs for QCIF/CIF are set to 30. When all inter-layer prediction tools are enabled, it has best coding performance but the hardware cost are doubled compared to normal prediction only. "Case 1" represents that the residual prediction is only performed on the normal prediction results. Although "case 1" only increase few hardware processing cycles on conventional H.264 designs, about 0.4 to 1 dB degradation is observed from simulation results in Fig. 2.

Based on the inference in [3] that residual prediction is more likely to improve coding performance while the motion vector (MV) for the block of current layer are similar to the MV of the corresponding base layer block, we assume that the normal prediction of IME can effectively find such motion trend and residual prediction is more efficient during fractional motion refinement. Thus we propose an IME-simplified residual prediction scheme. The IME of residual prediction in Fig. 3 is removed, and the integer MV from normal prediction are input to FME for residual prediction refinement. By the adopted IME-simplified scheme, the coding performance can be the same as "all ILP tools enable" at low and medium bitrate and at most 0.2 dB degradation at higher bitrate. Fig. 4 demonstrates our proposed residual prediction scheme. Since the same integer MVs are refined for both predictions, the loaded search candidates data are the same for both predictions' FME module. Thus, we will focus on integrating residual prediction into FME architecture in following sections.

3. LINEARITY OF HADAMARD TRANSFORM

In most FME algorithm and hardware design, sum-of-absolute-transformed-difference (SATD) by Hadamard transform is used as cost evaluation value for rate-distortion optimization to provide better coding performance, and this Hadamard transform module possesses about 40 to 50% FME cost [4, 8]. Since Hadamard transform is a linear transformation, it has following linear properties,

$$H(A+B) = HA + HB, \tag{1}$$

where H represents the 2-D Hadamard transform, and both A and B are square matrixes. Based on above linearity, it is possible to implement data reuse between normal prediction and residual prediction. In general, the SATD of each 4×4 block of normal prediction and residual prediction (RP) can be modeled as (2) and (3), respectively.

$$\begin{aligned} SATD_{n,4\times4} &= \sum |H(D_{n,4\times4})| \\ &= \sum |H(cur_{4\times4} - ref_{n,4\times4})|, \end{aligned} \tag{2}$$

$$\begin{aligned} SATD_{n,4\times4,RP} &= \sum |H(D_{n,4\times4,RP})| \\ &= \sum |H(cur_{4\times4} - res_{4\times4} - ref_{n,4\times4})|, \end{aligned} \tag{3}$$

where RP represents the data for residual prediction, n is the index of current search candidate, $cur_{4\times4}$, $ref_{n,4\times4}$, $res_{4\times4}$ are the current block, interpolated reference block and upsampled residual block, respectively. Based on the linearity of Hadamard transform, it is possible to substitute the term of (3) by (2) as follows,

$$\begin{aligned} H(D_{n,4\times4,RP}) &= H(cur_{4\times4} - res_{4\times4} - ref_{n,4\times4}) \\ &= H(cur_{4\times4} - ref_{n,4\times4}) - H(res_{4\times4}) \\ &= H(D_{n,4\times4}) - H(res_{4\times4}). \end{aligned} \tag{4}$$

Thus, the Hadamard transformed coefficients in (2) can now be reused to calculate cost of residual prediction without performing the Hadamard transform in (3) again. Since each

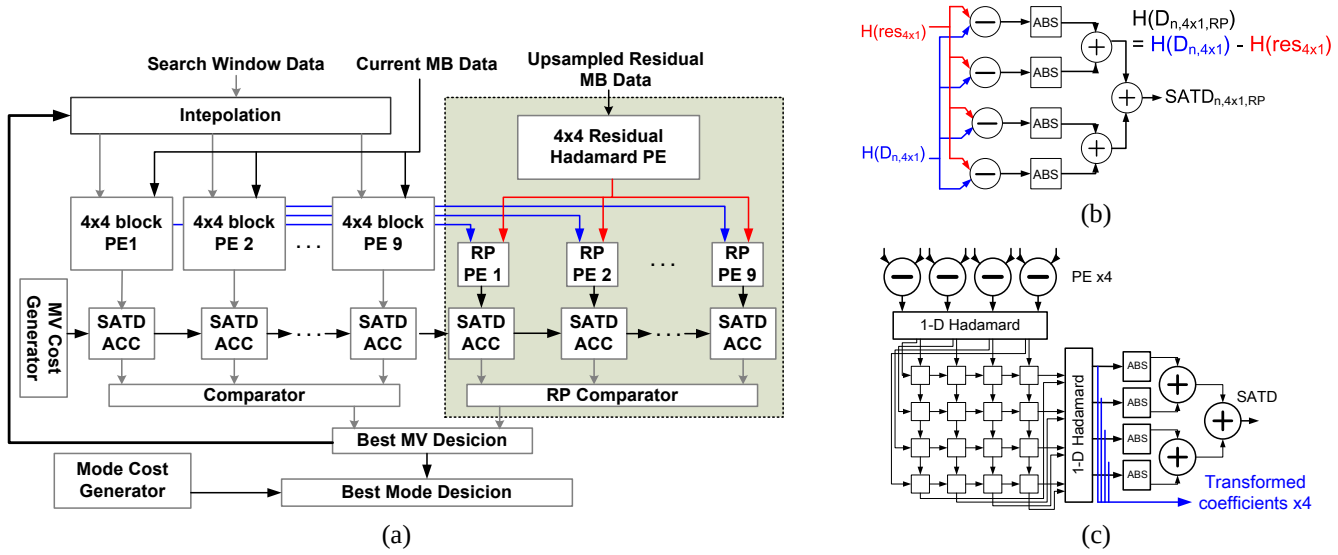


Fig. 5. (a)Block diagram of proposed residual prediction FME hardware (b) architecture of residual prediction (RP) PE (c) architecture of 4×4 PE
. ABS in above figures means absolute value.

search candidate applies the same upsampled residual block for residual prediction, the transformed coefficients of 4×4 residual block, $H(res_{4\times4})$, can be shared by every search candidate simultaneously.

4. PROPOSED RESIDUAL PREDICTION FME VLSI ARCHITECTURE

In this paper, we take the FME architecture in [4] as design example. Please note that the concept of proposed Hadamard-free architecture can be also applied in other existing FME design [8]. The proposed FME architecture with residual prediction is shown in Fig. 5. The region surrounded by dashed line is specified for residual prediction. The blue arrows and red arrows represent the data flow of transformed coefficients of differences and upsampled residual data, respectively. Originally, the $SATD_{RP}$ of each search candidate requires one 4×4 PE as shown in Fig. 5(c).

Based on (4), the Hadamard-transformed coefficients of difference are reused to derive $SATD_{RP}$, and the transformation for residual prediction (RP) can be eliminated. Thus, in our proposed RP PE in Fig. 5(b), the two 1-D Hadamard modules and 4×4 transposed registers are removed, and it can largely reduce the hardware cost compared to direct implementation. Since the transformed coefficients of residual block are required in (4), one 4×4 Hadamard transform PE in Fig. 5(c) is added. Besides, since the integer motion vectors for normal prediction and residual prediction are the same, the output of interpolation unit can be shared. It also save cost of one additional interpolation unit.

Table 1. Implementation results of proposed FME architecture with residual prediction function based on [4]

	TSMC 0.18um @100MHz		UMC 90nm @220MHz	
	[4]	Proposed	[4]	Proposed
Interpolation Unit	25501	25501	25593	25593
MVCost Core	5031	5030	5438	5424
4×4 PE $\times 9$	34863	34865	33391	33398
RP PE $\times 9$	0	8484	0	8891
Res. Transform PE	0	3076	0	2870
Others	5465	9399	5670	9663
Total	70860	86355	70092	85839

unit : NAND2 gate count

5. IMPLEMENTATION RESULTS

Based on our previous FME design [4] for H.264/AVC baseline profile, an architecture of FME with Hadamard-free residual prediction has been proposed and synthesized by UMC 90nm technology and TSMC 0.18um technology, respectively. Since our proposed architecture can parallel process both normal prediction and residual prediction, the processing capability can be the same as the referenced design. According to [4], each macroblock requires about 2000 cycles. Our proposed architecture can support 4CIF 30fps in 100MHz and 720p 30fps in 220MHz. The gate count distributions of two technologies are listed in Table 1. Based on Table 1, the proposed architecture can achieve both normal prediction and

residual prediction with only $(85839-70092)/70092 \cong 22\%$ hardware cost compared to original FME [4]. Please note that [4] can not process normal prediction and residual prediction simultaneously. Comparing to directly duplicating FME [4] for parallel processing normal and residual prediction, the proposed architecture can save the cost of 4×4 PE for residual prediction about 21000 gate counts and reuse the interpolation unit. Based on synthesized results in Table 1, $(70092 \times 2 - 85839)/(70092 \times 2) \cong 39\%$ hardware cost can be saved.

6. CONCLUSION

In this paper, we propose a cost efficient residual prediction architecture for H.264/AVC scalable extension. The IME-simplified scheme is adopted with less than 0.2 dB quality loss, and the linearity of Hadamard transform is applied to reduce the FME hardware cost while integrating residual prediction. Compared to directly duplicated implementation, it can reduce about 40% hardware cost. Moreover, the proposed design concept can be applied in other FME designs and software acceleration to support inter-layer prediction.

7. REFERENCES

[1] ISO/IEC JTC1, "Joint Draft 8 of SVC Amendment," ISO/IEC JTC1/SC29/WG11 and ITU-T SG16 Q.6, Doc. JVT-U201, Oct. 2006.

[2] ISO/IEC JTC1, "Joint Scalable Video Model 8.0," ISO/IEC JTC1/SC29/WG11 and ITU-T SG16 Q.6, Doc. JVT-U202, Oct. 2006.

[3] H. Schwarz, D. Marpe, and T. Wiegand, "SVC core experiment 2.1: Inter-layer prediction of motion and residual data," ISO/IEC JTC1/SC29/WG11/M11043, July 2004.

[4] T.-C Chen, Y.-W. Huang, and L.-G. Chen, "Fully utilized and reusable architecture for fractional motion estimation of H.264/AVC," in *Proceedings of IEEE ICASSP*, May 2004, pp. V–9–V–12.

[5] Y.-W. Huang and et al., "A 1.3tops H.264/AVC single-chip encoder for HDTV applications," in *Proc. of IEEE ISSCC*, 2005, pp. 128–588.

[6] T.-C. Chen and et al., "Analysis and architecture design of an HDTV720p 30 frames/s H.264/AVC encoder," *IEEE Trans. on CSVT*, vol. 16, no. 6, pp. 673–688, June 2006.

[7] Z. Liu, Y. Song, T. Ikenaga, and S. Goto, "A pipeline parallel tree architecture for full search variable block size motion estimation in H.264/AVC," in *Proc. of Picture Coding Symposium*, 2006.

[8] Y.-J. Wang, C.-C. Cheng, and T.-S. Chang, "A fast algorithm and its VLSI architecture for fractional motion estimation for H.264/MPEG-4 AVC video coding," *IEEE Trans. on CSVT*, vol. 17, no. 5, pp. 578–583, May 2007.