HIGH-THROUGHPUT AND MEMORY-AWARE HARDWARE OF A SUB-PIXEL INTERPOLATOR FOR MULTIPLE VIDEO CODING STANDARDS

Guilherme Paim, Jones Goebel, Wagner Penny, Bruno Zatt, Marcelo Porto, Luciano Agostini

Federal University of Pelotas (UFPel) - PPGC - Group of Architecture and Integrated Circuits (GACI)

ABSTRACT

Real-time operation and low-power dissipation in video coding systems have become important research challenges, especially in mobile devices with limited battery and computational resources. There are many video coding standards coexisting in the market nowadays, so it is important for current devices to support different video coding standards. This paper presents a multi-standard luminance sub-samples interpolator hardware design for the Motion Compensation (MC) and Fractional Motion Estimation (FME), with support to MPEG-2/4, H.264/AVC, HEVC, and AVS/2 video coding standards. Our design is able to save hardware resources through an optimized filter organization, totally compliant with the focused standards and capable to interpolate samples for UHD 4320p@60fps at real time. The 45nm standard-cell library implementation dissipates 10mW, when processing according MPEG-2 standard, up to 46.4mW when processing AVS2.

Index Terms— Multi-Standard Interpolator, MPEG-2/4, H.264/AVC, AVS/2, HEVC.

1. INTRODUCTION

Nowadays, the reproduction of digital videos has become an important challenge when developing real-time multi-standard codec architectures. Due the large number of video coding standards in the market, all devices that handle digital videos should be able to support different coding standards. Even so, several structures available in different standards are similar, allowing the design of multi-standard optimized architectures.

Several video coding techniques/algorithms were developed in the last years; each one introduces its own improvement to the video coding efficiency. These optimized solutions are combined leading to a new standard. In order to outperform the market-dominant video coding standard, the H.264/AVC (Advanced Video Coding) [1], huge efforts on research and standardization led to the specification of the High Efficiency Video Coding (HEVC) [2]. HEVC is the state-of-art video coding standard, reaching a reduction around 50% in bit rate, compared to the H.264/AVC [2], whilst keeping the same objective quality and improving subjective quality [3]. Besides, there are other video coding standards, such as the Chinese standards AVS2 (Advanced Video coding System) and its predecessor, the AVS (Audio Video coding Standard) [4], which also have a large market share. Furthermore, older standards such as MPEG-2 and MPEG-4 [5] are also important, since they are still in use in some applications.

The Fractional Motion Estimation (FME) and Motion Compensation (MC) are common tools of all these standards. The main process of MC and FME is the interpolation, present in both, coder and decoder. At the decoder side, the interpolation is responsible for about 25% of total computational effort [6]. The high complexity of the interpolation reinforces the need for dedicated hardware accelerators. A generic and multi-standard hardware for interpolation can reduce the time-to-market and the NRE (Non-Recurring Engineering) costs of design. It may be employed at both, single-standard and multiple-standard video coding systems.

This work presents the design of an ASIC-based multi-standard hardware architecture, targeting the MC and FME luminance interpolator for the MPEG-2, MPEG-4, H.264/AVC, HEVC, AVS, and AVS2 standards, through a high-throughput hardware design which is able to reach real-time processing for Ultra High Definition (UHD) videos in an energy-efficient implementation.

2. BACKGROUND AND RELATED WORKS

The FME is one of the features introduced to improve the coding efficiency in the current video encoders. FME allows the motion estimation to find better matching in fractional samples, improving the coding efficiency. The FME performs a search and comparison over the interpolated fractional samples, generating the Motion Vectors (MVs). The MV may point to an integer or to a fractional position with quarter-pixel precision for MPEG-4, HEVC, H.264, AVS, and AVS2, and half-pixel precision for MPEG-2. The MC tool is presented in both coder and decoder of all mentioned video coding standards and is responsible for the frame reconstruction, based on information from the MVs. The FME and MC require an interpolator to generate the fractional samples, by the use of FIR and/or bilinear filters with different coefficients for each standard, as summarized in Table I.

Some works in the literature present multi-standard hardware designs for the interpolation; these related works are presented in Table I. The works [5]-[8] focus on solutions for multi-standard motion compensation interpolators. Lee [5] proposes a reconfigurable MC architecture for three video coding standards, based on the top-down design methodology. Zhou [6] designs a hardware-efficient dual-standard VLSI architecture. Zheng [7] and Chen [8] also propose high performance MC architectures, targeting three

Table I. Filter coefficients for each evaluated standard and multi-standard related works support.

Standard	Coefficients		Supported Coding Standards						
	Half-pel	Quarter-pel	Lee [5]	Zhou [6]	Zheng [7]	Chen [8]	Lu [9]	Maich [10]	This work
MPEG-2	Bilinear	-	**X**	**X**		**X**	**X**		**X**
MPEG-4	{-1,3,-6,20,20,-6,3-1}	Bilinear	**X**				**X**		**X**
H.264/AVC	{1,-5,20,20,-5,1}	Bilinear	**X**	**X**	**X**	**X**	**X**	**X**	**X**
AVS	{-1,5,5,-1}	{1,7,7,1} and Bilinear		**X**	**X**	**X**	**X**		**X**
AVS2	{-1,4,-11,40,40,-11,4,-1}	{-1,4,-10,57,19,-7,3,-1}							**X**
HEVC	{-1,4,-11,40,40,-11,4,-1}	{-1,4,-10,58,17,-7,3,-1}						**X**	**X**

standards. Lu [9] proposes a hardware design for a multi-standard motion estimation targeting MPEG-2, MPEG-4, H.264/AVC, and AVS. Our previous work [10] presents an interpolation unit targeting MC/FME for HEVC and H.264/AVC. However, the current work presents the unique interpolator capable to provide support for six different standards (Table I), in MC or FME, at both encoder and decoder sides. Our solution targets high performance with power efficiency whereas reducing memory communication overhead through ideal basic processing unit selection.

3. MULTI-STANDARD INTERPOLATOR ARCHITECTURE

The MC must support all prediction modes and block sizes defined by each standard, since any mode/block size may be chosen as best prediction during the encoding process [2]. This requirement represents an important challenge when considering memory access pattern, control logic, and parallelism level. Thus, to deal with this challenge it is possible to decompose all prediction block sizes in smaller blocks, here called Basic Processing Units (BPU), leading to a more regular processing and memory access.

The designed multi-standard hardware architecture for luminance sample interpolation uses eight horizontal filters and eight vertical filters. This work introduces a novel hardware design for the interpolation filters, called **H**EVC/H.264 - **A**VS/2 - **M**PEG-2/4 (HAM) Filter. The subsection 3.1 discusses the choice of the best BPU size; subsection 3.2 explains the HAM Filter design whereas subsection 3.3 details the interpolation architecture.

3.1. Basic Processing Unit Sizing

The BPU is a processing block defined to have the ideal parallelism level for the designed architecture. Any BPU size could be chosen, but each size leads to a distinct memory and processing overhead. The overhead happens when the dimension of block to be processed are not multiple of the BPU dimensions; e.g., 4x8 block and 8x8 BPU. Also, smaller BPUs lead to higher memory communication due block borders multiple fetching. Thus, memory and processing overhead evaluation was performed, for all the standards, to find the ideal BPU size. As presented in [11], memory communication (amount of data required from the memory) and samples to process (amount of samples to be interpolated, including the overhead) should be considered to define the best BPU size. Moreover, we also consider a frequency increment factor, which represents the increase to the original frequency required to compensate for unnecessarily processed (discarded) samples.

Using 8x8 BPUs, the input matrix has 225 samples (15x15); integer samples to perform the interpolation process. The 8x8 BPU reduces by a factor of 52.6%, 47.6%, and 33.8% the memory communication (respectively for HEVC, H.264/AVC, and AVS2) in comparison with 4x4 BPU size, used by all multi-standard related works, without using local buffers, as adopted by [5]-[10]. Thereby, the overhead happens only on H.264/AVC, HEVC, and AVS2 standards, since they have support for coding blocks non-divisible or smaller than 8x8. The theoretical worst case overhead for 8x8 BPUs happens when only 4x4 blocks are processed and demands four times processing increase, i.e., 3/4 of the processed samples are discarded. However, this is not a real-world scenario even for very low-resolution videos. In order to quantify the overhead, we have counted the number of blocks non-divisible or smaller than 8x8 at the decoder. The choice of an 8x8 BPU results on overheads of 1.01, 1.11, and 1.41 times in the amount of samples to process, increasing the frequency architecture by same factors, for HEVC, H.264/AVC, and AVS2, respectively. All experiments were evaluated with the HEVC Test Model (HM) 13.0, the H.264/AVC Joint Model (JM) 19.0, and the AVS2 Reference Model (RM) 9.1 with all video sequences and Quantization Parameters (QPs) recommended in Common Test Conditions (CTCs) [12].

3.2. HAM Filter Design

Each standard defines different filters for the interpolation process. They may have distinct number of taps as well as different coefficients. However, it is possible to find many similarities among the filters of each standard, allowing the development of a unified filter for all standards, the HAM Filter. Fig. 1 shows the architectural design of the proposed HAM Filter architecture.

The HAM Filter implement additions of input samples multiplied by the standard-defined coefficients. The multiplications were performed with selective shift-adds operations (highlighted in gray rectangles in Fig. 1). These shift-adds operators are shared when the same coefficient is used in different conditions. Also, similar coefficients allow sharing partial subexpressions without any losses. A multiplexer is necessary to define the correct coefficient based on the desired standard.

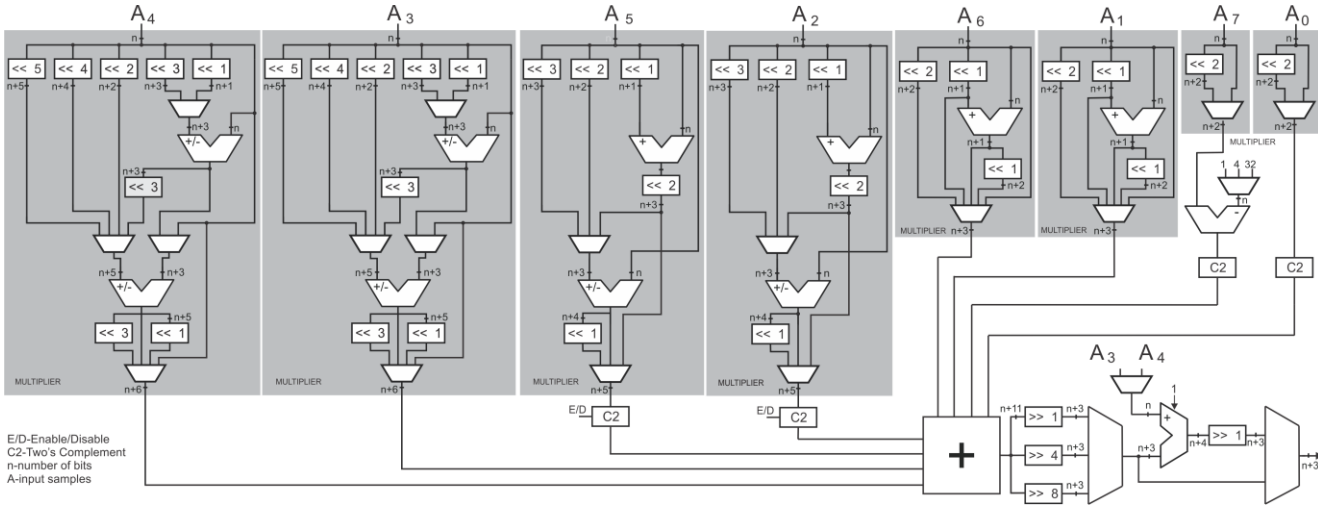

Fig. 1. *HAM Filter* Architecture.

Eight integer samples are fed per cycle as the inputs of the HAM filter. This solution implements a parallel calculation of one fractional sample per cycle (i.e., receives eight integer samples and delivers one fractional sample) in order to attend the performance requirements. Since there is a strong symmetry between the coefficients [2] [4] [9], it is possible an equal design for inputs from 0-3 and 4-7.

Take samples A1 and A6 as examples (see Fig. 1), they can be multiplied by -1, 0, 1, 3 or 4 [2] [4] [9], where 0 denotes the case where the standard does not evaluate that sample. Note the desired behavior can be obtained by using shifts and additions. A control signal defines, in the multiplexer, the correct weight for a certain standard. The same behavior can be observed in the other modules in Fig. 1. Through this strategy the designed HAM Filter is able to interpolate a fractional (half or quarter) sample for any of the focused standards.

3.3. Interpolator architecture

Fig. 2 shows the datapath of the proposed multi-standard luminance interpolator architecture. This new architecture implements line-level parallelism and explores pipelining and hardware reuse, leading to optimized parallelism and performance. Each sample is represented as a black square on the Fig. 2; they feed horizontal (H) filters, which are the first step of the interpolation process and, after that, the samples feed vertical (V) filters.

Inside horizontal and vertical filters there is one HAM Filter and a multiplexer which selects the datapath depending on MV information, which can be integer or fractional, as presented in Fig. 2. A control unit (omitted in the Fig. 2) is responsible to generate the control signals for the multiplexers, based on the information of the used standard.

At each clock cycle, 15 reference samples are read, corresponding to each line of the reference matrix. These samples are the inputs of horizontal filters, which have a set of eight input signals (see Fig.2). Each one of these horizontal filters has a bypass path, used when the motion vector in x axis (MV_X) points to an integer position, and a path with a HAM Filter, used when the MV_X points to a fractional position (a multiplexer selects the correct output based on MV_X information). The horizontal filter output is connected to eight shift registers and, after all registers are filled; all positions of the shift register are used as input of another filters, called vertical filters. These vertical filters have also the same two datapaths for bypass and HAM Filter, controlled by the vector in y axis (MV_Y). The outputs may have integer or fractional samples, with precision of half or quarter pixel. In the worst case, the system latency is eight clock cycles (from the shift registers), after that, 64 samples (one 8x8 BPU) are delivered at every 15 clock cycles. Thus, considering the 8x8 BPU block, the designed interpolator gives a throughput of 4.26 samples per cycle.

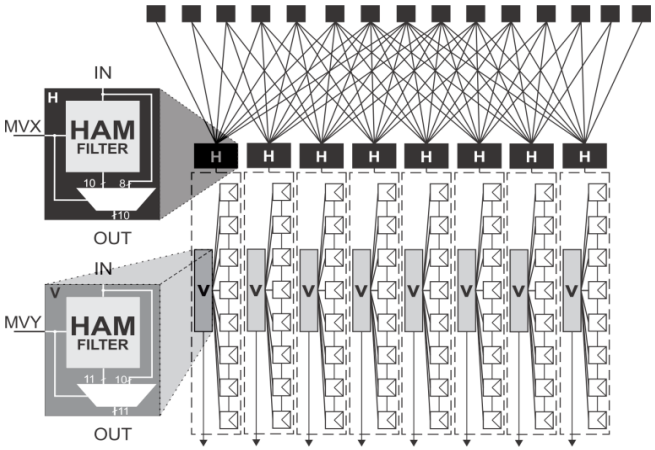

Fig. 2. Interpolator hardware architecture.

4. SYNTHESIS RESULTS AND COMPARISONS

The proposed architecture is fully compatible with MC/FME modules in all six standards. The interpolation coefficients of the standards were compared and their similarities were found and explored. As result, it was possible to develop an optimized filter architecture compatible with multiple standards. The proposed architecture was described in Verilog and synthesized using a 45nm@1.1V Nangate

and 180nm@1.98V X-FAB standard cell libraries with the Cadence Encounter RTL Compiler tool.

The operational frequencies (Table II) were defined as the minimum required to achieve real-time processing on different video resolutions. As discussed, the BPU 8x8 reduces the memory bandwidth whereas introduces an overhead of samples to process. This overhead increases the frequency of the architecture in 1.01, 1.11, and 1.41, for the HEVC, H.264, and AVS2, respectively.

The designed multi-standard interpolator was evaluated, considering area and power dissipation results, focusing on the performance from Full HD 1080p@30fps (1920x1080 pixels) up to UHD 4320p@60fps (8K) resolution (7680x4320 pixels). Table II presents the total power dissipation (leakage + dynamic) results for the 45nm and 180nm technology synthesis, considering each supported standard in the respective target and target performance/frequency. The power dissipation is estimated with a power supply of 1.25V, considering the specified target frequencies and with 50% of switching activity on inputs (same for Table III).

Table II. Power dissipation results with 50% input switching.

Standard	Resolution	Freq. (MHz)	Total Power (mW)	
			45nm	180nm
HEVC	1080@30fps	14.7	2.69	17.47
	2160@30fps	58.9	6.45	51.19
	4320@60fps	471.2	25.5	-
H.264	1080@30fps	16	2.31	13.82
	2160@30fps	64.2	5.24	40.4
	4320@60fps	513.2	20.1	-
AVS2	1080@30fps	20.4	3.07	20.69
	2160@30fps	81.7	7.69	67.3
	4320@60fps	653.2	41.5	-
AVS	1080@30fps	14.6	2.18	12.96
	2160@30fps	58.3	4.54	33.97
	4320@60fps	466.6	17.92	-
MPEG-4	1080@30fps	14.6	2.59	16.47
	2160@30fps	58.3	6.07	47.92
	4320@60fps	466.6	24.69	-
MPEG-2	1080@30fps	14.6	1.09	3.85
	2160@30fps	58.3	2.15	14.05
	4320@60fps	466.6	10.0	-

The power dissipation for high definition resolutions is similar for all video coding standards. However, as the resolution and throughput grow higher, the differences between the power dissipation for each standard increase. The highest power dissipation is observed for AVS2 standard, as expected, due its higher frequency overhead related to the use of 8x8 BPU size.

Table III presents the synthesis results of the developed multi-standard interpolator and a comparison between six related works. The gate count is calculated based on 2-input NANDs. It is possible to observe that the designed architecture is the unique capable to work with six video coding standards as well as to attend performance to process UHD 8K resolutions at 60 fps, reaching real time for this resolution. Comparing the cost in area, our design is larger than the other works, especially for 45nm synthesis. However, considering the number of supported standards and the performance, the increase in area is acceptable – remember that our multi-standard solution replaces six standard-specific interpolators. The solution proposed in this work dissipates half of power (in H.264/HEVC), reaches twice the performance and requires one third of area when compared with our previous work [10].

5. CONCLUSIONS

This paper presented a memory-aware and multi-standard interpolator hardware design for the luminance samples, used in MC and FME, for the MPEG-2/4, H.264/AVC, AVS/2, and HEVC standards. Many strategies, as the best choice of the basic processing unit block, efficient pipelining and hardware reuse, are used to reduce power and increase throughput. The HAM Filter was proposed to handle all the six standards.

The developed multi-standard interpolator architecture was synthesized using a 45nm Nangate and 180nm X-FAB standard cell libraries and the results were compared with related works. This work was the unique able to support six video coding standards and reached the highest processing rate among the related works. The developed architecture is able to process 4320p@60fps with a power dissipation ranging from 10 mW, in MPEG-2 mode (best case), up to 41.5 mW in AVS2 mode (worst case).

Table III. Synthesis results and comparison with related works.

Related Work	Lee [5]	Zhou [6]	Zheng [7]	Chen [8]	Lu [9]	Maich [10]	Developed	
Standards	MPEG-2 / MPEG-4 / H.264	H.264 / AVS	MPEG-2 / H.264 / AVS	MPEG-2 / H.264 / AVS	MPEG-2 / MPEG-4 / H.264 / AVS	HEVC / H.264	MPEG-2 / MPEG-4 / H.264 / HEVC / AVS / AVS2	
Technology	180nm	180nm	180nm	130nm	130nm	65nm	180nm	45nm
Frequency (MHz)	108	142.2	148.5	160	270	482.11	81.7	471.2
Gate Count (K)	29.6	14.4*	21.6	25.5	34.8*	166.8*	19.3*	46.4*
Max. Power (mW) (50% input switch.)	-	-	-	-	-	80.69	67.3	41.5
Max. Throughput	1080p @30fps	1080p @60fps	1080p @60fps	1080p @60fps	1080p @60fps	4320p @30fps	2160p @30fps	4320p @60fps

*Only luminance interpolator.

6. REFERENCES

[1] Sullivan, G., Marpe, D., Wiegand, T. “The H.264/MPEG4 Advanced Video Coding Standard and its Applications”. IEEE Communications Magazine, vol.44, 2006, pp. 134-143.

[2] Sullivan, G., et al. High Efficiency Video Coding. Springer, 2014.

[3] Grois, D., et. al. "Performance comparison of H.265/MPEG-HEVC, VP9, and H.264/MPEG-AVC encoders". PCS, 2013, pp. 394-397.

[4] Gao, W., Ma, S..Advanced Video Coding Systems. Springer, 2014.

[5] Lee, G., Yang, W., Wu, M., Lin, H. “Reconfigurable Architecture Design of Motion Compensation for Multi-Standard Video Coding”. ISCAS, 2010, pp. 2003-2006.

[6] Zhou, D., Liu, P.. “A Hardware-Efficient Dual-Standard VLSI Architecture for MC Interpolation in AVS and H.264”. ISCAS, 2007, pp.- 2910-2913.

[7] Zheng, J., Gao, W., Wu, D., Xie, D..”A Novel VLSI Architecture of Motion Compensation for Multiple Standards”. IEEE Transactions on Consumer Electronics, vol, 54, no. 2, 2008, pp. 687-694.

[8] Chen, X., et. al.”A High Performance and Low Bandwidth Multi-Standard Motion Compensation Design fo HD Video Decoder”. IEICE Trans. Electron., vol. E93-C, no. 3, 2010, pp. 253-260.

[9] Lu, L. et al. “Subpixel Interpolation Architecture for Multistandard Video Motion Estimation”. TCSVT, 2009, pp. 1897-1901.

[10] Maich, H.; Paim, G.; Afonso, V.; Agostini, L.; Zatt, B.; Porto, M., "A multi-standard interpolation hardware solution for H.264 and HEVC," IEEE International Conference on Image Processing (ICIP), 2015 , vol., no., pp.2910-2914, 27-30 Sept. 2015.

[11] Penny, W.; Paim, G.; Porto, M.; Agostini, L; Zatt, B. “Real-Time Architecture for HEVC Motion Compensation Sample Interpolator for UHD Videos”. in Proceedings of the 28th Symposium on Integrated Circuits and Systems Design (SBCCI), 2015.

[12] F. Bossen, Common Test Conditions and Software Reference Configurations (JCTVC-L1100), 2013.