High-Throughput Power-Efficient VLSI Architecture of Fractional Motion Estimation for Ultra-HD HEVC Video Encoding

VANAM BABU RAO[1], M. SUNEETHA[2]

[1]PG Scholar, Dept of ECE, Gokaraju Rangaraju Institute of Engineering and Technology, Bachupally, Hyderabad, India.
[2]Assistant Professor, Dept of ECE, Gokaraju Rangaraju Institute of Engineering and Technology, Bachupally, Hyderabad, India.

Abstract: The next-generation video coding standard of High-Efficiency Video Coding (HEVC) is especially efficient for coding high-resolution video such as 8K-ultra-high-definition (UHD) video. Fractional motion estimation in HEVC presents a significant challenge in clock latency and area cost as it consumes more than 40 % of the total encoding time and thus results in high computational complexity. With aims at supporting 8K-UHD video applications, an efficient interpolation filter VLSI architecture for HEVC is proposed in this paper. Firstly, a new interpolation filter algorithm based on the 8-pixel interpolation unit is proposed in this paper. It can save 19.7 % processing time on average with acceptable coding quality degradation. Based on the proposed algorithm, an efficient interpolation filter VLSI architecture, composed of a reused data path of interpolation, an efficient memory organization, and a reconfigurable pipeline interpolation filter engine, is presented to reduce the implement hardware area and achieve high throughput. The final VLSI implementation only requires 37.2k gates in a standard 90-nm CMOS technology at an operating frequency of 240 MHz. The proposed architecture can be reused for either half-pixel interpolation or quarter-pixel interpolation, which can reduce the area cost for about 131,040 bits RAM. The processing latency of our proposed VLSI architecture can support the real-time processing of 4:2:0 format 7680×4320@78fps video sequences.

Keywords: HEVC, AVC, Bitrates, Streaming Video, Prediction, Macro-Block And Quadtree.

I. INTRODUCTION

High Efficiency Video Coding (HEVC) is a video compression standard, a successor to H.264/MPEG-4 AVC (Advanced Video Coding), that was jointly developed by the ISO/IEC Moving Picture Experts Group (MPEG) and ITU-T Video Coding Experts Group (VCEG). (Sullivan, et al, 2012). The objective of this technique is to obtain high levels of coding efficiency, i.e. higher data compression while maintaining a threshold limit of video quality compensation. The coding efficiency relationship between two designs is typically best expressed in terms of percentage savings in bit rate for equal subjective perceptual quality. In addition to enabling service providers to deliver more content at a given quality (e.g., more television channels sent over the same data link or more video stored on the same storage medium), improved coding efficiency can alternatively be used to provide higher quality video (e.g., higher resolution or less distorted video) at a given bit rate, or to provide some other improved balance between bit rate and video quality. An analysis of the IP traffic by Cisco Visual Networking Index (VNI), India has revealed the following facts:

- An increasing number of devices like mobiles and tablets will increase the demand of connectivity to 18.9 billion by 2016.
- With increasing number of internet users, ubiquitous Wi-Fi growth and faster broadband speeds Global IP traffic is expected to reach 1.3 zettabytes per year by 2016, India expected to have highest IP growth rate.
- Almost 80% of this IP traffic will be consumer video.

This video coding technique almost doubles the data compression of a video without compensating the video quality. Hence, a 5GB Blu-Ray Movie encoded using current video encoding technologies will occupy a size of almost 2.5GB (having the same video quality) when encoded using HEVC. Using more efficient encoding decoding algorithms and exploiting parallel computing, HEVC will provide a standard to the industry for conformities at a global level. Thus, a need to benchmark the performance of HEVC in comparison to its predecessors like H.264 has arisen to verify the qualitative objectives are satisfied or not. Hence this paper deals with the key features of HEVC and compares its video quality and encoding performance through empirical observations. The major video coding standard directly preceding the HEVC project was H.264/MPEG-4 AVC, which was initially developed in the period between 1999 and 2003, and then was extended in several important ways from 2003–2009. H.264/MPEG-4 AVC has been an enabling technology for digital video in almost every area that was not previously covered by H.262/MPEG-2 Video and has substantially displaced the older standard within its existing application domains. It is widely used for many applications, including broadcast of

high definition (HD) TV signals over satellite, cable, and terrestrial transmission systems, video content acquisition and editing systems, camcorders, security applications, Internet and mobile network video, Blu-ray Discs, and real-time conversational applications such as video chat, video conferencing, and telepresence systems. However, an increasing diversity of services, the growing popularity of HD video, and the emergence of beyond HD formats (e.g., 4k×2k or 8k×4k resolution) are creating even stronger needs for coding efficiency superior to H.264/ MPEG-4 AVC's capabilities. The need is even stronger when higher resolution is accompanied by stereo or multi view capture and display. Moreover, the traffic caused by video applications targeting mobile devices and tablets PCs, as well as the transmission needs for video-on-demand services, are imposing severe challenges on today's networks. An increased desire for higher quality and resolutions is also arising in mobile applications.

II. LITERATURE SURVEY

Architecture Design for an H.264/AVC 4kx2k UHD Intra Prediction: In this dissertation, we first analyzed the design challenges of intra prediction, and proposed an efficient architecture based on H.264 for 4kx2k UHD. Due to the high data dependency of intra prediction, both pipelining and parallel processing techniques are limited to be applied. Moreover, it is difficult to get high hardware utilization and throughput because of the long MB-level and block-level reconstruction loops. The MB and block co-reordering is proposed to solve data dependency problem and improve pipeline utilization by almost 50%. The timing constraint of real-time 4096x2160 encoding can be achieved with negligible quality loss. 16x16 and 8x8 prediction engines work parallel for generating coefficients. A reordering interlaced reconstruction is also designed for fully pipelined architecture. Furthermore, PE-reusable 8x8 intra predictor and hybrid SAD & SATD mode decision are proposed to save hardware cost. It takes only 160 cycles to process one MB. Hardware utilization of the prediction and reconstruction modules achieve 90-100%. The design is implemented by 90nm CMOS technology with 113.2k gates and can encode 4096x2160 video sequences at 60 fps with operation frequency of 332MHz. Compared with previous designs of 1080p@30fps (Lin,TCVST'2009, Chuang,PCS'2007), the throughput of proposed design increases for 8 times with less 30% area overhead. For HD1080p 30 fps encoding requirement, this design can reduce 72% of operating frequency compared with Lin's work (TCVST'2009) because of high parallelism and pipeline efficiency.

Architecture Design for an H.264/AVC 8kx4k UHD Intra Prediction: In this dissertation, H.264 8kx4k UHD intra-prediction architecture is also presented. Due to this huge throughput requirement, design challenges such as complexity and data dependency, which currently exist for lower resolutions (e.g. 2160p and 1080p), become even more critical. Moreover, Pipeline latency influences the efficiency of pipelines with serious data dependency. In this work, we first propose an interlaced block reordering scheme together with a preliminary mode decision (PMD) strategy to resolve the data dependency between intra mode decision and reconstruction. In the meantime, hardware cost is reduced by PMD. We also propose a probability-based reconstruction scheme to solve the problem of long pipeline latency. In addition, hardware reuse strategies including a shared fine decision module and processing element-reusable prediction generator, are applied to further optimize the design. As a result, the hardware complexity (the product of hardware area and required operating frequency) is reduced by 77%, and it takes an average of 33 cycles to process an MB. The implementation result demonstrates that our design can support up to the specification of $7680 \times 4320p$ 60 f/s when running at 273 MHz. The 1080p 30 f/s encoding requires less than 9 MHz operating frequency, which is much lower than that used in previous works (Kuo: TVLSI'2011, Lin: TCSVT'2009).

Architecture Design for an HEVC 8kx4k UHD Fractional Motion Estimation: In this dissertation, 995Mpixels/s 0.2nJ/pixel fractional motion estimation architecture in HEVC for 8kx4k UHD is also presented. In this work, the design is co-optimized in algorithm and hardware architecture to reduce the complexity and achieve high throughput. They are characterized as\ follows.

- By using bilinear quarter pixel approximation, we reduce 76% interpolation complexity and save transform operation for quarter candidates.
- A 5T12S search pattern is proposed to achieve a tradeoff between hardware cost and coding quality. 48% hardware cost is reduced with negligible quality loss, compared with conventional 9T25S.
- Exhaustive size-hadamard transform (ES-HAD) is adopted in FME. It avoids unifying all blocks into small transform ones. Furthermore, it determines the best transform size, rather than using the complex RQT.

Besides, data reusing in ES-HAD is exploited and 58% hardware cost is reduced, compared with the straightforward implementation. This design is implemented in 65nm CMOS chip and verified by FPGA based evaluation system. It achieves 995Mpixels/s for 7680x4320 30fps encoding, at least 4.7 times faster than previous designs. Its power dissipation is 198.6mW at 188MHz, with 0.2nJ/pixel power efficiency. Despite high complexity in HEVC, the chip achieves 56% improvement on power efficiency than previous works in H.264(Tsung,ISSCC'2009, Kao, TVLSI' 2010).

III. PROPOSED SYSTEM

The proposed architecture is designed for 4096x2160 @60fps real-time application with reconstruction pixels predicting. Near full-mode intra 8x8 mode and 16x16 mode are chosen and good performance is achieved. In summary, this paper proposes the following techniques: (1) MB/block co-reordering to solve data dependency problem, (2) two prediction engines processing parallel and reordering interlaced reconstruction to enhance the throughput with high hardware utilization, and (3) PE-reusable 8x8 intra

predictor and hybrid SAD & SATD mode decision to save hardware cost.

A. Proposed Architecture

Our design target is very high throughput application such as 4096x2160 and 1080p. Experiment result shows that in this kind of video sequence, if 4x4 mode is removed we can get almost the same performance with reserving 4x4 mode prediction. In the experiment, intra prediction is processed with sum of absolute transformed difference (SATD) mode decisionwhich is more suitable for hardware implementation. Table 2 illustrates the comparison results with BD-bit rate and Fig3 shows the rate-distortion (RD) curves of the two sequences. By comparing these results, we can observe that 4x4 mode usually introduces large bit rate with unobvious PSNR increasing. 8x8 mode prediction in high profile may brings a little more residual in the bit rate scream, but generates much less head information than 4x4 for smooth MBs. Besides that, SATD mode decision cannot choose best mode accurately like rate-distortion optimization (RDO) and influences the comparison results. Without 4x4 prediction, quality loss could be caused for small resolution. While, the main target of this design is the high throughput application, 4x4 mode is removed in this architecture. In H.264/AVC reference software, Hadamard Transform (HT) is used to calculate SATD for mode decision since it has less computation complexity and no scaling effect. However, in the real encoding process, the DCT is adopted for reconstruction loop. In order to estimate bit rate more accurately, 4x4 and 8x8 integer DCT in cost function is used to approximate the effect of transform and quantization in H.264 encoding process. In this architecture which is designed for high throughput application, 8x8 mode and 16x16 mode with DCT-based SATD are implemented to simplify the architecture with good performance.

B. Code Tree Units (CTU)

Also referred to as Largest Coding Unit, CTU is the basic processing unit of HEVC replacing the earlier macroblock units used in AVC. While MacroBlock Units are a fixed 16x16 pixels, CTU's use variable sizes ranging from 64x64, 32x32 to 16x16. These larger block units facilitate in better division of discrete blocks, increasing coding efficiency. The CTU consists of a luma (It is brightness and represented by Y) and Chroma for Blue and Red.
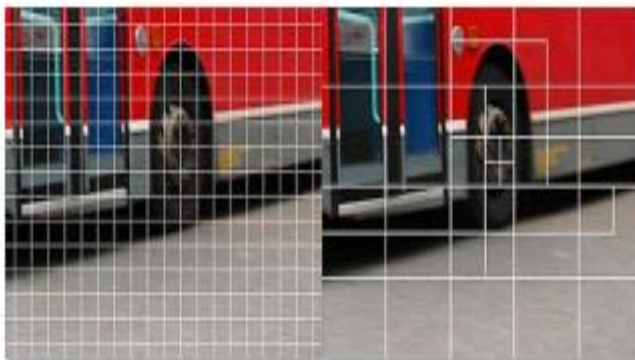
Fig1. Macro blocks in AVC vs. CTU in HEVC.

Chroma is half the size of luma as human eye is more sensitive to brightness. The size L×L of a luma CTB can be chosen as L = 16, 32, or 64 samples, with the larger sizes typically enabling better compression. The splitting of a CTU into luma and chroma CBs is signaled jointly. One luma CB in addition to two chroma CBs, form one Coding unit (CU). Prediction units and prediction blocks (PBs) Interpicture or intrapicture prediction can be made in H.265. This is resolved at the CU level. A PU partitioning structure originates from the CU level. Depending on the basic prediction-type decision, the luma and chroma CBs are also split in size using the luma and chroma prediction blocks. HEVC supports variable PB sizes ranging from 64×64 as well as 4×4 samples.
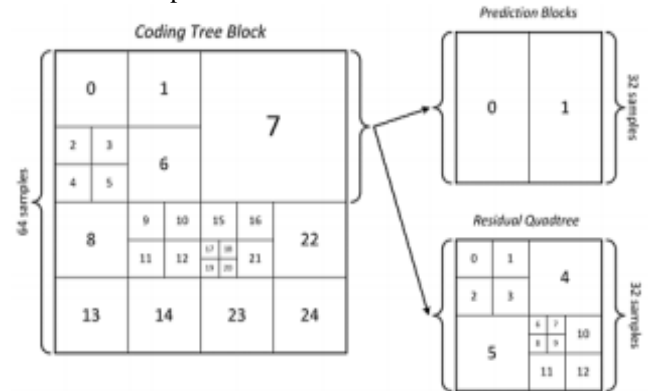
Fig2. Coding Tree Unit, Predictive Block, Residual Tree.

C. TUs and transform blocks

The coding of prediction residual is done using the block transforms created. A TU tree structure has its root at the CU level. The luma CB residual may be identical to the luma transform block (TB) or may be further split into smaller luma TBs. The same applies to the chroma TBs. Integer basis functions similar to those of a discrete cosine transform (DCT) are defined for the square TB sizes 4×4, 8×8, 16×16, and 32×32. For the 4×4 transform of luma intra picture prediction residuals, an integer transform derived from a form of discrete sine transform (DST) is alternatively specified.

D. Motion vector signaling

Advanced motion vector pre- diction (AMVP) is used, including derivation of several most probable candidates based on data from adjacent PBs and the reference picture. A merge mode for MV coding can also be used, allowing the inheritance of MVs from temporally or spatially neighbouring PBs. Improved skipped and direct motion inference make it superior to H.264.

E. Motion compensation

Quarter-sample precision is used for the MVs, and 7-tap or 8-tap filters are used for interpolation of fractional-sample positions (compared to six-tap filtering of half-sample positions followed by linear interpolation for quartersample positions in H.264/MPEG-4 AVC)(Sullivan, et al, 2012). PB's transmitting one motion vectors result in unpredictive coding whereas two motion vectors result in bi-predicitive

coding. Weighted prediction for offset and scaling is applied to the prediction signal.

F. Intra picture prediction

The decoded boundary samples of adjacent blocks are used as reference data for spatial prediction in regions where interpicture prediction is not performed. Intrapicture prediction supports 33 directional modes (compared to eight such modes in H.264/MPEG-4 AVC), plus planar (surface fitting) and DC (flat) prediction modes. The selected intrapicture prediction modes are encoded by deriving most probable modes (e.g., prediction directions) based on those of previously decoded neighbouring PBs.

G. Quantization control

As in H.264/MPEG-4AVC, uniform reconstruction quantization (URQ) is used in HEVC, with quantization scaling matrices supported for the various transform block sizes.

H. Entropy coding

Like H.264, the Context adaptive binary arithmetic coding (CABAC) is used for entophy encoding, but numerous improvements to its throughput speed and its compression performance, reducing memory requirements, making it more conducive for parallel processing architectures.
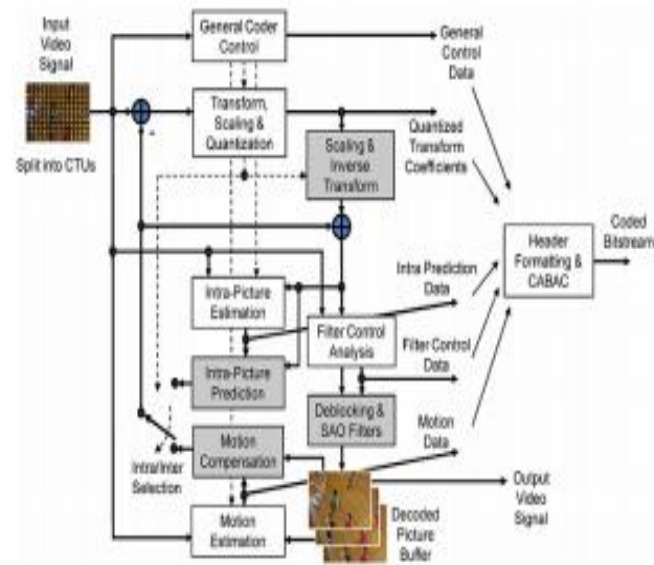

Fig3. HEVC Encoding.

I. In-loop deblocking filtering

Again, like H.264, the same deb locking filter serves within the interpicture prediction loop. However, by simplifying its filtering and decision-making designs, HEVC further exploits the parallel architectures for faster computation.

J. Sample adaptive offset (SAO)

The deb locking filter which uses the interpicture loop prediction passes on the signal to a nonlinear amplitude mapping which uses look up tables to imitate the original signal amplitudes. The look up table is constructed using additional attributes (at the encoder side) obtained from histogram analysis.

K. Fractional Motion Estimation Architecture for 8kx4k UHD

Now, the Joint Collaborative Team on Video Coding (JCT-VC) is developing the next-generation video coding standard, called High-Efficiency Video Coding (HEVC). It provides a significant rate-distortion improvement over its predecessor H.264/AVC and can save 40–50 % bit rates compared to H.264/AVC,especially for 4K(3840×2160)/ 8K (7680 × 4320)-ultra-high-definition(UHD)video applications. A number of new algorithmic tools have been proposed, covering many aspects of video compression technology, such as larger coding units, new tools, and more complex prediction schemes. Motion compensation (MC) is the key factor for efficient video compression. Compensation for motion with fractional-pel accuracy requires interpolation of reference pixels. Therefore, in order to increase the performance of integer pixel motion estimation, the sub-pixel (i.e., half and quarter) accurate variable block size motion estimation is applied in both H.264/AVC and HEVC. The H.264/AVC standard uses a six-tap finite impulse response (FIR) luma filtering at half-pixel positions followed by a linear interpolation at quarter-pixel positions. Chroma samples are computed by the weighed interpolation of four closest integer pixel samples. In HEVC standard, three different eight-tap or seven-tap FIR filters are used for the luma interpolation of half-pixel and quarter-pixel positions, respectively. Chroma samples are computed using four-tap filters. Sub-pixel interpolation is one of the most computationally intensive parts of HEVC video encoder and decoder.

In the high-efficiency and low-complexity configurations of HEVC decoder, 37 and 50 % of the HEVC decoder complexity is caused by sub-pixel interpolation on average, respectively. On the other hand, compared with the six-tap filters used in H.264/AVC standard, the seven-tap and eight-tap filters cost more area in hardware implementation and occupy 37~50 % of the total complexity for its DRAM access and filtering. Therefore, it is necessary to design a dedicated hardware architecture for MC interpolation filter to realize the real-time processing for high-resolution videos. The main contributions of this paper are summarized as follows:

- A fast and implementation-friendly interpolation algorithm is proposed, which skips the interpolation process of 4×8, 4×16, 8×4, 16×4, 16×12, and 12×16 sub-PU blocks to reduce the encoding time and hardware complexity.

- A reused three-level interpolation filter architecture is adopted for the half-pixel and quarter-pixel interpolations to store the intermediate result and thus can reduce the hardware cost.

- An efficient memory organization method is proposed in the paper to reduce the data access of SRAM and save the power of VLSI architecture.

- A five-step pipeline interpolation filter engine is proposed in the paper. It can shorten the critical path of the filter and improve the working speed.
- A reconfigurable interpolation unit is developed in the paper, and the two types of the filters can be carried out with the same hardware architecture by only reversing the order of input reference pixels. As a result, the proposed reconfigurable filter can reduce the area of the whole architecture.

L. Pipeline Interpolation Filter

This interpolator can support 8-pixel interpolation, which can adapt to most of the variable block sizes. One 16×16 block is split into two 8×16 blocks, and a 16×8 block is split into two 8×8 blocks. For the interpolation process of a 64×64 CU, $8\times$ block module can be reused by eight times.
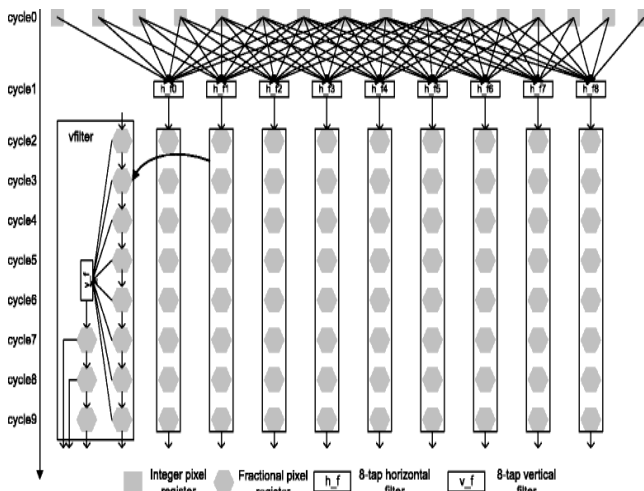

Fig. 4. The Proposed Pipeline interpolator architecture.

The pipeline 8-pixel interpolation filter engine. The proposed pipeline interpolator architecture. h_f represents the 8-tap horizontal interpolation filter, and v_f represents the 8-tap vertical interpolation filter. There are five steps in the operation of the interpolation filter pipeline.

Step1: The interpolation filter reads the reference integer pixels from the first line, and as a result, there are 16 reference data inputs from 0~15.

Step2: The horizontal interpolation filter h_f0 reads the integer pixels 0~ 7 of line 1 and the filter h_f1 read the integer pixels 1~8,and so on.These 16 pixels are interpolated by the corresponding horizontal interpolation filters.

Step3: The filtered data from the horizontal interpolation filter of line 1 are written into the registers of the vertical interpolation filter v_f. By repeating the same operations as in step 1 and step 2, the filtered data of following lines are written into the registers.

Step4: When the registers of v_f are filled with eight pixels, the 8-tap vertical interpolation starts to work and the filtered results of line 1 will be obtained.

Step5: When v_f executes filtering from line 1 to line 8, the input reference pixels of line 9 are interpolated by the horizontal interpolation filter h_f simultaneously. After the filtered data   of line 9 are written into the register of v_f

and the filtered results of line 1 are released by the register, the vertical interpolation filter starts to execute the filtering operation on the reference pixels from line 2 to line 9. According to the five steps above, the $8\times$ block interpolation engine performs the pipeline filtering operations and the ultimate interpolation filtered result will be obtained after one clock cycle.

IV. CONCLUSION

In this paper, high-performance VLSI architecture for luma interpolation in HEVC is proposed and it is implemented with 37.2k gates at an operating frequency of 240 MHz. It can support 8K-UHD (7680×4320)@78fps (4:2:0 format) real-time video processing. Our proposed architecture can be reused for half-pixel interpolation and quarter-pixel interpolation, and it reduces the area cost about 131,040 bits RAM with the reused interpolation architecture. Our proposed architecture can achieve high throughput for real-time encoding of ultra high-resolution videos with reduced hardware resources and is especially suitable for 8K-UHD video real-time encoding.

V. REFERENCES

[1] Lu YU, Jian-peng WANG, "Hot topic: Review of the current and future technologies for video compression", Journal of Zhejiang University-SCIENCE C (Computers & Electronics), 2010 11(1):1-13

[2] Hsueh-Ming Hang, Wen-Hsiao Peng, Chia-Hsin Chan and Chun-Chi Chen," Towards the Next Video Standard: High Efficiency Video Coding", Proceedings of the Second APSIPA Annual Summit and Conference, pages 609−618, Biopolis, Singapore, 14-17 December 2010.

[3] Bin Li, Gary J. Sullivan, and Jizheng Xu," Compression Performance of High Efficiency Video Coding (HEVC) Working Draft 4", pp-886-889 IEEE, 2012.

[4] JCT-VC, "Report of Subjective Test Results of Responses to the Joint Call for Proposals (CfP) on Video Coding Technology for High Efficiency Video Coding (HEVC)," Document JCTVC-A204, Dresden, DE, Apr. 2010.

[5] F. Bossen, "Common test conditions and software reference configurations," Document JCTVC-F900, Torino, IT, July 2011.

[6] Harilaos KOUMARAS, Michail-Alexandros KOURTIS, Drakoulis MARTAKOS, "Benchmarking the Encoding Efficiency of H.265/HEVC and H.264/AVC", IIMC International Information Management Corporation, 2012, ISBN: 978-1-905824-30-4

[7] Mahsa T. Pourazad, Colin Doutre, Maryam Azimi, and Panos Nasiopoulos, "HEVC: The New Gold Standard for Video Compression", IEEE CONSUMER ELECTRONICS MAGAZINE, JULY 2012.

[8] Detlev Marpe, Heiko Schwarz, Thomas Wiegand, Sebastian Boße, Benjamin Bross, Philipp Helle, Tobias Hinz, Heiner Kirchhoffer, Haricharan Lakshman,Tung Nguyen, Simon Oudin, Mischa Siekmann, Karsten Sühring, and Martin Winken," Improved Video Compression Technology and the Emerging High Efficiency Video Coding Standard", IEEE International Conference on Consumer Electronics - Berlin (ICCEBerlin), 2011.

[9] M. Winken, P. Helle, D. Marpe, H. Schwarz, and T. Wiegand, "Transform coding in the HEVC test model," in Proc. IEEE International Conference on Image Processing (ICIP), Sep. 2011.

[10] S. Oudin, P. Helle, J. Stegemann, C. Bartnik, B. Bross, D. Marpe, H. Schwarz, and T. Wiegand, "Block merging for quadtree-based video coding," in Proc. IEEE International Conference on Multimedia and Expo (ICME), Jul. 2011.

[11] H. Lakshman, B. Bross, H. Schwarz, and T. Wiegand, "Fractional-sample motion compensation using generalized interpolation," in Proc. Picture Coding Symposium (PCS), Dec. 2010.

[12] M. Siekmann, S. Boße, H. Schwarz, and T. Wiegand, "Separable Wiener filter based adaptive in-loop filter for video coding," in Proc. Picture Coding Symposium (PCS), Dec. 2010.

[13] Philippe Bordes, Gordon Clare, Félix Henry, Mickaël Raulet, Jérôme Viéron, "An overview of the emerging HEVC standard", IEEE, 2010.

[14] B. Bross, W. J. Han, J. R. Ohm, G. J. Sullivan and T. Weingand, "High efficiency video coding (HEVC) text specification draft 6," JCT-VC Document, JCTVCH1003-v21, April 2012.