41.7BN-PIXELS/S RECONFIGURABLE INTRA PREDICTION ARCHITECTURE FOR HEVC 2560X1600 ENCODER

Zhenyu Liu[†*] *Dongsheng Wang*[†] *Hongxiang Zhu*[⋆] *Xiaodong Huang*[⋆]

[†]TNList, Tsinghua University, Beijing 100084, China (liuzhenyu73,wds@tsinghua.edu.cn)

[⋆]Northwesten Polytechnical University, Xian 710072, China

ABSTRACT

The complexity of High Efficiency Video Coding (HEVC) intra prediction design mainly comes from two aspects. First, as compared with the predecessor H.264/AVC, HEVC increases the number of prediction angles from 9 up to 33. Second, HEVC employs 5 kinds of $n \times n$ prediction unit size, including 4×4, 8×8, 16×16, 32×32 and 64×64. The computation intensity of intra encoding is increased by one order. In this paper, we provide the high efficient reconfigurable VLSI architecture for all intra directional prediction modes. The proposed design possesses the following merits: (1) Our prediction engine is equipped with sixteen uniform modules, and can be configured to produce $2 \cdot m$ number row-wise n prediction samples in each cycle, where $n = \{4, 8, 16, 32, 64\}$ and $m = 64/n$; (2) As our design always produces the row-wise samples, the hardware consuming transpose register array between the prediction residue module and the following DCT engine is eliminated. This feature further avoids the bubble operations in the horizontal predictions. With TSMC 90nm CMOS technology, the proposed architecture achieves 357MHz operating frequency at the cost of 817.3k gates, and the corresponding power dissipation is 114mW. Our implementation can fulfill the throughput requirement of HD2560×1600@46fps real-time encoding.

Index Terms— HEVC, Unified Directional Intra Prediction, Reconfigurable, VLSI

1. INTRODUCTION

High Efficiency Video Coding (HEVC) standards [1, 2], being jointly developed by ISO/IEC Moving Picture Experts Group (MPEG) and ITU-T Video Coding Experts Group (VCEG), targets for saving up to 50% rate cost over the predecessor H.264/AVC [3], while maintaining the picture quality. The new standards provide the more efficient compression capability for High Definition (HD) as well as the coming Ultra-High Definition (UHD) video signals. To handle the large picture sizes, HEVC introduced the more flexible block based unit representation schemes, including quad-tree based coding unit (CU), variable block sizes of prediction unit (PU), and transform unit (TU). The CU is the basic processing unit. One $2n \times 2n$ CU can be split to four $n \times n$ sub-CU. PU is the prediction unit, and its size is the same as or smaller than the CU owner. The hierarchical TUs, whose sizes do not exceed the PU, are adopted in the transform of prediction residues. Moreover, HEVC employs the more accurate and computation consuming prediction mechanisms to reduce the residues, and consequently, promote the compression efficiency. As compared with H.264/AVC main-profile, the encoding time of HEVC is augmented by 470%. The additional computational complexity includes: First, the encoder traverses all combinations of CU, PU, and TU sizes to derive the candidate with the minimum rate-distortion cost. Second, the prediction signal generation requires more arithmetics. Specifically, HEVC intra prediction algorithm adopts 5 kinds of PU sizes, including 4×4, 8×8, 16×16, 32×32 and 64×64 block sizes. In contrast, H.264/AVC main profile merely has 4×4 and 16×16 blocks. Second, for each kind of PU size, the prediction modes number is extended to 35, which is composed of 33 directional predictions (unified direction intra), *DC* and *Planar* predictions [4, 5]. The high computational intensity makes ASIC accelerator essential in the real-time encoding, especially when facing HD and UHD specifications.

The rest of the paper is organized as follows. The related works and our contributions are discussed in Section 2. Next, Section 3 describes the overview of intra prediction mechanism and the corresponding encoding procedure. The proposed intra prediction architecture is provided in Section 3. The detailed performance analysis of our design are illustrated in Section 5. Finally, conclusions are drawn in Section 6

2. RELATED WORKS

In literature [6], Li et.al. noticed that 4×4-PUs account for 66% of the total PU number in the decoder side. In consequence, they devised the 4×4 intra prediction engine supporting 17 prediction directions and possessing 100M-pixel/S throughput. However, for the encoder side, especially when considering the HD specifications, as all PU sizes must be traversed to find the best candidate, billions of prediction pixels must be generated in each second. The high parallelism is a must in the prediction engine design. In addition, as many fast CU/PU mode decision algorithms[7, 8] have been proposed, PU size based reconfiguration is another desired feature. This is because that, if each PU size is equipped with the dedicated prediction engine, its hardware utilization will be degraded when this PU size is always skipped by the fast algorithm in the intra search procedure. Finally, in the HM reference software, the horizontal directional predictions always produce the column-wise signals. Because the following 2D-transform is constituted by the first row-wise and the following column-wise one-dimensional transforms, the $n \times n$ transpose register array is needed in the primitive implementation, which introduces the additional hardware cost and n-cycle bubbles operations ($n = \{4, 8, 16, 32, 64\}$).

In this paper, the parallel reconfigurable intra prediction architecture is provided. Our design is composed of sixteen process units, and it can be configured to generate $2 \cdot m$ number $n \times 1$ row-wise prediction pixels for $n \times n$ PU size, where $m = 64/n$. Because the generated signals is always row-wise, the inter stage transpose

*This work is funded by Huawei Technologies, TNList cross-discipline foundation, the Nature Science Foundation of China (Grant No.60833004 and 60902101), and the National 863 High-Tech Programs of China (No.2012AA010905).

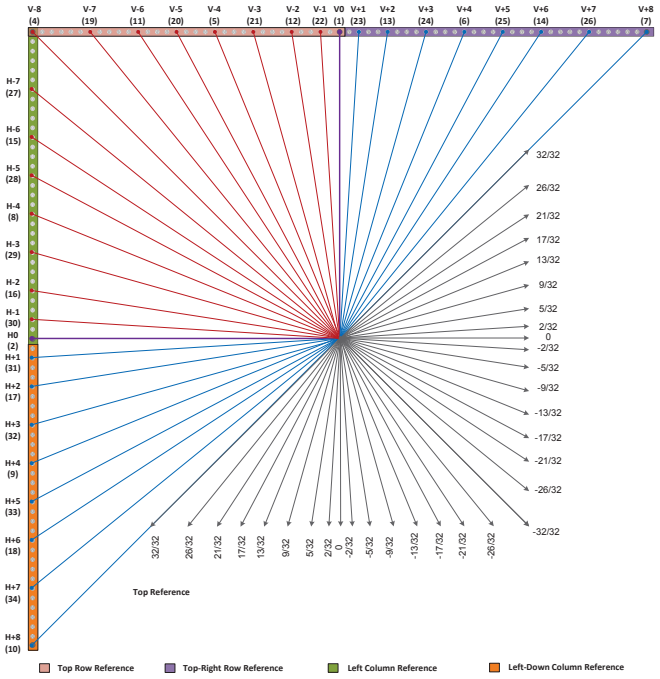

Fig. 1. Intra prediction directions and associate indices

register array is saved by our intra prediction engine.

3. OVERVIEW OF INTRA PREDICTION AND ENCODING

To reduce the prediction residual redundancy, and hence ameliorate the compression rate, HEVC refines the intra prediction angles to 33 modes, as shown by Fig. 1. In contrast, H.264/AVC is merely equipped with 9 intra prediction directions. The prediction directions are coarsely categorized to two sets, i.e., vertical predictions and horizontal predictions. In vertical predictions, the absolute value of the prediction angle θ is defined as

$$|\theta| = \arctan\left[\frac{\delta x}{\delta y}\right], \tag{1}$$

where, $\delta y = 32$ and $\delta x = [0, 2, 5, 9, 13, 17, 21, 26, 32]$. The horizontal predictions apply the similar mechanism. It should be noticed that θ could be negative or positive. The red lines indicate the negative directions and the blue lines represent the positive ones. In the most recent reference software HM9.0, for simplifying the decoder complexity, all PU sizes are entitled with 33 directional prediction modes. As shown by Fig. 1, the reference pixels for the current PU predictions include the neighboring left and left-down columns, and top and top-right rows. According to the primary prediction direction (vertical or horizontal), the references are defines as the main array and the side array. In case of vertical directions, the main array is composed of the top and top-right row-wise pixels, and the left column pixels are used as the side references. In the other case (horizontal predictions), the left and left-down column pixels are denoted as the main array, and consequently, the top row pixels are adopted as the side one.

When θ is positive, only the pixels in the main array are involved in the prediction processing. As θ becomes negative, the side array should be involved as the reference samples. To simplify the reference candidate determination process, HEVC standards project the side array onto the main extension array according to the prediction

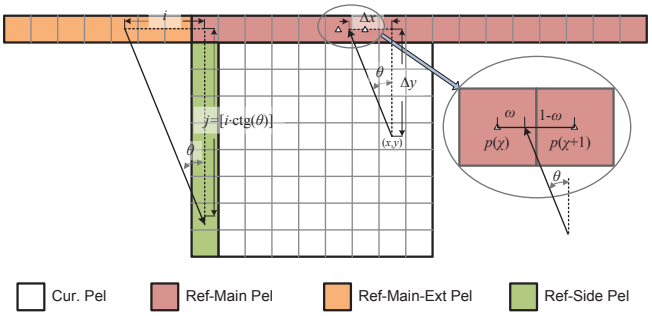

Fig. 2. Reference pixels projection mechanism in HEVC Intra directional prediction

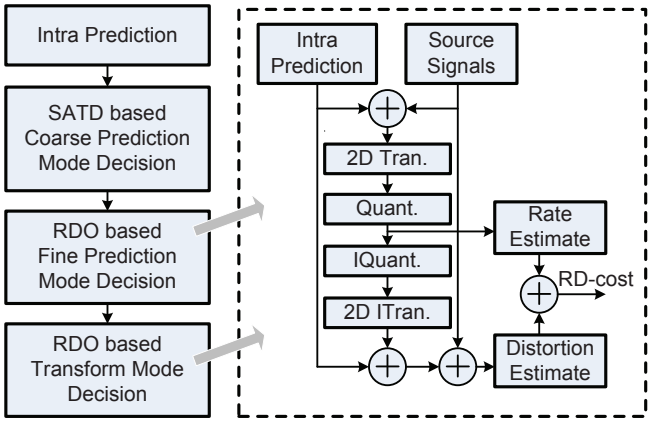

Fig. 3. HEVC intra encoding processing flow adopted by HM reference software

angle θ. Figure 2 depicts the side array projection in 8×8 PU prediction given the negative prediction angle. The khaki squares represent the main extension array. The ith pixel ($1 \le i < n$ for $n \times n$ PU) in the extension is projected from the jth pixel in the side array. The variable j is defined as

$$j = i \cdot [\cot(\theta)] = i \cdot \left[\frac{\delta y}{\delta x}\right]. \tag{2}$$

After constructing the entire reference main array, the prediction signals can be derived with the unified algorithm. Alone the prediction angle θ, any pixel in the current PU will be intercepted by the main array. The abscissa offset of the intercept point is labeled as Δx, which is derived as

$$\Delta x = \Delta y \cdot \frac{\delta x}{\delta y} \tag{3}$$

in which, Δy denotes the ordinate offset of the pixel to be predicted with respect to the main array. Two neighboring main reference pixels of the intercept point, i.e., $p(\chi)$ and $p(\chi + 1)$ (where $\chi = x + \lfloor \Delta x \rfloor$), are used to produce the prediction $\widetilde{p}(x, y)$ by using the simple linear interpolation formula as follows.

$$\widetilde{p}(x, y) = p(\chi)(1 - \omega) + p(\chi + 1)\omega \tag{4}$$

The intra encoding procedure adopted by the reference software is described as Fig. 3. The intra prediction mode decision includes prediction residue generation, SATD-based coarse prediction modes selection, RDO-based final prediction mode decision, and TU size decision. It can be observed that 2D-DCT and DST are utilized in

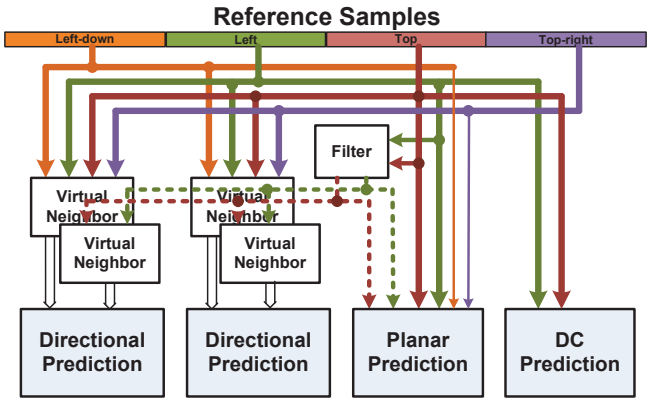

Fig. 4. Top block diagram of proposed HEVC intra prediction VLSI architecture

the last two steps. 2D-DCT/DST first processes row-wise and then the column-wise transforms. In the reference software, to work with the 2D transform, the $n \times n$ ($n \leq 64$) transpose register array is required for horizontal predictions, in which the samples are produced in column-wise. To eliminate the additional latency and hardware overhead of the transpose register, we devise the architecture providing the unified row-wise prediction mechanism.

4. VLSI ARCHITECTURE OF HEVC VARIABLE BLOCK SIZE INTRA PREDICTION ACCELERATOR

In this section, we first illustrate the top block of the overall intra prediction engine, and then describe the principle of the reconfigurable directional prediction and the hardware-saving *DC* and *Planar* accelerators.

4.1. Overview of Proposes Intra Prediction Engine

The overview of the top block diagram of the proposed intra prediction architecture is described in Fig. 4. As mentioned in Section 3, the source signals come from the neighboring left-down, left, top, and top-right totally 257 pixels. The prediction engine is composed of two directional prediction accelerators, one variable block size (VBS) *Planar* predictor, and one *DC* predictor. Each directional prediction accelerator can generate m horizontal or vertical modes row-wise $n \times 1$ prediction samples in each cycle, where $m = 64/n$ and $n = \{4, 8, 16, 32, 64\}$. The detailed circuits and principle of directional prediction accelerator will be discussed in the following paragraph. The *Planar* and *DC* predictor both can produce $n \times 1$ prediction samples in each cycle.

Recalling that, in modes 8×8, 16×16 and 32×32, when the value of $|\theta|$ is greater than the special threshold, the samples after 1-2-1 filtering are applied as the references. The filter in Fig 4 is dedicated to produce the 127 number filtered samples. Because the maximum block size using filtered samples is 32×32, the inputs to filter component merely include the top-row and left-column arrays, totally 129 pixels. Moreover, the *Planar* predictions in 8×8, 16×16 and 32×32 also use the filtered reference pixels.

As mentioned in Section 3, the main and side reference arrays are adaptively defined according to dominant prediction direction (vertical or horizontal). The function of the proposed "Virtual Neighbor" is to map the neighboring row and column pixels to the main and side arrays with the unified format. Let us use 4×4 predictions as an example. The neighboring pixel mapping schemes

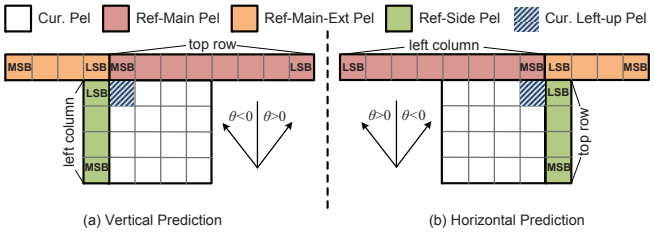

Fig. 5. Demonstration of neighboring reference pixel array mapping schemes for vertical and horizontal predictions (all symbols in (b) is rotated by 90 degree in clock-wise direction.)

for vertical and horizontal prediction are illustrated as Fig. 5(a) and (b), respectively. It should be noticed that the endianness types in Fig. 5(a) and (b) are different. For example, the first row pixel in the left column is the least significant byte (LSB) of the side array in the vertical predictions. In contrast, the horizontal ones use the same pixel as the most significant byte (MSB) of the main array. That is, the endianness of vertical predictions is clockwise, while the horizontal predictions apply the anti-clockwise endianness scheme. We can see that, for both cases, when $\theta < 0$, the prediction vector approaches the MSB of the main extension as the increase of $|\theta|$, while the prediction vector approaches the LSB of the main with the augment of $|\theta|$ when $\theta > 0$. In addition, in our implementation, the left-top corner neighboring pixel is always categorized as the main extension. The above optimization will simplify the reference selection circuits in the following directional prediction engines.

4.2. VLSI Architecture of Directional Prediction Engine

The source signals to the directional predictor come from "Virtual Neighbor" components. The directional intra predictions engine is primarily composed of sixteen 4×1-pel filters, which can be configured according to the current PU size. Specifically, when the PU size is n, the sixteen filters are deployed into $64/n$ groups.

For example, the top block diagram of $n = 16$ is depicted as Fig. 6. The inputs to each filter group include the original and the filtered main and side arrays. According to the current PU size and direction mode, the filter group first selects the original or the filtered signals as the references. When the prediction angle is negative, the "Mapper" component in each group is applied to generate the main extension pixels. It should be noticed that, because the main array can be shared by all predictions, there is no dedicated main array registers in each filter group. In contrast, because the prediction directions are different, each group must be equipped with the main extension registers. The whole 64-pel main-extension registers is also reconfigurable. In each group, n-pel registers are allocated to store the current projected extension pixels.

As aforementioned, the proposed design is guaranteed to produce row-wise signals. This feature mainly depends on the source signal selection mechanism design of the last stage filter. For the vertical predictions, each prediction pixel can determine its own intercept point according its position and its offset from the main array. Then, the two source pixels in the main array are determined, and then the prediction is derived from (4). In case of horizontal predictions, the offsets to the left neighboring column for prediction pixels in the same row are different. Therefore, each prediction pixel must have its own source selection logic. However, all prediction pixels in the same column always possess the same intercept offset in vertical direction. In this case, we configure the reference main registers as the shifting window. In each cycle of horizontal predictions, the

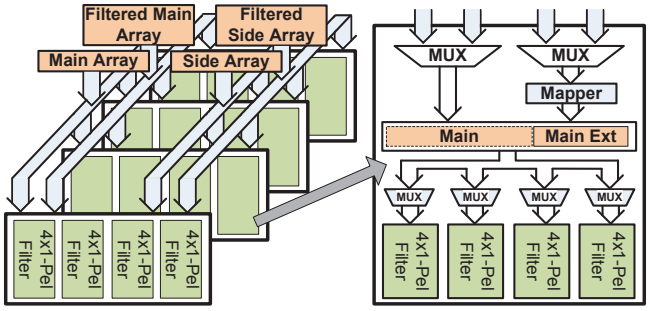

Fig. 6. Block diagram of directional prediction engine when $n = 16$

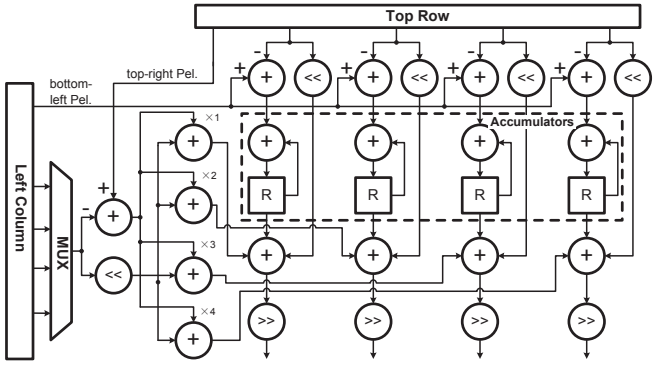

Fig. 7. Block diagram of Planar prediction engine when $n = 4$

project of each prediction in the same row maintains, while the main registers perform the right shift on the pixel pattern.

4.3. Simplified DC and Planar Prediction Engine

The *DC* mode prediction consists of two steps, the first step is calculating *DC* value of the reference samples, and the next one is filtering left and top edges of the prediction samples. To save the chip area, we applied an reusable adder module, which is composed of merely four three-input adders. The adder module is in charge of both computing the *DC* value and filtering the boundary pixels. Our *DC* engine requires initialization to derive the *DC* value and filter the first row pixels. Specifically, it requires a total of 1-, 6-, 12-, 24-, and 48-cycle initialization for 4×4, 8×8, 16×16, 32×32, 64×64 PUs, respectively. On the other hand, we schedule the *DC* prediction after the directional predictions. Therefore, its initialization overhead is hidden. For the *Planar* prediction always produces the row-wise pixels, we replace the multipliers with the accumulators in the column-wise interpolations. For instance, the block diagram for 4×4-PU *Planar* prediction is shown as Fig. 7.

5. EXPERIMENTAL RESULTS

The proposed architecture is implemented with TSMC 90nm 1P9M CMOS technology to verify its performance. Our design is described with Verilog-HDL, synthesized with SYNOPSYS Design-Compiler, and Place&Route using SYNOPSIS IC-Compiler. On the worst work conditions (0.9v, 125°C), the maximum operation frequency is 357MHz. The hardware cost analysis under different typical clock frequencies is illustrated as Table 1. The total hardware overhead of our design is composed of all component as shown in Fig. 4, i.e., two direction prediction engines, *Planar* engine, *DC* engine, reference filter, and virtual neighbor generator, in addition to 0.6-0.9k-gate control logic. It was observed that, under 166MHz clock speed, our design consumed 678.4k-gate standard cells. At the peak working frequency (357MHz), the corresponding hardware cost was augmented to 817.3k-gate.

Table 1. Hardware Cost Analysis (k-gate)

Component	Frequency(MHz)				
	166	233	300	333	357
DirPred	577.7	581.2	592.2	600.7	712.2
DC	10.6	10.7	10.8	10.9	11.1
Planar	44.8	44.9	47.4	47.6	48.4
Filter&VirtNbr	45.3	45.3	45.3	45.6	45.6
Total	678.4	682.1	695.7	704.8	817.3

DirPred: Directional Prediction; VirtNbr: Virtual Neighbor

Table 2. Power Consumption Analysis (mW)

Component	Frequency(MHz)				
	166	233	300	333	357
DirPred	40.7	56.7	74.5	86.0	92.1
DC	1.7	2.4	3.1	3.5	3.8
Planar	4.4	6.1	8.3	9.9	10.0
Filter&VirtNbr	3.7	5.2	6.7	7.4	8.1
Total	50.5	70.4	92.6	101.4	114.0

DirPred: Directional Prediction; VirtNbr: Virtual Neighbor

Our design applied the clock gating technique to disable the clock signals of idle registers, and saved 28-33% power dissipation. The power consumption analysis is given by Table 2. Directional prediction engines account for more than 80% of the overall power consumption.

Table 3. Performance Comparisons

Design	Tech. (nm)	Freq. (MHz)	PB (pel/cycle)	PU (mode)	Pred. (mode)	HC (k-gate)	HE (k)
Proposed	90	357	128	Full	Full	817.3	55.9
[6]	130	150	0.67	4×4	17	9.02	11.1

HE: hardware efficiency equal to (Freq.×PB/HW)

Table 3 summaries the performance comparisons of the proposed design with other counterparts in terms of prediction bandwidth (PB), supported PU/Prediction modes, hardware cost(HC), clock speed, and hardware efficiency (HE). As compared with previous work in [6], our design supports all PU sizes and prediction modes adopted by HEVC standards. In each cycle, our design can generate 128 prediction samples, which is 192 times of the previous work. Moreover, the clock speed is improved by 138%. If the hardware efficiency is defined as the ratio of the throughput to the gate count, it can be observed that we improved this indicator by 404%. The maximum throughput is listed as Table 4, which fulfills the specifications of HD2560x1600@46fps real-time encoding.

Table 4. Throughput Analysis (million-block/sec)

PU size	4×4	8×8	16×16	32×32	64×64
Throughput	434.81	163.05	33.97	7.64	1.91

6. CONCLUSIONS

The parallel reconfigurable intra prediction engine for HEVC stardands is proposed in this paper. Our design supports 4×4 to 64×64 variable PU sizes and full 35 prediction modes. 128 prediction pixels can be produced in each cycle. Using TSMC 90nm technology, 357MHz clock speed is achieved at the cost of 817.3k gates and 114.0mW power dissipation. Our design supports the 2560×1600 real-time encoding at the 46fps frame rate.

7. REFERENCES

[1] Ken McCann et al., "Samsungs response to the call for proposals on video compression technology," Dresden, DE, 2010, JCTVC-A124.

[2] G.J. Sullivan, J.R. Ohm, W.J. Han, and T. Wiegand, "Overview of the high efficiency video coding (hevc) standard," *IEEE Trans. Circuits and Systems for Video Tech*, 2012.

[3] Detlev Marpe, Thomas Wiegand, and Gary J. Sullivan, "The H.264/MPEG4 advanced video coding standard and its applications," *IEEE Commun. Mag.*, vol. 44, no. 8, pp. 134–143, Aug. 2006.

[4] J.H. Min, S. Lee, I.K. Kim, W.J. Han, J. Lainema, and K. Ugur, "Unification of the directional intra prediction methods in tmuc," *JCTVC-B100, Geneva, Switzerland*, 2010.

[5] T. Tan, M. Budagavi, and J. Lainema, "Summary report for te5 on simplification of unified intra prediction," *JCTVC-C046*, 2010.

[6] F. Li, G. Shi, and F. Wu, "An efficient vlsi architecture for 4×4 intra prediction in the high efficiency video coding (hevc) standard," in *Image Processing (ICIP), 2011 18th IEEE International Conference on*. IEEE, 2011, pp. 373–376.

[7] H. Sun, D. Zhou, and S. Goto, "A low-complexity hevc intra prediction algorithm based on level and mode filtering," in *Multimedia and Expo (ICME), 2012 IEEE International Conference on*. IEEE, 2012, pp. 1085–1090.

[8] G. Tian and S. Goto, "Content adaptive prediction unit size decision algorithm for hevc intra coding," in *Picture Coding Symposium (PCS), 2012*. IEEE, 2012, pp. 405–408.