


## Article

# Enhanced Intra Prediction Based on Adaptive Coding Order and Multiple Reference Sets in HEVC

Jin Young Lee 

School of Intelligent Mechatronics Engineering, Sejong University, Seoul, Korea; jinyounglee@sejong.ac.kr;  
Tel.: +82-2-6935-2672

Received: 5 June 2019; Accepted: 20 June 2019; Published: 22 June 2019



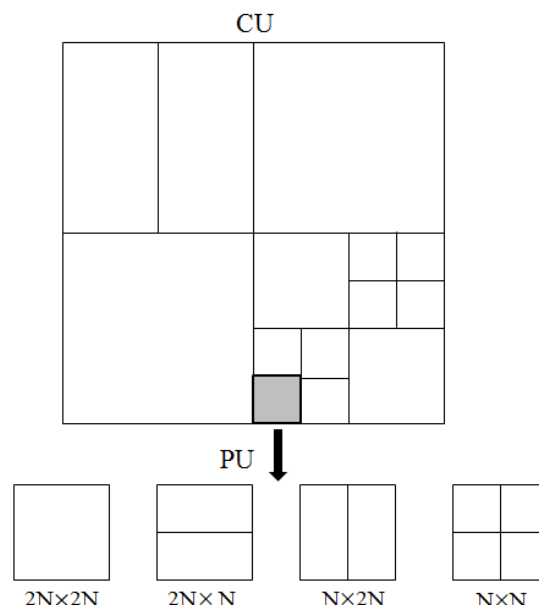
**Abstract:** High Efficiency Video Coding (HEVC) is the most recent video coding standard. It can achieve a significantly higher coding performance than previous video coding standards, such as MPEG-2, MPEG-4, and H.264/AVC (Advanced Video Coding). In particular, to obtain high coding efficiency in intra frames, HEVC investigates various directional spatial prediction modes and then selects the best prediction mode based on rate-distortion optimization. For further improvement of coding performance, this paper proposes an enhanced intra prediction method based on adaptive coding order and multiple reference sets. The adaptive coding order determines the best coding order for each block, and the multiple reference sets enable the block to be predicted from various reference samples. Experimental results demonstrate that the proposed method achieves better intra coding performance than the conventional method.

**Keywords:** HEVC; intra prediction; adaptive coding order; multiple reference sets

## 1. Introduction

With the advent of high-definition (HD) video contents, an advanced video coding standard that exceeds the coding capability of H.264/AVC [1] was strongly required by industry. To efficiently transmit HD videos in a limited bandwidth, the Joint Collaborative Team on Video Coding (JCT-VC), which was established by the ITU-T Video Coding Experts Group (VCEG) and ISO/IEC Moving Picture Experts Group (MPEG), developed the HEVC standard in 2013 [2,3]. Because it perfectly supports higher resolutions, higher frame rates, and higher image quality, it is currently being used in many applications, such as smart phones, tablet devices, and digital television. In addition, thanks to the high coding efficiency of the HEVC standard, it was successfully extended to 3D video coding by the Joint Collaborative Team on 3D Video Coding Extension Development (JCT-3V) in 2015 [4].

Most of the techniques in HEVC are very similar to those in H.264/AVC, but HEVC doubles the coding performance of H.264/AVC [5]. In order for the high improvement, HEVC introduces a novel coding structure such as a coding unit (CU), a prediction unit (PU), and a transform unit (TU). CU in HEVC is exactly the same as a macroblock in H.264/AVC, but it can be recursively split into four equal small size CUs up to a given split depth. For example, let us assume that the size of the largest CU, which is called a coding tree unit (CTU), is  $64 \times 64$  and the predefined split depth is three. CTU can be represented as one  $64 \times 64$  CU, four  $32 \times 32$  CUs, and sixteen  $16 \times 16$  CUs. Figure 1 shows an example of CUs and PUs. As displayed in Figure 1, the prediction partitions ( $16 \times 16$ ,  $16 \times 8$ ,  $8 \times 16$ , and  $8 \times 8$ ) in H.264/AVC are extended into four PU types ( $2N \times 2N$ ,  $2N \times N$ ,  $N \times 2N$ , and  $N \times N$ ), according to the relationship between CU in HEVC and the macroblock in H.264/AVC. In addition, four asymmetric PU types ( $2N \times nU$ ,  $2N \times nD$ ,  $nL \times 2N$ , and  $nR \times 2N$ ) are also employed in HEVC. TU selects a transform size and then performs the transform on the arbitrary partition of PU. As HEVC recursively splits coding blocks and checks them with the various prediction and transform styles, it can significantly improve the performance.



**Figure 1.** Example of coding units (CUs) and prediction units (PUs) in high efficiency video coding (HEVC).

To achieve the higher coding performance, HEVC includes many coding technologies, such as advanced motion vector prediction (AMVP) and sample adaptive offset (SAO) [2]. In particular, angular intra prediction is one of the main improvements in HEVC. In general, a prediction block is generated by copying the boundary pixels of previously compressed blocks along a variety of angular directions, and then residuals between the original block and the prediction block are compressed. H.264/AVC uses nine different intra prediction modes, which consist of a DC mode and eight angular intra prediction modes. However, HEVC uses planar, DC, and 33 angular modes to perform more accurate prediction. These fine-grained angular directions provide the significantly higher prediction accuracy than that of H.264/AVC.

In addition to the simple increase in the number of the angular directions, there have been several studies on efficient intra coding methods [6–14]. Intra prediction, in the method proposed in [6], is performed through template matching in reconstructed pixels. Bi-directional intra prediction (BIP) presents additional intra prediction modes, directional transform, and adaptive coefficient scanning to achieve the higher intra coding efficiency in the method introduced in [7]. In [8], a method was proposed in which reference samples used for intra prediction are generated by a conventional 2-tap filter and a 4-tap discrete cosine transform (DCT)-based interpolation filter. In the method introduced in [9], separable transforms in mode-dependent directional transforms (MDDT) are used to approximate ideal non-separable directional transforms, and derived from the Karhunen–Loeve transform (KLT). An adaptive intra prediction mode bit skip method was proposed in [10] for a small block. It skips intra prediction mode bits when the correlation of the neighboring boundary pixels is strong. In [11], a sampling-based intra prediction method was introduced that divides an original image into two sub-images through horizontal or vertical sampling. One sub-image is first compressed, and then the other sub-image is predicted from the reconstructed sub-image. In a similar way, the method proposed in [12] generates two sub-images, but each sub-image is compressed with different intra prediction modes, quantization parameters, and scanning patterns. Recently, deep learning based intra coding methods have been introduced to improve the coding performance [13]. For example, context-based prediction enhancement proposed in [14] reduces residual energy in intra prediction by employing an encoder-decoder convolutional neural network.

This paper introduces an enhanced intra prediction method based on adaptive coding order and multiple reference sets. Blocks are usually compressed in the coding order of above-left to bottom-right, but the correlation direction between pixels can differ for every block. It is highly possible that some

blocks provide a higher performance with the coding order of above-right to bottom-left or bottom-left to above-right, than that of above-left to bottom-right. Furthermore, the original intra prediction method always uses fixed reference samples for the angular prediction modes. In order to generate a variety of the intra predictors, the proposed method defines multiple reference sets, which support various combinations of reference samples. For instance, the angular prediction from above row samples of the left neighboring block and left column samples of the above neighboring block is used for higher prediction accuracy. Experimental results show that the proposed method based on adaptive coding order and multiple reference sets achieves better coding performance than the conventional method.

The remainder of this paper is organized as follows. Section 2 provides an overview of intra prediction in HEVC. In Section 3, the proposed method based on adaptive coding order and multiple reference sets is described in detail. Finally, experimental results are presented and I conclude the paper in Sections 4 and 5, respectively.

## 2. Overview of Intra Prediction in HEVC

Intra prediction exploits reconstructed boundary samples of spatially neighboring blocks to remove spatial redundancies effectively. HEVC employs 35 different intra prediction modes, which include planar, DC, and 33 angular modes. Each prediction mode has its index number, and the mode numbers represent their angular directions (2 to 34), except for the planar (0) and DC (1) modes [15]. For example, the DC mode of the number 1 uses an average of neighboring pixels as a predictor, but the planar mode of the number 0 uses a weighting average. The numbers 10 and 26 mean horizontal and vertical modes, respectively. Since the horizontal and vertical directions are frequently selected to be the best direction, the density around these directions is relatively high in comparison to that of the diagonal direction.

The directional modes perform the intra prediction by extrapolating reference samples along the angular directions. Figure 2 shows an extrapolation method for the horizontal and vertical directional modes. For example, in the case of the horizontal directional modes between 2 and 17, the reference samples in the above row (red) are projected onto the left column (blue). When the intra prediction modes are between 18 and 34, which belong to the vertical directional modes, the reference samples in the left column (red) are projected onto the above row (blue). After this extrapolation, the predicted sample values  $p(x, y)$  are obtained with the following Equations [15]:

$$\begin{aligned} p(x, y) &= ((32 - w) \times r[m + i + 1] + w \times r[m + i + 2] + 16) \gg 5, \\ i &= ((m + 1) \times d) \gg 5, \\ w &= ((m + 1) \times d) \& 31 \end{aligned} \quad (1)$$

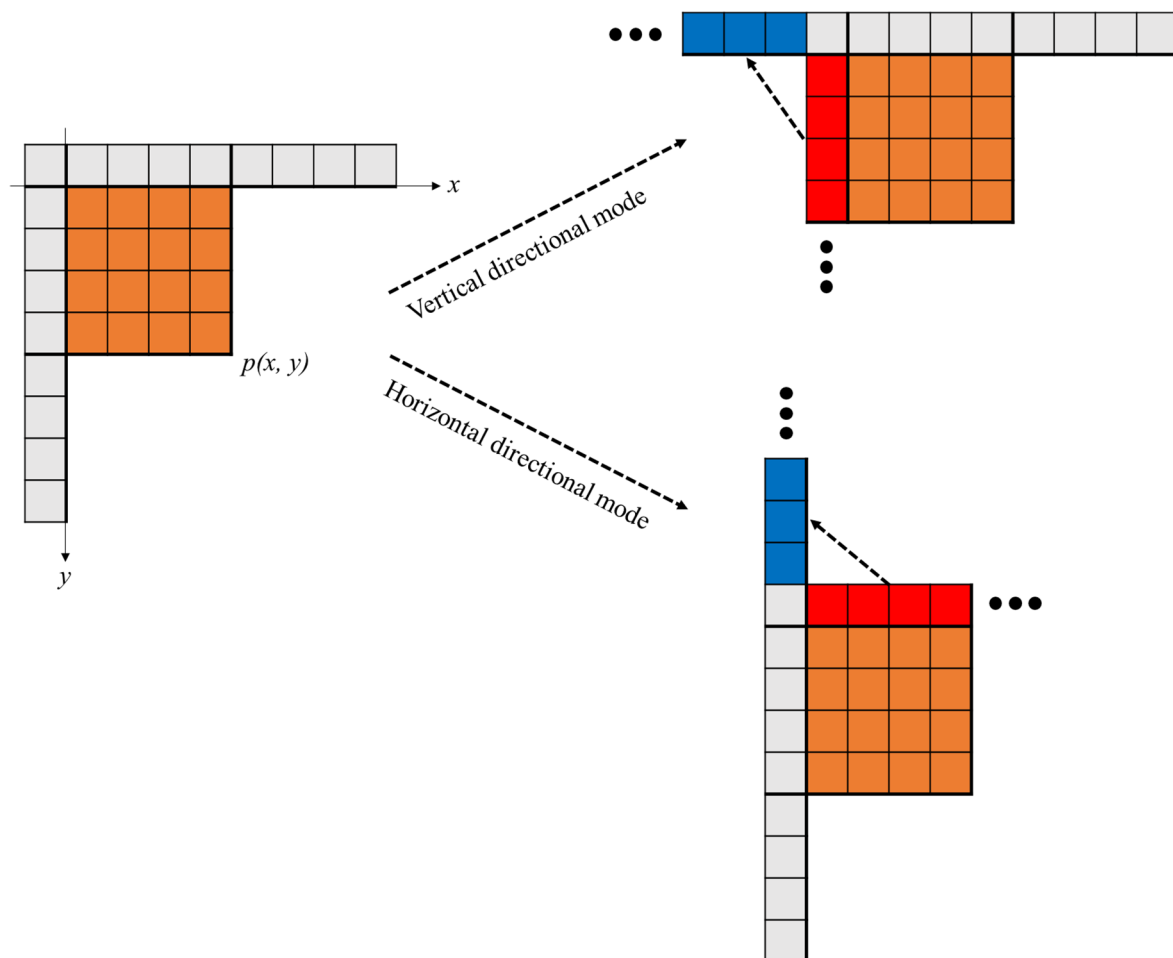
where  $x$  and  $y$  are spatial coordinates within a block. Here,  $r$  and  $i$  denote a reference sample and its index. When the intra prediction mode is between 2 and 17,  $m$  becomes  $x$ . When the mode is between 18 and 34,  $m$  becomes  $y$ .  $w$  denotes the weighting factor between the two reference samples  $r[m + i + 1]$  and  $r[m + i + 2]$ . The projected displacement  $d$  is expressed in an accuracy of  $1/32$ .

In order to determine the best mode among the 35 prediction modes without heavy complexity, many simplification processes have been studied [16,17]. First of all, sum of the absolute Hadamard transformed difference (SATD) is calculated for all possible prediction modes because the complexity of SATD is much lower than that of discrete cosine transform (DCT). Second, rough mode decision (RMD) is performed to find the modes that have smaller SATD costs. According to the block size, the number of the RMD modes is determined. Based on these RMD modes and the most probable mode (MPM) derived from the neighboring blocks, the rate-distortion (RD) list is constructed. In general, the MPM consists of the prediction modes of left and above blocks around the current block and one special mode, which is highly possible to be one of the planar, DC, horizontal, and vertical prediction modes. Next, RD costs [18] of the modes in the RD list are calculated and compared to each other. Finally, the

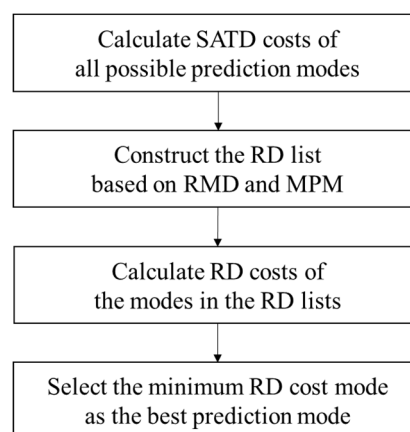
best prediction mode is chosen as the mode with the minimum RD cost. The RD cost is calculated with the following Equation:

$$RD\ cost = D + \lambda \times R \quad (2)$$

where  $D$  is the sum of the square difference between the original and reconstructed blocks,  $R$  is the number of coded bits, and  $\lambda$  is a predefined Lagrangian multiplier. It shows that the RD cost is calculated by considering not only the coding distortion ( $D$ ) but also the coding bitrate ( $R$ ). Figure 3 shows a flowchart of the intra prediction mode decision process.



**Figure 2.** Extrapolation for the horizontal and vertical directional modes.



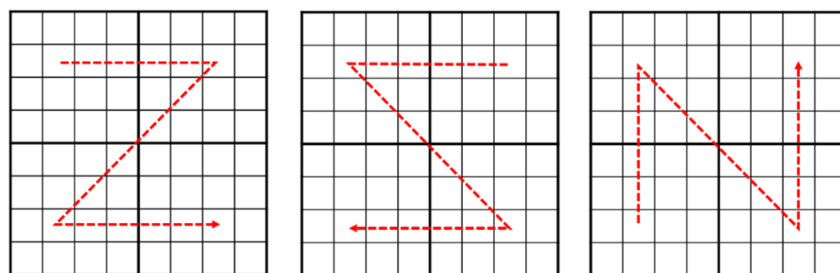
**Figure 3.** Flowchart of the intra prediction mode decision process.

### 3. Proposed Method

In general, pixels within an image have very strong correlations with each other. However, the original intra prediction method always uses fixed reference samples, as shown in Figure 2. To exploit the spatial correlation between the current block and its neighboring block as much as possible, the proposed method performs intra prediction from a variety of reconstructed samples by adaptively determining the best coding order and employing multiple reference sets. The adaptive coding order compresses CUs within each CTU, based on three different coding orders, and then compares their performance to find the best coding order. The multiple reference sets employ more reconstructed samples of the neighboring blocks as new reference samples in the prediction, so it can improve the prediction accuracy.

#### 3.1. Adaptive Coding Order

The goal of the adaptive coding order method is to find the best coding order that minimizes the RD cost of a frame or a block. Several related studies have been reported on top of H.264/AVC and HEVC [19–21]. Although the reported methods are somewhat different, their results are very similar. Therefore, three different coding orders, which are above-left to bottom-right, above-right to bottom-left, and bottom-left to above-right, are only proposed, as displayed in Figure 4. As the additional coding orders are investigated, the various correlation directions between the current and reference pixels can be compared to each other. The proposed method does not check the coding order of bottom-right to above-left, because our results showed that the additional performance was negligible on top of the other coding orders. It shows that these three coding orders are sufficient to find the correlation direction. The related results will be presented in Section 4.



**Figure 4.** Three different coding orders when the coding tree unit (CTU) has an  $8 \times 8$  size and it is split into  $4 \times 4$  CUs: Above-left to bottom-right (**left**), above-right to bottom-left (**middle**), and bottom-left to above-right (**right**).

Once the coding order is determined, all of the split CUs within the CTU are compressed in the corresponding order. Therefore, the index bits that indicate the best coding order are required for every CTU. As shown in Table 1, in the proposed method, when the coding order is above-left to bottom-right, a codeword of ‘0’ is signaled to a decoder. When the coding order is above-right to bottom-left and bottom-left to above-right, codewords of ‘10’ and ‘11’ are transmitted, respectively. The RD cost of each CTU is calculated as follows:

$$RD \cos t_{CTU}^i = \sum_{k \in CTU} RD \cos t_k \quad (3)$$

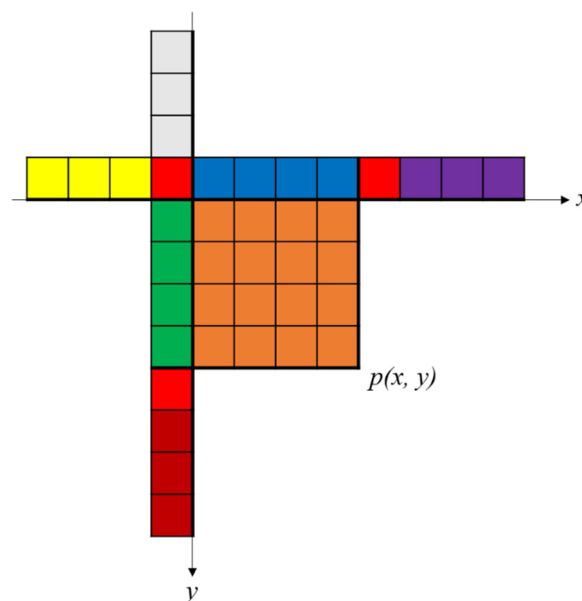
where  $k$  represents the indexes for the best CUs determined within the CTU and  $i$  means the coding order. The best coding order is selected to be the coding order that has the minimum RD cost.

**Table 1.** Codeword for the three different coding order.

Coding Order	Codeword
Above-left to bottom-right	0
Above-right to bottom-left	10
Bottom-left to above-right	11

### 3.2. Multiple Reference Sets

The original intra prediction modes use boundary samples adjacent to the current block [15]. Figure 5 illustrates all the possible reference samples that can be exploited for the intra prediction of the current block (orange) in the proposed method. For easy understanding, colors are employed to describe the reference pixels associated with the explanation. In Figure 5,  $x$  and  $y$  are a coordinate within the current block, and its origin is located in the above-left. This paper defines the red, blue, purple, green, and brown reference samples as an original reference set, which is always used to predict the current pixel values  $p(x, y)$  in the original method. If the correlation between the current block and the original reference set is very strong, residuals between the original and predicted pixel values will be small. On the other hand, if the correlation is very weak, the residuals will be large, which will result in performance degradation. Therefore, the performance highly depends on the correlation between the original and reference sample values. As some reconstructed pixels, such as yellow and grey, are not currently used in the prediction, there is still room for improvement in the intra coding performance.

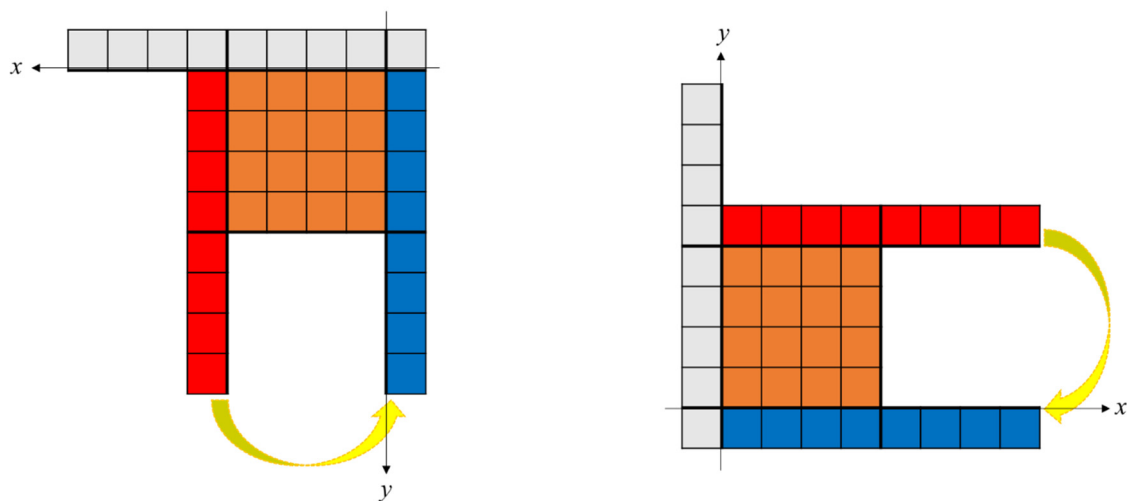
**Figure 5.** Possible reference samples that can be used in the proposed method.

To generate accurate predictors from various reference samples, the proposed method defines multiple reference sets, which consist of original, horizontal, and vertical reference sets. Table 2 shows which reference samples these three reference sets contain. Compared to the original reference set, the horizontal reference set uses above row samples of the left neighboring block, which are yellow samples in Figure 5. The vertical reference set includes left column samples of the above neighboring block, which are grey samples. In sum, yellow samples are added for the horizontal reference set, instead of purple, and grey samples are added for the vertical one, instead of brown. As these various reference samples can generate different predicted pixel values, it can contribute to enhancement of the intra prediction performance.

**Table 2.** Multiple reference sets.

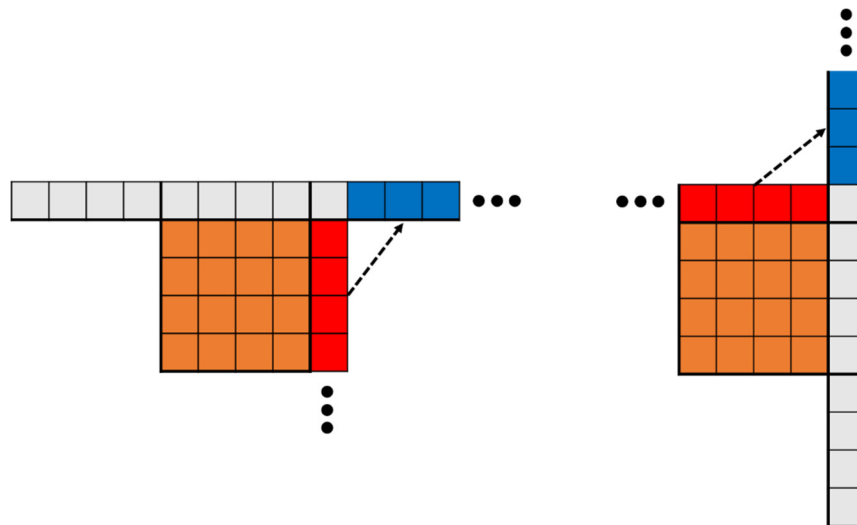
Reference Set	Reference Samples (Color in Figure 5)
Original reference set	Red, blue, purple, green, brown
Horizontal reference set	Red, blue, yellow, green, brown
Vertical reference set	Red, blue, purple, green, grey

Figure 6 shows how the current block is predicted with the additional reference sets. In the case of the horizontal reference set in Figure 6 (left), the origin of the  $xy$ -coordinate is first translated from above-left to above-right, and its axis is moved accordingly. Next, the left column samples (red) are copied to the right column of the current block (blue). Finally, prediction is performed with Equation (1). Please note that the  $xy$ -coordinate is different from the original one. In a similar way, prediction using the vertical reference set can be performed by translating the origin of the  $xy$ -coordinate from above-left to bottom-left. As shown in Figure 6 (right), the above row samples (red) are copied to the bottom row of the current block (blue) for the prediction. That is, some reference samples in both the horizontal and vertical reference sets are moved to directly utilize Equation (1) in the prediction.

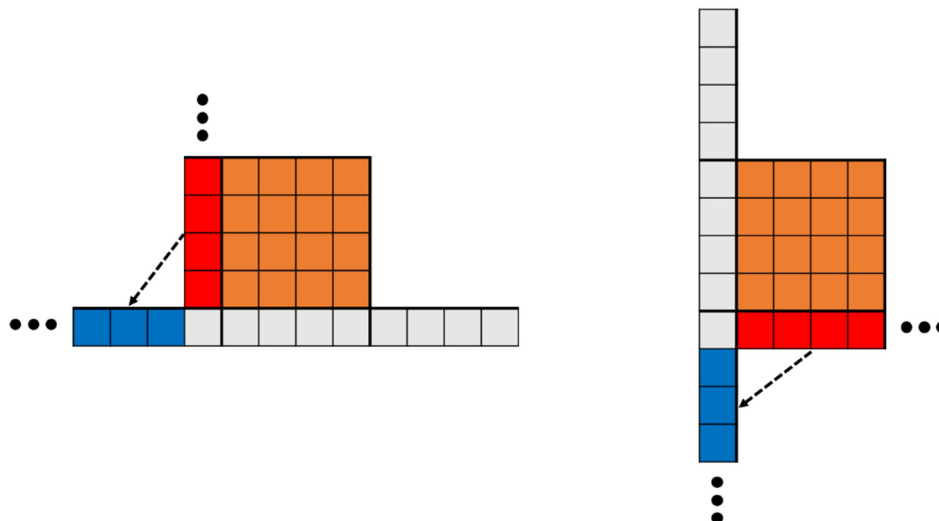


**Figure 6.** Translation of the  $xy$ -coordinate for the horizontal reference set (**left**) and the vertical reference set (**right**).

Figure 7 shows the extrapolation processes for the horizontal directional modes and the vertical directional modes in the horizontal reference set. It is very similar to the extrapolation in the original reference set in Figure 2. Because the origin of the  $xy$ -coordinate is translated into the above-right, in the case of the vertical directional modes, the reference samples in the right column (red) are projected onto the above row (blue). When the intra modes are the horizontal directional modes, the samples in the above row (red) are projected onto the right column (blue). Figure 8 shows the extrapolation in the vertical reference set. As the origin of the  $xy$ -coordinate is located in the bottom-left, the reference samples in the left column (red) are projected onto the bottom row (blue) in the extrapolation for the vertical directional modes. In the case of the horizontal directional modes, the reference samples in the bottom row (red) are projected onto the left column (blue).



**Figure 7.** Extrapolation for the vertical directional modes (**left**) and the horizontal directional modes (**right**) in the horizontal reference set.



**Figure 8.** Extrapolation for the vertical directional modes (**left**) and the horizontal directional modes (**right**) in the vertical reference set.

The intra prediction process is performed for every PU. Therefore, the index bits are required to indicate the type of the reference set used in each PU. Table 3 presents the codewords for the multiple reference sets. When the proposed method employs the original reference set, a codeword of ‘0’ is signaled to a decoder. In the case of the horizontal reference set, a codeword of ‘10’ is assigned. The vertical reference set uses a codeword of ‘11’. More reference sets can be defined by utilizing more row and column samples of the neighboring blocks. However, because additional bits are required for every PU, its overhead may be greater than the improvement. Therefore, the proposed method only uses the three different reference sets.

**Table 3.** Codeword for the multiple reference sets.

Reference Set	Codeword
Original reference set	0
Horizontal reference set	10
Vertical reference set	11



#### 4. Results

The proposed method was implemented in the reference software, HM 12.1. Five test sequences of a  $1920 \times 1080$  size, which are Kimono, ParkScene, Cactus, BasketballDrive, and BQTerrace, were used. Test sequence information, such as the total number of tested frames, frame rate (fps), and bit depth, is summarized in Table 4. All the test sequences were compressed as intra frames with four different quantization parameters (QPs) of 22, 27, 32, and 37, and other coding options followed the common test conditions (CTC) of the JCT-VC [22]. The coding performance is measured by the Bjontegaard Delta bit rate [23] in percentage. In the results, Y denotes the luminance component, and U and V indicate the chrominance components.

**Table 4.** Test sequence information [22].

Sequence	Total Frames	Frame Rate	Bit Depth
Kimono	240	24 fps	8
ParkScene	240	24 fps	8
Cactus	500	50 fps	8
BasketballDrive	600	60 fps	8
BQTerrace	500	50 fps	8

For evaluation of the coding performance, the proposed method, which consists of the adaptive coding order and the multiple reference sets, was compared to the sampling-based intra prediction method [11,12]. As the performance of these sampling-based prediction methods was reported on H.264/AVC, I implemented it on HEVC and re-optimized most of the parameters. In addition, it performs the frame-level sampling in the horizontal and vertical directions, but I modified it to the block-level to improve its coding performance. Table 5 presents the intra coding performance of the sampling-based prediction method and the proposed method. As shown in Table 5, the proposed method reduces the bit rates by 1.02% (Y), 1.27% (U), and 1.56% (V) on average, but the sampling-based prediction method reduces the bit rates by 0.37% (Y), 0.74% (U), and 0.76% (V). Because the proposed method considers the various coding orders and reference samples in the prediction, a consistent improvement could be achieved in all the test sequences. However, both methods require two additional encoding loops, such as the horizontally and vertically sampling based encoding in the sampling-based intra prediction method and above-right to bottom-left and bottom-left to above-right encoding in the proposed method. Therefore, their encoding complexity is about two or three times higher than the original method. Unfortunately, the complexity of the HEVC encoder is already high because of the high performance coding tools [24], so it is not desirable to add high computational tools in HEVC. In order to reduce the encoding complexity of the proposed method, it is possible to apply some fast intra prediction methods, such as traditional fast intra prediction methods [25,26] and fast algorithms using deep learning [27,28].

**Table 5.** Coding performance of (a) the sampling-based intra prediction method [11,12] and (b) the proposed method.

Sequence	(a)			(b)		
	Y	U	V	Y	U	V
Kimono	0.06	−0.45	−0.31	0.69	0.92	0.89
ParkScene	0.01	0.29	0.24	0.83	0.84	1.02
Cactus	0.11	0.08	0.15	0.97	0.68	1.22
BasketballDrive	1.43	3.50	3.46	1.26	2.41	2.59
BQTerrace	0.25	0.25	0.26	1.32	1.51	2.10
Avg.	0.37	0.74	0.76	1.02	1.27	1.56

Table 6 represents the detailed coding performance of the adaptive coding order. When the proposed method additionally applied the coding order of above-right to bottom-left, the performance is improved by 0.23% (Y), 0.43% (U), and 0.62% (V) on average. When the coding orders of above-right to bottom-left and bottom-left to above-right were used together, the proposed method achieves the bit rate saving of 0.42% (Y), 0.84% (U), and 1.05% (V) on average. However, when the coding order of bottom-right to above-left is further used, there is only negligible improvement due to the overhead of the index bits. Therefore, the proposed method only uses the coding orders of above-right to bottom-left and bottom-left to above-right in addition to the original coding order of above-left to bottom-right.

**Table 6.** Coding performance of the additional coding orders of (a) above-right to bottom-left, (b) above-right to bottom-left and bottom-right to above-right, and (c) above-right to bottom-left, bottom-right to above-right, and bottom-right to above-left.

Sequence	(a)			(b)			(c)		
	Y	U	V	Y	U	V	Y	U	V
Kimono	0.16	0.27	0.23	0.26	0.44	0.51	0.26	0.41	0.47
ParkScene	0.13	0.23	0.46	0.21	0.40	0.59	0.22	0.36	0.55
Cactus	0.17	0.23	0.45	0.34	0.50	0.72	0.36	0.45	0.68
BasketballDrive	0.21	0.78	0.84	0.54	1.66	1.62	0.63	1.62	1.62
BQTerrace	0.47	0.67	1.09	0.74	1.21	1.80	0.73	1.21	1.79
Avg.	0.23	0.43	0.62	0.42	0.84	1.05	0.44	0.81	1.02

Table 7 shows the detailed performance of each reference set. The results demonstrate that the additional use of the horizontal reference set provides the improvement of 0.28% (Y), 0.29% (U), and 0.24% (V) on average. In the case of the vertical reference set, the bit rate reduction of 0.41% (Y), 0.36% (U), and 0.42% (V) is achieved on average. When all the reference sets were employed together, the bit rate can be reduced by 0.64% (Y), 0.51% (U), and 0.58% (V) on average. This means that each reference set is effective, and the intra prediction based on the various reference samples can help to improve the coding performance. In addition, it can be seen that the multiple reference sets achieve a slightly better performance than the adaptive coding order, in terms of the luminance component. It can be also observed that the performance of the adaptive coding and the multiple reference sets are almost additive.

**Table 7.** Coding performance of (a) the original and horizontal reference sets, (b) the original and vertical reference sets, and (c) the original, horizontal, and vertical reference sets.

Sequence	(a)			(b)			(c)		
	Y	U	V	Y	U	V	Y	U	V
Kimono	0.22	0.32	0.25	0.28	0.26	0.17	0.47	0.43	0.39
ParkScene	0.30	0.37	0.23	0.36	0.23	0.22	0.64	0.55	0.47
Cactus	0.32	0.12	0.27	0.35	0.11	0.36	0.64	0.21	0.61
BasketballDrive	0.27	0.63	0.54	0.56	0.89	0.95	0.75	1.06	1.06
BQTerrace	0.29	0.01	−0.07	0.49	0.32	0.41	0.72	0.29	0.40
Avg.	0.28	0.29	0.24	0.41	0.36	0.42	0.64	0.51	0.58

## 5. Conclusions

This paper proposed the enhanced intra prediction method. The proposed method consists of the adaptive coding order and multiple reference sets. The adaptive coding order method finds the best coding order by investigating not only the original coding order of above-left to bottom-right but also the additional coding orders of above-right to bottom-left and bottom-left to above-right. The multiple reference sets improve the prediction performance by generating the various predictors.

The experimental results demonstrated that the proposed method is more efficient than the conventional method. However, the proposed method requires relatively high encoding complexity, so its impact is very limited in real-time applications of HEVC. Therefore, in future work, I will develop a fast intra prediction algorithm that can optimize the proposed method.

**Funding:** This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. 2018R1C1B5086072).

**Conflicts of Interest:** The author declares no conflicts of interest.

## References

1. Wiegand, T.; Sullivan, G.J.; Bjntegaard, G.; Luthra, A. Overview of the H.264/AVC video coding standard. *IEEE Trans. Circuit Syst. Video Technol.* **2003**, *13*, 560–576. [[CrossRef](#)]
2. Sullivan, G.J.; Ohm, J.-R.; Han, W.-J.; Wiegand, T. Overview of the high efficiency video coding (HEVC) standard. *IEEE Trans. Circuit Syst. Video Technol.* **2012**, *22*, 1649–1668. [[CrossRef](#)]
3. *High Efficiency Video Coding*; Doc. Recommendation ITU-T H.265; ITU-T: Geneva, Switzerland, 2015.
4. Sullivan, S.J.; Boyce, J.M.; Chen, Y.; Ohm, J.-R.; Segall, C.A.; Vetro, A. Standardized Extensions of High Efficiency Video Coding (HEVC). *IEEE J. Sel. Top. Signal Process.* **2013**, *7*, 1001–1016. [[CrossRef](#)]
5. Ohm, J.-R.; Sullivan, G.J.; Schwarz, H.; Tan, T.K.; Wiegand, T. Comparison of the coding efficiency of video coding standards including high efficiency video coding (HEVC). *IEEE Trans. Circuit Syst. Video Technol.* **2012**, *22*, 1669–1684. [[CrossRef](#)]
6. Tan, T.K.; Boon, C.S.; Suzuki, Y. Intra Prediction by Template Matching. In Proceedings of the IEEE International Conference on Image Processing, Atlanta, GA, USA, 8–11 October 2006.
7. Ye, Y.; Karczewicz, M.; Improved, H. 264 Intra Coding Based on Bi-directional Intra Prediction, Directional Transform, and Adaptive Coefficient Scanning. In Proceedings of the IEEE International Conference on Image Processing, San Diego, CA, USA, 12–15 October 2008.
8. Matsuo, S.; Takamura, S.; Jozawa, H. Improved Intra Angular Prediction by DCT-based Interpolation Filter. In Proceedings of the European Signal Processing Conference, Bucharest, Romania, 27–31 August 2012.
9. Ye, Y.; Karczewicz, M. *Improved Intra Coding*; Doc. VCEG-AG11; ITU-T Q.6/SG11: Shenzhen, China, 2007.
10. Kim, D.-Y.; Han, K.-H.; Lee, Y.-L. Adaptive Single-Multiple Prediction for H.264/AVC Intra Coding. *IEEE Trans. Circuit Syst. Video Technol.* **2010**, *20*, 610–615. [[CrossRef](#)]
11. Piao, Y.; Park, H. Adaptive Interpolation-Based Divide-and-Predict Intra Coding for H.264/AVC. *IEEE Trans. Circuit Syst. Video Technol.* **2010**, *20*, 1915–1921. [[CrossRef](#)]
12. Lee, J.Y.; Wey, H.-C.; Park, D.-S. A high performance and low complexity sampling-based intra coding method for H.264/AVC. *J. Vis. Commun. Image Represent.* **2012**, *23*, 1179–1188. [[CrossRef](#)]
13. Ma, S.; Zhang, X.; Jia, C.; Zhao, Z.; Wang, S.; Wang, S. Image and Video Compression with Neural Networks: A Review. *IEEE Trans. Circuit Syst. Video Technol.* **2019**, accepted. [[CrossRef](#)]
14. Wang, L.; Fiandrotti, A.; Purica, A.; Valenzise, G.; Cagnazzo, M. Enhancing HEVC Spatial Prediction by Context-based Learning. In Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, Brighton, UK, 12–17 May 2019.
15. Lainema, J.; Bossen, F.; Han, W.-J.; Min, J.; Ugur, K. Intra Coding of the HEVC Standard. *IEEE Trans. Circuit Syst. Video Technol.* **2012**, *22*, 1792–1801. [[CrossRef](#)]
16. Cho, S.; Kim, M. Fast CU Splitting and Pruning for Suboptimal CU Partitioning in HEVC Intra Coding. *IEEE Trans. Circuit Syst. Video Technol.* **2013**, *23*, 1555–1564. [[CrossRef](#)]
17. Saxena, A.; Fernandes, F.C. DCT/DST-based Transform Coding for Intra Prediction in Image/Video Coding. *IEEE Trans. Image Process.* **2013**, *22*, 3974–3981. [[CrossRef](#)] [[PubMed](#)]
18. Ortega, A.; Ranchandra, K. Rate-distortion methods for image and video compression. *IEEE Signal Process. Mag.* **1998**, *15*, 23–50. [[CrossRef](#)]
19. Matsuda, I.; Ohtake, Y.; Mochizuki, S.; Fukai, H.; Itoh, S. Interpolative intra prediction by adapting processing order in block-based image coding. In Proceedings of the IEEE International Conference on Image Processing, Melbourne, Australia, 15–18 September 2013.

20. Zheng, A.; Au, O.C.; Yuan, Y.; Yang, H.; Pang, J.; Ling, Y. Intra prediction with adaptive CU processing order in HEVC. In Proceedings of the IEEE International Conference on Image Processing, Paris, France, 27–30 October 2014.
21. Zheng, A.; Yuan, Y.; Zhou, J.; Guo, Y.; Yang, H.; Au, O.C. Adaptive Block Coding Order for Intra Prediction in HEVC. *IEEE Trans. Circuit Syst. Video Technol.* **2016**, *26*, 2152–2158. [[CrossRef](#)]
22. Bossen, F. *Common Test Conditions and Software Reference Configurations*. Doc. JCTVC-L110; ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11: Geneva, Switzerland, 2013.
23. Bjontegaard, G. *Calculation of Average PSNR Differences between Rd-Curves*; Doc. VCEG-M33; ITU-T SG 16/Q.6: Austin, Texas, 2001.
24. Correa, G.; Assuncao, P.; Agostini, L.; Cruz, L. Performance and Computational Complexity Assessment of High-Efficiency Video Encoders. *IEEE Trans. Circuit Syst. Video Technol.* **2012**, *22*, 1899–1909. [[CrossRef](#)]
25. BenHajyoussef, A.; Ezzedine, T.; Bouallegue, A. Gradient-based pre-processing for intra prediction in High Efficiency Video Coding. *Eurasip J. Image Video Process.* **2017**, *2017*, 9. [[CrossRef](#)]
26. Jamali, M.; Coulombe, S. Fast HEVC Intra Mode Decision Based on RDO Cost Prediction. *IEEE Trans. Broadcasting* **2019**, *65*, 109–122. [[CrossRef](#)]
27. Xu, M.; Li, T.; Wang, Z.; Deng, X.; Yang, R.; Guan, Z. Reducing Complexity of HEVC: A Deep Learning Approach. *IEEE Trans Image Process.* **2018**, *27*, 5044–5059. [[CrossRef](#)] [[PubMed](#)]
28. Li, N.; Zhang, Y.; Zhu, L.; Luo, W.; Kwong, S. Reinforcement learning based coding unit early termination algorithm for high efficiency video coding. *J. Vis. Commun. Image Represent.* **2019**, *60*, 276–286. [[CrossRef](#)]



© 2019 by the author. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).