Improved Sample Adaptive Offset for HEVC

Hong Zhang, Oscar C. Au, Yongfang Shi, Wenjing Zhu, Vinit Jakhetiya, Luheng Jia
Department of Electronic and Computer Engineering
Hong Kong University of Science and Technology
Clear Water Bay, Kowloon, Hong Kong
Email: {hzhangad, eeau, yshiab, wzhuaa, vjakhetiya, ljia}@ust.hk

***Abstract*—High-Efficiency Video Coding (HEVC) is the newest video coding standard which can significantly reduce the bit rate by 50% compared with existing standards. One new efficient tool is sample adaptive offset (SAO), which classifies reconstructed samples into different categories, and reduces the distortion by adding an offset to samples of each category. Two SAO types are adopted in HEVC: edge offset (EO) and band offset (BO). Four 1-D directional edge patterns are used in edge offset type, and only one is selected for each CTB. However, single directional pattern cannot remove artifacts effectively for the CTBs, which contain edges in different directions. Therefore, we analyze the performance of each edge pattern applied on this kind of CTB, and propose to take advantage of existing edge classes and combine some of the them as a new edge offset class, which can adapt to multiple edge directions. All the combinations are tested, and the results show that for Low Delay P condition, they can achieve 0.2% to 0.5% bit rate reduction.**

I. Introduction

Recently, the next generation video coding standard HEVC is being established by ITU-T VCEG and ISO/ICE MPEG organizations. The main goal of HEVC [1] is to reduce 50% bit rate in comparison with H.264/AVC [2], under the same perceptual video quality. In HEVC, the intra/inter prediction, transform and quantization are still processed based on block. Some artifacts existing in the previous coding standards also occur in HEVC, such as blocking artifacts, ringing artifacts and blurring artifacts.

In HEVC, two in-loop filters are adopted to remove these artifacts. The deblocking filter (DBF) is applied to the boundaries of reconstructed block to reduce the blocking artifacts. In addition to DBF, a new technique called sample adaptive offset (SAO) filter is applied adaptively to the reconstructed samples after DBF. SAO is helpful to reduce the ringing arifacts, which mainly come from the quantization errors of transform coefficients. Many proposals about SAO are studied in JCT-VC meetings.

In JCTVC-A124 [3], two tools extreme-value correction (EXC) and band-correction (BDC) are proposed to reduce the distortion between the reconstructed pictures and original ones. The key idea of these tools is to classify the reconstructed samples into different categories, and find optimal offsets for each category to minimize the mean distortion. The classification of EXC is based on the relationship of current reconstructed sample and neighbor ones. BDC uses pixel intensity to classify the reconstructed samples into different bands. Offsets are encoded into bit stream, and the classification is done at both encoder and decoder side, which can save bits for categories classification. However, the decoding time is too high to balance the coding gain.

JCTVC-C147 [4] and JCTVC-D122 [5] are proposed to combine sequential stages EXC and BDC into one stage and allow the encoder to select only one mode for each region adaptively. However, the processing of each sample is still too complex. JCTVC-D122 [5] and JCTVC-E049 [6] further simplify the sample classification and rename this tool as sample adaptive offset (SAO). Based on the proposal JCTVC-E049 [6], people improve SAO by applying it on coding tree unit (CTU) level, including one luma coding tree block (CTB) and two chroma CTBs.

The purpose of SAO is mainly to remove ringing artifacts, and reduce the mean distortion between reconstructed and original pictures. For current adopted SAO in HEVC, four 1-D edge patterns are used in edge offset type, however, if one CTB contains multiple edges, four 1-D EO classes are not efficient enough to remove artifacts in all directions. One solution of this problem is to combine different 1-D edge patterns adaptively for multiple edge types in one region.

In this paper, we apply four 1-D EO classes on a block with edges in different directions. The performance of each EO class is presented and analyzed. Based on the observation and analysis, we propose to implement different combinations of four 1-D edge patterns, and adaptively select some classes from all 11 combinations and add them into the edge offset classes. This method can help to remove artifacts along and around edges in different angles in one CTB.

The rest of the paper is organized as follow: In Section II, the technique details of SAO in HEVC are introduced. Section III analyzes the performance of each EO class and presents the proposed adaptive combination of edge patterns. The experiment results and further discussion are provided in Section IV. At last, a conclusion is draw in V section.

II. Sample Adaptive Offset in HEVC

In HEVC, SAO is an in-loop filter and located after deblocking filter as depicted in Fig.1. The idea of SAO is to compensate reconstructed samples by adding an offset to each pixel, so that the distortion between reconstructed picture and original one can be reduced. The key problem of this method is how to classify the reconstructed samples and how to select the offsets for each category. In current SAO, two different method: band offset (BO) and edge offset (EO) are selected.

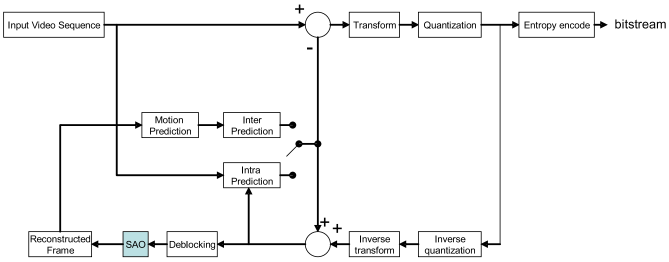

Fig. 1. Hybrid Video Encoder.

A. *Edge Offset*

Four 1-D edge classes are used for edge offset in SAO : horizontal, vertical, 135° and 45° diagonal, as shown in Fig. 2. For each class, the samples in one CTB are classified into five categories, based on the relationship of current sample c and two neighbor samples a and b. Table I lists the conditions for five categories, where category 0 means nothing is done for current sample, and for category 1 to 4, an optimal offset is computed and added to current sample c. From the statistical analysis [7]-[9], they discover that the majority of offsets for category 1 and 2 are positive, and for category 3 and 4 are negative, which indicates EO tries to reduce the distance between current sample and neighbor ones. This observation also helps to save the bit to encode sign of offsets. For each CTB, encoder tries every EO class and selects the best one based on rate-distortion performance. In order to reduce side information, the classification for samples are done at both encoder and decoder, only the class type and absolute offsets for every CTB are transmitted to decoder.

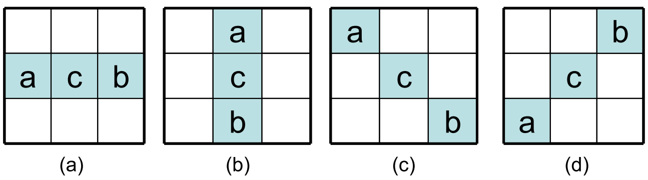

Fig. 2. Four 1-D directional patterns used in SAO for EO sample classification. (a) EO class 0: horizontal, (b) EO class 1: vertical, (c) EO class 2: 135°diagonal, (d) EO class 3: 45°diagonal

TABLE I
SAMPLE CATEGORY CLASSIFICATION FOR EACH EO CLASS

Category	Condition
0	None of the below
1	$c < a \,\&\&\, c < b$
2	$(c < a \,\&\&\, c == b) \,\|\, (c == a \,\&\&\, c < b)$
3	$(c > a \,\&\&\, c == b) \,\|\, (c == a \,\&\&\, c > b)$
4	$c > a \,\&\&\, c > b$

B. *Band Offset*

In band offset mode, sample value range is equally divided into 32 bands. If bit depth of sample is 8-bit (value range: 0-255), then the width of each band is 8 and sample values from $8k$ to $8k + 7$ (k : 0...31) belong to the kth band [10]. In one CTB, samples are classified into corresponding band, and the average difference of reconstructed samples and original ones is assigned as offset for each band. Different from EO, the offsets of BO can be positive or negative. Not all the bands are used, based on rate-distortion performance, only four consecutive bands are selected to compensate reconstructed samples. This is because that the intensity range is limited in one CTB, and tend to be concentrated in only a few of the bands. Type index, four offsets and the start point of selected bands are signaled to the decoder side.

SAO is a nonlinear filtering operation which compensates reconstructed samples and reduce the distortion with original samples. Band offset can help to reduce the artifacts in relatively smooth region, and edge offset can reduce the artifacts around the edges in the four directions. However, four single directions are not enough to deal with complicated situations, when there are different diagonal edges in one CTU. Therefore, we propose to take advantage of four 1-D patterns, and combine some of them to adapt to different situations.

III. Improved Sample Adaptive Offset

SAO is designed to reduce the ringing artifacts. The main cause of ringing artifacts is that signal is band-limited in frequency domain (cut high frequencies). In terms of time domain, the impulse response of the ideal low-pass filter in frequency domain is the sinc function, in Fig. 3. The ripples of the sinc function will cause the ringing artifacts.

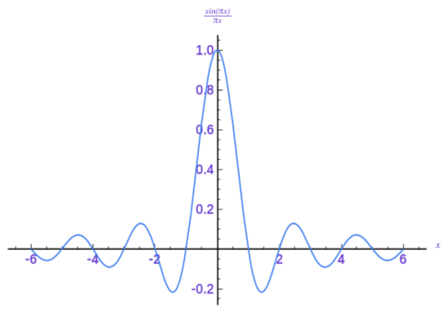

Fig. 3. Sinc function, impulse response for ideal low-pass filter

In HEVC, ringing artifacts mainly come from the quantization errors of transform coefficients. The edge offset mode of SAO can help to remove the ringing artifacts along and around edges. Four 1-D directional classes are designed and only one class is selected and applied to a CTB. When the edge directions in a CTB are similar, one edge class can perform well to reduce errors in this direction. However, if there are more than one edge directions in a CTB, single direction edge class is not enough to remove artifacts in all directions. One method for this problem is to take advantage of the existing edge classes, and combine some of them to adapt to different situations.

For example, in Fig. 4., (a) shows a 16×16 block X containing two edges: one horizontal and one vertical, illustrated in color blue. We apply 2-D DCT transform and quantization process to block X, in this example, the quantization step $Qstep$ is set 4. After inverse transform and inverse quantization, we get the reconstructed block X', shown in Fig. 4. (b). Quantization errors are indicated in different colors: red means that error of current pixel is negative, and green means positive error. We can see that the ringing artifacts occur along and around edges. SAO is designed to remove these artifacts. Four edge offset classes in HEVC are applied to the reconstructed block X', the results are shown in Fig. 4. (c) - (f). The performance using SAO is depicted in different colors: yellow means that the quantization errors are reduced, purple means errors are enlarged, and pink means absolute errors are same. No change for pixels in blue and green, which have the same meaning as before. Since there are no neighbour blocks in this example, the boundary pixels are not processed.

From the results, we can see that when horizontal edge class is applied to reconstructed block X', most errors along the horizontal edge and some errors around the edges are reduced, depicted in Fig. 4 (c). However, ringing artifacts sill exist along vertical edge and around edges. Similarly, in (d), the vertical edge is improved better. Diagonal edge classes cannot remove ringing artifacts along edges, as shown in (e) and (f), pixel values oscillate in two edges. So, this example shows that each edge offset class can remove artifacts along the corresponding edge direction, and some errors around the edges. However, single direction edge class cannot remove artifacts in all directions.

Based on the observation and analysis, we propose to combine different edge classes. For this example, since edge class 0 and 1 can remove artifacts along horizontal and vertical directions separately, we try to combine them together. Edge class 0 and 1 are applied to reconstructed block sequentially, using the same set of offsets. The results are shown in Fig. 4. (g), nearly no ringing artifacts along the edges, and most errors around are removed. The sum squared distortions (SSD) between reconstructed blocks and original one are shown in Table II. Proposed combination class performs the best.

TABLE II
SUM SQUARED DISTORTIONS FOR EACH EDGE CLASS

Edge Offset Class	Sum Squared Distortion
Without SAO	128
Class 0	61
Class 1	61
Class 2	70
Class 3	73
Class 0 & 1	50

This example helps to explain the shortcomings of using only one edge direction, and the benefits of our proposed method. In general, we propose to combine different edge classes together to form a new edge class. For each CTB, only one class is selected based on rate-distortion performance from the four existing edge classes and our proposed classes.

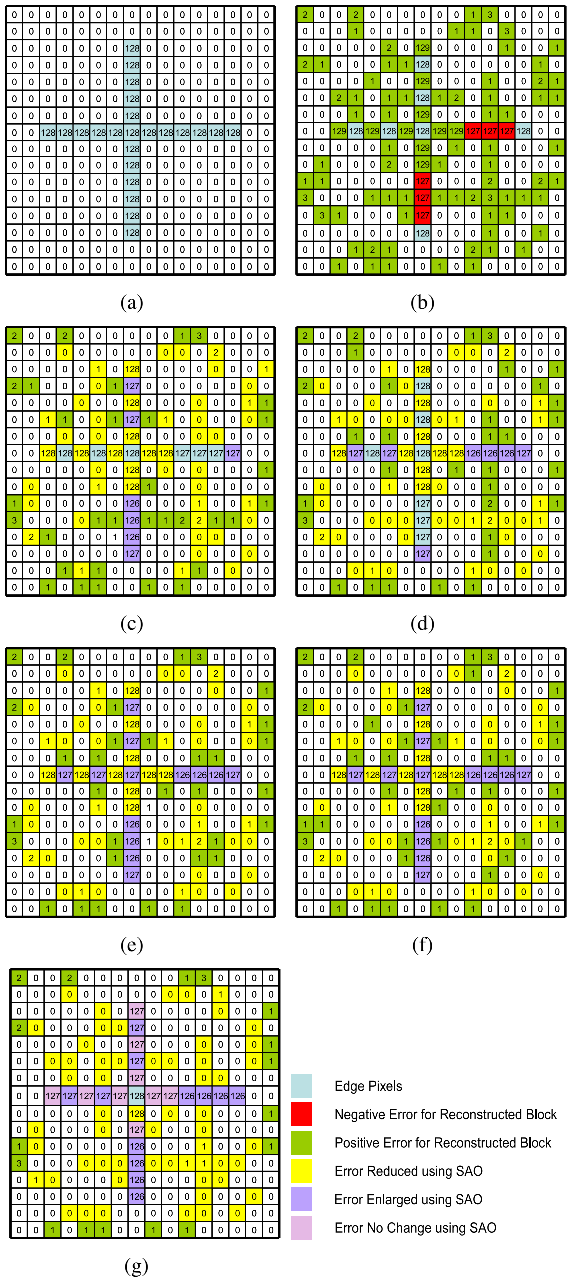

Fig. 4. Test block and reconstructed blocks. (a) Test block with size 16×16, (b) Reconstructed block, (c)-(f) Reconstructed block using edge offset class 0, 1, 2, and 3, (g) Reconstructed block using combined edge offset class (0 and 1)

IV. EXPERIMENT AND RESULTS

There are four edge offset classes, so $C_4^2 + C_4^3 + C_4^4 = 11$ combinations can be implemented. If all the combinations are added into edge offset classes, the computation complexity will increase greatly. Besides, more bits are needed to encode the index of each class. Therefore, we try to analyze the performance of each combination for all the test sequences. The reference software is HM9.0 of HEVC, and main conditions are tested. The results of Low Delay P (LP) condition are showed in Table III. The numbers in first row mean the

TABLE III
PERFORMANCE OF DIFFERENT EDGE OFFSET COMBINATIONS

Anchor: HM 9.0 Low Delay P (LP)		01	02	03	12	13	23	012	023	013	123	0123
		Y BD-rate(%)	Y BD-rate(%)	Y BD-rate(%)	Y BD-rate(%)	Y BD-rate(%)	Y BD-rate(%)	Y BD-rate(%)	Y BD-rate(%)	Y BD-rate(%)	Y BD-rate(%)	Y BD-rate(%)
ClassA 2560x1600	Traffic	-0.3	-0.2	-0.2	-0.2	-0.2	-0.2	-0.2	-0.1	-0.2	-0.2	-0.1
	People on Street	-0.8	-0.2	-0.3	-0.3	-0.4	-0.3	-0.4	-0.2	-0.5	-0.3	-0.1
	Nebuta	-1.6	-1	-1.1	-0.8	-0.8	-1.3	-1.5	-1.3	-1.6	-1.1	-0.7
	SteamLocomotive	-0.6	-0.5	-0.4	-0.2	-0.3	-0.4	-0.4	-0.4	-0.3	-0.3	-0.1
ClassB 1080p	Kimono	-0.4	-0.5	-0.4	-0.2	0	-0.5	-0.3	-0.6	-0.3	-0.4	-0.5
	ParkScene	-0.3	-0.2	-0.3	-0.2	-0.1	-0.2	-0.2	-0.2	-0.3	-0.1	0
	Cactus	-0.5	-0.2	-0.2	-0.2	-0.3	-0.3	-0.2	-0.2	-0.2	-0.1	-0.2
	BasketballDrive	-0.5	-0.4	-0.4	-0.4	-0.3	-0.3	-0.3	-0.2	-0.3	-0.2	-0.2
	BQTerrace	-0.9	-0.3	-0.5	-0.3	-0.4	-0.2	-0.2	-0.4	-0.5	-0.2	-0.1
ClassC WVGA	BasketballDrill	-0.1	-0.1	-0.1	-0.1	-0.1	-0.1	-0.2	0	-0.1	-0.1	-0.2
	BQMall	-0.4	-0.1	0.1	-0.1	0	-0.1	0.1	0	0	-0.1	0
	PartyScene	-0.2	-0.2	-0.1	-0.3	-0.1	-0.2	-0.2	-0.2	-0.1	-0.2	-0.2
	RaceHorses	-0.5	0	-0.1	-0.1	-0.1	-0.1	-0.1	-0.1	-0.1	-0.2	-0.1
ClassD WQVGA	BasketballPass	-0.4	-0.1	-0.2	-0.4	-0.1	-0.2	-0.2	-0.3	-0.3	-0.2	-0.3
	BQSquare	-0.1	0.2	0.1	0	0.1	0.2	0.1	-0.1	0.2	0.1	0.1
	BlowingBubbles	0	-0.1	0.1	0.1	-0.1	0	0	0	-0.2	0.2	0
	RaceHorses	-0.2	-0.2	-0.4	-0.3	-0.2	-0.2	-0.1	-0.1	0	-0.1	-0.1
ClassE 720P	FourPeople	-0.4	-0.1	-0.1	-0.1	-0.1	-0.1	0.1	0	0	-0.1	0.1
	Johnny	-0.3	-0.4	-0.1	0.1	-0.4	-0.1	-0.2	-0.4	-0.6	-0.3	-0.2
	KristenAndSara	-0.5	-0.1	0.1	-0.1	0	-0.1	-0.2	0	0	-0.4	-0.4
ClassF	BasketballDrillText	-0.3	0	0	0	-0.1	-0.2	0	0.1	-0.2	-0.1	-0.1
	ChinaSpeed	-0.6	-0.2	-0.2	-0.2	-0.2	-0.3	-0.2	-0.3	-0.2	-0.1	0
	SlideEditing	-0.7	-0.2	-0.1	-0.2	-0.3	-0.5	0	-0.1	-0.3	-0.3	-0.2
	SlideShow	-0.7	-0.2	-0.2	-0.2	-0.3	-0.1	-0.1	-0.3	-0.6	-0.3	-0.2
Summary	All	-0.5	-0.2	-0.2	-0.2	-0.2	-0.2	-0.2	-0.2	-0.3	-0.2	-0.2
	Encoding time(%)	102	101	101	102	102	102	103	102	103	103	105
	Decoding time(%)	101	100	100	100	100	101	100	100	101	101	102

combinations of different edge classes. For example, 01 means we combine edge offset class 0 and 1, and 0123 means the combination of edge class 0, 1, 2, and 3, and so on. The numbers in each column are the BD-rate reduction compared with anchor. The anchor is HM9.0 reference software with SAO, and the tested one is adding the corresponding combination edge class into original edge offset classes. The results show that for most of the test sequences, adding combination class 01 gives better performance, depicted in yellow. Maybe this is because horizontal and vertical edges appear more likely, so that most sequences contain them. For other sequences, such as "BlowingBubbles" and "Johnny", adding combination 013 performs better. Generally, considering the computation complexity, if the features of the encoded sequences are unknown, adding 01 combination into edge offset is more efficient. However, if we know the sequence features, we can select the best combination and add it into edge classes.

V. CONCLUSIONS

In this paper, we analyze the performance of each edge offset class, and find that single directional edge pattern is not efficient enough to remove artifacts for the CTBs, which contain multiple edges in different directions. According to the observation and analysis, we propose to combine different edge classes as one new edge class, which can remove errors in different directions. The performance of each combination is presented, and the average BD-rate reduction for low dealy P condition is from 0.2% to 0.5%. Specially, for most of the test sequences, combination of edge class 0 and 1 gives better performance. People also can select the best combination based on the features of encoded sequences.

REFERENCES

[1] G. J. Sullivan, J. Ohm, Woo-Jin Han, T. Wiegand, and T. Wiegand, "Overview of the high efficiency video coding (HEVC) standard," *Circuits and Systems for Video Technology, IEEE Transactions on,* vol. 22, no. 12, pp. 1649-1668, 2012.

[2] Joint Video Team of ITU-T VCEG and ISO/IEC MPEG, *ITU-T Rec. H.264, ISO/IEC 14496-10 AVC,* 2005.

[3] K. McCann, W.-J. Han, I.-K. Kim, J.-H. Min, E. Alshina, A. Alshin, T. Lee, J. Chen, V. Seregin, Y.-M. Hong S. Lee, M.-S. Cheon, and N. Shlyakhov, *Samsungs Response to the Call for Proposals on Video Compression Technolog,* document JCTVC-A124, Apr. 2010.

[4] C.-M. Fu, C.-Y. Chen, Y.-W. Huang, and S. Lei, *TE10 Subtest 3: Quadtree-Based Adaptive Offset,* document JCTVC-C147, Oct. 2010.

[5] C.-M. Fu, C.-Y. Chen, Y.-W. Huang, and S. Lei, *CE8 Subset 3: Picture Quadtree Adaptive Offset,* document JCTVC-D122, Jan. 2011.

[6] C.-M. Fu, C.-Y. Chen, C.-Y. Tsai, Y.-W. Huang, and S. Lei, *CE13: Sample Adaptive Offset with LCUIndependent Decoding,* document JCTVC-E049, Mar.2011.

[7] C.-M. Fu, Y.-W. Huang, S. Lei, I. S. Chong, and M. Karczewic, *Non-CE8: Offset Coding in SAO,* document JCTVC-G222, Nov. 2011.

[8] W.-S. Kim and D.-K. Kwon, *Non-CE8: Method of Visual Coding Artifact Removal for SAO,* document JCTVCG680, Nov. 2011.

[9] W.-S. Kim and D.-K. Kwon, *CE8 Subset c: Necessity of Sign Bits for SAO Offsets,* document JCTVC-H0434, Feb. 2012.

[10] C.-M. Fu, E. Alshina, A. Alshin, Y.-W. Huang, C.-Y. Chen, C.-Y. Tsai, C.-W. Hsu, S.-M. Lei, J.-H. Park, and W.-J. Han, "Sample adaptive offset in the HEVC standard," *Circuits and Systems for Video Technology, IEEE Transactions on,* vol. 22, no. 12, pp. 1755-1764, 2012.