NOISE REDUCTION FOR SCREEN CONTENT CODING BASED ON LOCAL HISTOGRAM

Kazuyuki Miyazawa *Akira Minezawa* *Shun-ichi Sekiguchi*

Mitsubishi Electric Corporation, 5-1-1, Ofuna, Kamakura, Japan

ABSTRACT

This paper proposes a noise reduction method for screen content coding using HEVC. The proposed method focuses on that the histograms of pixel values for screen content have distinct and sparse peaks respectively. This property is often degraded by the coding distortion, which arises some small peaks around the original peak. This paper presents a method for removing the small peaks derived from the noise, and its implementation to HEVC as an extension of SAO. Experimental evaluation shows that the proposed noise reduction method significantly improves the visual quality of decoded screen content with bit-rate reduction.

***Index Terms*—** Screen content coding, noise reduction, local histogram, HEVC, sample adaptive offset

1. INTRODUCTION

A new video coding standard, H.265/HEVC (High Efficiency Video Coding), developed by JCT-VC (Joint Collaborative Team on Video Coding) has been approved in January 2013 [1]. HEVC can reduce bit-rate requirements by half with comparable image quality compared to AVC/H.264, and will play an important role for transmitting high resolution videos such as 4K and 8K through broadband networks, while also making HD video streaming available even on bandwidth-constrained mobile networks.

Although HEVC has been developed mainly for the purpose of effective coding of natural video captured by cameras, the demand for *screen content coding* is greatly increasing due to rapidly growing video applications in areas such as wireless display and cloud computing [2]. Screen content coding can be defined as the coding of video containing a significant proportion of rendered graphics, text, or animation rather than, or in addition to, camera-captured video scenes [3]. HEVC still has a room for improvement in efficiency of screen content coding since it does not exploit many properties of screen content which are quite different from those of camera-captured content. Against this background, JCT-VC has started the work for extending HEVC for screen content coding, and published the first working draft specification in July 2014 [4].

Through the ongoing standardization process, various techniques have been proposed, and some of them provide a substantial gain for screen content coding. The most exploited feature of screen content is that patterns in screen content such as characters, icons and lines can repeat within a picture. So we can effectively remove the redundancy by finding a block (or line) similar to the one it is encoding on the already decoded region of the current picture for prediction [5], [6]. Another important property is that screen content has few (or even single) colors in local block. This property enables us to significantly improve the coding efficiency by first encoding small color table, and then translating the original pixel block into index map [7], [8].

In contrast to these techniques, this paper focuses on the histogram characteristics of screen content. One of the most distinguishable properties of screen content as compared to camera-captured content is that the histograms of pixel values for screen content have distinct and sparse peaks respectively. However, this property is often degraded by the coding distortion, which arises some small peaks around the original peak. In this paper, we propose a method for improving the decoded image quality by removing the small peaks derived from the noise in the histogram. This paper also presents a reasonable integration of the proposed method to HEVC by extending its new in-loop filter, SAO (Sample Adaptive Offset) [9]. Experimental evaluation under the conditions employed in the HEVC standardization process clearly demonstrates that the proposed noise reduction method significantly improves the subjective quality of decoded screen content with bit-rate reduction.

2. HISTOGRAM PROPERTY OF SCREEN CONTENT

Fig. 1 (a) and (b) show histograms of pixel values for camera-captured content and screen content, respectively. The histogram characteristics of screen content are quite different from those of camera-captured content, and show distinct and sparse peaks respectively.

Fig. 1 (c) also shows a histogram of compressed screen content, which has some small peaks around the original peak (surrounded by dashed ellipse). These small peaks are derived from the coding distortion such as mosquito noise as clearly shown in Fig. 1 (d), which should be removed for improving the decoded image quality. In the following section, we propose a method for reducing the coding distortion by restoring the shapes of histograms calculated from local regions of decoded image.

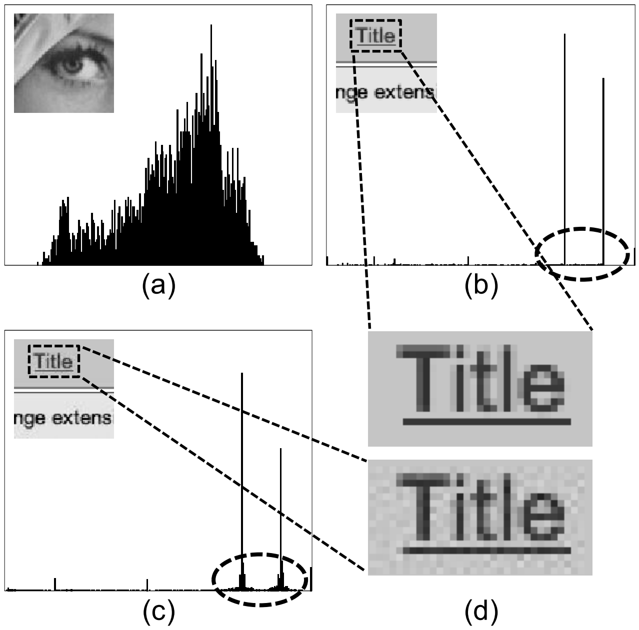


Fig. 1. Histograms calculated from (a) camera-captured content, (b) screen content and (c) compressed screen content, and (d) magnified view of (b) and (c).

3. PROPOSED METHOD

3.1. Overview of concept

The main idea of the proposed method is illustrated in Fig. 2. Fig. 2 (a) shows the histogram of pixel values for decoded screen content, where the original peaks are shown in black and the small peaks derived from the coding distortion are shown in gray. In order to reduce the noise in the decoded image, the proposed method recovers the sparsity of each peak as shown in Fig. 2 (b) by replacing the pixel values that correspond to the small peaks around the original peak with their original value (i.e., the pixel value of the original peak).

To be more specific, let N be the number of original peaks to be recovered, and p_i be the pixel value of each peak (i = 0, 1, ..., N − 1). If the peaks in the range between $p_i - r_i$ and $p_i + r_i$ can be regarded as the peaks derived from the coding distortion, we can remove the noise by replacing all the pixel values between $p_i - r_i$ and $p_i + r_i$ with p_i in the decoded image. Here, the important parameters which significantly affect the decoded image quality are p_i, r_i and N. For example, we cannot achieve a sufficient improvement if p_i is a weak peak in the histogram, or r_i (or N) is set to a very small (near zero) value. While, if r_i is inappropriately large, it may cause visual artifacts. So, ideally these parameters should be optimized in the encoder, and signaled to the decoder. However, signaling side information increases bit-rate and decreases coding efficiency. So we need to carefully select the parameters to be signaled, and also design the algorithm so that it can be easily integrated to HEVC.

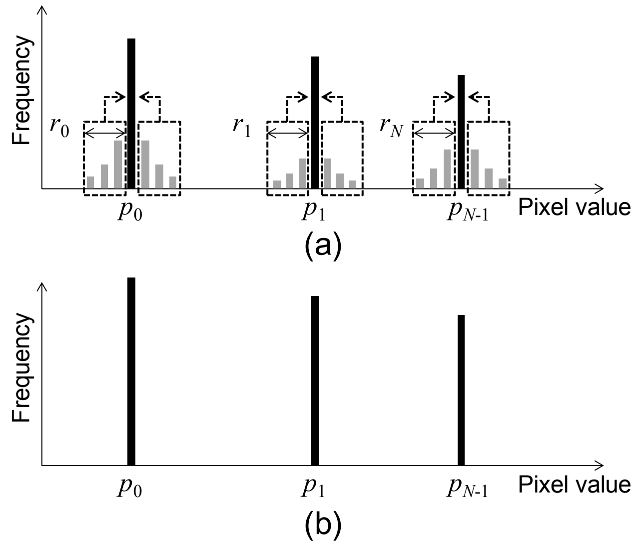


Fig. 2. Histograms of (a) before and (b) after restoration.

3.2. In-loop filtering in HEVC

Since our concept requires decoded image, it is reasonable to implement it as an extension of in-loop filtering in HEVC. There are two types of filters in HEVC, deblocking filter and SAO [9]. The deblocking filter is not suitable for the extension, because it signals almost no information to the decoder. While SAO, a newly added filter in HEVC, has several parameters to be signaled. Thus, we propose to implement our concept as an extension of SAO.

We here briefly describe the SAO process. SAO first classifies decoded pixels into different categories, and then adds an offset to the pixels belonging to the same category to reduce the distortion. SAO provides two classification methods, Edge Offset (EO) and Band Offset (BO). The EO classifies pixels based on the edge direction and structure of the decoded pixel, while the BO classifies pixels by intensity interval. The encoder determines which method to use, and calculates the offsets for reducing the coding distortion effectively. The number of categories to be compensated is four. It should be noted that the SAO parameters (e.g., an index specifying the classification method and four offsets) are calculated in the encoder and signaled to the decoder for each of the three color components in CTU (Coding Tree Unit), which is the largest processing unit (e.g., 64 × 64 pixels) in HEVC.

3.3. Proposed extension of HEVC

To implement our concept described in Section 3.1 as an extension of SAO, we add a new mode, Histogram Restoration (HR), as an alternative to the EO and BO. Thus, the encoder selects one from the three modes, EO, BO and HR, for each color component in CTU. The proposed HR applies the noise reduction to the highest N peaks in the histogram of pixel values calculated from the decoded CTU.

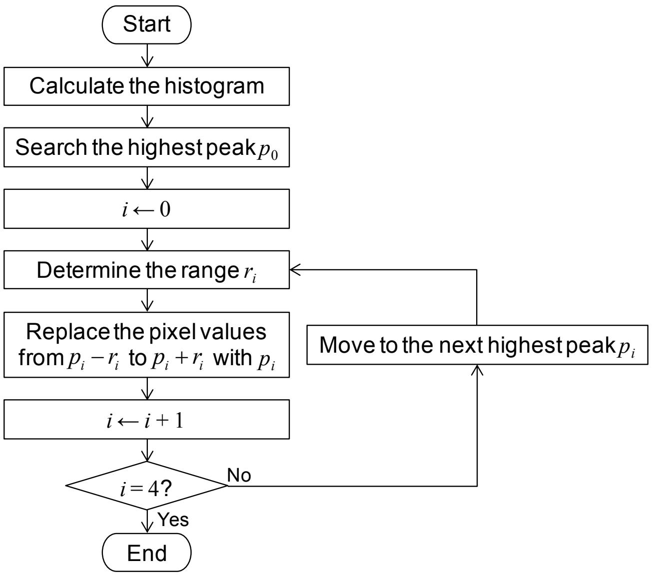


Fig. 3. Flowchart of HR in the encoder.

Since the histogram can also be calculated in the decoder, it is not necessary to signal the positions of the peaks p_i ($i = 0, 1, ..., N - 1$) to the decoder, and this saves side information significantly. The encoder determines the pixel value range r_i which is the range of peaks derived from the coding distortion around the original peak p_i (see Fig. 2). The number of peaks to be processed is set to four (i.e., $N = 4$) so that we can exploit the existing SAO syntax. When the encoder selects HR as the best mode, the encoder signals r_i ($i = 0, 1, 2, 3$) in the same manner as how the original SAO signals the four offsets of EO or BO. These processes for HR in the encoder are summarized in Fig. 3. The flow in the decoder is almost the same as Fig. 3, except that determining r_i is skipped since they are signaled from the encoder.

In HEVC, $B - 5$ bits and a sign bit are basically used for each of the four offsets in SAO, where B is the bits per pixel. According to the limit, we allocate $B - 4$ bits to r_i in HR since it does not require a sign bit ($r_i \geq 0$). In this paper, the encoder calculates distortions between original pixel values and decoded pixel values for all the possible values of r_i (i.e., $r_i = 0, 1, ..., 15$ when $B = 8$), and selects the one which minimizes the distortion to achieve the best performance. Also, the best mode selection from EO, BO and HR is done by rate-distortion optimization [10].

The original SAO provides an option for each CTU to reuse SAO parameters of the left or above CTU to reduce side information by skipping the parameter signaling. When the current CTU selects to reuse the parameters and if the neighboring CTU selects the HR, then the current CTU refers four pairs of peak positions and their ranges for noise removal (i.e., set of p_i and r_i) used in the neighboring CTU, and then replaces the pixel values around the four peaks. Since the peak positions are already known in this case, it is not necessary to calculate the histogram in the decoder.

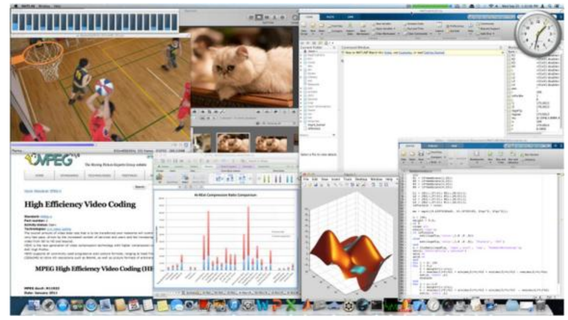


Fig. 4. Example frame from *Basketball_Screen*.

Table 1. BD-rates for Y component [%]

Sequence	AI	RA	LD
flyingGraphics	−1.14	−2.19	−3.26
desktop	−1.62	−1.46	−1.05
console	−4.03	−4.31	−3.51
web_browsing	−0.76	−0.41	−0.67
map	−0.46	−0.92	−1.19
programming	−0.73	−1.56	−2.25
SlideShow	−1.13	−1.21	−2.30
Basketball_Screen	−0.34	−0.75	−1.02
MissionControlClip2	−0.16	−0.22	−0.49
MissionControlClip3	−0.51	−0.97	−1.09
robot	0.08	−0.01	−0.01
EBURainFruits	0.02	0.05	0.05
Kimono1	0.05	0.01	0.12

4. EXPERIMENTAL RESULTS

The proposed method was implemented using the HEVC screen content coding test model (SCM 1.0) [11], and was tested against a reference using unmodified SCM 1.0, for the YUV444 sequences used in the common test conditions [12] defined during the HEVC standardization process. Fig. 4 shows an example frame of screen content used in the experiment. According to the common test conditions [12], we tested the three types of prediction structures: All Intra (AI), Random Access (RA) and Low Delay (LD).

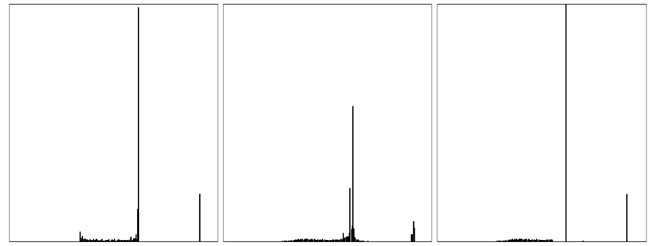
We used Bjøntegaard-Delta bit-rate (BD-rate) measure to evaluate objective coding efficiency [13]. The BD-rates (for Y) of the proposed method against SCM 1.0 are listed in Table 1, where negative BD-rate means bit-rate savings. Note that *EBURainFruits* and *Kimono1* are camera-captured contents, and others are screen contents. As seen from the table, the proposed method achieves up to more than 4% bit-rate reduction for screen contents, with negligible (0.05% on average) performance loss for camera-captured contents, which is due to the increase of the number of bits used for the index specifying EO, BO and HR. Regarding the run times, the proposed method increases about 0.5% encoding time and 6% decoding time on average. This can be further reduced by various methods such as limiting the number of pixels used in the histogram calculation.

(a)

(b)

Fig. 5. Subjective comparison of (a) *web_browsing* and (b) *MissionControlClip3*. (upper: original, middle: SCM, bottom: proposed)

To assess the visual quality, Fig. 5 shows a magnified view of parts of decoded images under AI configuration for SCM and the proposed method with original images for reference. As for Fig. 5 (a), it can be observed the mosquito noise around the characters in the result of SCM, while the proposed method significantly improves the visual quality by suppressing the noise. Also, as can be seen from Fig. 5 (b), SCM causes false edge in the top of the image and color artifact around the characters at the bottom. These coding distortions are clearly removed by the proposed method.



(a) (b) (c)

Fig. 6. Histograms calculated from (a) original image, (b) decoded image of SCM, and (c) decoded image of the proposed method.

Table 2. Percentage of each mode in SAO [%]

Sequence	**EO**	**BO**	**HR**
flyingGraphics	15.51	14.44	70.05
console	3.58	11.29	85.13

In order to see the effectiveness of the proposed method, Fig. 6 compares the shapes of histograms calculated from CTUs (64×64 pixels) extracted from an original image and decoded images for SCM and the proposed method, respectively (for *web_browsing*). The comparison clearly demonstrates that the coding distortion broadens the peaks in the histogram (Fig. 6 (b)) and the proposed method can restore the shape of peaks properly (Fig. 6 (c)).

Also, we calculated the percentage of EO, BO and HR in SAO for the two screen contents under AI configuration (QP = 27). The results are shown in Table 2, where the proposed HR accounts for more than 70% of the area applied SAO. Since the rate-distortion optimization [10] is used for selecting the best mode from EO, BO and HR in our implementation, these results highlight that the proposed method is far more suitable for screen content coding than EO and BO used in the original SAO.

5. CONCLUSIONS

This paper proposed a noise reduction method for screen content coding using a new video coding standard, HEVC. Unlike camera-captured content, the histograms of typical screen content have distinct and sparse peaks respectively. However, this property is degraded by the coding distortion, which arises some small peaks around the original peak. The proposed method reduces the noise by recovering the sparsity of histogram peaks, and is reasonably integrated to the HEVC standard as an additional mode of SAO. The proposed encoder signals the ranges for noise removal around the original peaks instead of the offsets of EO or BO used in the original SAO. Experimental evaluation clearly demonstrates that the proposed noise reduction method significantly improves the visual quality of decoded image with a reduction of up to 4% in bit-rate.

6. REFERENCES

[1] G. J. Sullivan, J.-R. Ohm, W.-J. Han, and T. Wiegand, "Overview of the high efficiency video coding (HEVC) standard," *IEEE Trans. Circuits Syst. Video Technol*., vol. 22, no. 12, pp. 1649–1668, Dec. 2012.

[2] Y. Lu, S. Li, and H. Shen, "Virtualized screen: a third element for cloud–mobile convergence," *IEEE Multimedia*, vol. 18, no. 2, pp. 4–11, Apr. 2011.

[3] *Standardization plan for HEVC extensions for screen content coding*, Standard ITU-T Q6/16 and ISO/IEC JTC1/SC29/WG11, MPEG N14520, Apr. 2014.

[4] R. Joshi, J. Xu, D. Flynn, M. Naccari, C. Rosewarne, K. Sharman, J. Sole, G. J. Sullivan, T. Suzuki, Y.-K. Wang, and B.Bross, "High Efficiency Video Coding (HEVC) screen content coding: draft 1," *Document of Joint Collaborative Team on Video Coding*, JCTVC-R1005, July 2014.

[5] M. Budagavi and D.-K. Kwon, "AHG8: Video coding using intra motion compensation," *Document of Joint Collaborative Team on Video Coding*, JCTVC-M0350, Apr. 2013.

[6] T. Lin, X. Chen, and S. Wang, "Pseudo-2D-matching based dual-coder architecture for screen contents coding," *Proc. IEEE Int'l Conf. Multimedia and Expo Workshops*, pp. 1–4, July 2013.

[7] L. Guo, M. Karczewicz, and J. Sole, "Palette mode for screen content coding," *Document of Joint Collaborative Team on Video Coding*, JCTVC-M0323, Apr. 2013.

[8] Z. Ma, W. Wang, M. Xu, and H. Yu, "Advanced screen content coding using color table and index map," *IEEE Trans. Image Processing*, vol. 23, no. 10, pp. 4399–4412, Oct. 2014.

[9] 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," *IEEE Trans. Circuits Syst. Video Technol*., vol. 22, no. 12, pp. 1755–1764, Dec. 2012.

[10] G. J. Sullivan and T. Wiegand, "Rate-distortion optimization for video compression," *IEEE Signal Processing Magazine*, vol. 15, issue. 6, pp. 74–90, Nov. 1998.

[11] R. Joshi, J. Xu, R. Cohen, S. Liu, Z. Ma, and Y. Ye, "Screen content coding test model 1 (SCM 1)," *Document of Joint Collaborative Team on Video Coding*, JCTVC-Q1014, Mar. 2014.

[12] H. Yu, R. Cohen, K. Rapaka, and J. Xu, "Common conditions for screen content coding tests," *Document of Joint Collaborative Team on Video Coding*, JCTVC-Q1015, Mar. 2014.

[13] G. Bjøntegaard, "Calculation of average PSNR differences between RD curves, document," *Document of Video Coding Expert Group*, VCEG-M033, Apr. 2001.