COLOR PALETTE FOR SCREEN CONTENT CODING

Liwei Guo[1]*, Wei Pu, Feng Zou, Joel Sole, Marta Karczewicz, and Rajan Joshi*
Qualcomm Technologies Inc.
{*wpu, fzou, joels, martak, rajanj*}@qti.qualcomm.com

ABSTRACT

Abstract—With the prevalence of high speed Internet access, emerging video applications such as remote desktop sharing, virtual desktop infrastructure, and wireless display require high compression efficiency of screen contents. However, traditional intra and inter video coding tools were designed primarily for natural contents. Screen contents have significantly different characteristics compared with nature contents, e.g. sharp edges, less or no noise, which makes those traditional coding tools less sufficient. In this research, a new color palette based video coding tool is presented. Different from traditionally intra and inter prediction that mainly removes redundancy between different coding units, palette coding targets at the redundancy of repetitive pixel values/patterns within the coding unit. In the palette coding mode, a lookup table named palette which maps pixel values into table indices (also called palette indices) is signaled first. Then the mapped indice for a coding unit (which we call index block) are coded with a novel three-mode run-length entropy coding. Some encoder-side optimization for palette coding is also presented in detail in this paper. Simulation has been performed using the common screen content coding test condition defined by JCT-VC and the results show that palette coding can effectively improve screen content coding efficiency for both lossless and lossy scenarios.

Index Terms— *HEVC, major color, palette, range extension, run length*

1. INTRODUCTION

Screen content coding (SCC) is an emerging new research topic in video coding literature due to its numerous potential applications. Different from natural content video, screen content video has some unique features which make SCC a challenging problem.

First, compared to natural video content, screen content video usually has much less (if any) noise while the edges tend to be much sharper. The natural video contents are usually captured by cameras with CCD/CMOS sensor. The capture process can smooth object edges (e.g. diffuse reflection) and introduced the noise (e.g. sensor thermal noise). In contrary, screen contents are normally generated by computers. No edge sharpness or image quality is compromised in the content generation process. Second, in screen content, for a local area, the number of different colors tends to be very limited. In another word, the color histogram of screen content video in a coding unit (CU) block usually has high peaks at only a few pixel values. Third, obvious repetitive patterns could be observed in screen content videos. For example, English characters and some logos used by the operating system may repeat many times within each video frame for screen content.

Based on the study of the unique characteristics of screen content video, this paper presents a new coding tool named palette coding for the compression efficiency of screen contents. Specifically, palette coding introduces a lookup table, i.e. color palette. This is based on the observation that in SCC, colors within one CU usually concentrate on a few peak values. After getting the palette, pixels within one CU are mapped to the palette indices. At the second stage, an effective three-mode run length entropy coding method is proposed to effectively compress the repetitive pattern in the indices of the block. No transformation process is invoked for palette coding to avoid blurring sharp edges that can significantly impact the visual quality of screen contents.

The contributions of the proposed palette coding method are summarized as follows:

1) Palette coding significantly improves SCC efficiency by exploiting the redundancy among similar pixel values and among local repetitive pixel patterns in the picture. This new kind of redundancy is not frequently observed in natural content video. Existing coding tools have been optimized for natural content and failed to a effectively handling screen contents.
2) In palette coding, a novel three-mode run length coding method is developed. The new entropy coding method is able to efficiently compress the pixels which have the same values as their left or top neighbors. It can also handle the case when a pixel value is too expensive to be included into the palette.
3) Palette coding is a very simple tool. It skips the traditional transform/inverse transform module, and improves the decoder throughput.

There were some existing major color table based SCC coding methods presented in [3][4][5][6]. Although conceptually similar, the palette coding method in this paper

---

[1] This work was carried out while L. Guo was with Qualcomm Technologies, Inc.

differentiates itself from the existing works in the following aspects: First, the existing method used line by line coding which cannot efficiently encode repetitive patterns – the patterns do not necessarily align with line boundaries. Without the restriction of line-alignment, the novel three-mode run-length coding method can handle more general scenarios. Second, several key optimization methods such as advanced palette prediction are included to boost the performance of the proposed palette coding method.

The rest of this paper is organized as follows. In Section 2, the proposed palette coding method is explained in detail. Section 3 reports simulation results using the HEVC RExt common test condition. Section 4 concludes the paper with a brief summary.

2. PALETTE CODING[2]

2.1 Run-Length Based Palette Coding Method

For an input CU, the palette coding is composed of two stages:

1. Deriving the palette: Palette is a look up table, where maps pixel value to table index. To derive palette, the histogram of the pixel values in the current CU is calculated. Then the peak values in the histogram are added into the palette (The implementation in this paper allows the palette table size to be up to 32). As an example, in Fig. 1, there are two peak values ‘r’ (red) and ‘b’ (blue), so two pairs (0, r) and (1, b) are added into the palette. There is an implicit quantization operation here. Not only pixels whose values are exactly equal to ‘r’ are mapped to index ‘0’, neighboring values around ‘r’ are mapped to index ‘0’ as well to improve rate-distortion performance. Low frequency pixel values (e.g. the white pixel in Fig. 1) are not included into the palette due to the same reason. The quantization step for palette table monotonically increases with QP.

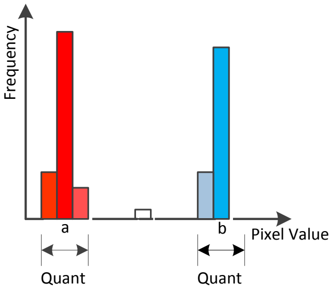


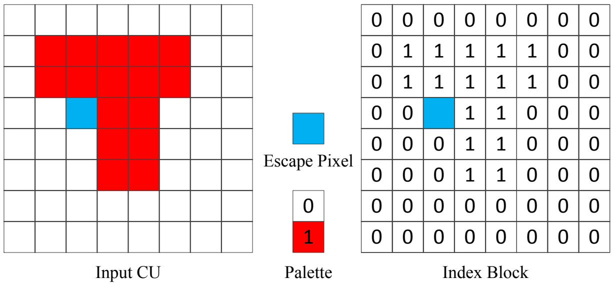
Figure 1. Deriving the color palette.

2. Coding CU using the palette: the mapped pixels (i.e. the Index Block in Fig. 2) in the CU are encoded in a raster scan order using 3 run-length modes specified as follows:

2.1 “Copy from Left” run mode (CL): In this mode, one palette index is first signaled followed by a non-negative value n indicating the run length, which means that the following n pixels have the same pixel index as the first signaled one. For example, in Fig. 3, $n = 4$.

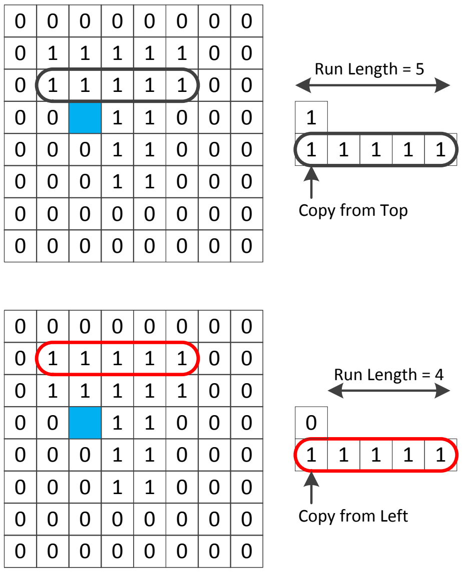
2.2 “Copy from Top” run mode (CT): In this mode, only a positive run length value m is transmitted to indicate that for the following m pixels including the current one, palette indexes are the same as their above neighbours. For example, in Fig. 3, $m = 5$.

2.3 “Escape” mode: Escape mode is used to code low frequency pixels which are not mapped into any index in the palette. Quantized pixels are directly coded into the bitstream. In Fig. 3, the blue pixel is coded in “Escape” mode.



0	0	0	0	0	0	0	0
0	1	1	1	1	1	0	0
0	1	1	1	1	1	0	0
0	0	🟦	1	1	0	0	0
0	0	0	1	1	0	0	0
0	0	0	1	1	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

Figure 2. Illustration of palette coding process.



0	0	0	0	0	0	0	0
0	1	1	1	1	1	0	0
0	1	1	1	1	1	0	0
0	0	🟦	1	1	0	0	0
0	0	0	1	1	0	0	0
0	0	0	1	1	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

0	0	0	0	0	0	0	0
0	1	1	1	1	1	0	0
0	1	1	1	1	1	0	0
0	0	🟦	1	1	0	0	0
0	0	0	1	1	0	0	0
0	0	0	1	1	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

Figure 3. Illustration of ‘copy from top’ run and ‘copy from left’ run modes in palette coding.

The method to code the run length variable m and n are aligned to coefficient coding method in HEVC, i.e. first coding ‘greater than 0’, ‘greater than 1’, and ‘greater than 2’. Then if necessary Golomb-Rice code with parameter equal to 3 is used to code the remaining values. Please refer to HEVC standard [8] for the details of coefficient coding.

---

[2] Due to limit of the space, not every technical detail is explained in this paper. Interested readers please find more comprehensive description and source code in reference [1].

2.2 Optimization Decisions

To further improve the efficiency of palette coding, several optimization technologies are introduced as follows.

Determine the Run Mode

In some cases, a sequence of pixels can be classified into either “copy from top” mode or “copy from left” mode. Theoretically, a rate-distortion checking process should be invoked at the encoder to determine the better run-length mode from them. However, such rate-distortion optimization process introduces too much computational cost. In practice, a simple mode decision method is developed based on the fact that “copy from top” mode has smaller signaling cost so that it is more preferred than “copy from left” mode. In our implementation, “copy from top” mode is selected if its run-length plus two is greater than “copy from left” run length.

Palette Prediction

For SCC, neighboring CUs have very strong correlation in their palettes. To exploit such correlation, and to avoid extra line buffer, the palette of the current CU is predicted from the one of its left neighboring CU given that the neighbor is coded using palette mode. Specifically, a binary vector whose size is the same as the left neighboring CU’s palette size is signaled in the bitstream. An entry ‘1’ specifies that the corresponding palette item of the left neighboring CU is reused for the current block, and an entry ‘0’ mans no reusing. For example, a vector of ‘101’ means that the first and the third palette items in the left neighboring CU’s palette is reused.

Escape Pixel Coding

The number of context coded bins for each pixel has impact on the decoding throughput and the hardware implementation complexity. Therefore, in the state-of-the-art High Efficiency Video Coding standard, such number is well controlled. However, totally removing contexts for escape pixel coding, i.e. always coding escape pixels with bypass mode (i.e., 0.5:0.5 probability model) will result in noticeable coding efficiency loss for ignoring the uneven probability distribution after escape pixel quantization. By jointly considering these two facts, in our implementation, only the most significant bit of the quantized escape pixel is coded using CABAC context while other bits are coded as bypass.

Table 1 Lossless Compression Results of Palette Coding v.s. HM RExt-5.1

	All Intra											
	Compression ratio								**Bit-rate increase**			
	Total		**Average**		**Min**		**Max**		Total	Avg	Min	Max
	Ref.	Tested	Ref.	Tested	Ref.	Tested	Ref.	Tested				
Class F	4.6	4.6	5.6	5.6	2.3	2.3	11.2	11.2	-0.5%	-0.4%	-1.2%	0.0%
Class B	2.2	2.2	2.3	2.3	2.1	2.1	2.4	2.4	0.0%	0.0%	0.0%	0.0%
RGB 4:4:4 SC	9.4	10.9	11.7	15.6	6.5	7.2	19.6	34.1	-13.8%	-17.5%	-42.4%	-0.3%
RGB 4:4:4 Animation	2.8	2.8	2.9	2.9	2.4	2.4	3.1	3.1	0.0%	0.0%	0.0%	0.0%
YC_bC_r 4:4:4 SC	11.2	12.7	13.5	17.7	8.0	8.6	21.6	36.1	-11.7%	-16.4%	-40.0%	-0.2%
YC_bC_r 4:4:4 Animation	3.1	3.1	3.2	3.2	2.6	2.7	4.0	4.0	-0.8%	-0.5%	-1.6%	0.0%
RGB 4:4:4 SC (Optional)	30.9	48.9	34.3	60.7	17.3	23.7	48.9	95.7	-36.8%	-39.1%	-48.9%	-27.2%
YC_bC_r 4:4:4 SC (Optional)	35.5	56.0	38.6	68.7	24.3	28.9	59.0	113.9	-36.6%	-37.6%	-48.6%	-15.9%
	Random Access											
	Compression ratio								**Bit-rate increase**			
	Total		**Average**		**Min**		**Max**		Total	Avg	Min	Max
	Ref.	Tested	Ref.	Tested	Ref.	Tested	Ref.	Tested				
Class F	8.6	8.6	35.8	35.8	3.0	3.0	88.9	88.5	0.0%	0.1%	-0.1%	0.4%
Class B	2.6	2.6	2.6	2.6	2.6	2.6	2.6	2.6	0.0%	0.0%	0.0%	0.0%
RGB 4:4:4 SC	56.8	58.4	140.5	189.3	20.0	20.1	291.0	498.7	-2.7%	-11.9%	-41.7%	-0.2%
RGB 4:4:4 Animation	3.7	3.7	3.7	3.7	3.5	3.5	3.9	3.9	0.0%	0.0%	0.0%	0.0%
YC_bC_r 4:4:4 SC	71.0	72.9	161.3	215.6	24.7	24.7	321.2	530.0	-2.6%	-12.1%	-39.4%	0.0%
YC_bC_r 4:4:4 Animation	3.9	3.9	4.5	4.5	2.8	2.8	5.7	5.7	-0.1%	0.0%	-0.2%	0.0%
RGB 4:4:4 SC (Optional)	70.9	79.3	329.6	531.4	19.9	21.5	498.7	797.8	-10.6%	-28.0%	-39.3%	-7.1%
YC_bC_r 4:4:4 SC (Optional)	109.0	118.3	356.5	626.9	32.5	32.8	601.5	1038.1	-7.9%	-29.7%	-46.2%	-0.8%
	Low Delay B											
	Compression ratio								**Bit-rate increase**			
	Total		**Average**		**Min**		**Max**		Total	Avg	Min	Max
	Ref.	Tested	Ref.	Tested	Ref.	Tested	Ref.	Tested				
Class F	8.8	8.8	56.2	56.2	3.0	3.0	163.1	162.9	0.0%	0.0%	0.0%	0.1%
Class B	2.6	2.6	2.6	2.6	2.6	2.6	2.6	2.6	0.0%	0.0%	0.0%	0.0%
RGB 4:4:4 SC	63.6	64.1	485.6	685.3	21.0	21.0	2111.8	3244.6	-0.7%	-8.2%	-34.9%	-0.1%
RGB 4:4:4 Animation	3.7	3.7	3.7	3.7	3.5	3.5	3.8	3.8	0.0%	0.0%	0.0%	0.0%
YC_bC_r 4:4:4 SC	80.6	81.0	544.8	753.5	25.9	25.9	2333.7	3491.0	-0.6%	-8.3%	-33.2%	0.0%
YC_bC_r 4:4:4 Animation	3.9	3.9	4.5	4.5	2.8	2.8	5.8	5.8	-0.1%	0.0%	-0.1%	0.0%
RGB 4:4:4 SC (Optional)	77.2	83.0	1334.9	1726.5	20.1	21.5	3012.9	3874.5	-7.0%	-17.6%	-24.3%	-6.3%
YC_bC_r 4:4:4 SC (Optional)	125.2	127.7	1294.3	1940.5	33.1	33.1	2516.7	3782.1	-1.9%	-22.4%	-33.6%	-0.1%

3. SIMULATION RESULTS

In this section, the proposed palette coding method is implemented on top of HM 12.0 RExt-5.1 reference software and evaluated using standard HEVC RExt test condition for screen content coding specified in [7]. The test conditions were defined for a comprehensive evaluation over a test set consisting of sequences with different characteristics. Class F has screen content/mixed content captured in 420 format. Class B is natural video contents. As name suggestions, other sequences are screen content or animation captures in different color spaces.

Table 1 shows the lossless compression performance and Table 2 is the lossy compression performance (measured in terms of BD-rate reduction). The maximum size of palette is restricted to 32.

It can be observed from the simulation results that palette coding is effective for all of the three test configurations, i.e. all intra, random access, and low delay B. However, it is also observed that higher gain is achieved for the all intra configuration. The reason is that palette coding is essentially a tool to exploit the redundancy within each CU, rather than CUs in different frames. Another observation is that palette coding is more effective for typical screen contents and less effective for animation in which the number of colors in one CU is more diversified and repetitive pixel patterns are not very strong.

4. CONCLUSION

This paper presents an effective new tool named palette coding for compressing video contents of computer screen activities such as browsing web page or using google map. Palette coding showed significant coding gain by reducing the redundancy of repetitive pixel values and pixel patterns within a coding unit, which could not be well exploited by traditional inter or intra coding modes. Palette coding bypasses most of the existing coding processes such as residual transformation and intra directional prediction so that it can be efficiently implemented using either software or hardware.

5. REFERENCES

[1]. W. Pu, F. Zou, J. Sole, M. Karczewicz, and R. Joshi, “Non-RCE4: Refinement of the palette in RCE4 Test 2,” in ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11 Document JCTVC-P0231, San Jose, CA, USA, Jan. 9-17, 2014.

[2]. L. Guo, W. Pu, M. Karczewicz, J. Sole, R. Joshi, and F. Zou, “RCE4: Results of Test 2 on palette mode for screen content coding,” in ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11 Document JCTVC-P0198, San Jose, CA, USA, Jan. 9-17, 2014.

[3]. X, Guo, Y. Lu, and S. Li, “RCE4: Test 1 Major-color-based screen content coding,” in ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11 Document JCTVC-P0108, San Jose, CA, USA, Jan. 9-17, 2014.

[4]. Z. Pan, H. Shen, Y. Lu, N. Yu, and S. Li, “A low-complexity screen compression scheme,” in *Proc. IEEE VCIP’12*, San Diego, CA, USA, Nov. 27-30, 2012.

[5]. Z. Pan, H. Shen, Y. Lu, S. Li, and N. Yu, “A low-complexity screen compression scheme for interactive screen sharing,” *IEEE Trans. Circuits Sys. Video Technol.*, vol. 23, no. 6, June

Table 2 Lossy Compression Results of Palette Coding v.s. HM RExt-5.1

	All Intra Main-tier			**All Intra High-tier**			**All Intra Super-High-tier**		
	Y	U	V	Y	U	V	Y	U	V
Class F	-1.5%	-3.4%	-2.7%	-1.7%	-2.8%	-2.4%	-1.9%	-2.2%	-2.0%
Class B	0.1%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%
RGB 4:4:4 SC	-13.4%	-12.4%	-12.8%	-16.0%	-15.4%	-15.4%	-18.7%	-19.2%	-19.1%
RGB 4:4:4 Animation	0.0%	-0.1%	0.0%	0.0%	-0.1%	0.0%	0.0%	-0.1%	0.0%
YC_bC_r 4:4:4 SC	-9.3%	-12.4%	-13.8%	-13.7%	-15.8%	-16.5%	-17.1%	-19.5%	-20.5%
YC_bC_r 4:4:4 Animation	0.1%	0.0%	-0.1%	0.0%	-0.1%	-0.1%	0.0%	-0.1%	-0.1%
RGB 4:4:4 SC (Optional)	-30.0%	-28.9%	-30.1%	-34.0%	-33.6%	-34.5%	-40.6%	-40.9%	-41.9%
YC_bC_r 4:4:4 SC (Optional)	-23.4%	-29.3%	-28.1%	-30.2%	-33.8%	-34.0%	-37.2%	-42.4%	-43.7%
	Random Access Main-tier			**Random Access High-tier**					
	Y	U	V	Y	U	V			
Class F	-0.8%	-3.6%	-2.1%	-0.9%	-2.4%	-1.6%			
Class B	0.2%	0.2%	0.2%	0.1%	0.1%	0.1%			
RGB 4:4:4 SC	-11.7%	-10.8%	-11.1%	-12.9%	-12.4%	-12.6%			
RGB 4:4:4 Animation	0.0%	0.0%	0.1%	0.0%	0.0%	0.0%			
YC_bC_r 4:4:4 SC	-8.7%	-11.6%	-13.1%	-11.9%	-14.1%	-14.9%			
YC_bC_r 4:4:4 Animation	0.0%	-0.1%	-0.1%	0.0%	0.0%	0.0%			
RGB 4:4:4 SC (Optional)	-22.1%	-20.9%	-22.1%	-24.9%	-24.1%	-24.9%			
YC_bC_r 4:4:4 SC (Optional)	-19.3%	-27.1%	-27.2%	-23.5%	-29.0%	-30.5%			
	Low delay B Main-tier			**Low delay B High-tier**					
	Y	U	V	Y	U	V			
Class F	0.0%	-3.3%	-0.9%	-0.2%	-1.8%	-0.6%			
Class B	0.1%	-0.2%	-0.2%	0.1%	-0.1%	-0.1%			
RGB 4:4:4 SC	-4.8%	-4.6%	-4.8%	-5.9%	-5.8%	-5.9%			
RGB 4:4:4 Animation	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%			
YC_bC_r 4:4:4 SC	-3.6%	-4.5%	-5.2%	-4.9%	-5.6%	-6.2%			
YC_bC_r 4:4:4 Animation	0.1%	0.1%	0.1%	0.1%	0.1%	0.1%			
RGB 4:4:4 SC (Optional)	-10.9%	-10.5%	-11.1%	-15.0%	-14.6%	-15.7%			
YC_bC_r 4:4:4 SC (Optional)	-10.7%	-15.1%	-15.4%	-14.3%	-18.1%	-18.5%			

2013.

[6]. H. Shen, Y. Lu, "A high-performance remote computing platform," in *Proc. IEEE PerCom'09*, Galveston, TX, SUA, March 9-13, 2009.

[7]. L. Guo, X. Guo, and A. Saxena, "HEVC Range Extensions Core Experiment 4 (RCE 4): Palette coding for screen content," in ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11 Document JCTVC-O1124, Geneva, Switzerland, Oct. 2013.

[8]. J. Sole, R. Joshi, N. Nguyen, M. Karczewicz, G. Clare, F. Henry, and A. Duenas,, "Transform coefficient coding in HEVC," *IEEE Trans. Circuits Sys. Video Technol.*, vol. 22, no. 12, Dec 2012.