

Neighboring Sample Prediction Coding for HEVC Screen Content Coding

Yao-Jen Chang¹, Ching-Chieh Lin¹, Jih-Sheng Tu¹,
Chun-Lung Lin¹(✉), Chao-Hsiung Hung¹, and Pei-Hsuan Tsai²

¹ Industrial Technology Research Institute, Hsinchu, Taiwan
{britpablo, JackLin, sunriseJSTu,
Chunlung, chhung}@itri.org.tw

² Institute of Manufacturing Information and Systems,
National Cheng Kung University, Tainan, Taiwan
peihsuan.tsai@gmail.com

Abstract. High Efficiency Video Coding (HEVC) Screen Content Coding (SCC) is under standardizing for the screen-captured contents. Because many areas are composed of the texts and lines featuring non-smooth textures, the traditional intra prediction is not suitable for those areas. Therefore, this paper proposes a neighboring sample prediction coding (NSPC) for HEVC SCC, which represents the samples of the coding unit (CU) by the indexes of samples selected from the neighboring CUs. A unified sample selection scheme (USSS) based on a neighboring sample list is also proposed to determine the priority ordering of the selected neighboring samples. The index coding with a single sample index skipping method is also designed for coding the converted indexes. Different NSPC signaling methods are evaluated under the common test condition for HEVC SCC, and the results show that the NSPC improves the Bjontegaard-Delta bitrate saving of Screen Content Coding Test Model 2.0 up to 1.0 %.

Keywords: Screen content coding · High efficiency video coding · Intra prediction · Palette coding

1 Introduction

In recent years, screen-content applications, such as wireless screen mirroring, remote desktop, remote play and cloud gaming, have attracted attention in the video standard community. In 2014, Joint Collaborative Team on Video Coding (JCT-VC), established by Moving Picture Experts Group (MPEG) and Video Coding Experts Group (VCEG), held a joint call-for-proposals[1] to extend High Efficiency Video Coding (HEVC) standard [2] to screen content coding (SCC). Now the HEVC SCC [3] is being standardized under the HEVC coding platform with newly additional tools specializing in coding screen contents.

Screen contents generally include a random mixture of text and computer-generated graphic. This emerging screen-captured video contents exhibit very different characteristics from camera-captured video. Since the objects, such as texts and lines with

sharp edges, mainly feature non-smooth textures with irregular directions, the traditional intra prediction which predicts the samples of a coding unit (CU) by the intra samples with a regular direction is not suitable for these objects. To address the feature of screen contents, major-color-based coding tools, called palette mode (PM) [4–6], are proposed and comprehensively evaluated using the test sequences released by JCT-VC. The basic idea of the PM is to select a number of major colors which are the most representative pixel samples for the CU. Then each pixel in the CU is converted to an index corresponding to the nearest one of the major colors.

Usually, there is high correlation of pixel samples between the CU and the neighboring CUs. Therefore, this paper proposes a neighboring sample prediction coding (NSPC) to represent the pixel samples of the current CU by the indexes of the samples selected out of the neighboring CUs. A unified sample selection scheme (USSS) based on a neighboring sample list is also proposed to determine the priority ordering of the selected neighboring samples. The index coding with a single sample index skipping method is also designed for coding the converted indexes. For comparing the performance, we propose two versions of NSPC for implementation on the reference software Screen Content Coding Test Model 2.0 (SCM2.0) [7]: one views it as an additional tool which is independent of the other ones, and the other incorporates it into the existing PM, where the signaling are within the signaling of the PM. Due to the information on the neighboring CUs available for the decoder, there is no need to signal the neighboring samples and thus no burden on the bitstream. Both versions of NSPC are evaluated under the common test condition (CTC) [8] for HEVC SCC, and the evaluation results show that the NSPC can improve the Bjontegaard-Delta bitrate (BD-rate) [9] saving up to 1.0 % as compared to SCM 2.0

The rest of this paper is organized as follows. Section 2 introduces the palette mode in HEVC SCC. Section 3 presents the proposed NSPC. Section 4 shows the analyses of the NSPC. Section 5 presents two versions of NSPC to be evaluated in Sect. 6. Section 7 concludes this paper.

2 Palette Mode in HEVC SCC

The PM tool [4–6] is one of new coding tools adopted in the HEVC SCC. The concept of the PM is to analyze the pixel samples in the CU first, select several major colors out of the pixel samples to represent the CU, and convert each pixel in the CU to the index which corresponds to the major color nearest to the pixel sample.

The encoder first selects the most representative major colors by considering the combined pixel samples of the three components (Y/G, U/B and V/R) in the CU, and then establishes a look-up table of major colors, called major color table. Each major color would be given an index. A group of pixels gathering around a major color are represented by this major color and converted to the index corresponding to the major color. In the SCM2.0, the maximum amount of major colors is set to thirty-one.

Another important stage in the encoder is to code and transmit the major color table and the converted indexes. The major colors listed in the major color table must be coded by using a lossless coding because the decoder will need the major color table to reconvert the indexes to the major colors. Subsequently, the converted indexes are

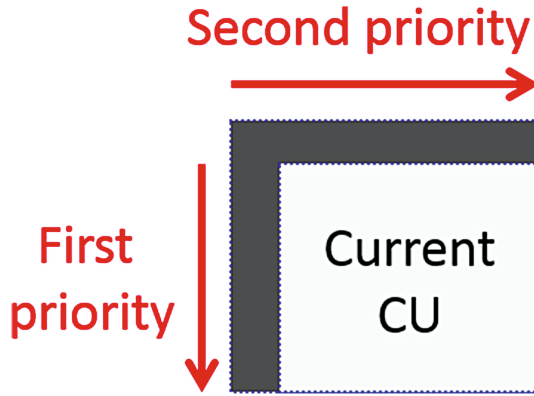


Fig. 1. The fixed order for selecting the neighboring samples of N_2, N_3, \dots, N_{10} in the proposed USSS

coded based on two copy run modes: one is copy above run mode, and the other is copy left run mode. A flag is first signaled to indicate which copy run mode is selected. The copy above run mode is activated if the current and following color indexes are identical to the indices of the above pixel line. The copy left run mode is used when there are identical color indexes followed by the color index of the current coding pixel. For the copy left run mode, the color index is signaled to decoder to indicate the value of the index. For two copy run modes, a run value is further signaled to indicate the amount of pixels that copied from the above pixel line or the left pixel.

On the decoder side, it first decodes the major colors to reconstruct the major color table of the CU. Meanwhile, the indexes are also decodes to represent each pixel of the CU. By looking up the major color table, each index is reconverted to one of the major colors. Finally, the reconstructed samples of the CU are those colors reconverted from indexes.

3 Proposed Neighboring Sample Prediction Coding

The pixels in the CU are able to be predicted by the samples out of the neighboring blocks. The traditional intra prediction coding in the HEVC is the prediction method employing neighboring samples under the assumption of the CU with a regular texture or the smooth area. However, many areas of the screen captured content are composed of texts and lines located in the software such as websites and word-editing, which features a non-smooth texture with irregular directions. Therefore, the original intra prediction coding is no longer suitable for those areas. In this section, the NSPC is proposed to represent each pixel sample of the CU by the nearest one of the neighboring samples, which are selected from the nearest neighboring CUs and the previously encoded palette CU. Differing from the traditional intra prediction coding, the NSPC does not follow the rules of the texture direction or smoothing, and thus is suitable for the screen captured content.

The NSPC includes a number of neighboring sample (NS) modes, which are 1-NS mode, 2-NS mode, and so on. For M -NS mode, a neighboring sample list is established based on M neighboring samples, i.e., $\{S_1, S_2, \dots, S_M\}$, where M could be 1, 2, and so on, and S_m is the m -th neighboring sample of M -NS mode. Then each sample in the CU is predicted by one of M neighboring samples. The neighboring samples in the neighboring sample list are mainly selected out of the neighboring reference blocks, which are inside the nearest neighboring CUs relative to the current CU. As shown in Fig. 1, the grey region is the neighboring reference blocks, where the parameter H is the height of the grey area in the above CU and the above-left CU, and the parameter W is the width of the grey area in the left CU and the above-left CU. In this paper, the maximum value of M is set as 10 and W and H are set as 2 based on the performance on BD-rate.

To unify the establishment of the neighboring sample lists for different NS modes, the USSS is proposed to select the neighboring samples following a certain ordering. Because there are ten neighboring samples at most for the NS modes in the proposed NSPC, the USSS orderly selects ten neighboring samples and denotes the ordering samples as the list $\{N_1, N_2, \dots, N_{10}\}$, where N_i is defined as the i -th neighboring sample. The first sample N_1 is the most frequently occurring sample out of the neighboring reference blocks, and then the rest of samples N_2, N_3, \dots, N_{10} are filled with the neighboring samples from the neighboring reference blocks in a fixed order as shown in Fig. 1, where the first priority is the top sample to the bottom sample in the left neighboring reference block, and the second priority is the sample of the above-left CU, the left sample to the right sample of the above neighboring reference block. Note that a neighboring sample will not be filled with the list if its value is identical to the value of the previously derived sample N_i . If empty entry exists in the list followed by the filling process, the color samples from the major color table of the previously coded CU are employed to fill in those empty entries. After setting up the ordering sample list $\{N_1, N_2, \dots, N_{10}\}$, the neighboring sample lists for ten different NS modes can be easily established as follows:

- 1-NS mode: $\{N_1\}$,
- 2-NS mode: $\{N_1, N_2\}$,
- 3-NS mode: $\{N_1, N_2, N_3\}$,

and so on. The last NS mode is 10-NS mode, whose neighboring sample list is $\{N_1, N_2, \dots, N_{10}\}$.

After the USSS establishes the neighboring sample lists, each NS mode will convert each pixel of the CU to the index corresponding to the nearest neighboring sample out of its own neighboring sample list. The converting process for the M -NS mode can be illustrated as follows:

Step 1: The pixel $p(x,y)$ is predicted by the nearest neighboring sample $n(x,y)$, where (x,y) is the coordinate in the CU:

$$n(x, y) = \arg \min_N \text{abs}(N_i - p(x, y)).$$

Step 2: The prediction sample $n(x,y)$ is converted to the corresponding index $r(x,y)$ based on the look up table as shown in Table 1. The indexes $r(x,y)$ for all (x,y) in the CU are the results of the converting process.

During the signaling process at the encoder, a CU-level enabling flag is signaled to indicate whether to enable the proposed NSPC. If the NSPC tool is enabled, a flag is further signaled to indicate which NS mode is selected. Note that the best mode out of different NS modes will be selected based on the rate distortion (RD) optimization. The indexes $r(x,y)$ for all (x,y) in the CU are also required to be coded. The copy left run mode and copy above run mode proposed in the PM are applied to the index coding for different NS modes. However, for the 1-NS mode, a single sample index skipping method is designed to skip the signaling of the copy left run mode and copy above run mode. Because the indexes $r(x,y)$ for 1-NS mode must be a constant value '0', the decoder can readily parse the indexes $r(x,y)$ to be zeros without any further signaling of copy run modes after it knows the mode is 1-NS mode.

Table 1. The look up table of the index corresponding to each neighboring samples in the M -NS mode $\{N_1, N_2, \dots, N_M\}$

The neighboring sample	The sample index
N_1	0
N_2	1
\vdots	\vdots
N_M	$M-1$

Compared to the traditional intra prediction coding, the assignment of the prediction sample $n(x,y)$ of the pixel $p(x,y)$ for the NSPC is not based on a regular texture direction. Therefore, the NSPC is suitable for the texts and lines with irregular texture direction, which is often appeared in the screen-captured contents. The PM selects a number of prediction samples, i.e., the major colors, out of the current CU to represent each pixel samples in the CU. However, the PM requires to transmit the prediction samples to the decoder by a lossless coding, and thus the syntax of the PM would code a huge amount of bits to carry the information on the prediction samples. Differing from the PM, the NSPC uses the neighboring samples to predict the pixels of the CU. Because the neighboring samples from the neighboring blocks are available for the decoder, the syntax of the NSPC does not carry the information on the prediction samples, and thus NSPC results in a shorter syntax than PM.

4 Analysis for Neighboring Sample Prediction Coding

The number of neighboring samples for NSPC is chosen between one and ten as mentioned above. The number of major color samples for the PM, however, is set by the value between one and thirty-one. A large number of samples for each tool would be required when the CU is composed of much more colors, but a small number of samples would be preferable as the CU only covers a few colors. To understand the percentage of the number of samples for the NSPC-based the PM-based CUs, we analyze both tools under the sequence category of "text & graphic with motion", which

Table 2. Percentages of the number of the major color samples employed for the PM-based CU for different screen-content sequences

Sequences	The number of samples			
	1	2	3	4 ~ 31
YUV, FlyingGraphics, 1080p	0.54 %	15.27 %	21.02 %	63.17 %
YUV, Desktop, 1080p	0.06 %	30.33 %	6.00 %	63.61 %
YUV, Console, 1080p	0.13 %	48.36 %	12.60 %	38.91 %
YUV, web_browsing, 720p	0.00 %	2.23 %	4.56 %	93.21 %
YUV, Map, 720p	0.37 %	4.88 %	6.47 %	88.28 %
YUV, Programming, 720p	0.71 %	38.01 %	5.10 %	56.18 %
YUV, SlideShow, 720p	0.87 %	46.14 %	13.71 %	39.28 %
RGB, FlyingGraphics, 1080p	0.64 %	13.07 %	19.57 %	66.72 %
RGB, Desktop, 1080p	0.03 %	22.62 %	6.94 %	70.41 %
RGB, Console, 1080p	0.20 %	43.91 %	15.16 %	40.73 %
RGB, web_browsing, 720p	0.05 %	2.63 %	4.72 %	92.60 %
RGB, Map, 720p	0.19 %	2.43 %	3.86 %	93.52 %
RGB, Programming, 720p	0.84 %	30.58 %	3.62 %	64.96 %
RGB, SlideShow, 720p	0.64 %	26.32 %	14.10 %	58.94 %

Table 3. Percentages of the number of the neighboring samples employed for the NSPC-based CU for different screen-content sequences

Sequences	The number of sample			
	1	2	3	4 ~ 10
YUV, FlyingGraphics, 1080p	85.71 %	1.41 %	1.70 %	11.18 %
YUV, Desktop, 1080p	44.12 %	37.07 %	14.01 %	4.8 %
YUV, Console, 1080p	6.63 %	63.40 %	20.86 %	9.11 %
YUV, web_browsing, 720p	96.89 %	0.21 %	0.38 %	2.52 %
YUV, Map, 720p	69.12 %	7.72 %	7.25 %	15.91 %
YUV, Programming, 720p	89.89 %	0.58 %	1.39 %	8.14 %
YUV, SlideShow, 720p	95.52 %	1.28 %	0.50 %	2.7 %
RGB, FlyingGraphics, 1080p	80.00 %	0.24 %	1.74 %	18.02 %
RGB, Desktop, 1080p	37.45 %	43.63 %	14.64 %	4.28 %
RGB, Console, 1080p	7.58 %	63.92 %	19.87 %	8.63 %
RGB, web_browsing, 720p	97.74 %	0.22 %	0.28 %	1.76 %
RGB, Map, 720p	76.41 %	1.66 %	3.43 %	18.50 %
RGB, Programming, 720p	93.47 %	0.63 %	0.67 %	5.23 %
RGB, SlideShow, 720p	95.81 %	1.32 %	0.56 %	2.31 %

comprises 14 screen-content video sequences for the SCC CTC issued by JCT-VC [8]. All-intra configuration and quantization parameter 22 are specified for simulations.

The percentage results for the PM-based CU and the NSPC-based CU are listed in Tables 2 and 3, respectively. In Table 2, there are around 60 ~ 90 % CUs in a frame to use 4 or above color samples for the PM, but less than 1 % CUs are represented by

1 color sample. The results suggest that the PM could efficiently address the CU composed of many diverse colors. However, Table 3 shows a converse results, where there are around 70 ~ 95 % CUs in a frame to use 1 or 2 neighboring samples for the NSPC, but around 2 ~ 15 % CUs are represented by 4 or above neighboring samples. The results in Table 3 infer that the NSPC would be more effective for the CU with few colors. This is attributes to the high possibility of the neighboring samples hitting the pixel samples in the current CU as the number of the pixel samples is rather less. According to the percentages of one sample in both tables, it suggests that the CU with one or few colors can be more efficiently predicted by the NSPC than the PM which requires a huge number of bits to signal the major colors to the decoder. The analyses show the importance of the single sample index skipping method for the NSPC mentioned in Sect. 3. Because of forcing to signal the major colors for the PM, the results in Table 2 and the bitrate will remain unchanged even if the single sample index skipping method is applied to the PM.

5 Signaling of Neighboring Sample Prediction Coding for HEVC SCC

The proposed NSPC is integrated into the reference software SCM2.0 [7] for evaluation in HEVC SCC. This section contributes to studying the syntax integration between the NSPC and the existing PM tool in HEVC SCC. We propose two implementation methods for the integration: one is to view the NSPC as an independent mode, and the other is to incorporate the NSPC into the existing PM. The former transmits the NSPC signals independently, and the latter inserts the signaling of the NSPC within the signaling of the PM.

The signaling [7] of the PM in the HEVC SCC can be shown in Fig. 2. A CU-level PM flag is signaled to indicate whether to enable the PM tool. If the PM tool is enabled, the major colors in the established major color table and the indexes of the pixels in the CU are coded and transmitted. The first implementation method is as shown in Fig. 3, which puts the signaling of the NSPC on the same level of the signaling of the PM. A CU-level NSPC flag is signaled to indicate whether to enable the independent NSPC tool. Suppose that the independent NSPC tool is enabled and the M -NS mode is selected. Then the number of M and the indexes of the pixels in the CU are coded and transmitted. The second implementation method is as shown in Fig. 4, which puts the signaling of the NSPC within the signaling of the PM. In this method, the NSPC signaling is incorporated into the PM signaling. Based on the CU-level NSPC flag, the PM tool can decide how to predict the pixel samples of the CU. If the CU-level NSPC flag is enabled, the PM selects the neighboring samples from the neighboring reference blocks to represent the pixel samples of the CU. Otherwise, the PM selects some of the pixel samples from the current CU to represent the whole pixel samples of the CU. This paper will take advantage of the both implementation methods of the NSPC for performance evaluation. The evaluation results will be discussed in Sect. 6, which shows that there is no conflicts between the NSPC and the PM even when the NSPC is implemented within the PM.

```

A coding unit (CU) {
  ...
  Signaling CU-level_PM_flag
  If (CU_level_PM_flag) {
    Signaling the major colors
    Signaling the indexes of the pixels in CU
  }
  ...
}

```

Fig. 2. The signaling of the PM in HEVC SCC

```

Suppose that M-NS mode is selected:
A coding unit (CU) {
  ...
  Signaling CU_level_NSPC_flag
  If(CU_level_NSPC_flag) {
    Signaling the selected value of M
    Signaling the indexes of the pixels in CU
  }
  else {
    ...
    Signaling CU-level_PM_flag
    If (CU_level_PM_flag) {
      Signaling the major colors
      Signaling the indexes of the pixels in CU
    }
    ...
  }
}

```

Fig. 3. The signaling integration of the NSPC and the PM by viewing the NSPC as an independent tool

6 Simulation Results

The proposed NSPC is evaluated based on the CTC [8] for SCC reference software version SCM2.0 [7] with the following configurations:

1. All-intra (AI) configuration is used.
2. The intra block copy (IBC) tool is enabled with search range up to full frame.
3. Four quantization values are evaluated: 22, 27, 32 and 37.
4. There are totally 26 test sequences [8], which are classified into ten categories according to the video content, video size, and color format. Among 26 sequences, those in class “text & graphic with motion” are the standard screen-content videos, and those in class “mixed content” are the videos composed of screen content and natural content.

BD-rate measure [9] is used to evaluate objective coding efficiency. A negative BD-rate number means that the tool has coding gains.

Table 4. BD-rate results of NSPC-I versus anchor SCM2.0 under AI-lossy coding

	All Intra		
	G/Y	B/U	R/V
RGB, text & graphics with motion, 1080p	-1.0 %	-0.9 %	-1.0 %
RGB, text & graphics with motion,720p	-1.0 %	-1.0 %	-1.0 %
RGB, mixed content, 1440p	-0.4 %	-0.3 %	-0.4 %
RGB, mixed content, 1080p	-0.4 %	-0.3 %	-0.3 %
RGB, Animation, 720p	0.0 %	0.0 %	0.0 %
RGB, camera captured, 1080p	-0.1 %	0.0 %	0.0 %
YUV, text & graphics with motion, 1080p	-0.7 %	-0.6 %	-0.6 %
YUV, text & graphics with motion,720p	-0.8 %	-0.8 %	-0.8 %
YUV, mixed content, 1440p	-0.5 %	-0.5 %	-0.6 %
YUV, mixed content, 1080p	-0.5 %	-0.4 %	-0.4 %
YUV, Animation, 720p	0.0 %	-0.2 %	-0.2 %
YUV, camera captured, 1080p	-0.1 %	-0.1 %	-0.1 %
Enc Time[%]	103 %		
Dec Time[%]	96 %		

```

Suppose that M-NS mode is selected:
A coding unit (CU) {
    ...
    Signaling CU-level_PM_flag
    If (CU_level_PM_flag) {
        Signaling CU_level_NSPC_flag
        If (CU_level_NSPC_flag) {
            Signaling the selected value of M
        }
        else {
            Signaling the major colors
        }
        Signaling the indexes of the pixels in CU
    }
    ...
}
    
```

Fig. 4. The signaling integration of the NSPC and the PM by inserting the NSPC signaling underneath the PM signaling

For comparisons, both versions of the NSPC discussed in Sect. 5 are evaluated in this section. The two versions are, respectively, denoted as NSPC-I and NSPC-II. The NSPC-I is the first implementation version which puts the NSPC signaling onto an identical level to the PM signaling. The NSPC-II is the second implementation version which puts NSPC underneath the PM signaling. The BD-rate results for NSPC-I and NSPC-II are, respectively, summarized in Tables 4 and 5. It is seen that the BD-rate saving for the NSPC-I and the NSPC-II can both achieve up to 1.0 % coding gain for

“text & graphic with motion”. The results in Table 5 show that the NSPC-II is no conflict with the PM when the NSPC is viewed as a special mode of the PM. It means that the NSPC without any signaling of the major colors can be beneficial to improve the coding performance of the PM which requires to signal a huge number of the major colors. Comparing the two implementation methods with each other, the results show that the NSPC-I as an independent mode is still superior to the NSPC-II in terms of the BD-rates.

Table 5. BD-rate results of NSPC-II versus anchor SCM2.0 under AI-lossy coding

	All Intra		
	G/Y	B/U	R/V
RGB, text & graphics with motion, 1080p	-0.9 %	-0.8 %	-0.8 %
RGB, text & graphics with motion,720p	-1.0 %	-0.9 %	-0.9 %
RGB, mixed content, 1440p	-0.4 %	-0.3 %	-0.4 %
RGB, mixed content, 1080p	-0.4 %	-0.3 %	-0.3 %
RGB, Animation, 720p	0.0 %	0.0 %	0.0 %
RGB, camera captured, 1080p	0.0 %	0.0 %	0.0 %
YUV, text & graphics with motion, 1080p	-0.6 %	-0.4 %	-0.5 %
YUV, text & graphics with motion,720p	-0.6 %	-0.7 %	-0.5 %
YUV, mixed content, 1440p	-0.4 %	-0.4 %	-0.4 %
YUV, mixed content, 1080p	-0.4 %	-0.3 %	-0.3 %
YUV, Animation, 720p	0.0 %	-0.2 %	-0.2 %
YUV, camera captured, 1080p	-0.1 %	-0.1 %	-0.1 %
Enc Time[%]	102 %		
Dec Time[%]	97 %		

Table 6. BD-rate results of the modified PM versus anchor SCM2.0 under AI-lossy coding

	All Intra		
	G/Y	B/U	R/V
RGB, text & graphics with motion, 1080p	-0.2 %	-0.2 %	-0.2 %
RGB, text & graphics with motion,720p	-0.2 %	-0.2 %	-0.1 %
RGB, mixed content, 1440p	-0.1 %	-0.2 %	-0.2 %
RGB, mixed content, 1080p	-0.2 %	-0.3 %	-0.2 %
RGB, Animation, 720p	0.0 %	-0.1 %	-0.1 %
RGB, camera captured, 1080p	0.0 %	0.0 %	0.0 %
YUV, text & graphics with motion, 1080p	-0.2 %	-0.4 %	-0.4 %
YUV, text & graphics with motion,720p	-0.1 %	-0.3 %	-0.6 %
YUV, mixed content, 1440p	-0.1 %	-0.4 %	-0.4 %
YUV, mixed content, 1080p	-0.1 %	-0.1 %	-0.3 %
YUV, Animation, 720p	0.0 %	-0.3 %	-0.2 %
YUV, camera captured, 1080p	0.0 %	0.0 %	0.0 %
Enc Time[%]	104 %		
Dec Time[%]	100 %		

The NSPC uses the proposed USSS to establish a sample list $\{N_1, N_2, \dots, N_{10}\}$, whose samples are composed of the neighboring samples out of the neighboring CUs. There are ten NS modes based the sample list to run the RD costs as described in Sect. 3. Then the best NS mode with the lowest RD cost will be selected. To verify the effectiveness of the NSPC being along with the USSS, we propose to modify the encoder of the PM in the original reference software SCM2.0 by adding a sample list $\{C_1, C_2, \dots, C_{10}\}$, whose samples are the major colors in the original major color table. Similarly, there are ten additional PM modes: $\{C_1\}, \{C_1, C_2\}, \dots, \{C_1, C_2, \dots, C_{10}\}$. Each PM mode has different size of major color table. Then the best PM mode with the lowest RD cost will be selected. Note that it is a non-normative modification of the PM in HEVC SCC. The results of the modified PM are summarized in Table 6. It shows that the modified PM can achieve up to 0.2 % BD-rate saving for the screen-captured video categories “text & graphic with motion”. However, there is 0.8 % BD-rate loss for the modified PM compared with the both versions of the NSPC. It verifies that encoder-only modification of the PM can achieve a minor gain, but cannot hit a higher gain achieved by the NSPC.

7 Conclusion

The proposed NSPC for HEVC SCC has been presented in this paper. The NSPC represents the samples of the CU by the indexes of samples selected from the neighboring CUs, such as the nearest CUs and previously encoded palette CU. The USSS based on a neighboring sample list is also proposed for determining the priority ordering of the selected neighboring samples. Two signaling methods of the NSPC are evaluated under the CTC for HEVC SCC, where the former is to view the NSPC as an independent mode, and the latter is to incorporate the NSPC into the existing PM mode. The experimental results show that both signaling methods of NSPC can improve the BD-rate saving of SCM2.0 up to 1.0 %, and the NSPC without any signaling of the major colors can be beneficial to improve the coding performance of the PM which requires to signal a huge number of the major colors. The analyses are also given in this paper to understand the statistical distribution of the number of the neighboring samples, and the results indicate that the NSPC is more effective than the PM for coding the CU with a small number of pixel samples. It verifies that there is no conflict between the NSPC and the PM on the BD-rate performance.

References

1. Joshi, R., Cohen, R., Yu, H.: BoG report on summary of objective performance and tools for SCC CfP responses. Document JCTVC-Q0239 (2014)
2. Boyce, J., Chen, J., Chen, Y., Flynn, D., Hannuksela, M.M., Naccari, M., Rosewarne, C., Sharman, K., Sole, J., Sullivan, G.J., Suzuki, T., Tech, G., Wang, Y.-K., Wegner, K., Ye Y.: Edition 2 Draft Text of High Efficiency Video Coding (HEVC), Including Format Range (RExt), Scalability (SHVC), and Multi-View (MV-HEVC) Extensions. Document JCTVC-R1013 (2014)

3. Joshi, R., Xu, J.: HEVC Screen Content Coding Draft Text 1. Document JCTVC-R1005 (2014)
4. Guo, X., Lu, Y., Li, S.: RCE4: Test1. Major-color-based screen content coding. Document JCTVC-P0108 (2014)
5. Pu, W., Guo, X., Onno, P., Lai, P., Xu, J.: AHG10: Suggested software for palette coding based on RExt6.0. Document JCTVC-Q0094 (2014)
6. Onno, P., Xiu, X., Huang, Y.-W., Joshi, R.: Suggested combined software and text for run-based palette mode. Document JCTVC-R0348 (2014)
7. Joshi, R., Xu, J., Cohen, R., Liu, S., Ma, Z., Ye, Y.: Screen content coding test model 2 encoder description (SCM2). Document JCTVC-R1014 (2014)
8. Yu, H., Cohen, R., Rapaka, K., Xu, J.: Common test conditions for screen content coding. Document JCTVC-R1015 (2014)
9. Bjøntegaard, G.: Calculation of average PSNR differences between RD curves. Document VCEG-M33 (2001)