Adjusted 4:2:2 Chroma Subsampling Strategy for Compressing Mosaic Videos with Arbitrary RGB Color Filter Arrays in HEVC

Chien-Hsiung Lin*, Kuo-Liang Chung*, and Jian-Ping Fang*

*Department of Computer Science and Information Engineering
National Taiwan University of Science and Technology, Taipei, Taiwan, R.O.C.
E-mail: d9409301@mail.ntust.edu.tw; klchung01@gmail.com; m10115008@mail.ntust.edu.tw

***Abstract*—To reduce the cost, most of surveillance cameras and web cameras are equipped with a single sensor covered by a red-green-blue (RGB) color filter array (CFA) for capturing only one RGB color component per pixel and hence produce so-called mosaic images. Conventionally, such cameras perform demosaicking, the transform from RGB domain to YUV domain, the chroma subsampling with 4:2:2 format, and the video coding to compress the captured mosaic images for storage saving and transmission over the Internet. However, in the conventional compression scheme, the commonly used 4:2:2 chroma subsampling strategies always sample U and V components from the fixed positions without considering the significance of the sampled U and V components for reconstructing R, G and B pixels, which results in the quality degradation of the reconstructed videos. In this paper, to remedy this weakness, we propose an adjusted 4:2:2 chroma subsampling strategy for compressing mosaic videos with arbitrary RGB-CFA structures in HEVC. The novelty of our work lies on that the positions of the sampled U and V components are dynamically adjusted according to the ordered significance of the U component for reconstructing B and G pixels as well as the V component for reconstructing R and G pixels. Experimental results demonstrate that the proposed adjusted 4:2:2 chroma subsampling strategy outperforms, in terms of the quality of the reconstructed mosaic videos and the reconstructed full-color videos, the commonly used ones in the cases of middle and high bitrate and is competitive with the commonly used ones in the case of low bitrate.**

***Index Terms*—Arbitrary RGB color filter arrays, Chroma subsampling, HEVC, Mosaic videos.**

I. Introduction

For the purposes of policing and social communication, surveillance cameras and web cameras have been widely used in human everyday life. To reduce the cost, these cameras are usually equipped with a single charge-coupled device (CCD) or complementary metal oxide semiconductor (CMOS) sensor, the surface of which is covered with a red-green-blue (RGB) color filter array (CFA), for capturing only one RGB color component per pixel. Since each pixel in the image frames of the acquired video has only one RGB color component, this kind of video is referred to as a mosaic video. For a mosaic video, the adopted RGB-CFA structure which means the arrangement of the color filters is demanded in the following video processing such as demosaicking [17], [15], [11], super-resolution [7], [12] or compression [8], [5], [6]. In general, the information about the adopted RGB-CFA structure can be obtained from the camera manufacturer or the header of the raw CFA video in TIFF-EP format. Ten typical types of RGB-CFA structures [16] are shown in Fig. 1, among which the Bayer CFA [1] is the most well-known RGB-CFA structure.

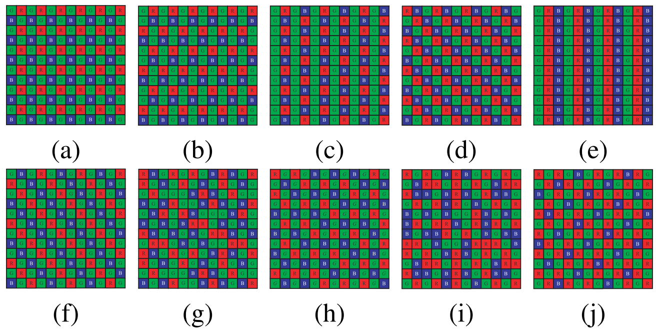


Fig. 1: Ten typical RGB-CFA structures: (a) Bayer CFA. (b) Lukac and Plataniotis CFA. (c) Yamanaka CFA. (d) Diagonal stripe CFA. (e) Vertical stripe CFA. (f) Modified Bayer CFA. (g) HVS-based CFA. (h) Type I Pseudo-random CFA. (i) Type II Pseudo-random CFA. (j) Type III Pseudo-random CFA.

Constrained by limited storage and bandwidth, compression of the acquired mosaic videos is necessary for the surveillance cameras and web cameras. In recent years, several compression schemes for mosaic videos with the Bayer CFA have been proposed. Gastaldi *et al.* [8] presented the first mosaic video coders for compressing the videos with the Bayer CFA under the MPEG-2 environment [9], where each mosaic image frame is decomposed into three distinct rectangular color planes which are individually compressed to increase the compression ratio. Since H.264/AVC video coders [4] provide better compression performance than MPEG-2 video coders, Doutre *et al.* [5] incorporated the structure conversion technique in [8] with the prediction schemes in H.264/AVC to compress the mosaic videos with the Bayer CFA. Based on the arrangement of the Bayer CFA, Doutre and Nasiopoulos [6] further modified the intra prediction scheme in H.264/AVC to enhance the quality of the reconstructed mosaic video.

978-616-361-823-8 © 2014 APSIPA

However, due to the difficulty in converting the mosaic videos with irregular RGB-CFA structures into three rectangular color planes, these structure conversion based compression schemes cannot be used to compress mosaic videos with non-Bayer CFAs.

As an alternative, researchers suggested that through performing the demosaicking, the RGB to YUV color domain transformation, and the chroma subsampling on mosaic videos with the Bayer CFA or non-Bayer CFAs, the resultant subsampled videos can be directly compressed using a general video coder, leading to a universal compression scheme for mosaic video with arbitrary RGB-CFA structures. In such a universal mosaic video compression scheme, the two key components influencing the quality of the reconstructed mosaic videos are the chroma subsampling strategy and the adopted video coder. Since most existing surveillance cameras and web cameras use the capturing application programming interfaces to access the camera driver and hence usually deliver video signals in uncompressed YUV 4:2:2 format, a 4:2:2 chroma subsampling strategy for implementing the above universal mosaic video compression scheme in surveillance cameras and web cameras is required. In addition, as the state-of-the-art video coding standard, high efficiency video coding (HEVC) [3], released, the HEVC video coders provide better compression performance than the H.264/AVC ones. Thus, the conventional approach is to adopt the commonly used 4:2:2 chroma subsampling strategies designed for full-color videos and the HEVC video coder into the universal mosaic video compression scheme. However, the commonly used 4:2:2 chroma subsampling strategies designed for full-color videos always sample the U and V components from the fixed positions without considering the characteristics of mosaic videos, resulting in a limited increase in quality of the reconstructed mosaic videos.

In this paper, we propose an adjusted 4:2:2 chroma subsampling strategy for compressing mosaic videos with arbitrary RGB-CFA structures in HEVC. Based on the ordered significance of the U component for reconstructing B and G pixels as well as the V component for reconstructing R and G pixels, we dynamically adjust the positions of the sampled U and V components for achieving better quality of the reconstructed mosaic videos. Experimental results on four test videos with the ten RGB-CFA structures demonstrate that the universal mosaic video compression scheme in HEVC using the proposed adjusted 4:2:2 chroma subsampling strategy does achieve, under the same bitrate, better quality of the reconstructed mosaic videos and the reconstructed full-color videos than the conventional universal mosaic video compression scheme.

The remainder of this paper is organized as follows. In Section II, we review the conventional universal mosaic video compression scheme in HEVC and look into the weakness of the adopted 4:2:2 chroma subsampling strategies. In Section III, we present the proposed adjusted 4:2:2 chroma subsampling strategy for compressing mosaic videos with arbitrary RGB CFA structures in HEVC. Based on four test videos with the ten typical RGB-CFA structures, Section IV gives the empirical results of the compression performance in terms of the quality of the reconstructed mosaic videos and the reconstructed full-color videos as well as the bitrate requirement. Concluding remarks are given in Section V.

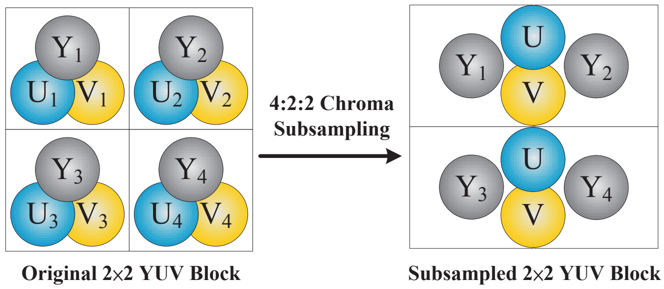


Fig. 2: Procedure of the 4:2:2 chroma subsampling.

II. Conventional Universal Mosaic Video Compression Scheme in HEVC and its 4:2:2 Chroma Subsampling Strategies

This section begins with a brief description of the conventional universal compression scheme for mosaic videos with arbitrary RGB-CFA structures in HEVC and then points out the weakness of the 4:2:2 chroma subsampling strategies used in the conventional compression scheme.

For each mosaic image I_M in the captured mosaic video with an arbitrary RGB-CFA, the conventional compression scheme, at the encoder side, first demosaics I_M by a universal demosaicing technique to obtain the demosaiced full-color RGB image and then performs the transform from RGB domain to YUV domain defined as

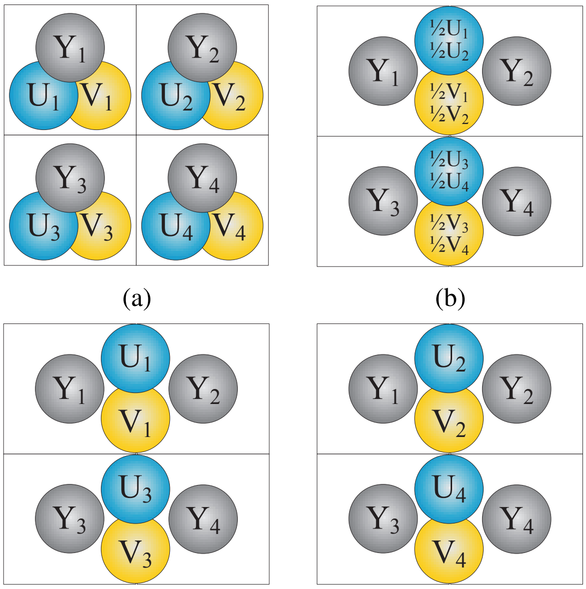
$$Y = 0.257R + 0.504G + 0.098B + 16 \quad (1)$$

$$U = -0.148R - 0.291G + 0.439B + 128 \quad (2)$$

$$V = 0.439R - 0.368G - 0.071B + 128 \quad (3)$$

to obtain the corresponding YUV image I_{YUV}. After this, a 4:2:2 chroma subsampling strategy is applied to the U and V chroma components in I_{YUV} such that for each 2×2 block in I_{YUV}, only one sampled U component and one sampled V component are kept per row, as shown in Fig. 2. Then the resultant subsampled YUV image is conveyed to the HEVC encoder for compression with 4:2:2 format.

In the conventional compression scheme, there exist three commonly used 4:2:2 chroma subsampling strategies, respectively, denoted by 4:2:2(A), 4:2:2(L), and 4:2:2(R) here and illustrated in Fig. 3. For the 4:2:2(A) shown in Fig. 3(b), the one sampled U and V components associated with each row of the 2×2 block are determined as the average of the two U and V components in the row, respectively. For the 4:2:2(L) shown in Fig. 3(c), the one sampled U and V components associated with each row of the 2×2 block are determined as the U and V components in the left column of the row, respectively. As for the 4:2:2(R) shown in Fig. 3(d), its determination process of the sampled U and V components is similar to that of the



(a) (b) (c) (d)

Fig. 3: Subsampling examples using the three commonly used 4:2:2 chroma subsampling strategies in the conventional compression scheme. (a) Original 2×2 YUV block. (b) Subsampled block by the 4:2:2(A). (c) Subsampled block by the 4:2:2(L). (d) Subsampled block by the 4:2:2(R).

4:2:2(L) but chooses the U and V components in the right column of the row.

At the decoder side, to reconstruct the compressed mosaic video, the HEVC decoder first reconstructs the subsampled YUV images with the specified 4:2:2 format and then the conventional scheme duplicates, respectively, the sampled U and V components as the discarded U and V components for each row of the 2×2 blocks in the reconstructed subsampled YUV images. Next, the transformation from YUV domain to RGB domain defined as

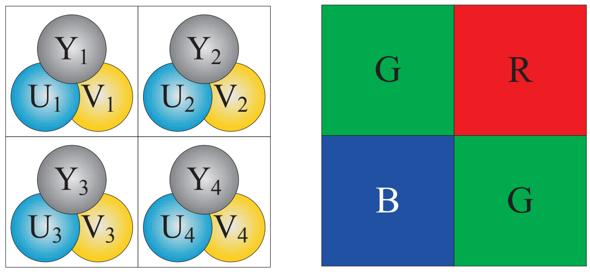
$$R = 1.164(Y - 16) + 1.596(V - 128) \quad (4)$$

$$G = 1.164(Y - 16) - 0.391(U - 128) - 0.813(V - 128) \quad (5)$$

$$B = 1.164(Y - 16) + 2.018(U - 128) \quad (6)$$

is performed on the reconstructed YUV images to obtain the reconstructed RGB images. Finally, the reconstructed mosaic video is delivered by mosaicking these reconstructed RGB images with the associated RGB-CFA structure.

Now we point out the weakness of the above three commonly used 4:2:2 chroma subsampling strategies. From Eqs. (4)–(6), it can be observed that at the decoder side, only reconstructing the R and G components involve the V component. Moreover, the coefficients of the V component in Eqs. (4) and (5) reveal that the V component possesses more impact on reconstructing the R component than on reconstructing the G component. Thus, to better reconstruct the compressed mosaic video, the two sampled V components



(a) (b)

Fig. 4: (a) One 2×2 YUV block and (b) its associated mosaic block.

for each 2×2 YUV block at the encoder side should be preferentially determined as the V component(s) of the pixel(s) corresponding to the R pixel(s) in the associated co-located mosaic block. Similarly, the two sampled U components for the current 2×2 YUV block should be preferentially determined as the U component(s) of the pixel(s) corresponding to the B pixel(s) in the associated co-located mosaic image block. However, the above three commonly used 4:2:2 chroma subsampling strategies ignore these observations and always sample the U and V components from the fixed positions, resulting in quality degradation of the reconstructed mosaic videos. To remedy this weakness, we propose an adjusted 4:2:2 chroma subsampling strategy where the positions of the sampled U and V components are dynamically adjusted according to the ordered significance of the U component for reconstructing B and G pixels as well as the V component for reconstructing R and G pixels, as will be shown in the next section.

III. Proposed Adjusted 4:2:2 Chroma Subsampling Strategy for Compressing Mosaic Videos with Arbitrary RGB-CFA Structures in HEVC

Given a mosaic video with an arbitrary RGB-CFA, after the demosaicing and the color domain transformation as described earlier, we have the associated YUV video. Considering a 2×2 block in the YUV image frame, let $\mathbb{B}^K_{\text{YUV}}(i,j), 1 \le i,j \le 2$, denote the value of component K ($K \in \{Y, U, V\}$) of the pixel at position (i,j) in which i and j indicate row i and column j of the current block, respectively. For the pixel at position (i,j) of the current YUV block, let $\mathbb{B}_M(i,j)$ denote the RGB color component of the pixel at the same position of the co-located 2×2 block in the associated mosaic image frame. For example, considering a 2×2 YUV block and its associated mosaic block shown in Fig. 4, we have $\mathbb{B}_M(1,1) = G$, $\mathbb{B}_M(1,2) = R$, $\mathbb{B}_M(2,1) = B$, and $\mathbb{B}_M(2,2) = G$.

For the current 2×2 YUV block, all the four Y components remain retained, but based on the proposed adjusted chroma subsampling strategy, the two pairs of the sampled U and V components, one for the upper 1×2 sub-block and the other for the lower 1×2 sub-block , are determined by dynamically adjusting the positions of the sampled U and V components according to the ordered significance of the U component for reconstructing B and G pixels as well as the V component for reconstructing R and G pixels. Let (U_u, V_u) denote the pair of the sampled U and V components for the upper sub-block and (U_l, V_l) the one for the lower sub-block. In what follows, we first describe how to determine (U_u, V_u) using the proposed adjusted chroma subsampling strategy.

Let $\mathbb{S}_u^C, C \in \{R, G, B\}$ denote the set of position(s) of the pixel(s) with $\mathbb{B}_M(i,j) = C$ in the upper sub-block and similarly define $\mathbb{S}_l^C$ for the lower sub-block. According to the observation from Eqs. (4)–(6), for better reconstructing the compressed mosaic video, when determining U_u, we preferentially consider the pixel(s) with $\mathbb{B}_M(i,j) = B$ in the upper sub-block and determine U_u by

$$U_u = \frac{1}{|\mathbb{S}_u^B|} \sum\nolimits_{(i,j)\in\mathbb{S}_u^B} \mathbb{B}_{\mathrm{YUV}}^U(i,j). \tag{7}$$

However, for the case that there exist no pixels with $\mathbb{B}_M(i,j) = B$ in the upper sub-block, i.e., $\left|\mathbb{S}_u^B\right| = 0$, determining U_u by Eq. (7) results in a calculation error. In this situation, further considering that the determination of U_u also impacts the reconstruction of the G component(s) in the upper sub-block and the spatial correlation of the U components of adjacent pixels with $\mathbb{B}_M(i,j) = B$ or $\mathbb{B}_M(i,j) = G$ in the lower sub-block can be exploited, we thus modify Eq. (7) into Eq. (8) for yielding U_u, where

$$w_1 = \frac{1}{\sqrt{2}} \cdot e^{-\frac{1}{2}{d_{GB}}^2} \tag{9}$$

with d_{GB} denoting the shortest Euclidean distance from the pixels at positions in $\mathbb{S}_u^G$ to the pixels at positions in $\mathbb{S}_l^B$.

Next, for the determination of V_u , since the sampled V component affects the reconstruction of the R component more than that of the G component, we preferentially refer to the V component(s) of the pixel(s) with $\mathbb{B}_M(i,j) = R$ in the upper sub-block if available; otherwise we refer to the V components of the pixels with $\mathbb{B}_M(i,j) = G$ in the upper sub-block and the pixels with $\mathbb{B}_M(i,j) = R$ and $\mathbb{B}_M(i,j) = G$ in the lower sub-block. Thus, we determine the sampled V component V_u for the upper sub-block by Eq. (10), where

$$w_2 = \frac{1}{\sqrt{2}} \cdot e^{-\frac{1}{2}{d_{GR}}^2} \tag{11}$$

with d_{GR} denoting the shortest Euclidean distance from the pixels at positions in $\mathbb{S}_u^G$ to the pixels at positions in $\mathbb{S}_l^R$. After determining (U_u, V_u) for the upper sub-block, following the similar principles, we can calculate (U_l, V_l) for the lower sub-block by completely replacing the subscript u by l and the subscript l by u in Eqs. (8) and (10) as well as in the definitions of d_{GB} and d_{GR}.

To better understand the above chroma subsampling procedure, we take an example to explain how to determine (U_u, V_u) and (U_l, V_l) for one 2×2 YUV block using the proposed adjusted 4:2:2 chroma subsampling strategy. Consider the 2×2 YUV block and its associated mosaic block shown in Fig. 4. First, since there exist no pixels with $\mathbb{B}_M(i,j) = B$ in the upper sub-block but exist one pixel with $\mathbb{B}_M(i,j) = G$ in the upper sub-block and one pixel with $\mathbb{B}_M(i,j) = B$ in the lower sub-block (i.e., $|\mathbb{S}_u^B| = 0$, $|\mathbb{S}_u^G| = 1$, and $|\mathbb{S}_l^B| = 1$), U_u is determined as $(1-w_1)\cdot U_1 + w_1 \cdot U_3$ where $w_1 = 0.242$ due to $d_{GB} = 1$. Next, since there exist one pixel with $\mathbb{B}_M(i,j) = R$ in the upper sub-block, V_u is directly determined as V_2. As for (U_l, V_l), U_l is determined as U_3 since there exist one pixel with $\mathbb{B}_M(i,j) = B$ in the lower sub-block, while V_l is determined as $(1 - w_2) \cdot V_4 + w_2 \cdot V_2$ with $w_2 = 0.242$ and $d_{GR} = 1$ since there exist no pixels with $\mathbb{B}_M(i,j) = R$ in the lower sub-block but exist one pixel with $\mathbb{B}_M(i,j) = G$ in the lower sub-block and one pixel with $\mathbb{B}_M(i,j) = R$ in the upper sub-block.

The same strategy can be applied to all the 2×2 YUV blocks in the image frames of the YUV video for the determination of their sampled U and V components. Although the positions of the sampled U and V components by the proposed adjusted chroma subsampling strategy are different from those by the three commonly used strategies shown in Figs. 3(b)–3(d), the resulting subsampled YUV images still can be compressed in 4:2:2 format by the HEVC encoder and the compressed mosaic video can be reconstructed at the decoder side via the same reconstruction process as described in Section II.

IV. Experimental Results

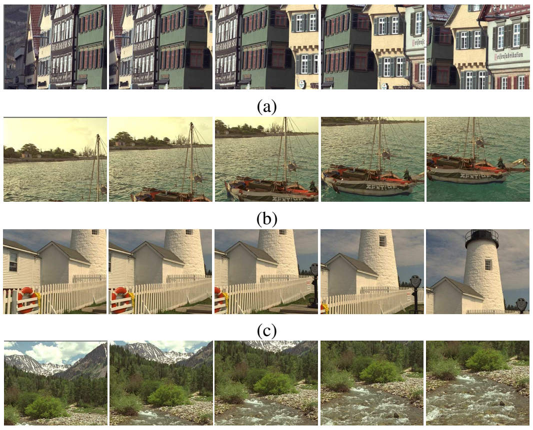
In order to evaluate the quality and compression performance of the proposed adjusted 4:2:2 chroma subsampling strategy, we conduct several experiments on the four test videos shown in Fig. 5. The four test videos, each with one hundred 352×288 RGB image frames normalized to 8-bit per color channel, are adopted from Kodak collection [13]. Following the standard practice in [14], [15], [10], [18], for each test video, we first subsample, according to the specific RGB-CFA structure, appropriate color for each pixel to generate the required mosaic videos with the ten RGB-CFA structures shown in Fig. 1. We then perform, in HEVC, the compression and reconstruction processes with the proposed adjusted 4:2:2 chroma subsampling strategy and the three commonly used ones, 4:2:2(A), 4:2:2(L), and 4:2:2(R), on these mosaic videos. Note that in the experiments, we adopt Yang *et al.*'s universal demosaicking technique [18] to recover the full RGB colors for the mosaic videos.

All the experiments are implemented on an IBM compatible computer with an Intel Core i7-3770 CPU 3.4GHz and 8GB RAM. The operating system is Microsoft Windows 7 64-bit operating system. The program development environment is Visual C++ 2008. The HEVC reference software used for video compression is HM-13.0-RExt-6.0. In addition, the GOP size is set to 8, the GOP structure adopts the random access main profile, the intra period is set to 32, and the ten different quantization parameters (QPs) considered in compression process are 2, 6, 10, 14, 18, 22, 26, 30, 34 and 38.

The quality of the reconstructed mosaic video and the storage requirement of the compressed video are the main performance measures for comparison. Since the mosaic video eventually needs to be reconstructed to the full-color video for display purpose, the quality of the reconstructed full-color video is also considered for comparison. Here, we adopt the average peak signal-to-noise ratio (PSNR) to measure

$$
U_u = \begin{cases} \frac{1}{|\mathbb{S}_u^B|}\sum_{(i,j)\in\mathbb{S}_u^B}\mathbb{B}_{\text{YUV}}^U(i,j), & \text{if } \left|\mathbb{S}_u^B\right| \neq 0, \\ (1-w_1)\times\frac{1}{|\mathbb{S}_u^G|}\sum_{(i,j)\in\mathbb{S}_u^G}\mathbb{B}_{\text{YUV}}^U(i,j) + w_1\times\frac{1}{|\mathbb{S}_l^B|}\sum_{(i,j)\in\mathbb{S}_l^B}\mathbb{B}_{\text{YUV}}^U(i,j), & \text{if } \left|\mathbb{S}_u^B\right| = 0 \text{ and } \left|\mathbb{S}_u^G\right|, \left|\mathbb{S}_l^B\right| \neq 0, \\ \frac{1}{|\mathbb{S}_u^G|}\sum_{(i,j)\in\mathbb{S}_u^G}\mathbb{B}_{\text{YUV}}^U(i,j), & \text{if } \left|\mathbb{S}_u^B\right|, \left|\mathbb{S}_l^B\right| = 0 \text{ and } \left|\mathbb{S}_u^G\right| \neq 0, \\ \frac{1}{|\mathbb{S}_l^B|}\sum_{(i,j)\in\mathbb{S}_l^B}\mathbb{B}_{\text{YUV}}^U(i,j), & \text{if } \left|\mathbb{S}_u^B\right|, \left|\mathbb{S}_u^G\right| = 0 \text{ and } \left|\mathbb{S}_l^B\right| \neq 0, \\ \frac{1}{|\mathbb{S}_l^G|}\sum_{(i,j)\in\mathbb{S}_l^G}\mathbb{B}_{\text{YUV}}^U(i,j), & \text{otherwise.} \end{cases} \tag{8}
$$

$$
V_u = \begin{cases} \frac{1}{|\mathbb{S}_u^R|}\sum_{(i,j)\in\mathbb{S}_u^R}\mathbb{B}_{\text{YUV}}^V(i,j), & \text{if } \left|\mathbb{S}_u^R\right| \neq 0, \\ (1-w_2)\times\frac{1}{|\mathbb{S}_u^G|}\sum_{(i,j)\in\mathbb{S}_u^G}\mathbb{B}_{\text{YUV}}^V(i,j) + w_2\times\frac{1}{|\mathbb{S}_l^R|}\sum_{(i,j)\in\mathbb{S}_l^R}\mathbb{B}_{\text{YUV}}^V(i,j), & \text{if } \left|\mathbb{S}_u^R\right| = 0 \text{ and } \left|\mathbb{S}_u^G\right|, \left|\mathbb{S}_l^R\right| \neq 0, \\ \frac{1}{|\mathbb{S}_u^G|}\sum_{(i,j)\in\mathbb{S}_u^G}\mathbb{B}_{\text{YUV}}^V(i,j), & \text{if } \left|\mathbb{S}_u^R\right|, \left|\mathbb{S}_l^R\right| = 0 \text{ and } \left|\mathbb{S}_u^G\right| \neq 0, \\ \frac{1}{|\mathbb{S}_l^R|}\sum_{(i,j)\in\mathbb{S}_l^R}\mathbb{B}_{\text{YUV}}^V(i,j), & \text{if } \left|\mathbb{S}_u^R\right|, \left|\mathbb{S}_u^G\right| = 0 \text{ and } \left|\mathbb{S}_l^R\right| \neq 0, \\ \frac{1}{|\mathbb{S}_l^G|}\sum_{(i,j)\in\mathbb{S}_l^G}\mathbb{B}_{\text{YUV}}^V(i,j), & \text{otherwise.} \end{cases} \tag{10}
$$



(a)
(b)
(c)
(d)

Fig. 5: Four test videos: (a) *Houses*. (b) *Boat*. (c) *Lighthouse*. (d) *Nature*.

the quality of the one hundred mosaic image frames in the reconstructed video. Denote by $\mathbb{P} = \{(m,n)|1 \leq m \leq H, 1 \leq n \leq W\}$ the set of pixel coordinates in one image frame of size $W \times H$. The PSNR of the reconstructed mosaic video with N image frames of size $W \times H$ is expressed as

$$
\text{PSNR} = \frac{1}{N}\sum_{n=1}^{N} 10\log_{10}\frac{255^2}{\frac{1}{WH}\sum\limits_{p\in\mathbb{P}}\left[F_M^n(p) - \widetilde{F}_M^n(p)\right]^2}, \tag{12}
$$

where $F_M^n(p)$ denotes the color value of the pixel at position p in the n-th mosaic image frame of the original mosaic video and $\widetilde{F}_M^n(p)$ represents the reconstructed analogue. To evaluate the quality of the reconstructed full-color video, the original mosaic video and the reconstructed mosaic video are first demosaicked to full-color videos and then the color PSNR (CPSNR) is calculated between them. The CPSNR of the reconstructed full-color video is expressed as

$$
\text{CPSNR} = \frac{1}{N}\sum_{n=1}^{N} 10\log_{10}\frac{255^2}{\text{CMSE}} \tag{13}
$$

with

$$
\text{CMSE} = \frac{1}{3WH}\sum_{p\in\mathbb{P}}\sum_{C\in\{R,G,B\}}\left[F_{RGB}^{n,C}(p) - \widetilde{F}_{RGB}^{n,C}(p)\right]^2 \tag{14}
$$

where $F_{RGB}^{n,C}(p)$ and $\widetilde{F}_{RGB}^{n,C}(p)$ denote the $C \in \{R,G,B\}$ color value of the pixel at position p in the n-th image frame of the original and reconstructed full-color videos, respectively. With the frame rate F, the bitrate of the compressed video is defined as

$$
\text{bitrate} = \frac{T}{N} \times F, \tag{15}
$$

where T represents the total number of bits used for compressing the mosaic video with N frames. Higher values of PSNR and CPSNR indicate, respectively, better quality of the reconstructed mosaic video and the reconstructed full-color video, while lower values of bitrate imply less storage requirement for the compressed video.

To provide quantitative comparison results, Bjøntegaard delta PSNR (BD-PSNR) [2] is used to measure the average difference in PSNR or CPSNR over similar values of bitrate between two compared 4:2:2 chroma subsampling strategies. The results of BD-PSNR for the 4:2:2(L), the 4:2:2(R), and the proposed adjusted 4:2:2 chroma subsampling strategy are tabulated in Table 1, in which the 4:2:2(A) is supposed as the comparison basis. A positive value of BD-PSNR in Table 1 indicates, compared with the 4:2:2(A), the quality superiority of the corresponding 4:2:2 chroma subsampling strategy for compressing mosaic videos in HEVC. It is clear that the proposed subsampling strategy almost yields better quality of the reconstructed mosaic videos and the reconstructed full-color videos than the 4:2:2(A), especially in the low QP situation. Although the BD-PSNR performance of the proposed subsampling strategy decays for the high QP situation, it seldom occurs in practice since encoding mosaic videos produced from surveillance cameras or web cameras usually needs to keep higher reconstructed quality for the purposes of life policing and social communication. In contrast, the 4:2:2(L) and the 4:2:2(R) yield negative values of BD-PSNR for all the concerned QP situations. This is mainly because determining the sampled U and V components by selecting the U and V components from single fixed positions actually discards more chroma information than that by averaging the U and V components in all available positions and hence results in severe quality loss of the reconstructed videos.

In summary, the proposed adjusted 4:2:2 chroma subsampling strategy for compressing mosaic videos with arbitrary RGB-CFA structures do have sufficient quality benefit in practice, especially for low and middle QP situations.

V. Conclusion

In this paper, we have presented an adjusted 4:2:2 chroma subsampling strategy for compressing mosaic videos with arbitrary RGB-CFA structures in HEVC. For the determination of the sampled U and V chroma components, different from the three commonly used 4:2:2 chroma subsampling strategies which always sample the U and V components from the fixed positions, we dynamically adjust the positions of the sampled U and V components according to the ordered significance of the U component for reconstructing B and G pixels as well as the V component for reconstructing R and G pixels. Experimental results on the four test videos with the ten RGB-CFA structures demonstrate that the proposed adjusted 4:2:2 chroma subsampling strategy outperforms, in terms of the quality of the reconstructed mosaic videos and the reconstructed full-color videos, the three commonly used ones in the cases of middle and high bitrate. Although the proposed strategy may achieve slightly lower quality in the case of low bitrate, this case seldom occurs in the practical usage of surveillance cameras and web cameras. We conclude that the proposed adjusted 4:2:2 chroma subsampling strategy is practical and effectively improves the quality of the reconstructed mosaic videos and the reconstructed full-color videos.

Acknowledgment

The work of C. H. Lin and K. L. Chung was supported by the National Science Council of R. O. C. under the Contract 102-2221-E-011-055-MY3.

References

[1] B.E. Bayer, Color imaging array, U.S. Patent No. 3971065, 1976.

[2] G. Bjontegaard, Calculation of average PSNR differences between RD curves, VECG-M33, ITU-T VECG meeting, Austin, Texas, 2–4, 2001.

[3] B. Bross, W.J. Han, J.R. Ohm, G.J. Sullivan, and T. Wiegand, High efficiency video coding (HEVC) text specification draft 10, document JCTVC-L1003, 2013.

[4] Draft ITU-T Recommendation and Final Draft International Standard of Joint Video Specification, document ITU-T Rec. H.264/ISO/IEC 14 496-10 AVC, Joint Video Team of ISO/IEC and ITU-T, 2003.

[5] C. Doutre, P. Nasiopoulos, K.N. Plataniotis, H.264-based compression of Bayer pattern video sequences, IEEE Trans. Circuits and Systems for Video Technology 18(6) (2008) 725–734.

[6] C. Doutre, P. Nasiopoulos, Modified H.264 intra prediction for compression of video and images captured with a color filter array, in Proc. of IEEE International Conference on Image Processing, 2009, pp. 3401–3404.

[7] S. Farsiu, M. Elad, P. Milanfar, Multiframe demosaicing and super-resolution of color images, IEEE Trans. Image Processing 15(1) (2006) 141–159.

[8] F. Gastaldi, C.C. Koh, M. Carli, A. Neri, S.K. Mitra, Compression of videos captured via Bayer patterned color filter arrays, in Proc. of 13th European Signal Processing Conference, 2005, pp. 983–992.

[9] Generic coding of moving pictures and associated audio (MPEG-2), *ISO/IEC 13818*, 1995.

[10] B.K. Gunturk, J. Glotzbach, Y. Altunbasak, R.W. Schaffer, R.M. Murserau, Demosaicking: color filter array interpolation, IEEE Signal Processing Magazine 22(1) (2005) 44–54.

[11] D. Heiss-Czedik, R. Huber-Mörk, D. Soukup, H. Penz, B.L. García, Demosaicing algorithms for area- and line-scan cameras in print inspection, Journal of Visual Image Communicaiton and Representation 20(6) (2009) 389–398.

[12] B.K. Karch, R.C. Hardie, Adaptive wiener filter super-resolution of color filter array images, Optics Express 21(16) (2013) 18820–18841.

[13] Kodak True Color Image Collection [Online]. Available: http://r0k.us/graphics/kodak/.

[14] R. Lukac, K. Martin, K.N. Plataniotis, Digital camera zooming based on unified CFA image processing steps, IEEE Trans. Consumer Electronics 50(1) (2004) 15–24.

[15] R. Lukac, K.N. Plataniotis, Universal demosaicking for imaging pipelines with an RGB color filter array, Pattern Recognition 38(11) (2005) 2208–2212.

[16] R. Lukac, K.N. Plataniotis, Color filter arrays: Design and performance analysis, IEEE Trans. Consumer Electronics 51(4) (2005) 1260–1267.

[17] S.C. Pei, I.K. Tam, Effective color interpolation in CCD color filter arrays using signal correlation, IEEE Trans. Circuits and Systems for Video Technology 13(6) (2003) 503–513.

[18] W.J. Yang, K.L. Chung, W.N. Yang, L.C. Lin, Universal chroma subsampling strategy for compressing mosaic video sequences with arbitrary RGB color filter arrays in H.264/AVC, IEEE Trans. Circuits and Systems for Video Technology 23(4) (2013) 591–606.

TABLE 1: BD-PSNR results (in dB) of the 4:2:2(L), the 4:2:2(R), and the proposed adjusted 4:2:2 chroma subsampling strategy over the 4:2:2(A) for the four test videos and different QP intervals.

QP Interval	Test Video	BD-PSNR					
		Reconstructed mosaic video			Reconstructed full-color video		
		4:2:2(L)	4:2:2(R)	Proposed	4:2:2(L)	4:2:2(R)	Proposed
2–14	Houses	-1.7884	-1.2509	2.3242	-1.6226	-0.9451	2.0395
	Boat	-1.1596	-1.0446	1.1218	-1.0234	-0.7981	1.0170
	Lighthouse	-1.3475	-1.0019	1.6263	-1.2808	-0.7087	1.5264
	Nature	-1.3141	-1.1337	1.5654	-1.1956	-0.9114	1.3496
	Average	-1.4024	-1.1078	1.6594	-1.2806	-0.8408	1.4831
10–22	Houses	-1.0181	-0.6385	1.0297	-0.9893	-0.4805	0.9502
	Boat	-0.5277	-0.4937	0.3548	-0.5084	-0.3823	0.3686
	Lighthouse	-0.7264	-0.5284	0.6374	-0.7576	-0.3576	0.6932
	Nature	-0.5742	-0.5263	0.4932	-0.5708	-0.4347	0.4501
	Average	-0.7116	-0.5467	0.6288	-0.7065	-0.4138	0.6155
18–30	Houses	-0.4786	-0.2780	0.3383	-0.5031	-0.2088	0.3359
	Boat	-0.2099	-0.2024	0.0556	-0.2178	-0.1637	0.0758
	Lighthouse	-0.3546	-0.2813	0.1441	-0.4052	-0.1920	0.2140
	Nature	-0.1967	-0.1918	0.0365	-0.2139	-0.1684	0.0318
	Average	-0.3100	-0.2384	0.1436	-0.3350	-0.1832	0.1644
26–38	Houses	-0.1766	-0.1073	0.0315	-0.1933	-0.0834	0.0420
	Boat	-0.0919	-0.0790	-0.0480	-0.0951	-0.0664	-0.0363
	Lighthouse	-0.1735	-0.1540	-0.0587	-0.2050	-0.1175	-0.0188
	Nature	-0.0727	-0.0623	-0.0611	-0.0769	-0.0567	-0.0591
	Average	-0.1287	-0.1006	-0.0341	-0.1425	-0.0810	-0.0181
Average		-0.6382	-0.4984	0.5994	-1.2323	-0.3797	0.5612