Effective Content-Aware Chroma Reconstruction Method for Screen Content Images

Kuo-Liang Chung, *Senior Member, IEEE*, Yan-Cheng Liang, and Ching-Sheng Wang

***Abstract*— In this paper, we propose an effective novel content-aware chroma reconstruction (CACR) method for screen content images (SCIs). After receiving the decoded downsampled YUV image on the client side, our fast chroma-copy approach reconstructs the missing chroma pixels in the flat regions of SCI. Then, for non-flat regions, a non-flat region-based winner-first voting (NRWV) strategy is proposed to identify the chroma subsampling scheme used on the server side prior to compression. Further, an effective adaptive hybrid approach is proposed to reconstruct each missing chroma pixel in the non-flat region by fusing the two reconstructed results, one from our modified NRWV-based chroma subsampling-binding and luma-guided chroma reconstruction scheme, which favors the sharp edges in SCI, as well as the other from the bicubic interpolation scheme, which favors blurred and continuous-tone textures. Further, based on the identified chroma subsampling scheme, a geometry alignment-based error compensation approach is proposed to enhance the reconstructed chroma image. Based on typical test SCIs and JCT-VC screen content videos, comprehensive experiments are carried out in HEVC-16.17 to demonstrate that in terms of quality, visual effect, and quality-bitrate tradeoff of the reconstructed SCIs, our CACR method significantly outperforms the existing state-of-the-art methods.**

***Index Terms*— Chroma-copy, content-aware chroma reconstruction, chroma subsampling identification, high efficiency video coding (HEVC), quantitative and qualitative quality, quality-bitrate tradeoff, screen content image (SCI).**

I. Introduction

DUE to the increasing popularity of mobile phones, tablets, and touch-screen laptops in the consumer electronics market, computer-generated screen content images (SCIs) have received growing attention in many applications, such as cloud computing, remote conferencing, and screen sharing. In addition to the camera-captured nature subimages with blurred edges, fat lines, and continuous-tones, SCIs often comprise flat regions, sharp edges, texts, repeated patterns, thin lines, and computer-rendered graphics with few colors.

Because the human visual perception system is more sensitive to luminance than to chroma, prior to compression, subsampling the chroma image is an effective way to achieve better compression performance by the encoder [6]. From the following YUV-to-RGB color conversion, we have

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 1.164 & 0 & 1.596 \\ 1.164 & -0.391 & -0.813 \\ 1.164 & 2.018 & 0 \end{bmatrix} \begin{bmatrix} Y-16 \\ U-128 \\ V-128 \end{bmatrix} \quad (1)$$

where Y denotes the luma image and UV denotes the chroma image. From Eq. (1), we know that all R, G, and B color components contain the luminance part. Therefore, encoding the RGB image directly is not suggested. Accordingly, prior to compression, the input RGB full-color SCI is first transformed to a YUV image. The commonly used chroma subsampling formats include 4:4:4, 4:2:2, and 4:2:0. For each 2×2 UV block, the 4:4:4 scheme has no compression and maintains both luma and chroma data entirely; this approach is often used for video games, high-end display devices, and post-production in studio. The 4:2:2 scheme determines one sampled U and V components for each row of the 2×2 UV block. To achieve better compression performance for fitting the limited internet bandwidth, instead of 4:4:4 and 4:2:2, 4:2:0 is adopted; this scheme has been widely used in Blu-ray discs (BDs), digital versatile discs (DVDs), movies, sports, TV shows, and remote conferencing; the 4:2:0 scheme determines one sampled U and V components of the 2×2 UV block as the subsampled (U, V)-pair. Considering the aforementioned applications of SCIs, our research considers only the 4:2:0 format.

The five most commonly used chroma 4:2:0 subsampling schemes are 4:2:0(A), 4:2:0(L), 4:2:0(R), 4:2:0(DIRECT), and 4:2:0(MPEG-B). 4:2:0(A) subsamples the (U, V)-pair by averaging the U and V components of the 2×2 UV block. 4:2:0(L) and 4:2:0(R) subsample the (U, V)-pair by averaging the chroma components in the left column and right column, respectively, of the 2×2 UV block. 4:2:0(DIRECT) selects the top-left (U, V)-pair of the 2×2 UV block as the subsampled solution. 4:2:0(MPEG-B) determines the subsampled (U, V)-pair by performing a 13-tap filter with mask [2, 0, −4, −3, 5, 19, 26, 19, 5, −3, −4, 0, 2]/64 on the top-left location of the 2×2 UV block. For completeness, we also include the two currently existing chroma subsampling schemes CS_{BIL} and CS_{BIC} [17], which are introduced in the fifth paragraph of Section I.A. For convenience, we designate the seven aforementioned chroma subsampling schemes by the set **CS** = {4:2:0(A), 4:2:0(L), 4:2:0(R), 4:2:0(DIRECT), 4:2:0(MPEG-B), CS_{BIL}, and CS_{BIC}}.

Manuscript received December 28, 2017; revised July 20, 2018 and August 28, 2018; accepted October 5, 2018. Date of publication October 10, 2018; date of current version October 29, 2018. This work was supported by MOST under Grant 105-2221-E-011-118-MY3. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Feng Wu. *(Corresponding author: Kuo-Liang Chung.)*

The authors are with the Department of Computer Science and Information Engineering, National Taiwan University of Science and Technology, Taipei 10672, Taiwan (e-mail: klchung01@gmail.com).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIP.2018.2875340

Accordingly, the subsampled YUV image is fed into the encoder, e.g. High Efficiency Video Coding (HEVC) reference software HM-16.17+SCM-8.6 used in this research, and then the encoded subsampled YUV image is transmitted to the decoder via the Internet. On the client side, the decoded subsampled UV image must be upsampled to the original size; then, the upsampled chroma image and the decoded equal-sized Y image compose the reconstructed YUV image. Finally, the target RGB full-color image is reconstructed by applying Eq. (1) to the reconstructed YUV image. In this research, we focus on the key issue: how to upsample the decoded subsampled UV image effectively for SCIs.

A. Motivation

We first introduce the 14 existing chroma reconstruction methods in **CR** = {COPY, BIL (bilinear interpolation), BIC (bicubic interpolation), 8-TAP, NEDI (new edge-directed interpolation) [11], SAI (soft decision adaptive interpolation) [21], LAU (luma-assisted chroma upsampling) [10], GF (guided-filter method) [8], GCR (guided chroma reconstruction method) [16], LACDU (luma aware chroma downsampling and upsampling) [17], ASBLG (adaptive chroma subsampling-binding and luma-guided method) [2], sparse coding-based super-resolution (ScSR) [19], cascade of sparse coding-based network (CSCN) [18], modified super-resolution convolutional neural network (MSRCNN) [5]}. In **CR**, COPY, BIL, BIC, 8-TAP, NEDI, and SAI are luma-independent approaches, whereas LAU, GF, GCR, LACDU, and ASBLG are luma-dependent approaches. ScSR is a sparse representation approach. CSCN and MSRCNN are deep learning-based approaches. This subsection ends with an explanation of the motivation of the research of this paper. All 14 methods in **CR** are included in the comparative methods to show the superiority of our proposed method in terms of image quality.

In COPY, the three missing chroma pixels of the 2×2 chroma block copy the subsampled (U, V)-pair. BIL, BIC, and NEDI [11] applied the bilinear, bicubic, and modified Wiener filter-based interpolations, respectively, to reconstruct the chroma image. As a building function in HEVC, 8-TAP with 1/2 motion precision applies the mask $[-1, 4, -11, 40, 40, -11, 4, -1]/64$, which is centered at each missing chroma pixel, to reconstruct all the missing chroma pixels. SAI [21] interpolates the missing chroma pixels by training the relationship between the known and the missing chroma pixels, but also between the missing chroma pixels themselves. For SCIs, NEDI has better quality performance than SAI, but for nature images, SAI is superior to NEDI. The common weakness of the abovementioned six luma-independent chroma reconstruction methods is that they ignore the luma information as a way to enhance the quality of the reconstructed chroma image, leading to limited quality improvement.

To improve the CPSNR (color peak-signal-to-noise ratio) quality performance of BIL under 4:2:0(A), LAU [10] employs the mean squared luma difference between the neighboring chroma pixels of the current missing chroma pixel and the co-located luma pixel into BIL. The weakness of LAU is that it does not consider the textural correlation between luma and chroma images, which hinders quality improvement. Considering the correlation between the downsampled decoded luma image and the decoded downsampled chroma image, GF [8] uses the linear regression technique to enhance the reconstructed chroma quality. The weakness in GF is that for nature images, the correlation is not particularly high, leading to limited quality improvement.

For SCIs, GCR [16] uses 4-TAP to reconstruct the missing chroma pixels that correspond to the co-located non-textural luma block, and it uses the linear regression technique and the luma-guided approach to reconstruct the missing chroma pixels corresponding to the co-located textural luma block, thus satisfying one of the following two assumptions. The first assumption is that the block must have at most four different luma value; under this condition, if two luma pixels in the block have the same luma value, they also have the same chroma value. The second assumption is that the chroma component has the same (or inverse) texture as the luma component between the minimum and maximum values. Experimental results have shown the quality superiority of GCR over 4-TAP. However, the lack of the use of the chroma subsampling identification and the out-of-range problem, namely, the luma-guided anomaly problem [2], constrain the quality performance of GCR. Based on the winner-first voting strategy to identify the used chroma subsampling scheme and the solution to the out-of-range problem, ASBLG [2] outperforms the above-introduced chroma reconstruction methods for SCIs. However, ASBLG does not show favor to natural subimages in SCI, which often have blurred edges, thick lines, and continuous-tone colors, because the correlation between the subsampled luma and chroma block-pair is not particularly high, which degrades the performance quality.

In LACDU [17], Wang *et al.* first incorporated the BCIM (base color and index map) technique, i.e., the Palette mode [14], into the representations of the luma and chroma images; they proposed two effective combinations, CS_{BIL}-BIL and CS_{BIC}-BIC, in which CS_{BIL} and CS_{BIC} denote the bundled chroma subsampling schemes under the chroma reconstruction methods, BIL and BIC, respectively. Both combinations proposed by Wang *et al.* outperform the combination $IDID_{NEDI}$-NEDI [22], in which $IDID_{NEDI}$ indicates the bundled chroma subsampling scheme corresponding to the chroma reconstruction NEDI, and the other combinations such as 4:2:0(A)-BIL, 4:2:0(A)-BIC, and 4:2:0(A)-GF. For convenience and to avoid confusion, CS_{BIL}-BIL and CS_{BIC}-BIC are designated CS_{BIL}-LACDU and CS_{BIC}-LACDU, respectively. The constraint in LACDU is that it must bundle with CS_{BIL} or CS_{BIC}.

Unlike the aforementioned reconstruction methods, based on sparse representation, Yang *et al.* [19] presented the first sparse coding-based super-resolution (ScSR) method. Here, the two terms 'upsampling' and 'super-resolution' denote the same thing. To improve ScSR, Wang *et al.* [18] proposed a cascade of sparse-coding-based network (CSCN) in which the end-to-end training step adopts a multi-scale cost function. Recently, Dong *et al.* [5] proposed a modified super-resolution convolutional neural network-based (MSRCNN-based) method associated with a compact hourglass-shape

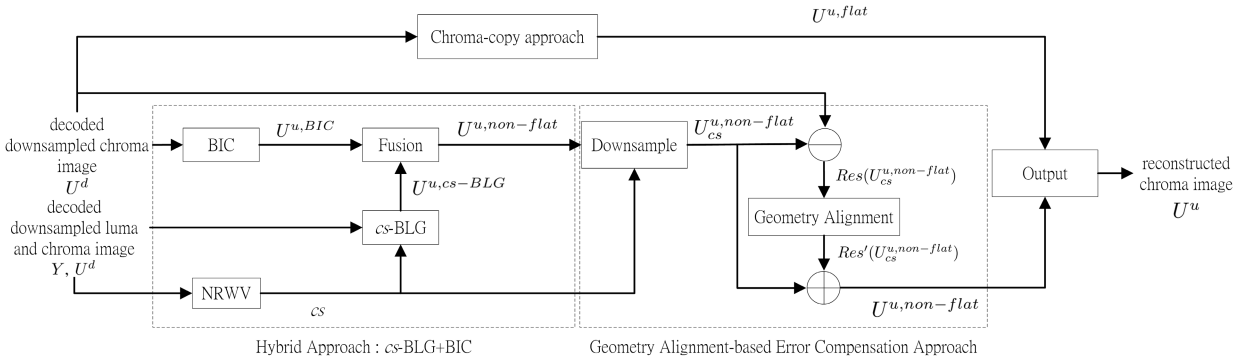


Fig. 1. The flowchart of the proposed CACR chroma reconstruction method.

CNN structure to improve the quality of the upsampled images by their previous SRCNN-based method [4]. The experimental results demonstrated that both CSCN and MSRCNN are superior to the previous method [15].

To overcome the weaknesses and constraints in the abovementioned chroma reconstruction methods, the main motivation of this research is to develop a novel and effective chroma reconstruction method for SCIs.

B. Contributions

In this paper, we propose a novel effective content aware-based chroma reconstruction (CACR) method for SCIs. Relative to the newly published work in [2], the main novel aspects and significant contributions of our CACR method include the following four aspects.

The first novel aspect of our CACR method is that we partition all missing chroma pixels in the initial upsampled chroma image into flat regions and non-flat regions. This is the main reason that our region-based chroma reconstruction method is also called the content-aware chroma reconstruction method. In the previous ASBLG method [2], all missing chroma pixels are treated the same. In our CACR method, all missing chroma pixels in flat regions are reconstructed by our simple chroma-copy approach, leading to satisfactory reconstruction quality and computation-saving merits because, on average, 50% of all missing chroma pixels are in flat regions.

The second novel aspect in CACR is that for those missing chroma pixels in non-flat regions, we propose a non-flat region-based winner-first voting (NRWV) strategy to identify what kind of chroma subsampling is used, say $cs \in$ **CS**, prior to compression. Relative to the previous voting strategy [2] in which the voting was performed on the flat and non-flat regions, our NRWV strategy has similar robustness in addition to computation-saving merit.

The third novel aspect in CACR is that a hybrid approach is proposed to reconstruct all the missing chroma pixels in the non-flat region by fusing the two reconstructed results, one from the *cs*-binding and luma-guided chroma reconstruction scheme, denoted by *cs*-BLG to distinguish from ASBLG [2], which benefits reconstructing sharp textural parts in computer-generated objects in SCIs, and the other from BIC, which benefits reconstructing the blurred textural parts and continuous-tone colors within objects.

The fourth novel aspect is that we propose a *cs*-binding and geometry alignment-based error compensation approach to further enhance the reconstructed chroma image quality.

In addition, the computational complexity analysis and actual execution time cost of the proposed CACR method are provided. Based on the 26 typical test SCIs [12], [17] and 9 JCT-VC SCVs [20], comprehensive experiments are carried out to demonstrate that in terms of CPSNR, CESIM (color edge similarity index) [13], visual effect, and quality-bitrate tradeoff of the reconstructed SCIs, our CACR method significantly outperforms the 14 abovementioned state-of-the-art methods. All experiments are realized in the HEVC test model HM-16.17+SCM-8.6 reference codec [1], [9].

Under the first five chroma subsampling schemes in **CS**, on average, our CACR method provides 1.84 dB and 1.49 dB CPSNR quality improvement for SCIs and SCVs, respectively, relative to ASBLG. In addition, our CACR method provides at least 3.28 dB and 2.07 dB CPSNR quality improvement for SCIs and SCVs, respectively, relative to any other chroma reconstruction method in **CR**\{ASBLG, LACDU}. In particular, for the reconstructed SCIs and SCVs, the combinations CS_{BIC}-CACR and CS_{BIL}-CACR provide the best CPSNR and CESIM quality performance, respectively. Note that the CS_{BIL} and CS_{BIC} chroma subsampling schemes were proposed by Wang *et al.* [17].

The rest of this paper is organized as follows. In Section II, our CACR method is presented. In Section III, several experiments are conducted to demonstrate the quality, quality-bitrate tradeoff, and visual effect merits of our CACR method. In Section IV, some concluding remarks are addressed.

II. Proposed Content-Aware Chroma Reconstruction Method: CACR

This section consists of five subsections. The four contributions of the our CACR method are presented in the first four subsections. In the fifth subsection, the related computational complexity analysis and the actual execution time cost are provided. The whole CACR method is shown in Fig. 1; for convenient depiction, we take only the decoded downsampled chroma U image as the input example, e.g., U^d, although the procedure can be easily applied to upsample the decoded downsampled chroma V image. We designate the final upsampled chroma U image $U^{u,CACR}$ and the luma image Y.

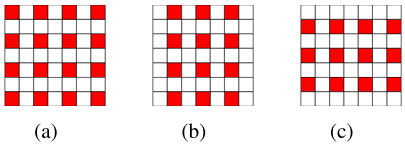


Fig. 2. Three block types. (a) Type A. (b) Type B. (c) Type C.

A. Chroma-Copy Approach to Reconstruct the Missing Chroma Pixels in Flat Regions

On the client side, we first map the decoded downsampled chroma image U^d to an upsampled one U^u whose size is four times as large as U^d, and each pixel $U^d(i, j)$ is moved to the location $(2i, 2j)$ in U^u. For each missing pixel $U^u(m, n)$, we apply a 7×7 window W centered at the location (m, n), and we let the configuration of the decoded downsampled pixels in W be the type A block with 16 downsampled chroma pixels in Fig. 2(a), where the red squares denote the downsampled chroma pixels, the type B block with 12 downsampled chroma pixels in Fig. 2(b), or the type C block with 12 downsampled chroma pixels in Fig. 2(c). The 'chroma-copy' condition used in the proposed chroma-copy approach is defined below.

Definition 1: For the missing pixel $U^u(m, n)$, if all the downsampled chroma pixels in the 7×7 window W have the same chroma value, say $U^{u,cc}(m, n)$, then the 'chroma-copy' condition is held, and the chroma-copy status map at location (m, n) is set to 1, i.e., $CC(m, n) = 1$; otherwise, we set $CC(m, n) = 0$.

If the block type of $U^u(m, n)$ is type A, as shown in Fig. 2(a), it takes 15 comparisons to identify whether the missing chroma pixel $U^u(m, n)$ is in one flat region. Considering the three block types in Fig. 2, on average, it takes 12.3$(= (15 + 11 + 11)/3)$ comparisons to identify whether $U^u(m, n)$ is in one flat region. Once the chroma-copy status map is set, the missing chroma pixel $U^u(m, n)$ with $CC(m, n) = 1$ is thus reconstructed by the value $U^{u,cc}(m, n)$; otherwise, $U^u(m, n)$ is reconstructed by the fused value determined by the proposed hybrid approach, as presented in the next subsection. After reconstructing all the missing pixels in flat regions by the proposed chroma-copy approach, let these reconstructed chroma pixels $U^u(i, j)$s with $CC(i, j) = 1$ in flat regions be denoted by $U^{u,flat}$ without any confusion.

According to the above description of the proposed chroma-copy approach, it is easy to verify that it takes constant operations, i.e., O(1) time, where the complexity definition in terms of big-O is suggested to refer to [3], to complete the chroma-copy for each missing chroma pixel $U^u(m, n)$ with $CC(m, n) = 1$. For a W×H SCI, it takes O(WH) time to complete the whole chroma-copy approach and assemble the complete chroma-copy status map.

B. Non-Flat Region-Based Winner-First Voting Strategy to Identify the Chroma Subsampling Scheme Used on the Server Side: NRWV

Before presenting the proposed hybrid method for reconstructing these missing chroma pixels in non-flat regions, we first present the proposed NRWV strategy, which modifies the previous voting strategy in [2] while offering faster performance and similar robustness. Differing from the voting strategy on the whole Y and UV images in [2], NRWV only performs voting on the non-flat regions to reduce the computational complexity. In NRWV, a CDD (correlation distortion degree) threshold is specified to discard invalid votes to further speed up the voting process while preserving similar robustness.

For the completeness of this subsection, we revisit the calculation of the CDD value in [2]. Let the correlation point set (CPS) representation denote the correlation between the subsampled decoded luma block Y_b^d and the co-located decoded subsampled chroma block U_b^d. Because $|\mathbf{CS}| = 7$, the seven correlations between the 5×5 subsampled chroma block U_b^d and the seven co-located 5×5 subsampled luma blocks, which are generated by performing the seven chroma subsampling schemes in **CS** on Y_b, form seven CPSs. For each CPS corresponding to one correlation between U_b^d and Y_b^d, in the 2-D space, the X-axis denotes the subsampled chroma value, and the Y-axis denotes the subsampled luma value. For the seven correlations, seven CDD values are calculated. First, according to each correlation, the fitting line can be obtained using the least square technique, and then we calculate the distance from each point in the CPS to the fitting line. Further, we sum up these calculated distances as the CDD value of that CPS. Consequently, seven CDD values are calculated for the seven CPSs.

The proposed NRWV procedure is listed below, as shown in Algorithm 1.

Algorithm 1 NRWV

Input: Y, U^d, CC, and T. */*CC denotes the chroma-copy status map, and T denotes the CDD threshold value for discarding the invalid vote*/*
Output: cs. */*the identified chroma subsampling scheme*/*
for *each non-flat location* (i, j) *with* $CC(i, j) = 0$ **do**
  for *each* $cs \in$ **CS do**
    Apply a 5×5 window centered at $Y(i, j)$.
    Generate the 5×5 downsampled luma block corresponding to the co-located 5×5 chroma block in U^u.
    Calculate the CDD value between the luma and chroma block-pair.
  Let cs_{min} be the chroma subsampling scheme with minimum CDD value, say CDD_{min}, among the concerned $|\mathbf{CS}|$ CDD values. Here, $|\mathbf{CS}| = 7$.
  if $CDD_{min} < T$ **then**
    Add one vote for the candidate chroma subsampling scheme cs_{min}, and add CDD_{min} to the accumulated CDD (ACDD) of cs_{min}.
Let cs be the final identified chroma subsampling scheme with maximum votes and minimum ACDD value.
return cs;

According to the description of our NRWV procedure, because we only consider each pixel location in the non-flat region with zero chroma-copy status, it takes O($|\mathbf{CS}|$) time

to compute the concerned |**CS**| CDD values and to determine the accumulated CDD (ACDD) value for the candidate chroma subsampling scheme cs_{min}. The total time complexity required in NRWV for processing the whole W×H SCI is bounded by O(|**CS**|αWH), in which |**CS**| = 8 and α (< 1) denotes the ratio of the number of pixels in the non-flat regions in the initial U^u to the image size W×H.

C. Hybrid Approach to Fuse the Missing Chroma Pixels in Non-Flat Regions: cs-BLG+BIC

In this subsection, we propose a novel hybrid approach that combines our improved chroma reconstruction method, *cs*-BLG, and BIC to maximize the quality of the reconstructed pixel for the missing chroma pixel $U^u(m, n)$ in the non-flat region. Specifically, the proposed hybrid approach compromises the quality of the reconstructed chroma pixel on the sharp textural part in computer-generated objects by *cs*-BLG and the reconstructed chroma pixel quality on the blurred textural part or continuous-tone colors by BIC. Based on the identified chroma subsampling information *cs* by NRWV, the proposed *cs*-BLG procedure is listed in Algorithm 2.

Algorithm 2 *cs*-BLG

Input: Y, U^d, and cs.
Output: $U^{u,cs-BLG}$. /*the reconstructed result by *cs*-BLG*/
for *each location* (i, j) *with* $CC(i, j) = 0$ **do**
  Apply a 5×5 window centered at $Y(i, j)$.
  Generate the 5×5 downsampled luma block by the chroma subsampling scheme cs, denoted by Y^d_{cs}, corresponding to the co-located 5×5 chroma block in U^u.
  Let $k = 5$ and Y^d_{block} be the set of downsampled luma values in the $k \times k$ downsampled luma block.
  while $Y(i, j) < min(Y^d_{block})$ *or* $Y(i, j) > max(Y^d_{block})$ **do**
    /*out-of-range anomaly alarm occurs*/
    if *the window size* $> 11 \times 11$ **then**
      break. /*jump out of the while loop*/
    Enlarge the window size from $k \times k$ to $(k + 2) \times (k + 2)$.
    Reset Y^d_{block}.
  Let (a, b) be the solved correlation parameter pair after applying the linear regression technique to determine the fitting line between the downsampled luma and chroma block-pair.
  $U^{u,cs-BLG}(i, j) = a \cdot Y(i, j) + b$. /*reconstruct the missing chroma pixel in the non-flat region*/
return $U^{u,cs-BLG}$;

Let these missing chroma pixels $U^u(i, j)$s with $CC(i, j) = 0$ in non-flat regions be denoted by $U^{u,non-flat}$. After performing the above *cs*-BLG method in Algorithm 2 on $U^{u,non-flat}$, the reconstructed chroma result is designated $U^{u,cs-BLG}$. Similarly, after performing BIC on the original $U^{u,non-flat}$, the reconstructed chroma result is designated $U^{u,BIC}$. For each missing chroma pixel $U^u(i, j)$ in the original non-flat region, to better fuse $U^{u,cs-BLG}(i, j)$ and $U^{u,BIC}(i, j)$, the estimation errors for $U^{u,cs-BLG}(i, j)$ and $U^{u,BIC}(i, j)$ are calculated in advance. Then, the proper weights for the two fused results can be determined such that the estimation accuracy of $U^u(i, j)$ can be maximized. If the error of $U^{u,cs-BLG}(i, j)$, namely $Err(U^{u,cs-BLG}(i, j))$, is larger than the error of $U^{u,BIC}(i, j)$, i.e., $Err(U^{u,BIC}(i, j))$, it implies that BIC provides more accurate estimation contribution to the missing chroma pixel $U^{u,non-flat}(i, j)$ than *cs*-BLG. In this case, $U^{u,BIC}(i, j)$ should be assigned a larger weight, say $W_{BIC}(i, j)$. In contrast, $U^{u,cs-BLG}(i, j)$ should be assigned a smaller weight, say $W_{cs-BLG}(i, j)$, when fusing the two reconstructed results, $U^{u,cs-BLG}(i, j)$ and $U^{u,BIC}(i, j)$, to estimate $U^u(i, j)$.

To calculate the two estimation errors, $Err(U^{u,cs-BLG}(i, j))$ and $Err(U^{u,BIC}(i, j))$, we take the decoded downsampled chroma image U^d as the comparative basis because U^d is the only data that we can receive from the decoder. It is known that $U^{u,cs-BLG}$ and $U^{u,BIC}$ are four times as large as U^d. Based on the identified chroma subsampling scheme cs, the two reconstructed upsampled chroma images, $U^{u,cs-BLG}$ and $U^{u,BIC}$, are first downsampled by the chroma subsampling scheme cs. Designate the two resultant downsampled chroma images $U^{d,cs-BLG}_{cs}$ and $U^{d,BIC}_{cs}$. The estimation error $Err(U^{d,cs-BLG}_{cs}(i, j))$ can be calculated by

$$Err(U^{d,cs-BLG}_{cs}(i, j)) = |U^d(i, j) - U^{d,cs-BLG}_{cs}(i, j)| \quad (2)$$

Similarly, the estimation error $Err(U^{d,BIC}_{cs}(i, j))$ is followed.

Therefore, the aforementioned weight $W_{U^{d,cs-BLG}_{cs}(i,j)}$ can be calculated by

$$W_{U^{d,cs-BLG}_{cs}(i,j)} = \frac{1/Err(U^{d,cs-BLG}_{cs}(i, j))}{1/Err(U^{d,cs-BLG}_{cs}(i, j)) + 1/Err(U^{d,BIC}_{cs}(i, j))} \quad (3)$$

Similarly, the weight $W_{U^{d,BIC}_{cs}(i,j)}$ is followed. By Eq. (3), the two weights for $U^{u,cs-BLG}(i', j')$ and $U^{u,BIC}(i', j')$, $(i', j') \in \{(2i, 2j), (2i+1, 2j), (2i, 2j+1), (2i+1, 2j+1)\}$, are given by

$$\begin{aligned} W_{U^{u,cs-BLG}(i',j')} &= W_{U^{d,cs-BLG}_{cs}(i,j)} \\ W_{U^{u,BIC}(i',j')} &= W_{U^{d,BIC}_{cs}(i,j)} \end{aligned} \quad (4)$$

By Eq. (4) and the two reconstructed results, $U^{u,cs-BLG}(i', j')$ and $U^{u,BIC}(i', j')$, the missing chroma pixel $U^{u,non-flat}(i', j')$ in the non-flat region can be fused by

$$\begin{aligned} U^{u,non-flat}(i', j') = W_{U^{u,cs-BLG}(i',j')} \times U^{u,cs-BLG}(i', j') \\ + W_{U^{u,BIC}(i',j')} \times U^{u,BIC}(i', j') \end{aligned} \quad (5)$$

According to the description of the above proposed hybrid approach, the computational complexity analysis is similar to that of NRWV. Because we only reconstruct each missing chroma pixel in the non-flat region and both *cs*-BLG and BIC take O(1) time for reconstructing each concerned missing chroma pixel, fusing the two results by Eq. (5) also takes

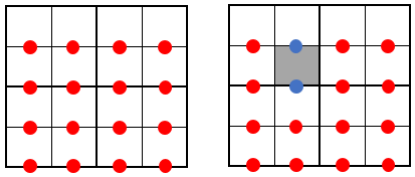


Fig. 3. Depiction of the proposed geometry alignment-based technique for error compensation. (a) Each pixel located in a non-pixel position in the upsampled residual image for cs = 4:2:0(L). (b) Geometry aligned re-estimated pixel in the upsampled residual image.

O(1) time. Therefore, the total time complexity required in the proposed hybrid approach is bounded by O(αWH).

D. cs-Binding Geometry Alignment-Based Error Compensation Approach to Further Enhance the Quality of the Upsampled Chroma Pixels in Non-Flat Regions

In this subsection, to further enhance the quality of the reconstructed result in non-flat regions, a new cs-binding geometry alignment-based error compensation approach is proposed. First, we perform chroma subsampling cs on the fused upsampled chroma image $U^{u,non-flat}$, which has been obtained by Eq. (5); the obtained downsampled image is designated $U_{cs}^{u,non-flat}$. Next, the residual between $U_{cs}^{u,non-flat}$ and the initial decoded downsampled chroma image U^d is computed by

$$Res(U_{cs}^{u,non-flat}) = U^d - U_{cs}^{u,non-flat} \tag{6}$$

Furthermore, BIC is applied to upsample the residual image $Res(U_{cs}^{u,non-flat})$ to obtain the upsampled residual image, which is used to enhance the fused upsampled chroma image $U^{u,non-flat}$.

To achieve better error compensation effects for $U^{u,non-flat}$, in what follows, a cs-binding and geometry alignment-based error compensation approach is proposed. For convenient exposition, we assume that cs = 4:2:0(L). After applying BIC to upsample $Res(U_{cs}^{u,non-flat})$, as shown in Fig. 3(a), each pixel in the upsampled result is depicted by a red circle located in the non-pixel position due to adopting chroma 4:2:0(L) subsampling. The key point of our cs-binding and geometry alignment-based error compensation approach is to apply BIC to re-estimate the upsampled residual image. As shown in Fig. 3(b), each residual pixel in the re-estimated upsampled residual image $Res'(U_{cs}^{u,non-flat})$ is denoted by a gray box by averaging the two vertical residual pixel values in blue in Fig. 3(b). The re-estimated residual pixel in gray is located exactly at the integer (x, y)-coordinate, i.e., the normal pixel position. Consequently, the upsampled chroma result for non-flat regions is compensated by

$$U^{u,non-flat} = U^{u,non-flat} + Res'(U_{cs}^{u,non-flat}) \tag{7}$$

Finally, the resultant upsampled chroma image is equal to the union of $U^{u,non-flat}$ in Eq. (7) and $U^{u,flat}$ obtained by our chroma-copy approach. It is straightforward to confirm that the total computational complexity is bounded by O(WH) after performing the chroma subsampling cs on the fused upsampled chroma image $U^{u,non-flat}$, the residual computation between $U_{cs}^{u,non-flat}$ and the decoded downsampled chroma image U^d, BIC to upsample the residual image $Res(U_{cs}^{u,non-flat})$, the linear interpolation to re-estimate the upsampled residual image, named $Res'(U_{cs}^{u,non-flat})$, and the error compensation in Eq. (7).

E. Total Computational Complexity Analysis

We now analyze the computational complexity of our CACR method. As verified in Subsection II-A and Subsection II-B, our chroma-copy approach and NRWV strategy take O(WH) time and O(8αWH) time, respectively. As verified in Subsection II-C and Subsection II-D, our hybrid approach and cs-binding and geometry alignment-based error compensation approach take O(αWH) time. Consequently, the total time complexity of our CACR chroma reconstruction method takes O(WH) time.

In addition, under the same test SCIs, for one image, the average execution time requirements of the 15 chroma upsampling methods of interest, namely, COPY, 8-TAP, BIL, BIC, NEDI [11], SAI [21], GF [8], GCR [16], LAU [10], MSRCNN [5], ScSR [19], CSCN [18], LACDU [17], and ASBLG [2], and our CACR method, are 0.006, 0.016, 0.009, 0.227, 5.232, 6.995, 0.142, 0.107, 1.772, 32.282, 10.504, 1165.230, 8.130, 1.681, and 2.264 seconds, respectively. From these actual time costs, we observe that the actual time cost of CACR is much less than that of SAI, MSRCNN, ScSR, CSCN, and LACDU, although CACR takes more time than the other methods. However, as described in the next section, our CACR method has the best CPSNR, CESIM, quality-bitrate tradeoff, and visual effect performance among the methods compared here.

III. Experimental Results

In this section, the 26 test SCIs (see Fig. 4) with sizes ranging from 624 × 624 to 1920 × 1080 and the 9 JCT-VC test SCVs, each sequence with 16 image frames with sizes ranging from 640 × 576 to 1920 × 1080, are used to report the quality and the quality-bitrate tradeoff merits of our CACR method relative to the methods of interest in **CR**. The quality metrics used are CPSNR, CESIM, and the visual effect; the quality-bitrate tradeoff is expressed in RD curves. All experiments are implemented using a computer with an Intel Core i7-4790 CPU running at 3.6 GHz with 8 GB RAM. The operating system is Microsoft Windows 7 64-bit. The program development environment is Visual C++ 2013.

A. CPSNR and CESIM Quality Merits

When setting QP (quantization parameter) to 0, the first experiment is carried out to demonstrate the CPSNR and CESIM merits of our CACR method. In this experiment, after performing CACR on the server side, we bypass the encoder and go directly to the client side to upsample the subsampled YUV image. The CPSNR metric is defined by

$$\text{CPSNR} = \frac{1}{N}\sum_{n=1}^{N} 10\log_{10}\frac{255^2}{\text{CMSE}} \tag{8}$$

Fig. 4. The 26 test SCIs.

TABLE I

AVERAGE CPSNR GAINS OF *cs*-CACR OVER $\mathbf{CS}^1 \times \mathbf{CR}^1\backslash\{\text{CACR}\}$ FOR THE RECONSTRUCTED UV IMAGES, SCIS, AND SCVS

	COPY	8-TAP	BIL	BIC	NEDI [11]	SAI [21]	GF [8]	GCR [16]	LAU [10]	MSRCNN [5]	ScSR [19]	CSCN [18]	ASBLG [2]	**CACR**
4:2:0(A)	-3.8747	-4.7874	-4.8433	-4.7420	-4.9225	-5.2989	-4.3880	-3.7603	-3.5847	-3.1000	-3.0532	-2.9943	-2.3636	**39.2040**
	(-3.5700)	(-4.4717)	(-4.5750)	(-4.4531)	(-4.6717)	(-4.8632)	(-4.2676)	(-3.6093)	(-3.4350)	(-2.6565)	(-2.6363)	(-2.6119)	(-2.3231)	**(35.4432)**
	[-2.0272]	[-2.9338]	[-3.0465]	[-2.9347]	[-3.0550]	[-3.2630]	[-2.8594]	[-2.7531]	[-2.0743]	[-1.0954]	[-1.0347]	[-1.0471]	[-2.2452]	**[32.7811]**
4:2:0(L)	-4.5125	-3.6887	-3.3754	-3.4031	-3.3980	-3.9658	-3.6028	-3.0409	-3.1327	-5.1080	-4.8941	-4.7885	-1.4910	**38.0726**
	(-4.1311)	(-3.4716)	(-3.2117)	(-3.1983)	(-3.2659)	(-3.6119)	(-3.5900)	(-2.7810)	(-3.0596)	(-4.5360)	(-4.3434)	(-4.2844)	(-1.5803)	**(34.4546)**
	[-2.6105]	[-2.0261]	[-1.7688]	[-1.8114]	[-1.7127]	[-2.0332]	[-2.1415]	[-1.8658]	[-2.0331]	[-3.0050]	[-2.7672]	[-2.7382]	[-1.3312]	**[31.7226]**
4:2:0(R)	-4.7330	-6.0997	-5.8670	-5.9184	-5.9075	-6.1683	-3.7052	-3.3623	-3.2893	-5.3609	-5.1937	-5.1632	-1.6341	**38.2923**
	(-4.3865)	(-5.7019)	(-5.5397)	(-5.5563)	(-5.6018)	(-5.7495)	(-3.6537)	(-3.7097)	(-3.1801)	(-4.8582)	(-4.6983)	(-4.6937)	(-1.6539)	**(34.6246)**
	[-2.1918]	[-3.4492]	[-3.3363]	[-3.3465]	[-3.3654]	[-3.4595]	[-1.6402]	[-3.1168]	[-1.4408]	[-2.4437]	[-2.3433]	[-2.3508]	[-0.8352]	**[31.1893]**
4:2:0(DIRECT)	-6.0637	-3.9851	-3.3914	-3.4946	-3.3811	-3.9117	-4.2789	-3.7588	-3.9006	-5.5394	-7.0510	-6.9366	-1.9502	**38.4708**
	(-5.3907)	(-3.4882)	(-3.0442)	(-3.0865)	(-3.0764)	(-3.4728)	(-4.0610)	(-3.2395)	(-3.6230)	(-6.3734)	(-6.2011)	(-6.1223)	(-1.8402)	**(34.6730)**
	[-3.3868]	[-1.8738]	[-1.3267]	[-1.4935]	[-1.1903]	[-1.7520]	[-2.1535]	[-1.8568]	[-2.2201]	[-4.3067]	[-4.0759]	[-4.0409]	[-1.2389]	**[31.4285]**
4:2:0(MPEG-B)	-5.3634	-3.4734	-3.6492	-3.4821	-3.7690	-4.1126	-4.5115	-3.3660	-3.9863	-5.2929	-5.2973	-5.1451	-1.8756	**38.6967**
	(-5.0002)	(-3.1855)	(-3.4246)	(-3.2257)	(-3.5644)	(-3.7374)	(-4.3789)	(-3.0689)	(-3.8286)	(-4.8893)	(-4.8913)	(-4.7760)	(-1.8486)	**(34.9709)**
	[-3.2243]	[-1.7043]	[-1.8183]	[-1.6990]	[-1.8130]	[-1.9445]	[-2.7456]	[-1.9036]	[-2.5592]	[-3.0899]	[-3.0656]	[-2.9634]	[-1.8471]	**[32.0614]**
Average CPSNR Gain	-4.9095	-4.4069	-4.2253	-4.2080	-4.2756	-4.6915	-4.0973	-3.4577	-3.5787	-4.8803	-5.0979	-5.0055	-1.8629	**38.5473**
	(-4.4957)	(-4.0548)	(-3.9590)	(-3.9040)	(-4.0360)	(-4.2870)	(-3.9902)	(-3.2817)	(-3.4253)	(-4.6627)	(-4.5541)	(-4.4977)	(-1.8492)	**(34.8333)**
	[-2.6881]	[-2.3974]	[-2.2593]	[-2.2570]	[-2.2273]	[-2.4905]	[-2.3080]	[-2.2992]	[-2.0655]	[-2.7881]	[-2.6573]	[-2.6281]	[-1.4995]	**[31.8366]**

with

$$\text{CMSE} = \frac{1}{2XY}\sum_{l\in\mathcal{L}}\sum_{C\in\{U,V\}}[I_{n,C}^{ori,UV}(l) - I_{n,C}^{rec,UV}(l)]^2$$

in which $\mathcal{L} = \{(x, y)|1 \le x \le W, 1 \le y \le H\}$ denotes the set of pixel coordinates in one W×H SCI, N (= 26 + 270) denotes the number of total test SCIs, and $I_{n,C}^{ori,UV}(l)$ denotes the C-color value of the pixel at location l in the nth original UV image and $I_{n,C}^{rec,UV}(l)$ the reconstructed analogue.

For evaluating the CPSNR merit of the our CACR method, we consider all the 70 combinations in $\mathbf{CS}^1 \times \mathbf{CR}^1$ with $\mathbf{CS}^1 = \mathbf{CS}\backslash\{\text{CS}_{\text{BIL}}, \text{CS}_{\text{BIC}}\}$ and $\mathbf{CR}^1 = \mathbf{CR}\backslash\{\text{LACDU}\}$. In Table I, the average CPSNR gains of the reconstructed SCIs and SCVs by the combination *cs*-CACR over *cs-cr*, $cs \in \mathbf{CS}^1$ and $cr \in \mathbf{CR}^1\backslash\{\text{CACR}\}$, are enclosed by '()' and '[]', respectively. From Table I, we observe that our CACR method in boldface has the best CPSNR performance and in particular that CACR offers 1.86 dB, 1.85 dB, and 1.50 dB quality improvement for the reconstructed UV images, SCIs, and SCVs, respectively, relative to the previously reported method ASBLG [2].

CESIM [13] is a useful quality assessment metric for evaluating the edge-preservation performance of the reconstructed SCIs and is defined by

$$\text{CESIM} = \frac{1}{N}\sum_{n=1}^{N}\frac{1}{3}(\text{ESIM}_\text{R} + \text{ESIM}_\text{G} + \text{ESIM}_\text{B}) \qquad (9)$$

where ESIM_R, ESIM_G, and ESIM_B indicate the ESIM values of the reconstructed R, G, and B images, respectively, with

$$\text{ESIM} = \frac{\sum_{(x,y)}\text{W}(x,y)\text{S}(x,y)}{\sum_{(x,y)}\text{W}(x,y)},$$

$\text{S}(x, y) = [\text{ECS}(x, y)] \cdot [\text{EWS}(x, y)] \cdot [\text{EDS}(x, y)]$, and $\text{W}(x, y) = max(\text{EWM}_r, \text{EWM}_d)$ where ECS, EWS, and EDS denote the similarity of edge contrast, edge width, and edge direction, respectively. EWM_r and EWM_d denote the edge width maps of the test SCI and reconstructed SCI, respectively. The detailed definition of the ESIM metric can be found in [13].

Table II demonstrates the CESIM performance comparison among the 14 methods of interest. The average CESIM gains

TABLE II

AVERAGE CESIM GAINS OF *cs*-CACR OVER $\mathbf{CS}^1 \times \mathbf{CR}^1 \backslash \{\text{CACR}\}$ FOR THE RECONSTRUCTED SCIS AND SCVS

	COPY	8-TAP	BIL	BIC	NEDI [11]	SAI [21]	GF [8]	GCR [16]	LAU [10]	MSRCNN [5]	ScSR [19]	CSCN [18]	ASBLG [2]	**CACR**
4:2:0(A)	-0.0407 [-0.0433]	-0.0737 [-0.1038]	-0.0438 [-0.0810]	-0.0411 [-0.0823]	-0.0683 [-0.0793]	-0.0718 [-0.0814]	-0.0728 [-0.0729]	-0.0723 [-0.0707]	-0.0391 [-0.0248]	-0.0365 [-0.0391]	-0.0375 [-0.0378]	-0.0332 [-0.0321]	-0.0211 [-0.0517]	**0.7288** **[0.6846]**
4:2:0(L)	-0.0714 [-0.0680]	-0.0630 [-0.0900]	-0.0529 [-0.0548]	-0.0550 [-0.0576]	-0.0523 [-0.0525]	-0.0566 [-0.0593]	-0.0746 [-0.0783]	-0.0488 [-0.0492]	-0.0435 [-0.0361]	-0.0944 [-0.1064]	-0.0907 [-0.1020]	-0.0869 [-0.0957]	-0.0235 [-0.0477]	**0.7269** **[0.6825]**
4:2:0(R)	-0.0779 [-0.0652]	-0.1231 [-0.1476]	-0.1108 [-0.1188]	-0.1146 [-0.1226]	-0.1082 [-0.1144]	-0.1148 [-0.1208]	-0.0788 [-0.0734]	-0.0982 [-0.1054]	-0.0476 [-0.0301]	-0.1054 [-0.0985]	-0.1029 [-0.0966]	-0.0973 [-0.0899]	-0.0237 [-0.0407]	**0.7317** **[0.6768]**
4:2:0(DIRECT)	-0.0950 [-0.0957]	-0.0568 [-0.0883]	-0.0433 [-0.0430]	-0.0437 [-0.0484]	-0.0409 [-0.0348]	-0.0437 [-0.0427]	-0.0775 [-0.0859]	-0.0475 [-0.0477]	-0.0539 [-0.0529]	-0.1345 [-0.1577]	-0.1291 [-0.1510]	-0.1253 [-0.1454]	-0.0342 [-0.0416]	**0.7258** **[0.6830]**
4:2:0(MPEG-B)	-0.0823 [-0.0935]	-0.0404 [-0.0825]	-0.0418 [-0.0538]	-0.0379 [-0.0545]	-0.0421 [-0.0473]	-0.0406 [-0.0484]	-0.0661 [-0.0723]	-0.0429 [-0.0540]	-0.0467 [-0.0561]	-0.0980 [-0.1038]	-0.0973 [-0.1042]	-0.0912 [-0.0955]	-0.0234 [-0.0359]	**0.7141** **[0.6735]**
Average CESIM Gain	-0.0734 [-0.0732]	-0.0714 [-0.1024]	-0.0585 [-0.0703]	-0.0585 [-0.0731]	-0.0623 [-0.0657]	-0.0655 [-0.0705]	-0.0740 [-0.0766]	-0.0619 [-0.0654]	-0.0462 [-0.0400]	-0.0937 [-0.1011]	-0.0915 [-0.0983]	-0.0868 [-0.0917]	-0.0252 [-0.0435]	**0.7255** **[0.6801]**

TABLE III

CPSNR AND CESIM QUALITY COMPARISON AMONG $\mathbf{CS}^2 \times \mathbf{CR}^2$ FOR THE RECONSTRUCTED UV IMAGES, SCIS, AND SCVS

	CS_{BIL}-LACDU	CS_{BIC}-LACDU	CS_{BIL}-ASBLG	CS_{BIC}-ASBLG	CS_{BIL}-CACR	CS_{BIC}-CACR
CPSNR	35.9215 (32.6666) [31.7091]	35.9358 (32.6543) [31.6884]	37.1053 (33.4037) [31.0851]	37.0422 (33.3356) [31.0069]	39.3721 (35.7294) [33.5840]	**39.4414** **(35.8004)** **[33.6443]**
CESIM	(0.6889) [0.6368]	(0.6824) [0.6322]	(0.6804) [0.6317]	(0.6684) [0.6242]	**(0.7390)** **[0.6945]**	(0.7326) [0.6902]
CPSNR Gain	-3.5199 (-3.1338) [-1.9352]	-3.5056 (-3.1461) [-1.9560]	-2.3361 (-2.3967) [-2.5863]	-2.3992 (-2.4648) [-2.6374]	-0.0693 -0.0710 -0.0603	
CESIM Gain	(-0.0501) [-0.0576]	(-0.0565) [-0.0622]	(-0.0586) [-0.0627]	(-0.0706) [-0.0703]		(-0.0064) [-0.0042]

of *cs*-CACR over any other combinations for the reconstructed SCVs are enclosed by '()'. From Table II, we observe that for any chroma subsampling scheme *cs* in $\mathbf{CS}^1$, the combination *cs*-CACR in boldface always has the best CESIM performance for the reconstructed SCIs and SCVs relative to all existing combinations in $\mathbf{CS}^1 \times \mathbf{CR}^1 \backslash \{\text{CACR}\}$. On the other hand, our CACR method has the best edge-preservation effect among the methods of interest.

Due to the particularity in LACDU, we now evaluate the CPSNR and CESIM quality performance for the six combinations in $\text{CS}^2 \times \text{CR}^2$ = ({CS_{BIL}-LACDU, CS_{BIC}-LACDU, CS_{BIL}-ASBLG, CS_{BIC}-ASBLG, CS_{BIL}-CACR, CS_{BI}-CACR}). From Table III, we observe that in terms of CPSNR, CS_{BIC}-CACR in boldface has the best performance among the six combinations for the reconstructed UV images, SCIs, and SCVs. In particular, CS_{BIC}-CACR also offers superior CPSNR performance relative to any other combination in $\mathbf{CS}^1 \times \mathbf{CR}^1$, as illustrated in Table I. In terms of CESIM, Table III indicates that CS_{BIL}-CACR in boldface has the best performance among the six combinations. Here, the chroma subsampling schemes CS_{BIL} and CS_{BIC} were proposed by Wang *et al.* [17].

B. Visual Effect Merit

In this subsection, in addition to the three chroma reconstruction methods, LACDU, ASBLG, and CACR, discussed in the last paragraph of Subsection III-A, we include one simple chroma reconstruction method, COPY, in the visual effect comparison. Thus, the four considered comparative combinations are CS_{BIC}-COPY, CS_{BIC}-LACDU, CS_{BIC}-ASBLG, and CS_{BIC}-CACR.

As shown in Fig. 5(a), the 22nd SCI test image is taken as the comparison example. For visual comparison, as shown in Figs 5(b)-(c), two magnified subimages are decoupled from the character-containing and color-containing regions in Fig. 5(a), respectively. After performing the four combinations on Fig. 5(a), the eight reconstructed magnified subimages for Fig. 5(b) and Fig. 5(c) are shown in Figs. 5(d)-(g) and Figs. 5(h)-(j), respectively. As shown in the regions marked by ellipses, we observe that CS_{BIC}-CACR has the best color and edge preservation effects in the four combinations. As expected, although COPY is simple, CS_{BIC}-COPY has the worst visual effects, notably containing color shifting and blurred edge artifacts. The detailed visual effects comparison of all the test SCIs can be found in [7].

TABLE IV

KEY PARAMETER SETTINGS OF THE HEVC CODEC: HM-16.17+SCM-8.6

Parameter	Value	Parameter	Value
CTU Size/Depth	64/4	IntraPeriod	4
QPoffset	0	GOP Size	4
Fast Search	1	Search Range	32
Entropy Coding	CABAC	SAO	1
AMP	1	SAOLcuBoundary	0

C. Quality-Bitrate Trade-Off Merit

The HEVC codec used is the reference software HM-16.17+SCM-8.6. The coding structure used in our experiment is 'Low Delay P'. Table IV tabulates the key parameter settings; the remaining parameters are set as default. In Table IV, the parameter 'CTU Size/Depth' is set to 64/4, indicating that the coding tree unit (CTU) is of size 64×64 and of depth 4. The term 'IntraPeriod' is set to 4, indicating the period of intra prediction. The parameter 'QPoffset', which indicates the QP offset from the base QP value, is set to 0. The parameter 'GOP size' denotes the size of the group of

Fig. 5. The visual effect comparison for the 22nd reconstructed SCI. (a) The groundtruth of 22nd SCI. (b) The characters part of (a). (c) The colorful part of (a). (d) The reconstructed image of (b) by CS_{BIC}-COPY. (e) The reconstructed image of (b) by CS_{BIC}-LACDU. (f) The reconstructed image of (b) by CS_{BIC}-ASBLG. (g) The reconstructed image of (b) by CS_{BIC}-CACR. (h) The reconstructed image of (c) by CS_{BIC}-COPY. (i) The reconstructed image of (c) by CS_{BIC}-LACDU. (j) The reconstructed image of (c) by CS_{BIC}-ASBLG. (k) The reconstructed image of (c) by CS_{BIC}-CACR.

pictures and is set to 4. The parameter 'Fast Search' is set to 1, indicating that the diamond search [23] is adopted. The parameter 'Search Range' is set to 32, indicating that the size of the search window is 32×32. The parameter 'Entropy Coding' is set to CABAC (context-adaptive binary arithmetic coding). The coding tools contain three parameters: 'SAO' (sample adaptive offset), 'AMP' (asymmetric motion partitions), and 'SAOLcuBoundary' (SAO largest coding unit boundary), which are set to 1, 1, and 0, respectively.

To plot the RD curves, 11 different QPs, namely, 0, 4, 8, 12, 16, 20, 24, 28, 32, 36, and 40, are considered to demonstrate the quality-bitrate tradeoff performance comparison among the four combinations CS_{BIC}-COPY, CS_{BIC}-LACDU, CS_{BIC}-ASBLG, and CS_{BIC}-CACR. For the 9 test JCT-VC SCVs, the four RD curves, as shown in Fig. 6, indicate that under the same QP value, CS_{BIC}-CACR has the best CPSNR performance of the reconstructed SCVs. On the other hand, CS_{BIC}-CACR has the best quality-bitrate tradeoff among the four combinations.

Finally, we come to a conclusion for the abovementioned experimental results. From Tables I-II, we observe that when QP = 0, the ASBLG method [2] has the second best CPSNR and CESIM performance for the test SCIs and SCVs, and our CACR method is the first best among the concerned methods in $\mathbf{CS}^1 \times \mathbf{CR}^1$; from Table III, CACR and ASBLG still have the first and second best CPSNR and CESIM performance among the concerned methods in $\mathbf{CS}^2 \times \mathbf{CR}^2$. Fig. 6 indicates

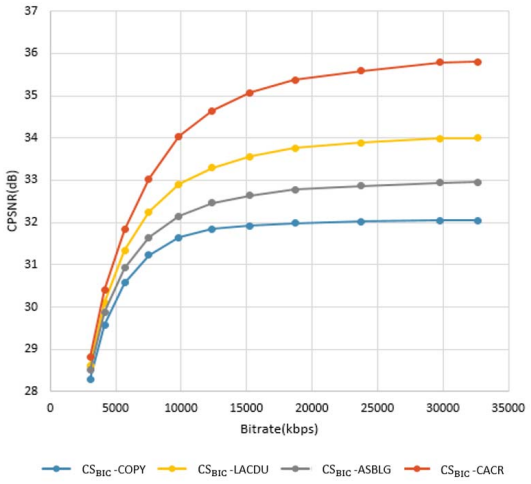


Fig. 6. RD curves for CS_{BIC}-COPY, CS_{BIC}-LACDU, CS_{BIC}-ASBLG, and CS_{BIC}-CACR under the 9 test JCT-VC SCVs.

that CACR, LACDU [17], and ASBLG have the first, second, and third best quality-bitrate tradeoff performance under the parameter settings in Table IV.

IV. Conclusion

We have presented our effective CACR method for upsampling decoded downsampled chroma images on the client side.

The four novel aspects and contributions of CACR have also been clarified in Subsection I-B. Based on 26 typical test SCIs and 9 JCT-VC test SCVs, comprehensive experiments have been carried out to show that in terms of CPSNR, CESIM, the visual effect, and the quality-bitrate tradeoff in the RD curve, our CACR method is significantly superior to the 14 state-of-the-art reconstruction methods COPY, 8-TAP, BIL, BIC, NEDI [11], SAI [21], GF [8], GCR [16], LAU [10], MSRCNN [5], ScSR [19], CSCN [18], LACDU [17], and ASBLG [2].

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] K.-L. Chung, C.-C. Huang, and T.-C. Hsu, "Adaptive chroma subsampling-binding and luma-guided chroma reconstruction method for screen content images," *IEEE Trans. Image Process.*, vol. 26, no. 12, pp. 6034–6045, Dec. 2017.

[3] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to Algorithms, Section 3.1: Asymptotic notation*, 3rd ed. London, U.K.: MIT Press, 2009.

[4] C. Dong, C. C. Loy, K. He, and X. Tang, "Image super-resolution using deep convolutional networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 2, pp. 295–307, Feb. 2016.

[5] C. Dong, C. C. Loy, and X. Tang, "Accelerating the super-resolution convolutional neural network," in *Proc. Eur. Conf. Comput. Vis.*, Aug. 2016, pp. 1–16.

[6] D. A. Kerr. (Jan. 2005). *Chrominance Subsampling in Digital Images*. [Online]. Available: http://dougkerr.net/Pumpkin/articles/Subsampling.pdf

[7] *Experimental Results*. Accessed: Oct. 2018. [Online]. Available: ftp://140.118.175.164/CACR_Exp

[8] K. He, J. Sun, and X. Tang, "Guided image filtering," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 6, pp. 1397–1409, Jun. 2013.

[9] *HM16.17+SCM-8.6*. Accessed: Jan. 2018. [Online]. Available: https://hevc.hhi.fraunhofer.de/svn/svn/HEVCSoftware/tags/HM-16.17+SCM-8.6/

[10] J. Korhonen, "Improving image fidelity by luma-assisted chroma subsampling," in *Proc. IEEE Int. Conf. Multimedia Expo.*, Jul. 2015, pp. 1–6.

[11] X. Li and M. T. Orchard, "New edge-directed interpolation," *IEEE Trans. Image Process.*, vol. 10, no. 10, pp. 1521–1527, Oct. 2001.

[12] Z. Ni, L. Ma, H. Zeng, C. Cai, and K.-K. Ma, "Gradient direction for screen content image quality assessment," *IEEE Signal Process. Lett.*, vol. 23, no. 10, pp. 1394–1398, Oct. 2016.

[13] Z. Ni, L. Ma, H. Zeng, J. Chen, C. Cai, and K.-K. Ma, "ESIM: Edge similarity for screen content image quality assessment," *IEEE Trans. Image Process.*, vol. 26, no. 10, pp. 4818–4831, Oct. 2017.

[14] W. Pu *et al.*, "Palette mode coding in HEVC screen content coding extension," *IEEE J. Emerg. Sel. Topics Circuits Syst.*, vol. 6, no. 4, pp. 420–432, Dec. 2016.

[15] R. Timofte, V. De Smet, and L. Van Gool, "A+: Adjusted anchored neighborhood regression for fast super-resolution," in *Proc. Asian Conf. Comput. Vis.*, Nov. 2014, pp. 111–126.

[16] T. Vermeir *et al.*, "Guided chroma reconstruction for screen content coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 26, no. 10, pp. 1884–1892, Oct. 2016.

[17] S. Wang, K. Gu, S. Ma, and W. Gao, "Joint chroma downsampling and upsampling for screen content image," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 26, no. 9, pp. 1595–1609, Sep. 2016.

[18] Z. Wang, D. Liu, J. Yang, W. Han, and T. Huang, "Deep networks for image super-resolution with sparse prior," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2015, pp. 370–378.

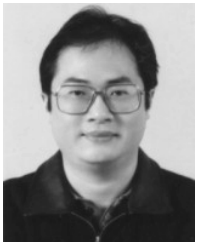
[19] J. Yang, J. Wright, T. S. Huang, and Y. Ma, "Image super-resolution via sparse representation," *IEEE Trans. Image Process.*, vol. 19, no. 11, pp. 2861–2873, Nov. 2010.

[20] H. Yu, R. Cohen, K. Rapaka, and J. Xu, *Common Test Conditions for Screen Content Coding*, document JCTVC-X1015, Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11 24th Meeting, Geneva, Switzerland, May/Jun. 2016.

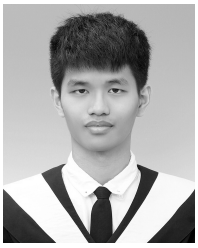
[21] X. Zhang and X. Wu, "Image interpolation by adaptive 2-D autoregressive modeling and soft-decision estimation," *IEEE Trans. Image Process.*, vol. 17, no. 6, pp. 887–896, Jun. 2008.

[22] Y. Zhang, D. Zhao, J. Zhang, R. Xiong, and W. Gao, "Interpolation-dependent image downsampling," *IEEE Trans. Image Process.*, vol. 20, no. 11, pp. 3291–3296, Nov. 2011.

[23] S. Zhu and K.-K. Ma, "A new diamond search algorithm for fast block-matching motion estimation," *IEEE Trans. Image Process.*, vol. 9, no. 2, pp. 287–290, Feb. 2000.



Kuo-Liang Chung (SM'01) received the B.S., M.S., and Ph.D. degrees from National Taiwan University, Taipei, Taiwan, in 1982, 1984, and 1990, respectively. He has been the Chair Professor with the Department of Computer Science and Information Engineering, National Taiwan University of Science and Technology, Taipei, since 2009. His research interests include deep learning, image processing, and video compression. He was a recipient of the Distinguished Research Award (2004–2007) and the Distinguished Research Project Award (2009–2012) from the National Science Council of Taiwan. In 2017, he received the Scientific Paper Award from the Far Eastern Y. Z. Hsu Science and Technology Memorial Foundation. He has been an Associate Editor of the *Journal of Visual Communication and Image Representation* since 2011. His research interests include deep learning, image processing, and video compression.



Yan-Cheng Liang received the B.S. degree from the Department of Applied Mathematics, National Sun Yat-sen University, Kaohsiung, Taiwan, in 2016. He is currently pursuing the M.S. degree in computer science and information engineering with the National Taiwan University of Science and Technology, Taipei, Taiwan. His research interests include image processing and video compression.

Ching-Sheng Wang received the B.S. degree from the Department of Computer Science and Information Engineering, National Taipei University of Education, Taipei, Taiwan, in 2017. He is currently pursuing the M.S. degree in computer science and information engineering with the National Taiwan University of Science and Technology, Taipei. His research interests include image processing and video compression.