Adaptive Effective Wiener Filter- and Regression-Based Upsampling for Asymmetric Resolution Stereoscopic Video Coding

Kuo-Liang Chung, *Senior Member, IEEE*, Shin-Bei Tsai, Yu-Ling Tseng, and Chi-Chao Huang

***Abstract*— In asymmetric resolution stereoscopic video coding (ARSVC), a stereoscopic video consists of one full-sized left-view video sequence and the synchronized quarter-sized right-view video sequence for achieving a bitrate reduction effect by the encoder. Prior to displaying 3D scenes on the screen, it is necessary to upsample the decoded downsampled right-view video sequence at the client side. In this paper, we propose an effective adaptive upsampling method for ARSVC. First, we employ the resolution- and texture-consistency consideration in the conventional Wiener filter-based interpolation scheme, called RTCWF, to enhance the upsampling accuracy in the spatial domain. Second, we propose a linear regression-based interview prediction scheme with residual compensation called LRIPRC, to increase the upsampling accuracy in the interview domain. Third, we propose an adaptive fusion-based approach to integrate RTCWF and LRIPRC, called RTCWF-LRIPRC, to maximize the quality improvement of the upsampled image. Based on seven typical test stereoscopic video sequences, in 3D-HEVC, the experimental results demonstrated that in terms of six well-known quality metrics and execution time requirements, our RTCWF-LRIPRC method outperforms the state-of-the-art upsampling methods for ARSVC.**

***Index Terms*— Asymmetric resolution stereoscopic video coding (ARSVC), convolutional neural network, interview prediction, linear regression, performance comparison, 3D-HEVC, upsampling, Wiener filter.**

I. Introduction

AS TECHNIQUES used by three-dimensional televisions (3D-TV) [11] are becoming more mature and 3D content is growing rapidly in the multimedia and entertainment markets, to reduce the bitrate requirements while preserving quality, several effective 3D video coding schemes have been developed. Among these schemes, the color plus depth video coding (CDVC) model [30] and the stereoscopic video coding (SVC) model [1] are the two most well-known models. In CDVC, the two input sequences consist of one real color video sequence and the synchronized quarter-sized depth video sequence. After decoding the two video sequences and upsampling the decoded depth video sequence, the depth image-based rendering (DIBR) technique [12] is used to generate the virtual color video sequence. Further, the real and virtual color video sequences are synthesized to provide 3D scenes.

SVC contains the left-view and the synchronized equal-sized right-view video sequences. According to the suppression theory of the human binocular vision system [29], it is suggested that the right-view image in SVC be encoded at a lower bitrate than the left-view image to reduce the bitrate. This model is called the asymmetric SVC (ASVC). Usually, ASVC is classified into asymmetric resolution stereoscopic video coding (ARSVC) [4], [13] and asymmetric quality stereoscopic video coding. In this paper, we focus on the upsampling of the downsampled right-view image for ARSVC.

Prior to compression in ARSVC, the right-view image is downsampled to a quarter-sized image. Then, the left-view image and the downsampled right-view image are fed into the encoder. In [27], a disparity compensation-based encoding scheme was presented using the left-view image information. In [6], an efficient adaptive filter-based scheme was developed to generate a picture-level interview predictor for encoding the downsampled right-view image. In [5], a regionally adaptive filter-based scheme was proposed to generate a predictor of an MB. Simulation results showed the bitrate-saving merit of these encoding schemes. After encoding the left-view image and the downsampled right-view image, the encoded bitstream is transmitted to the decoder via the internet.

When receiving the decoded left-view and downsampled right-view image pair at the client side, an upsampling process is necessary to upsample the downsampled right-view image to its original size. In H.264/AVC, the 6-TAP filter [18] can be used to upsample the downsampled right-view image. Li and Orchard [25] presented a new edge-directed interpolation (NEDI) to modify the conventional Wiener filter (WF)-based upsampling method and the edge-directed interpolation (EDI) [2]. Zhang [40] presented a soft decision adaptive interpolation (SAI) to interpolate the missing pixels in each 2×2 downsampled block by jointly training the relationship not only between the known pixels and the missing pixels in the block but also between the missing pixels themselves to improve NEDI. Gao *et al.* [15] presented a spatial interview- and WF-based upsampling method. In their method, each missing pixel is estimated as the linear weighted summation

Manuscript received July 31, 2017; revised May 8, 2018, June 16, 2018, July 5, 2018, and July 18, 2018; accepted July 20, 2018. Date of publication July 25, 2018; date of current version August 21, 2018. This work was supported under Contract MOST-105-2221-E-011-118-MY3. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Chia-Kai Liang. *(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.2018.2859579

of 13 known pixels, in which the four known pixels come from the downsampled right-view image and the nine known pixels come from the best matched block in the left-view image. Then, an over-determined system is constructed. Further, the 13 solved parameters are used as the 13 weights to estimate the missing pixel, achieving better quality when compared with NEDI.

Unlike the aforementioned upsampling methods, Yang *et al.* [38] presented the first sparse representation-based upsampling method. To alleviate the deblocking effect, Kang *et al.* [21] presented an improved method to model the similarity relation between the sparse representation of the high-resolution image patch and that of the corresponding low-resolution image patch. In contrast to the previous depth map-based methods [16], [32], which handle the upsampling process and the depth map estimation process separately, Zhang *et al.* [39] presented an iterative depth map-based method for ARSVC. In their iterative method, the downsampled right-view image is first upsampled using the bilinear interpolation, and then, the depth map is estimated using the cooperative optimization for an energy function. Further, the estimated depth map is used to guide the conjugate gradient-based upsampling process. The above iterative process is repeated until the upsampled right-view image is convergent. However, the above sparse representation-based upsampling methods and the depth map-based upsampling methods are time consuming. Jeong *et al.* [20] extended their multi-frame example-based upsampling method [19], therein achieving better quality compared with the bilinear interpolation, Cho *et al.*'s method [7], and Freeman *et al.*'s method [14] for the test video sequences.

Recently, several convolutional neural network-based (CNN-based) upsampling methods [22], [33] have been developed. Dong *et al.* [10] proposed a super-resolution CNN (SRCNN) method associated with a compact hourglass-shaped CNN structure to reduce the computational complexity and improve the quality of the upsampled images. Experimental data showed that SRCNN outperforms the related methods [9], [34]. Wang *et al.* [36] proposed an end-to-end deep and shallow network-based (EEDS-based) method that jointly learns the feature extraction, upsampling, and high-resolution reconstruction modules. Experimental data showed the superior quality of EEDS over the related methods [17], [34].

A. Common Weaknesses in Existing WF-Based Upsampling Methods and Motivation

In NEDI [25], also called WF for convenience, as shown in Fig. 1, the circle-marked pixels in the current 3×3 block B_c are known in the decoded downsampled right-view image. Among the missing pixels of the right-view image, the square-marked pixels will be restored first, then the triangle-marked ones, and finally the cross-marked ones. For the current missing square-marked pixel $\overline{R}_{m,n}$, if the variance of the window centered at $\overline{R}_{m,n}$ is smaller than a specified threshold T, where $T = 8$ empirically, the bilinear interpolation is applied to estimate $\overline{R}_{m,n}$; otherwise, it is estimated by WF.

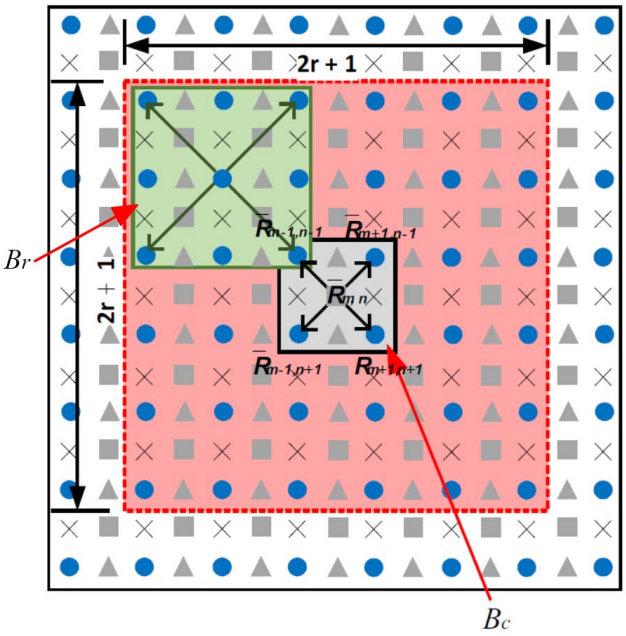


Fig. 1. The 5×5 resolution-inconsistent reference block B_r and the 3×3 current block B_c in the 11×11 window depicting the RI problem of WF.

The first weakness in WF is the resolution-inconsistency (RI) problem. In Fig. 1, we observe that the distance from the square-marked pixel $\overline{R}_{m,n}$ to each neighboring circle-marked pixel in B_c is $\sqrt{2}$, while that between the corner pixel and the central pixel in the 5×5 reference block B_r is $2\sqrt{2}$. This leads to an RI problem causing the quality degradation of the upsampled image. The second weakness in WF is the texture-inconsistency (TI) problem. For each B_r in the 11×11 window, the four corner pixels in B_r may form a different texture pattern from that of B_c. This TI problem also degrades the quality of the upsampled image. The above RI and TI problems occurring in WF are simply called the RTI problem.

After describing the RTI problem of WF in the spatial domain, we now highlight the two weaknesses in the WF-based interview prediction (WFIP) scheme [8] in the interview domain. The first weakness in WFIP is the rugged residual field problem. Here, the residual field is defined as the difference between the upsampled right-view image by WFIP and the equal-sized ground-truth image. In WFIP, the current block B_c in the right-view image first finds the best matched 3×3 block in the 41×41 search window of the left-view image. Since the left-view image is a full-sized image, in the 6×6 window centered at the center of the best matched 3×3 block, an overdetermined system with 16 equations can be constituted. Then, by WF, the missing square-marked pixel $\overline{R}_{m,n}$ in B_c is estimated, but the rugged residual field caused by WFIP implies the limited quality improvement. The second weakness in WFIP is the lack of an intellectual fusion strategy for integrating the estimation results, one from the spatial domain and the other from the interview domain, confining the quality improvement of the upsampled image.

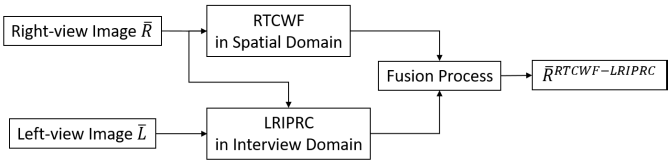


Fig. 2. The flowchart of our RTCWF-LRIPRC method for ARSVC.

B. Contributions

To remedy the above-mentioned RTI problem in WF, in this paper, we propose an RT-consistency (RTC)-based WF, called RTCWF, to improve the upsampling accuracy. Then, to remedy the two drawbacks in the interview domain of WFIP, we first propose a new linear regression-based IP (LRIP) with residual compensation (RC) scheme, called LRIPRC, to increase the upsampling accuracy. Second, an adaptive fusion strategy is proposed to integrate the two estimation results, one from RTCWF and the other from LRIPRC, called RTCWF-LRIPRC, to further improve the upsampling accuracy. The flowchart of our RTCWF-LRIPRC upsampling method for ARSVC is depicted in Fig. 2.

Based on seven typical test stereoscopic video sequences, in 3D-HEVC, thorough experiments have been performed to demonstrate the execution time (in seconds) and quality performance merits of RTCWF-LRIPRC. Here, the quality performance is evaluated by the six quality metrics: the peak signal-to-noise ratio (PSNR), structural similarity (SSIM), motion-based video integrity evaluation (MOVIE), low-complexity version of stereoscopic video quality metric (StSD_{LC}), quality-bitrate tradeoff in rate-distortion (RD) curves, and visual effect. When compared with eight state-of-the-art upsampling methods, such as the Anchor, i.e., the 8-TAP filter [3], the bicubic interpolation (Bicubic) [23], SAI [40], NEDI [25], WFIP [8], Gao *et al.*'s method [15], SRCNN [10], and EEDS [36], our RTCWF-LRIPRC method has the best overall execution time and quality for ARSVC.

The remainder of this paper is organized as follows. In Section II, our RTCWF method in the spatial domain is presented. In Section III, our LRIPRC method in the interview domain is presented. In Section IV, our RTCWF-LRIPRC method for maximizing the quality improvement of the upsampled image is presented. In Section V, the experimental results are reported and used to show the quality and execution time performance superiority of RTCWF-LRIPRC for ARSVC. In Section VI, some concluding remarks are provided.

II. The Proposed Resolution- and Texture-Consistency-Based WF: RTCWF

Initially, we apply Bicubic on the enlarged downsampled right-view image to obtain the interpolated right-view image. In the proposed RC-based WF, called RCWF, if the variance of the window centered at $\overline{R}_{m,n}$ is less than a specified threshold T_{var}, where $T_{var} = 8$ empirically, $\overline{R}_{m,n}$ inherits the estimated value determined by Bicubic in the initial step; otherwise,

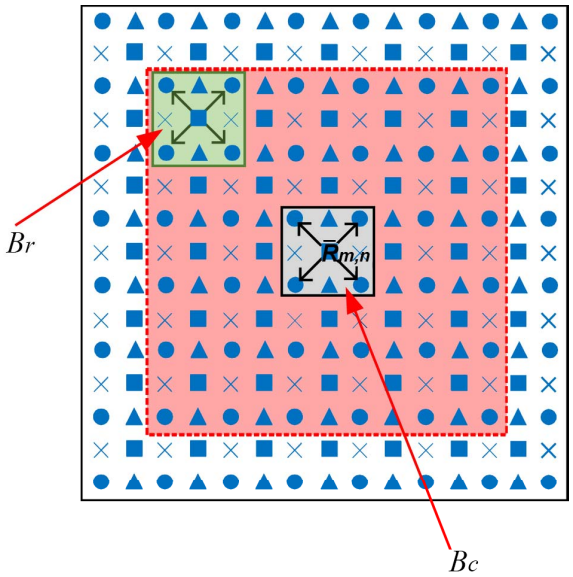


Fig. 3. The 3×3 resolution-consistent reference block B_r and the 3×3 current block B_c in the 11×11 window used to depict the idea of RCWF.

$\overline{R}_{m,n}$ is re-estimated by

$$\overline{R}_{m,n}^{RCWF} = \alpha_0 \times \overline{R}_{m-1,n-1} + \alpha_1 \times \overline{R}_{m+1,n-1} + \alpha_2 \times \overline{R}_{m-1,n+1} + \alpha_3 \times \overline{R}_{m+1,n+1} \quad (1)$$

As shown in Fig. 3, we adopt an 11×11 window W_r to cover the required 81 3×3 reference blocks, in which each a 3×3 reference block is resolution-consistent with the 3×3 current block B_c. The four unknown coefficients, α_0, α_1, α_2, and α_3, in Eq. (1) can be solved by the normal equation [37].

In what follows, based on the proposed TC-based RCWF, called RTCWF, we deliver an elimination strategy to remove those texture-inconsistent 3×3 reference blocks relative to B_c for increasing the accuracy of the upsampled image and reducing the execution time requirement in RTCWF. In Fig. 4, let the central pixel of B_c have a value of 49, which was estimated by Bicubic, and the four reference circle-marked pixel values are 52, 8, 88, and 53. Let B_c be denoted by (49, 52, 8, 88, 53). Comparing the three blocks, B_{n1}(=(50, 48, 10, 90, 52)), B_{n2}(=(8, 10, 5, 8, 6)), and B_{n3}(=(84, 95, 88, 98, 92)), with B_c, we find that B_{n1} is similar to B_c, but B_{n2} and B_{n3} are quite different from B_c. Therefore, B_{n1} could be included in the over-determined system, but B_{n2} and B_{n3} should be excluded. We treat the other remaining 3×3 reference blocks in W_r in the same way to filter out all the improper 3×3 reference blocks and only include the proper 3×3 reference blocks in the over-determined system. Expectantly, the central square-marked pixel in B_c can be estimated by our RTCWF scheme to achieve higher upsampling accuracy and lower computational cost in the spatial domain.

The texture similarity (SIM) between B_c and the reference block B_n centered at location (m', n'), $m - 4 \leq m' \leq m + 4$

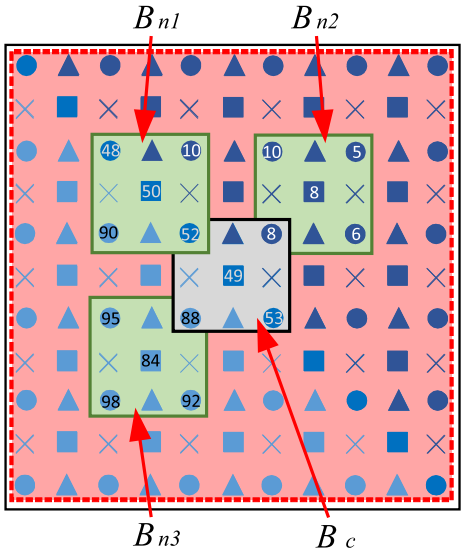


Fig. 4. The example used to depict the proposed RTCWF scheme.

and $n-4 \le n' \le n+4$, in W_r can be defined by

$$SIM(B_c, B_n) = \frac{1}{\prod_{(p,q)\in S}(|\overline{R}_{m+p,n+q} - \overline{R}_{m'+p,n'+q}| + \epsilon) + 1.0} \quad (2)$$

where S = {(0, 0), (−1, −1), (1, −1), (−1, 1), (1, 1)}; ϵ is set to 0.01 to prevent the product of the four absolute differences appearing in the denominator of (2) from becoming zero. After calculating all the concerned texture similarity values for all 19,660,800 11×11 windows in the Alt Moabit video sequence with 100 images, the average texture similarity value of one 11×11 window W_r can be defined by

$$\mu_s = \frac{\sum_{B_n \in W_r} SIM(B_c, B_n)}{81} \quad (3)$$

where '81' denotes the total number of texture similarity values in W_r.

Fig. 5(a) illustrates the distribution of these μ_s-values for all 19,660,800 11×11 windows, where the x-axis denotes the μ_s-value and the y-axis denotes the number of proper 3×3 reference blocks in an 11×11 window to be selected. Because the distribution of these μ_s-values is scattered, we attempt to transform the definition of μ_s in (3) to the following new form:

$$\mu'_s = \frac{\sum_{B_n \in W_r} \ln \frac{1}{SIM(B_c, B_n)}}{81} \quad (4)$$

The distribution of these μ'_s-values for all the 11×11 windows is illustrated in Fig. 5(b). Interestingly, the curve function

$$y = -21.84 \ln(\mu'_s) + 80.515 \quad (5)$$

can well fit the distribution in Fig. 5(b). From Fig. 5(b), we can see that higher μ'_s values in W_r result in fewer 3×3 reference blocks in W_r that are selected. By (5), after determining the number of proper reference 3×3 blocks in W_r, these y RTC blocks are included in the final over-determined system. Further, by (1), the square-marked pixel in B_c can be better estimated by our RTCWF-only method, which is already a competitive solution. We refer readers to the experimental section.

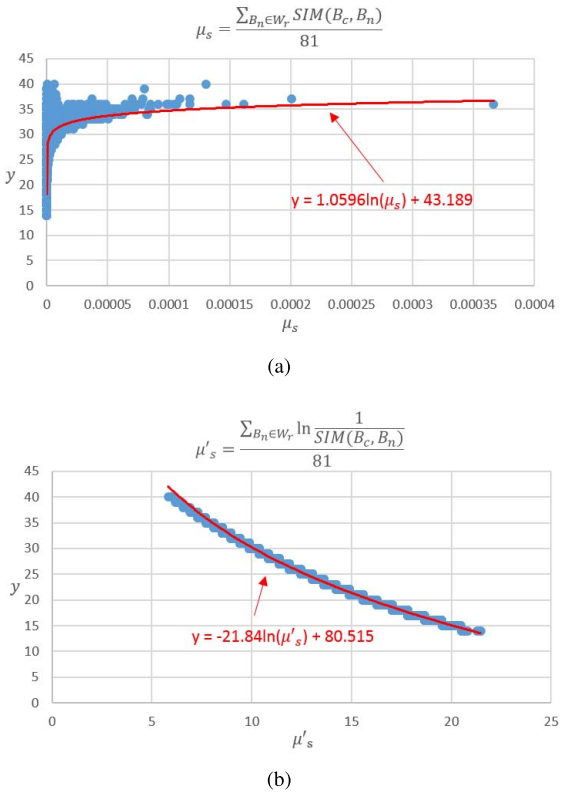
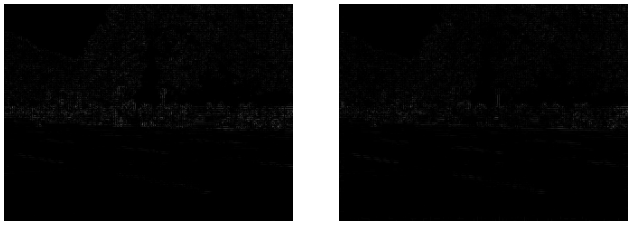


Fig. 5. Determining the number of the proper reference 3×3 blocks in a 11×11 window by curve-fitting technique for RTCWF. (a) By (3), the distribution of these μ_s values for all 11×11 windows in the Alt Moabit video sequence with 100 images. (b) By (4), the distribution of these μ'_s values.



(a) (b)

Fig. 6. Residual field comparison between WFIP_1_EC and LRIPRC. (a) Residual field of WFIP_1_EC. (b) Residual field of LRIPRC.

III. The Proposed Linear Regression-Based Interview Prediction With Residual Compensation: LRIPRC

In this section, our LRIPRC method is presented. As shown in Fig. 6 for one real test subimage, the residual field of LRIP is flatter than that of WFIP_1_EC [8], indicating the upsampling quality merit of the linear regression (LR) technique used in our interview prediction. The workflow of LRIPRC is depicted in Fig. 7, where the blue circle-marked pixel in the right-view image $\overline{R}$ denotes the ground-truth pixel; $\overline{R}^{LRIP}$ denotes the estimated right-view image by LRIP;

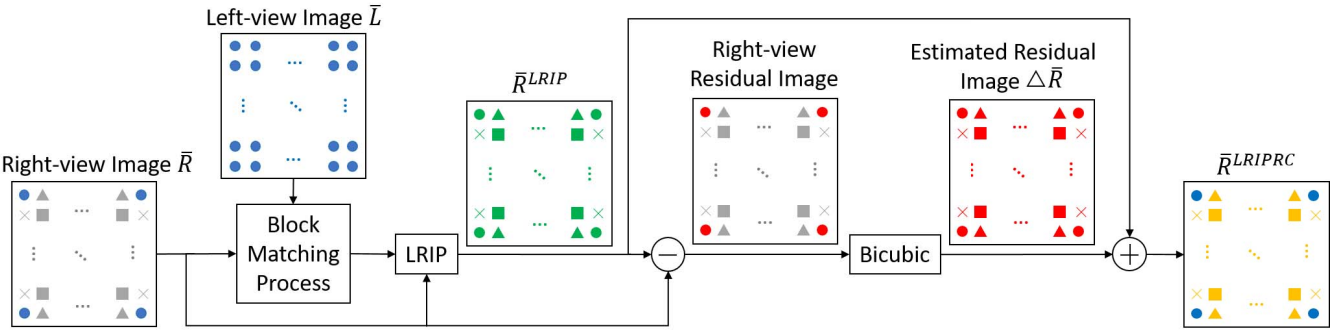


Fig. 7. The workflow of the proposed LRIPRC method.

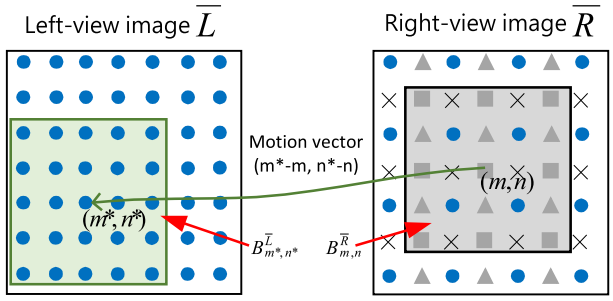


Fig. 8. Interview correlation between $\overline{L}$ and $\overline{R}$ in LRIPRC.

the red circle-marked pixel in the right-view residual image denotes the estimated error between the ground-truth pixel and the co-located estimated pixel by LRIP; $\Delta\overline{R}$ denotes the interpolated residual image by Bicubic, and $\overline{R}^{LRIPRC}$ denotes the upsampled right-view image obtained by first performing LRIP and then performing RC.

A. The Proposed LRIP Method

As shown in Fig. 8, for estimating the current square-marked pixel in the current 5×5 block $B^{\overline{R}}_{m,n}$, LRIP first determines the location of the best matched block $B^{\overline{L}}_{m*,n*}$, i.e., (m^*, n^*), by the minimum criterion of the sum of absolute difference (SAD), which yields

$$(m^*, n^*) = \arg \min_{\substack{m-10 \le m' \le m+10 \\ n-10 \le n' \le n+10}} SAD(B^{\overline{R}}_{m,n}, B^{\overline{L}}_{m',n'}) \tag{6}$$

with

$$SAD(B^{\overline{R}}_{m,n}, B^{\overline{L}}_{m',n'}) = \sum_{i=1}^{4} \left| B^{\overline{R}}_{m,n}(i) - B^{\overline{L}}_{m',n'}(i) \right| \tag{7}$$

where, as shown in Fig. 9, $B^{\overline{R}}_{m,n}(i)$ and $B^{\overline{L}}_{m',n'}(i)$, $1 \le i \le 4$, denote the ith circle-marked pixel values in $B^{\overline{R}}_{m,n}$ and $B^{\overline{L}}_{m',n'}$, respectively. Next, by applying the LR technique to

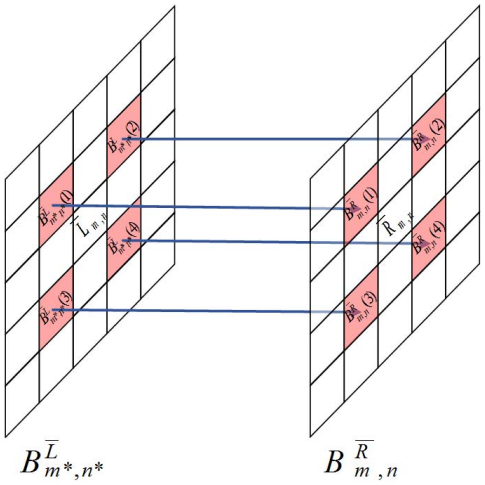


Fig. 9. The interview correlation between the four circle-marked pixels in $B^{\overline{R}}_{m,n}$ and the four circle-matched pixels in $B^{\overline{L}}_{m^*,n^*}$.

the interview correlation, we obtain

$$\begin{aligned} \alpha + \beta \times B^{\overline{L}}_{m^*,n^*}(1) &= B^{\overline{R}}_{m,n}(1) \\ \alpha + \beta \times B^{\overline{L}}_{m^*,n^*}(2) &= B^{\overline{R}}_{m,n}(2) \\ \alpha + \beta \times B^{\overline{L}}_{m^*,n^*}(3) &= B^{\overline{R}}_{m,n}(3) \\ \alpha + \beta \times B^{\overline{L}}_{m^*,n^*}(4) &= B^{\overline{R}}_{m,n}(4). \end{aligned} \tag{8}$$

The correlation parameter-pair (α, β) can be solved by

$$\begin{cases} \alpha = \dfrac{\sum_{i=1}^{4} B^{\overline{R}}_{m,n}(i)}{4} - \beta \dfrac{\sum_{i=1}^{4} B^{\overline{L}}_{m^*,n^*}(i)}{4} \\ \beta = \dfrac{4 \cdot \sum_{i=1}^{4} \left(B^{\overline{R}}_{m,n}(i) \cdot B^{\overline{L}}_{m^*,n^*}(i) \right) - \sum_{i=1}^{4} B^{\overline{R}}_{m,n}(i) \cdot \sum_{i=1}^{4} B^{\overline{L}}_{m^*,n^*}(i)}{4 \cdot \sum_{i=1}^{4} \left(B^{\overline{L}}_{m^*,n^*}(i)^2 \right) - \left(\sum_{i=1}^{4} B^{\overline{L}}_{m^*,n^*}(i) \right)^2} \end{cases} \tag{9}$$

Then, the square-marked pixel $\overline{R}_{m,n}$ in $B^{\overline{R}}_{m,n}$ is estimated by

$$\overline{R}_{m,n} = \alpha + \beta \times \overline{L}_{m^*,n^*} \tag{10}$$

After performing WFIP and RTCWF-LRIP on the seven test video sequences, the PSNR gain of RTCWF-LRIP over WFIP is 0.86 dB.

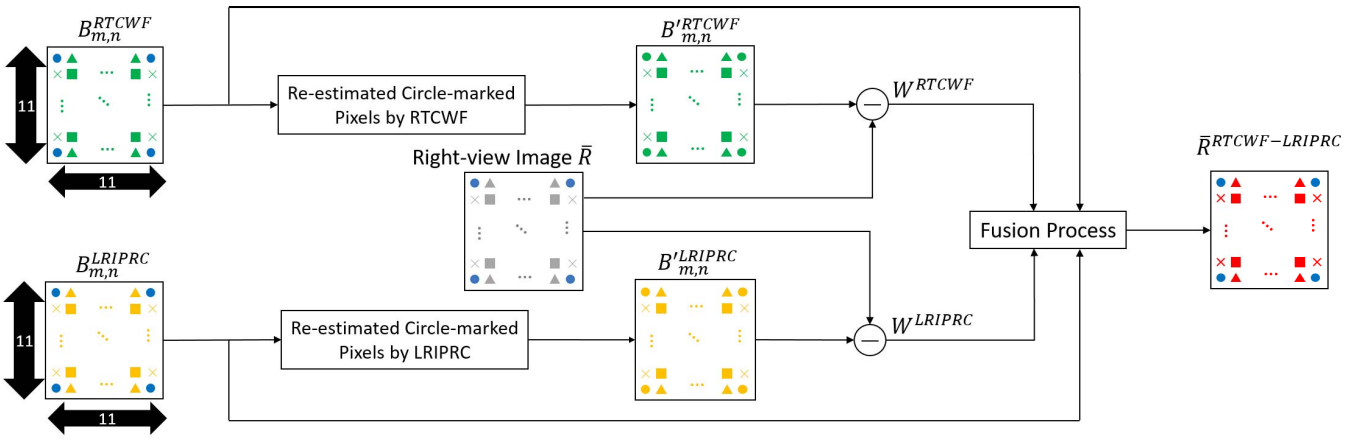


Fig. 10. The workflow of the proposed fusion strategy for RTCWF-LRIPRC.

B. The Proposed RC Process

An RC process is proposed to further enhance the quality of the estimated square-marked pixel $\overline{R}_{m,n}$. First, the four circle-marked pixels in $B^{\overline{R}}_{m,n}$ are re-estimated in the same way as we estimated $\overline{R}_{m,n}$, and the resultant $B^{\overline{R}}_{m,n}$ is denoted as $B'^{\overline{R}}_{m,n}$. Then, the residual block $\Delta B^{\overline{R}}_{m,n}$ is given by

$$\Delta B^{\overline{R}}_{m,n} = B^{\overline{R}}_{m,n} - B'^{\overline{R}}_{m,n} \tag{11}$$

After constructing all the residual blocks, the residual value of the square-marked pixel in the block $B^{\overline{R}}_{m,n}$ can be estimated by Bicubic based on the four residual values in $\Delta B^{\overline{R}}_{m,n}$, and the estimated residual value of the square-marked pixel in $B^{\overline{R}}_{m,n}$ is denoted as $\Delta\overline{R}_{m,n}$. Further, the estimated squared-marked pixel value of $\overline{R}_{m,n}$ is compensated by

$$\overline{R}^{LRIPRC}_{m,n} = \overline{R}_{m,n} + \Delta\overline{R}_{m,n} \tag{12}$$

IV. The Proposed Adaptive RTCWF-LRIPRC-Based Upsampling Method: RTCWF-LRIPRC

We propose an adaptive fusion strategy to integrate RTCWF and LRIPRC, called RTCWF-LRIPRC, to further improve the quality of the upsampled image. The workflow of RTCWF-LRIPRC is depicted in Fig. 10, where $B^{RTCWF}_{m,n}$ and $B^{LRIPRC}_{m,n}$ denote the 11×11 block centered at $\overline{R}^{RTCWF}_{m,n}$ and $\overline{R}^{LRIPRC}_{m,n}$, respectively, where $\overline{R}^{LRIPRC}_{m,n}$ has been defined in Eq. (12) and $\overline{R}^{RTCWF}_{m,n}$ is analogous but by RTCWF.

A. Predicting Errors for RTCWF and LRIPRC

The circle-marked pixels in $B^{RTCWF}_{m,n}$ and $B^{LRIPRC}_{m,n}$ are re-estimated by RTCWF and LRIPRC, respectively, and we obtain two modified blocks: $B'^{RTCWF}_{m,n}$ and $B'^{LRIPRC}_{m,n}$. The errors of $B'^{RTCWF}_{m,n}$ and $B'^{LRIPRC}_{m,n}$ are denoted by $E(B^{RTCWF}_{m,n})$ and $E(B^{LRIPRC}_{m,n})$, respectively, and they are calculated by

$$E(B^{RTCWF}_{m,n}) = \frac{\sum_{i=1}^{36}\left|B^{RTCWF}_{m,n}(i) - B'^{RTCWF}_{m,n}(i)\right|}{36} \tag{13}$$

and

$$E(B^{LRIPRC}_{m,n}) = \frac{\sum_{i=1}^{36}\left|B^{LRIPRC}_{m,n}(i) - B'^{LRIPRC}_{m,n}(i)\right|}{36}, \tag{14}$$

where '36' denotes the total number of the circle-marked pixels in the 11×11 window W_r. $B^{RTCWF}_{m,n}(i)$ and $B'^{RTCWF}_{m,n}(i)$, $1 \le i \le 36$, denote the ith circle-marked pixel values in $B^{RTCWF}_{m,n}$ and $B'^{RTCWF}_{m,n}$, respectively.

B. Adaptive Fusion Strategy for RTCWF-LRIPRC

Consequently, the final value of the square-marked pixel located at location (m,n) is fused by

$$\overline{R}^{RTCWF-LRIPRC}_{m,n} = W^{RTCWF}\times\overline{R}^{RTCWF}_{m,n} + W^{LRIPRC}\times\overline{R}^{LRIPRC}_{m,n} \tag{15}$$

where the two weights are defined by

$$W^{RTCWF} = \frac{E(B^{LRIPRC}_{m,n})}{E(B^{RTCWF}_{m,n}) + E(B^{LRIPRC}_{m,n})} \tag{16}$$

and

$$W^{LRIPRC} = \frac{E(B^{RTCWF}_{m,n})}{E(B^{RTCWF}_{m,n}) + E(B^{LRIPRC}_{m,n})}. \tag{17}$$

V. Experimental Results

In this section, thorough experiments are performed to demonstrate the quality and execution time merits of the proposed RTCWF-LRIPRC upsampling method for ARSVC when compared with the eight concerned methods [3], [8], [10], [15], [23], [25], [40], [36]. The six adopted quality metrics are PSNR, SSIM, MOVIE, StSD_{LC}, the quality-bitrate tradeoff in the RD curves, and the visual effect. The execution time required in each method is measured by seconds per image on average. A total of seven test stereoscopic video sequences, each with 100 image frames, were downloaded from the MOBILE3DTV website [26] for the performance comparison. The first test video sequence is Alt Moabit, with each image being 512×384 pixels; for simplicity, it is denoted by Alt_Moabit (512×384). The other six test video sequences

TABLE I
COMPARISON OF AVERAGE PSNR FOR THE SEVEN TEST VIDEOS AMONG THE CONCERNED UPSAMPLING METHODS

Method \ Video	Alt_Moabit	Bullinger	Car	Horse	Rollerblade	GT-Fly	Undo_Dancer	Average PSNR	Average PSNR Gain
Anchor [3]	30.46	38.50	38.03	30.98	29.84	31.56	27.36	32.39	3.40
Bicubic [23]	30.72	38.71	38.18	31.15	30.03	31.84	27.64	32.61	3.18
SAI [40]	30.79	39.93	38.14	31.06	29.76	32.09	27.90	32.81	2.98
NEDI [25]	30.45	38.50	37.45	30.37	29.15	32.21	28.08	32.32	3.47
Gao *et al.* [15]	33.56	39.48	39.11	31.75	30.58	34.32	29.50	34.04	1.75
WFIP_1_EC [8]	32.07	41.07	39.43	31.86	30.67	33.05	28.77	33.84	1.94
SRCNN [10]	33.75	**41.82**	39.77	32.68	32.02	34.36	30.57	35.00	0.79
EEDS [36]	32.86	40.70	38.82	32.00	31.10	34.04	30.34	34.27	1.52
RTCWF-only	30.86	39.20	38.18	31.09	29.87	32.00	27.79	32.71	3.07
RTCWF-LRIPRC	**34.84**	40.84	**41.27**	**34.66**	**32.72**	**35.43**	**30.76**	**35.79**	

TABLE II
COMPARISON OF AVERAGE SSIM FOR THE SEVEN TEST VIDEOS AMONG THE CONCERNED UPSAMPLING METHODS

Method \ Video	Alt_Moabit	Bullinger	Car	Horse	Rollerblade	GT-Fly	Undo_Dancer	Average SSIM	Average SSIM Gain
Anchor [3]	0.9820	0.9941	0.9958	0.9756	0.9867	0.9688	0.9831	0.9837	0.0092
Bicubic [23]	0.9830	0.9943	0.9959	0.9763	0.9872	0.9706	0.9841	0.9845	0.0084
SAI [40]	0.9832	0.9957	0.9959	0.9758	0.9864	0.9722	0.9850	0.9849	0.0080
NEDI [25]	0.9817	0.9940	0.9952	0.9711	0.9841	0.9728	0.9855	0.9835	0.0094
Gao *et al.* [15]	0.9908	0.9953	0.9967	0.9800	0.9884	0.9835	0.9897	0.9892	0.0037
WFIP_1_EC [8]	0.9875	0.9967	0.9969	0.9798	0.9889	0.9775	0.9876	0.9879	0.0050
SRCNN [10]	0.9916	**0.9972**	0.9972	0.9836	0.9920	0.9835	0.9919	0.991	0.0019
EEDS [36]	0.9897	0.9965	0.9966	0.9807	0.9901	0.9823	0.9915	0.9896	0.0033
RTCWF-only	0.9835	0.9948	0.9959	0.9759	0.9867	0.9716	0.9846	0.9847	0.0082
RTCWF-LRIPRC	**0.9935**	0.9965	**0.9980**	**0.9896**	**0.9931**	**0.9872**	**0.9922**	**0.9929**	

TABLE III
COMPARISON OF AVERAGE MOVIE FOR THE SEVEN TEST VIDEOS AMONG THE CONCERNED UPSAMPLING METHODS

Method \ Video	Alt_Moabit	Bullinger	Car	Horse	Rollerblade	GT-Fly	Undo_Dancer	Average MOVIE	Average MOVIE Gain
Anchor [3]	0.0019	0.0006	0.0006	0.0008	0.0013	0.0018	0.0030	0.0014	0.0008
Bicubic [23]	0.0020	0.0006	0.0006	0.0009	0.0015	0.0017	0.0029	0.0015	0.0008
SAI [40]	0.0022	0.0006	0.0007	0.0011	0.0019	0.0016	0.0030	0.0016	0.0009
NEDI [25]	0.0025	0.0007	0.0009	0.0014	0.0028	0.0017	0.0030	0.0018	0.0012
Gao *et al.* [15]	0.0009	0.0010	0.0003	0.0010	0.0008	0.0008	0.0022	0.0010	0.0003
WFIP_1_EC [8]	0.0019	0.0004	0.0004	0.0009	0.0015	0.0013	0.0023	0.0013	0.0006
SRCNN [10]	**0.0005**	**0.0002**	**0.0002**	**0.0004**	**0.0005**	**0.0003**	**0.0006**	**0.0004**	-0.0003
EEDS [36]	0.0009	0.0003	0.0003	0.0007	0.0011	0.0005	0.0009	0.0007	0.0000
RTCWF-only	0.0021	0.0006	0.0006	0.0010	0.0019	0.0016	0.0029	0.0015	0.0008
RTCWF-LRIPRC	0.0009	0.0004	0.0003	0.0005	0.0006	0.0007	0.0011	0.0007	

are Bullinger (432×240), Car (432×240), Horse (432×240), Rollerblade (320×240), GT-Fly (960×544), and Undo_Dancer (960×544).

The related experiments have been performed under the Windows 7 operating system. Excluding SRCNN [10] and EEDS [36], the concerned methods are implemented in Visual C++ 2015 on the following platform: 3.6 GHz Intel i7-4790 CPU with 8 GB of RAM. SRCNN [10] and EEDS [36] are implemented in Matlab R2013a and Python 3.6, respectively, on the following platform: GeForce GTX 1080 Ti GPU with 24 GB of RAM. The compression platform used is HTM 2.0, and the GOP size is set to 32. A total of 11 different QPs, 4, 8, 12, 16, 20, 24, 28, 32, 36, 40, and 44, are used for encoding the test video sequences.

A. PSNR and SSIM Quality Merits

To evaluate the quality of the upsampled right-view image, the traditional PSNR metric is used. Table I tabulates the average PSNR values of the concerned upsampling methods. Here, the direct downsampling scheme is applied to each original right-view image frame. From Table I, we observe that our RTCWF-LRIPRC method has the best average PSNR performance. When compared with the other concerned methods, the PSNR gain of RTCWF-LRIPRC is more than 0.79 dB. Note that the PSNR performance of SRCNN and EEDS is superior to that of Gao *et al.*'s method.

SSIM is used to measure the joint effects of the luminance, contrast, and structure similarity preservation between the upsampled right-view image and the equal-sized ground-truth right-view image. We suggest that readers refer to [35] for a detailed definition of SSIM. Table II tabulates the average SSIM performance for the concerned upsampling methods. From Table II, we still observe that our RTCWF-LRIPRC method has the highest average SSIM performance among the concerned methods. In addition, the SSIM performance of SRCNN and EEDS remains superior to Gao *et al.*'s method.

TABLE IV

COMPARISON OF AVERAGE $StSD_{LC}$ FOR THE SEVEN TEST VIDEOS AMONG THE CONCERNED UPSAMPLING METHODS

Method \ Video	Alt_Moabit	Bullinger	Car	Horse	Rollerblade	GT-Fly	Undo_Dancer	Average $StSD_{LC}$	Average $StSD_{LC}$ Gain
Anchor [3]	0.654	0.585	0.601	0.656	0.660	0.415	0.613	0.598	0.121
Bicubic [23]	0.658	0.575	0.632	0.659	0.661	0.314	0.583	0.583	0.106
SAI [40]	0.659	0.574	0.644	0.660	0.661	0.384	0.606	0.598	0.122
NEDI [25]	0.660	0.622	0.658	0.661	0.661	0.546	0.648	0.637	0.160
Gao *et al.* [15]	0.478	0.555	0.461	**0.647**	0.658	0.224	0.641	0.523	0.047
WFIP_1_EC [8]	0.655	0.546	0.621	0.659	0.661	0.504	0.633	0.611	0.134
SRCNN [10]	**0.465**	**0.434**	**0.447**	0.658	**0.620**	**0.030**	**0.059**	**0.388**	-0.089
EEDS [36]	0.634	0.570	0.609	0.661	0.656	0.067	0.180	0.482	0.005
RTCWF-only	0.659	0.616	0.637	0.661	0.660	0.366	0.604	0.600	0.123
RTCWF-LRIPRC	0.583	0.536	0.458	0.650	0.645	0.138	0.328	0.477	

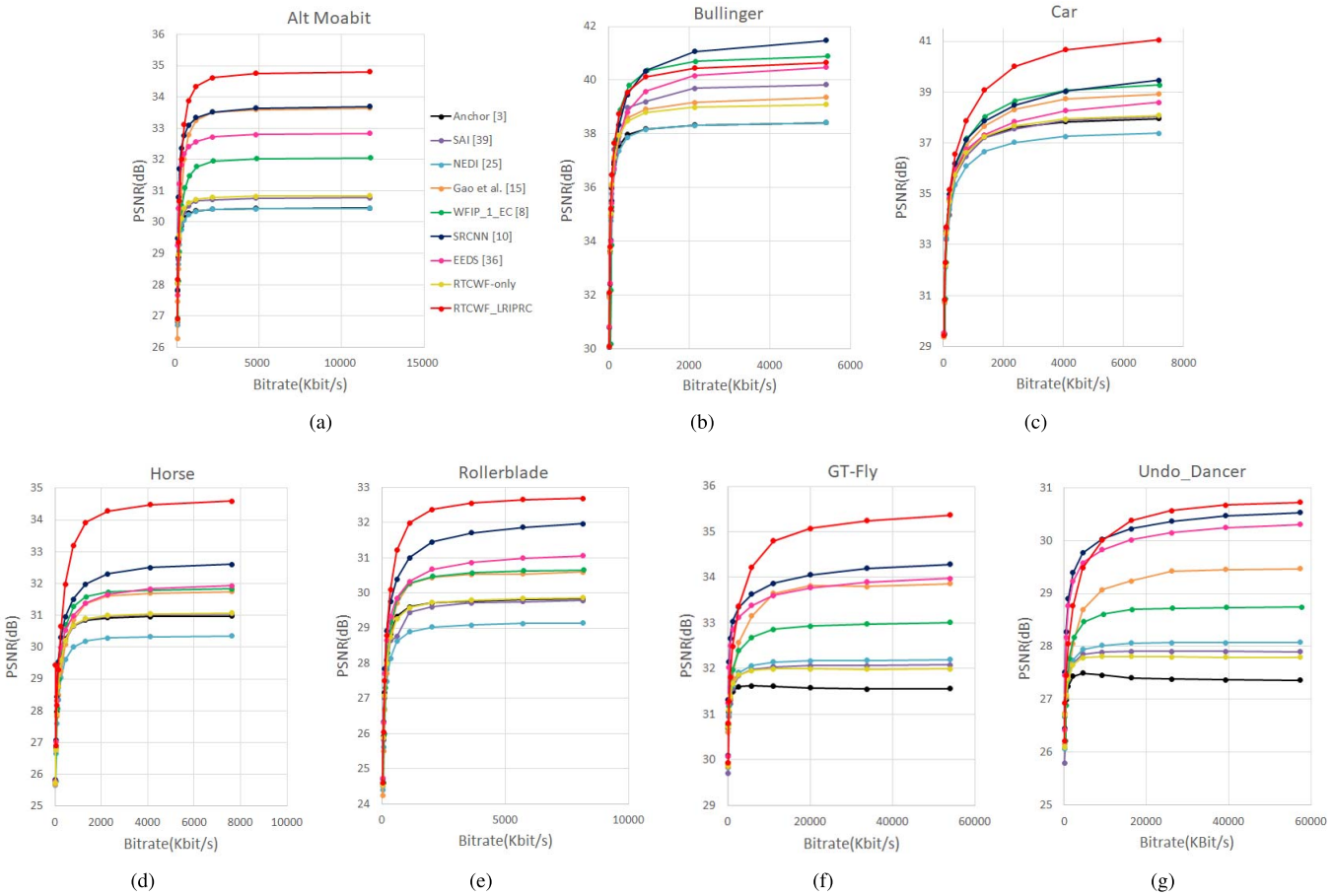


Fig. 11. The quality-bitrate tradeoff comparison. (a) Alt Moabit. (b) Bullinger. (c) Car. (d) Horse. (e) Rollerblade. (f) GT-Fly. (g) Undo_Dancer.

B. MOVIE and $StSD_{LC}$ Quality Merits

The motion-based video integrity evaluation (MOVIE) index [28] is used to measure the reconstructed video quality by considering both spatial and temporal distortion assessment. A lower MOVIE index means a better reconstructed video quality. The MOVIE index is defined by

$$\text{MOVIE} = \text{MOVIE}_{\text{spatial}} \times \text{MOVIE}_{\text{temporal}} \tag{18}$$

where the detailed definitions of the average spatial domain error, $\text{MOVIE}_{\text{spatial}}$, and the temporal domain error, $\text{MOVIE}_{\text{temporal}}$, can be found in [28]. Table III tabulates the average MOVIE performance for all the concerned methods. From Table III, we observe that SRCNN has the best MOVIE performance, i.e., the lowest average MOVIE value, and our RTCWF-LRIPRC method and EEDS are the second.

$StSD_{LC}$ [31] is a useful metric for measuring the upsampled stereoscopic video quality, and it considers the structural distortions, blur measurement, and content complexity in the upsampled stereoscopic video. A lower $StSD_{LC}$ means a better upsampled video quality. A detailed definition of the $StSD_{LC}$ metric can be found in [31]. Table IV lists the average $StSD_{LC}$ performance and indicates that SRCNN is in first place, our RTCWF-LRIPRC method is second, EEDS is third, and Gao *et al.*'s method is fourth.

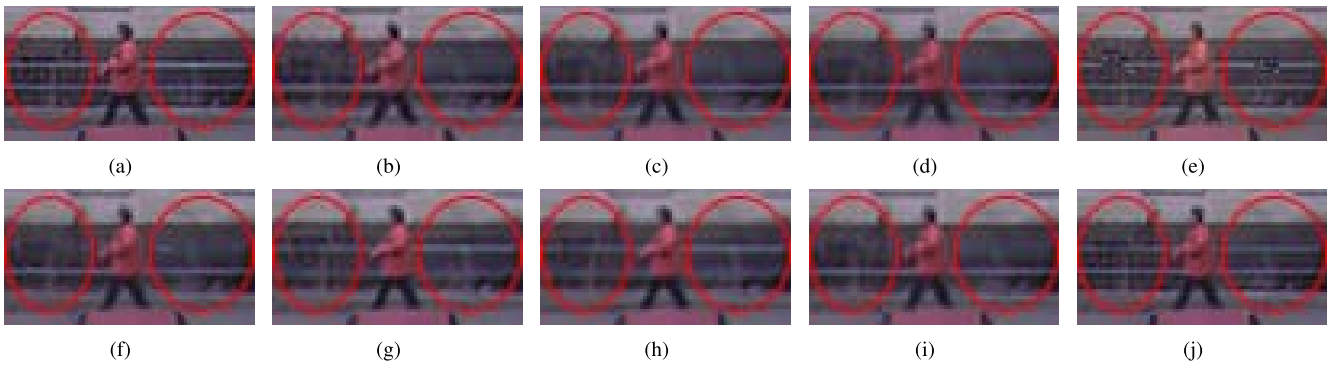


Fig. 12. The first visual effect comparison. (a) The magnified subimage cut off from the Alt Moabit image. (b) Anchor. (c) SAI. (d) NEDI. (e) Gao *et al*. (f) WFIP_1_EC. (g) SRCNN. (h) EEDS. (i) RTCWF-only. (j) RTCWF-LRIPRC.

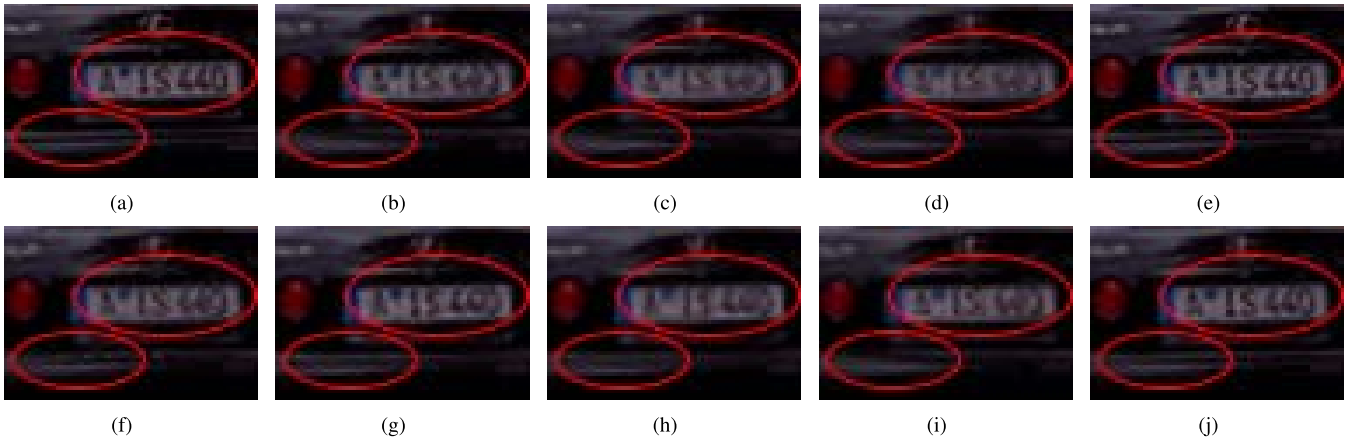


Fig. 13. The second visual effect comparison. (a) The magnified subimage cut off from the Car image. (b) Anchor. (c) SAI. (d) NEDI. (e) Gao *et al*. (f) WFIP_1_EC. (g) SRCNN. (h) EEDS. (i) RTCWF-only. (j) RTCWF-LRIPRC.

TABLE V

COMPARISON OF AVERAGE EXECUTION TIME FOR THE SEVEN TEST VIDEOS AMONG THE CONCERNED UPSAMPLING METHODS

Method \ Video	Alt_Moabit	Bullinger	Car	Horse	Rollerblade	GT-Fly	Undo_Dancer	Average Time	Average Time Gain
Anchor [3]	**0.032**	**0.018**	**0.017**	**0.017**	**0.014**	**0.087**	**0.086**	**0.039**	-0.923
Bicubic [23]	0.035	0.019	0.018	0.019	0.016	0.088	0.088	0.040	-0.921
SAI [40]	1.235	0.638	0.619	0.633	0.464	3.246	3.318	1.451	0.489
NEDI [25]	0.472	0.184	0.204	0.286	0.174	0.995	1.279	0.513	-0.448
Gao *et al.* [15]	2.581	0.680	0.982	2.275	1.057	3.222	8.596	2.770	1.809
WFIP_1_EC [8]	2.666	1.044	1.182	1.624	1.045	5.384	7.042	2.855	1.894
SRCNN [10] (on GPU)	8.916	6.503	6.463	6.303	6.267	15.283	14.839	9.225	8.246
EEDS [36] (on GPU)	2.300	2.337	2.403	2.332	2.330	2.350	2.743	2.399	1.438
RTCWF-only	0.456	0.086	0.130	0.377	0.199	0.536	1.053	0.405	-0.556
RTCWF-LRIPRC	0.987	0.272	0.369	0.734	0.399	1.529	2.439	0.961	

C. Bitrate-Distortion Tradeoff Merit in RD Curves

The RD curves are plotted to demonstrate the quality-bitrate tradeoff performance of the concerned methods under the aforementioned eleven QPs. Here, the bitrate of one compressed test video is defined by

$$\text{bitrate} = \frac{B}{N} \tag{19}$$

where B denotes the total number of bits used to compress the test video with N (= 100) images. Excluding Bicubic, Fig. 11 plots the nine RD curves of the concerned upsampling methods. In Fig. 11, the X-axis denotes the required number of Kbits per second, and the Y-axis denotes the PSNR value. From Fig. 11, we observe that except for the Bullinger video sequence, our RTCWF-LRIPRC method has the best quality-bitrate tradeoff performance for the other six video sequences, and SRCNN is the second best. For the Bullinger video sequence, SRCNN has the best quality-bitrate tradeoff performance, and RTCWF-LRIPRC is the second best.

D. Visual Effect Merit

As shown in Fig. 12(a), as the ground-truth example, we take one magnified subimage cut off from the first image of the Alt Moabit video with QP=0. After applying Anchor, SAI,

NEDI, Gao *et al.*, WFIP_1_EC, SRCNN, EEDS, our RTCWF-only method, and our RTCWF-LRIPRC method to Fig. 12(a), Figs. 12(b)-(j) demonstrate the nine upsampled subimages, respectively. As shown in the regions marked by red ellipses, RTCWF-LRIPRC has the best visual effect, especially for horizontal textures. The second visual effect comparison is illustrated in Fig. 13, and we observe that RTCWF-LRIPRC and Gao *et al.*'s method have the best visual effect, especially for license numbers.

E. Execution Time Comparison

Table V illustrates the execution time comparison among the concerned methods. From Table V, we observe that, excluding RTCWF-only, our RTCWF-LRIPRC method, which is implemented on a CPU, is the fourth best among the concerned methods and is much faster than SRCNN and EEDS, which are implemented on a GPU. On average, RTCWF-LRIPRC needs 0.961 seconds per image on a CPU, EEDS needs 2.399 seconds in Python 3.6 on a GPU under the Keras framework, and SRCNN needs 9.225 seconds in Matlab R2013a on a GPU under the Caffe framework. The number of MAC operations of SRCNN is 570.42k, as counted by the Netscope CNN Analyzer, indicating why the execution time performance of SRCNN is the worst among the concerned methods.

Tables I-IV and Figs. 11-13 indicate that among the concerned methods, the proposed RTCWF-LRIPRC method has the best PSNR, SSIM, quality-bitrate tradeoff, and visual effect performance and is the second best in the MOVIE and $\mathrm{StSD_{LC}}$ performance comparison. According to the execution time and overall quality performance comparison for seven typical test stereoscopic video sequences, we conclude that our proposed RTCWF-LRIPRC upsampling method performs best among the concerned methods for ASRVC.

VI. Conclusion

We have presented the proposed RTCWF-LRIPRC upsampling method for ARSVC. The resolution- and texture-inconsistency problem in the conventional WF-based upsampling scheme in the spatial domain has been resolved by RTCWF. The rugged residual field occurring in the previous WF-based IP prediction scheme in the interview domain has been improved by LRIPRC. An adaptive fusion-based strategy is proposed to integrate RTCWF and LRIPRC such that the quality improvement of the upsampled image can be maximized. Based on seven typical test stereoscopic video sequences, the experimental results have shown that, in terms of six quality metrics and the execution time performance on two implementation platforms, our RTCWF-LRIPRC method achieves the best overall performance among the concerned methods for ARSVC. Recently, Zhang *et al.* [42] proposed a successful super-resolution CNN method for multiple degradations (SRMD), in which the blur and noise degradations [41] are considered. Future research work will integrate RTCWF-LRIPRC and the CNN architecture to handle multiple degradations in upsampling for ARSVC.

Acknowledgment

The authors appreciate the programming help of Mr. C. Cheng, the proofreading help of Ms. C. Harrington, and the valuable comments of the Associated Editor Dr. Chia-Kai Liang and the three anonymous reviewers for improving the manuscript.

References

[1] P. Aflaki, M. M. Hannuksela, J. Häkkinen, P. Lindroos, and M. Gabbouj, "Subjective study on compressed asymmetric stereoscopic video," in *Proc. IEEE Int. Conf. Image Process.*, Sep. 2010, pp. 4021–4024.

[2] J. Allebach and P. W. Wong, "Edge-directed interpolation," in *Proc. 3rd IEEE Int. Conf. Image Process.*, Sep. 1996, pp. 707–710.

[3] B. Bross, W.-J. Han, G. J. Sullivan, J. R. Ohm, and T. Wiegand, "High efficiency video coding (HEVC) text specification draft 9," document JCTVC-K1003, Beijing, China, Oct. 2012.

[4] H. Brust, A. Smolic, K. Mueller, G. Tech, and T. Wiegand, "Mixed resolution coding of stereoscopic video for mobile devices," in *Proc. 3DTV Conf. True Vis.-Capture, Transmiss. Display 3D Video*, May 2009, pp. 1–4.

[5] Y. Chen, Y.-K. Wang, M. Gabbouj, and M. M. Hannuksela, "Regionally adaptive filtering for asymmetric stereoscopic video coding," in *Proc. IEEE Int. Symp. Circuits Syst.*, May 2009, pp. 2585–2588.

[6] Y. Chen, Y.-K. Wang, M. M. Hannuksela, and M. Gabbouj, "Picture-level adaptive filter for asymmetric stereoscopic video," in *Proc. IEEE Int. Conf. Image Process.*, Oct. 2008, pp. 1944–1947.

[7] C. Cho, J. Jeon, and J. Paik, "Example-based super-resolution using self-patches and approximated constrained least squares filter," in *Proc. IEEE Int. Conf. Image Process.*, Oct. 2014, pp. 2140–2144.

[8] K.-L. Chung, Y.-H. Huang, and W.-C. Liu, "Quality-efficient upsampling method for asymmetric resolution stereoscopic video coding with interview motion compensation and error compensation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 24, no. 3, pp. 430–442, Mar. 2014.

[9] 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.

[10] C. Dong, C. C. Loy, and X. Tang, "Accelerating the super-resolution convolutional neural network," in *Proc. Eur. Conf. Comput. Vis.*, Aug. 2016, pp. 391–407.

[11] DVB Consortium. (Jul. 2010). *DVB Commercial Requirements for DVB 3-D-TV*. [Online]. Available: http://www.dvb.org

[12] C. Fehn, "Depth-image-based rendering (DIBR), compression, and transmission for a new approach on 3D-TV," *Proc. SPIE*, vol. 5291, p. 524762, May 2004.

[13] C. Fehn, P. Kauff, S. Cho, H. Kwon, N. Hur, and J. Kim, "Asymmetric coding of stereoscopic video for transmission over T-DMB," in *Proc. 3DTV Conf.*, May 2007, pp. 1–4.

[14] W. T. Freeman, T. R. Jones, and E. C. Pasztor, "Example-based super-resolution," *IEEE Comput. Graph. Appl.*, vol. 22, no. 2, pp. 56–65, Mar./Apr. 2002.

[15] M. Gao, S. Ma, D. Zhao, and W. Gao, "A spatial inter-view auto-regressive super-resolution scheme for multi-view image via scene matching algorithm," in *Proc. IEEE Int. Symp. Circuits Syst.*, May 2013, pp. 2880–2883.

[16] J. Ge, J. Liu, Y. Zhao, and B. Zhang, "A resolution enhancement algorithm for an asymmetric resolution stereo video," *EURASIP J. Image Video Process.*, vol. 2015, no. 1, p. 23, 2015.

[17] S. Gu, W. Zuo, Q. Xie, D. Meng, X. Feng, and L. Zhang, "Convolutional sparse coding for image super-resolution," in *Proc. Int. Conf. Comput. Vis.*, Dec. 2015, pp. 1823–1831.

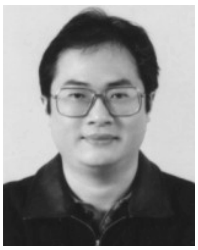
[18] *Adaptive Basic Unit Layer Rate Control for JVT*, Document JVT-G012, ISO/IEC JTC1/SC29/WG11 and ITU-T SG16/Q.6, Pattaya, Thailand, 2003.

[19] S. Jeong, I. Yoon, J. Jeon, and J. Paik, "Multi-frame example-based super-resolution using locally directional self-similarity," in *Proc. IEEE Int. Conf. Consum. Electron.*, Jan. 2015, pp. 631–632.

[20] S. Jeong, I. Yoon, and J. Paik, "Multi-frame example-based super-resolution using locally directional self-similarity," *IEEE Trans. Consum. Electron.*, vol. 61, no. 3, pp. 353–358, Aug. 2015.

[21] L.-W. Kang, C.-C. Hsu, C.-W. Lin, and C.-H. Yeh, "Learning-based joint super-resolution and deblocking for a highly compressed image," *IEEE Trans. Multimedia*, vol. 17, no. 7, pp. 921–934, Jul. 2015.

[22] A. Kappeler, S. Yoo, Q. Dai, and A. K. Katsaggelos, "Video super-resolution with convolutional neural networks," *IEEE Trans. Comput. Imag.*, vol. 2, no. 2, pp. 109–122, Jun. 2016.
[23] R. G. Keys, "Cubic convolution interpolation for digital image processing," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-29, no. 6, pp. 1153–1160, Dec. 1981.
[24] J. Kim, J. K. Lee, and K. M. Lee, "Accurate image super-resolution using very deep convolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2016, pp. 1646–1654.
[25] X. Li and M. T. Orchard, "New edge-directed interpolation," *IEEE Trans. Image Process.*, vol. 10, no. 10, pp. 1521–1527, Oct. 2001.
[26] MOBILE3DTV Stereo Video Contents. (Feb. 2009). *Mobile 3DTV Content Delivery Optimization Over DVB-H System*. [Online]. Available: http://sp.cs.tut.fi/mobile3dtv/stereo-video
[27] S.-N. Park and D.-G. Sim, "View-dependency video coding for asymmetric resolution stereoscopic views," *Opt. Eng.*, vol. 48, no. 7, p. 077009, Jul. 2009.
[28] K. Seshadrinathan and A. C. Bovik, "Motion tuned spatio-temporal quality assessment of natural videos," *IEEE Trans. Image Process.*, vol. 19, no. 2, pp. 335–350, Feb. 2010.
[29] P. Seuntiens, L. Meesters, and W. Ijsselsteijn, "Perceived quality of compressed stereoscopic images: Effects of symmetric and asymmetric JPEG coding and camera separation," *ACM Trans. Appl. Perception*, vol. 3, no. 2, pp. 95–109, Apr. 2006.
[30] F. Shao, G. Jiang, M. Yu, K. Chen, and Y.-S. Ho, "Asymmetric coding of multi-view video plus depth based 3-D video for view rendering," *IEEE Trans. Multimedia*, vol. 14, no. 1, pp. 157–167, Feb. 2012.
[31] V. De Silva, H. K. Arachchi, E. Ekmekcioglu, and A. Kondoz, "Toward an impairment metric for stereoscopic video: A full-reference video quality metric to assess compressed stereoscopic video," *IEEE Trans. Image Process.*, vol. 22, no. 9, pp. 3392–3404, Sep. 2013.
[32] J. Tian, L. Chen, and Z. Liu, "Dual regularization-based image resolution enhancement for asymmetric stereoscopic images," *Signal Process.*, vol. 92, no. 2, pp. 490–497, 2012.
[33] R. Timofte *et al.*, "NTIRE 2017 challenge on single image super-resolution: Methods and results," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops*, Jul. 2017, pp. 1110–1121.
[34] R. Timofte, V. De Smet, and L. Van Gool, "A+: Adjusted anchored neighborhood regression for fast super-resolution," in *Proc. Asian Conf. Comput. Vis.*, Singapore, Nov. 2014, pp. 111–126.
[35] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004.
[36] Y. Wang, L. Wang, H. Wang, and P. Li, "End-to-end image super-resolution via deep and shallow convolutional networks," in *Proc. Comput. Vis. Pattern Recognit.*, Jul. 2016, pp. 1–10.
[37] D. S. Watkins, *Fundamentals of Matrix Computations*. New York, NY, USA: Wiley, 1991, sec. 3.5.
[38] 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.
[39] J. Zhang, Y. Cao, Z.-J. Zha, Z. Zheng, C. W. Chen, and Z. Wang, "A unified scheme for super-resolution and depth estimation from asymmetric stereoscopic video," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 26, no. 3, pp. 479–493, Mar. 2016.
[40] 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.
[41] K. Zhang, W. Zuo, S. Gu, and L. Zhang, "Learning deep CNN denoiser prior for image restoration," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 3929–3983.
[42] K. Zhang, W. Zuo, and L. Zhang, "Learning a single convolutional super-resolution network for multiple degradations," in *Proc. Comput. Vis. Pattern Recognit.*, 2018.



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 is currently a Chair Professor with the Department of Computer Science and Information Engineering, National Taiwan University of Science and Technology, Taipei. 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. He received the Scientific Paper Award from the Far Eastern Y. Z. Hsu Science and Technology Memorial Foundation in 2017. He has been an Associate Editor of the *Journal of Visual Communication and Image Representation* since 2011.

Shin-Bei Tsai received the B.S. degree in electronic engineering from the National Yunlin University of Science and Technology, Yunlin, Taiwan, in 2015. She is currently pursuing the M.S. degree in computer science and information engineering with the National Taiwan University of Science and Technology, Taipei, Taiwan. Her research interests include image processing and video compression.

Yu-Ling Tseng received the B.S. degree in computer science and information engineering from the National Taipei University of Education, Taipei, Taiwan, in 2016. She is currently pursuing the M.S. degree in computer science and information engineering with the National Taiwan University of Science and Technology, Taipei. Her research interests include image processing and video compression.

Chi-Chao Huang received the B.S. degree in computer science and information engineering from National Chengchi University, Taipei, Taiwan, in 2015. 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.