J. Vis. Commun. Image R. 30 (2015) 350–362

Contents lists available at ScienceDirect

J. Vis. Commun. Image R.

journal homepage: www.elsevier.com/locate/jvci

Fast and quality-efficient scheme for asymmetric multi-view video plus depth coding under the bitrate constraint ☆

Chien-Hsiung Lin [a], Kuo-Liang Chung [a,*], Jiann-Jone Chen [b], Yung-Hsiang Chiu [a], Yan-Nan Chen [a]

[a] Department of Computer Science and Information Engineering, National Taiwan University of Science and Technology, No. 43, Section 4, Keelung Road, Taipei 10672, Taiwan, ROC
[b] Department of Electrical Engineering, National Taiwan University of Science and Technology, No. 43, Section 4, Keelung Road, Taipei, 10672, Taiwan, ROC

ARTICLE INFO

Article history:
Received 18 November 2014
Accepted 30 April 2015
Available online 27 May 2015

Keywords:
3D video
3D multi-view video plus depth coding
Adaptive weighted mode in-loop filter
Asymmetric video coding
Bitrate-constrained video coding
Depth image-based rendering
JMVM
Texture image-assisted inter prediction

ABSTRACT

Recently, multi-view video plus depth based 3D video (3D-MVD) coding has been studied extensively. This paper presents a fast and efficient asymmetric 3D-MVD coding scheme under specified bitrate constraints. To efficiently encode depth images, a texture image-assisted inter prediction strategy is proposed to determine whether to use a direct copy of the co-located block as the inter prediction so as to skip the normal encoding process. In addition, an adaptive weighted mode in-loop filter strategy is proposed to refine the reconstructed depth image quality. Both strategies help to significantly reduce the encoding time and improve the reconstructed depth image quality under the bitrate constraint. Experiment results showed that the proposed asymmetric 3D-MVD coding scheme does achieve better quality of the reconstructed depth videos and the rendered virtual views at a less encoding-time requirement when compared with the state-of-the-art scheme by Shao et al. and the traditional 5:1 bitrate allocation scheme.

1. Introduction

With the advances in video coding technology, application systems such as 3D television (3DTV) [2,18] and free viewpoint television (FTV) [25,18] enable viewers to experience real 3D scenes through specific display terminals. The former can render 3D scenes with a depth map on a stereoscopic 3D displayer, while the latter allows viewers to experience 3D scenes from arbitrary viewpoints. To improve the perception quality of 3D videos, more than one video view of the 3D scenes needs to be captured as the input for either 3DTV or FTV. To this end, more capturing and storage devices are required, leading to a significant increase in the cost of 3D videos. One alternative is to synthesize the captured videos of specific views to yield virtual views for 3D scenes. For practical 3D scene display, various representation formats can be adopted to render 3D videos, among which the multi-view video plus depth (MVD) [32] format is the most widely used for both 3DTV and FTV applications.

One MVD based 3D video, abbreviated as one 3D-MVD, comprises multiple views of videos, each of which is represented by a texture video with its synchronized depth video. For 3D-MVD display, virtual views from arbitrary viewing positions can be synthesized from the acquired videos with depth information via depth image-based rendering (DIBR) technology [1]. Furthermore, the occlusion and holes in the virtual views can be compensated for by using the information contained in the acquired views. Since the 3D-MVD comprises 2D texture and depth information of 3D scenes, the conventional 2D video coding and transmission facilities can be directly applied to 3D-MVDs.

Due to the enormous amount of information in 3D-MVDs, they need to be compressed for transmission over the Internet. Conventional compression schemes for a 3D-MVD tackle the texture and depth videos individually without exploring the correlation between them. For texture information of the 3D-MVD, simulcast coding and multi-view video coding are two commonly used compression schemes. The former independently compresses the texture video for each view using H.264/AVC [28], while the latter exploits the inter-view correlations to jointly compress the texture videos of all views [13,15,23,26,4]. From the viewpoint of compression, the former is easy to implement but inefficient, while the latter demonstrates better compression performance but at high computational cost. As for depth videos, since they can be treated as the luminance component of texture videos, the compression schemes for texture videos can also be applied to compress depth videos.

In general, the quality of synthesized videos depends on the quality of the reconstructed texture and depth videos. Although the texture and depth videos can be compressed individually,

☆ This paper has been recommended for acceptance by "M.T. Sun".
* Corresponding author.
E-mail address: klchung01@gmail.com (K.-L. Chung).

http://dx.doi.org/10.1016/j.jvcir.2015.04.016
1047-3203/

the correlation between them can be exploited to yield a better rate-distortion performance [14]. To exploit this correlation, the joint compression schemes that utilize the motion vectors of the texture image as the counterpart of the depth image were developed in [3,20]. Due to the difference in characteristics between texture and depth videos, directly applying the motion vectors of the texture image to the depth image usually results in large residuals around object boundaries and edges. In addition to the above compression schemes using the motion vectors, another joint compression schemes to skip selected blocks of the depth image [9] and fast determine the inter prediction mode of the depth image [24] were proposed based on the temporal and inter-view correlations of the corresponding texture images. These joint compression schemes can effectively reduce the encoding bitrate complexity, but the quality of the synthesized video was not well addressed.

Transmitting compressed 3D-MVDs via the Internet for users with different available bandwidths would impose different bitrate constraints on the encoding. Previous research on constrained bitrate compression has focused on performing bitrate allocation between texture and depth videos to improve the quality of the synthesized video. A European Information Society Technologies project [1] recommends a 5:1 ratio for video bitrate allocation between texture and depth information from empirical results. However, since the texture and depth characteristics of videos are dynamic, this fixed bitrate allocation ratio cannot yield satisfactory synthesized video quality. Morvan et al. [17] proposed an optimal bitrate allocation scheme to minimize the distortion between the synthesized and the real videos from the same viewpoint. The optimal bitrate allocation is carried out by an exhaustive search over all possible quantization parameter (QP) pairs for texture and depth videos, which is time-consuming. Since the real view video at the synthesized viewpoint may not be available in practice, Liu et al. [11] proposed to estimate, through a developed model, the distortion between the synthesized and the real videos from the same viewpoint and then to search exhaustively for the optimal QP pairs for texture and depth videos. Although the bitrate allocation schemes proposed in [17,11] can yield higher synthesized video quality, the time complexity of the optimized search is too high for the schemes to be feasible for practical 3D-MVD applications.

According to the binocular suppression theory [5], the higher reconstruction quality view will compensate for the lower one to the human eye such that the perceptual quality degradation on the stereoscopic display terminal for the latter can be neglected. This visual suppression property enables the development of more efficient coding schemes for achieving comparable 3D perception quality at a lower bitrate. When the encoder is performing bitrate allocation among multiple view videos under a bitrate constraint, the saved bitrate from utilizing the visual suppression property can be used to further improve the quality of the synthesized video. Shao et al. [22] proposed an asymmetric 3D-MVD coding scheme (AMVDC) to improve the quality of the reconstructed depth videos for some views by utilizing the bitrate saved from not encoding the chroma component of the texture videos for the other views. With the enhanced depth videos at the decoder, the AMVDC proposed by Shao et al. can provide more accurate pixel correspondences, when reconstructing the discarded chroma component, to yield better synthesized video quality. However, unlike texture images, depth images generally exhibit high spatial pixel correlations except on object boundaries or edges. These edge regions in the depth images are more sensitive for generating artifacts of the rendered views, and need higher bitrates than smooth regions such as backgrounds. In other words, equally enhancing the compression of edges and smooth regions in the depth videos without considering the above special characteristics in [22] leads to limited quality enhancement for depth videos.

In this paper, we propose a fast and quality-efficient scheme for AMVDC with bitrate constraints to solve the above problem in the state-of-the-art AMVDC proposed by Shao et al. In the proposed scheme, specifically for coding depth images, two novel strategies, the texture image-assisted inter prediction (TIAIP) strategy and the adaptive weighted mode in-loop filter (AWMIF) strategy, are delivered to speed up the encoding process and improve the quality of the reconstructed depth images. Since the distortion which occurs at the edges of depth images often causes more artifacts in the synthesized views, higher bitrates should be allocated to the edges, as compared to smooth regions, to reduce reconstruction distortion. To this end, the proposed TIAIP first determines whether the current depth block can use the direct copy of the co-located reference depth block without normal encoding in consideration of the temporal correlations of the depth images and those of the associated texture images. The resultant saved bitrate from these skipped depth blocks, most of which appear in smooth regions, can then be re-allocated to enhance the compression of the remaining unencoded depth blocks on the edges and, meanwhile a considerable acceleration of the encoding process can be achieved. Furthermore, since the edges of the depth images often coincide with those of the associated texture images, the proposed AWMIF further improves the quality of the reference depth image by utilizing the edge coherence between the patches of the compressed texture and depth images. Experiments on two typical 3D-MVDs, *Ballet* and *Breakdancers*, show that the proposed asymmetric coding scheme for 3D-MVDs outperforms the state-of-the-art scheme by Shao et al. [22] in terms of quality of both the reconstructed depth and synthesized view videos by 5.7 dB and 1.58 dB on average, respectively, and in encoding time reduction by about 19.53%. As compared to the traditional 5:1 bit allocation scheme, the improved PSNRs (peak signal-to-noise ratios) are 8.97 dB and 2.25 dB on average, respectively, for the reconstructed depth and synthesized view videos, and there is an encoding time reduction of 21.79%.

The rest of this paper is organized as follows. The AMVDC proposed by Shao et al. is reviewed in Section 2. In Section 3, the proposed fast and quality-efficient scheme is described. Section 4 demonstrates the experimental results and performance evaluation. Section 5 concludes this paper.

2. The asymmetric 3D-MVD coding scheme by Shao et al.

The asymmetric 3D-MVD coding (AMVDC) scheme proposed by Shao et al. is designed to optimize the quality of synthesized view videos by performing the bitrate allocation among different acquired view videos under the specified bitrate. For easy explanation, the 3D-MVD example comprises only two video views, the fourth and sixth views of the *Ballet* and *Breakdancers* videos, which can be extended to deal with more views in the same way. The complete flowchart of the AMVDC is demonstrated in Fig. 1, where JMVM denotes the abbreviation of the joint multi-view video model, and the bitrate allocation and compression operations are described as follows.

In general, better depth image quality would lead to a more accurate warping result which provides better rendered views or other acquired views. For coding the 3D-MVD with two acquired views, left and right, under a certain bitrate constraint, the AMVDC was designed to allocate more bitrates to encode the left-view depth images from the bitrates for the chroma component of the right-view texture images. For practical implementation, the first two GOPs of the 3D-MVD are first encoded under a bitrate constraint R_C to estimate the target bitrates for the left-view and right-view

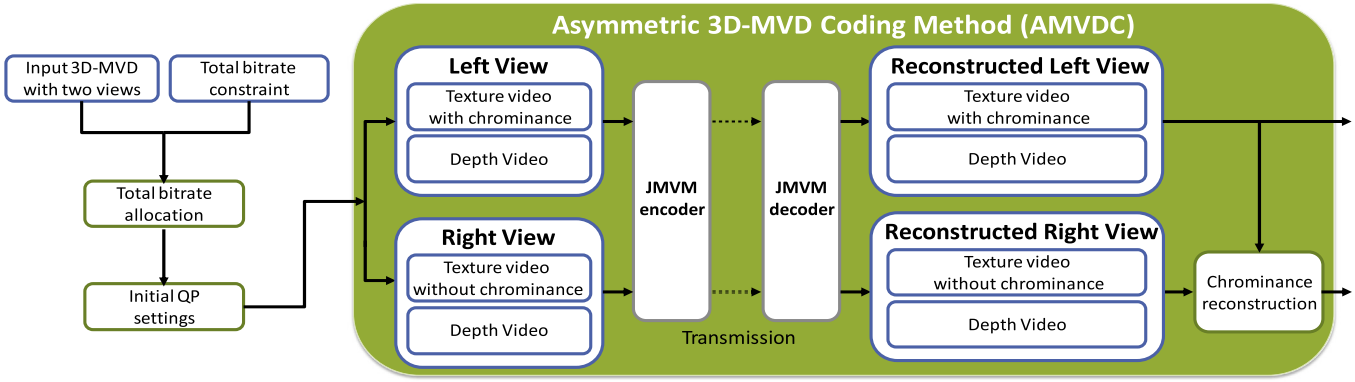

Fig. 1. Flowchart of the asymmetric 3D-MVD video coding.

videos, respectively. Let R_L denote the used bitrate of the left-view after encoding several GOPs of the 3D-MVD, $R_{L,t}$ and $R_{L,d}$ denote those of the left-view texture and depth images, while $R_{L,t,y}$ and $R_{L,t,uv}$ denote those of the left-view luminance and chroma components. The same notation can also be applied to the right-view video. Under the constrained bitrate, R_C, the target allocated bitrates for the left-view and right-view texture and depth images by the AMVDC can be expressed as

$$\widehat{R}_{L,t} = R_C \cdot \frac{R_L}{R_L + R_R} \cdot \frac{R_{L,t}}{R_{L,t} + R_{L,d}} \tag{1}$$

$$\widehat{R}_{L,d} = R_C \cdot \frac{R_L}{R_L + R_R} \cdot \frac{R_{L,d}}{R_{L,t} + R_{L,d}} + R_C \cdot \frac{R_R}{R_L + R_R} \cdot \frac{R_{R,t}}{R_{L,t} + R_{RL,d}} \cdot \frac{R_{R,t,uv}}{R_{R,t,y} + R_{R,t,uv}} \tag{2}$$

$$\widehat{R}_{R,t} = R_C \cdot \frac{R_R}{R_L + R_R} \cdot \frac{R_{R,t}}{R_{L,t} + R_{RL,d}} \cdot \frac{R_{R,t,y}}{R_{R,t,y} + R_{R,t,uv}} \tag{3}$$

$$\widehat{R}_{R,d} = R_C \cdot \frac{R_R}{R_L + R_R} \cdot \frac{R_{R,d}}{R_{R,t} + R_{R,d}}. \tag{4}$$

Note that the bitrate for the right-view chroma components, U and V, is moved from the target bitrate $\widehat{R}_{R,t}$ in Eq. (3) to the left-view depth video $\widehat{R}_{L,d}$ in Eq. (2). To further investigate the effect of the bitrate allocation between the texture video and depth video on the quality of the synthesized images, a view synthesis distortion model, $D_{sv}(R_t, R_d)$ [30,21,7], is employed to find the optimal bitrate allocation between the texture and depth videos for each acquired view. The distortion model can be described by the following formulas:

$$(R^*_{L,t}, R^*_{L,d}) = \arg \min_{0.2 \leqslant \frac{R_t}{R_t + R_d} \leqslant 0.8} D_{sv}(R_t, R_d)$$

$$s.t.\ R_t + R_d = R_C \cdot \frac{R_L}{R_L + R_R}, \tag{5}$$

$$(R^*_{R,t}, R^*_{R,d}) = \arg \min_{0.2 \leqslant \frac{R_t}{R_t + R_d} \leqslant 0.8} D_{sv}(R_t, R_d)$$

$$s.t.\ R_t + R_d = R_C \cdot \frac{R_R}{R_L + R_R}. \tag{6}$$

By substituting $R_{L,t}, R_{L,d}, R_{R,t}$, and $R_{R,d}$ in Eqs. (1)–(4) with the optimal values $R^*_{L,t}, R^*_{L,d}, R^*_{R,t}$, and $R^*_{R,d}$ from Eqs. (1) and (3), the modified bitrate allocation formulas can yield the final target bitrates allocated for the texture and depth videos in the left and right views.

After determining the target bitrates for the left-view and right-view texture and depth videos, the initial guess of QP for each video to be encoded will be carried out. For accurate bitrate control, a linear rate-quantization model [12] is employed to yield good QP guesses for all texture and depth videos, i.e.,

$$\widehat{R}_{L,t} = \frac{K_{L,t}}{Q^{step}_{L,t}} + C_{L,t} \tag{7}$$

$$\widehat{R}_{L,d} = \frac{K_{L,d}}{Q^{step}_{L,d}} + C_{L,d} \tag{8}$$

$$\widehat{R}_{R,t} = \frac{K_{R,t}}{Q^{step}_{R,t}} + C_{R,t} \tag{9}$$

$$\widehat{R}_{R,d} = \frac{K_{R,d}}{Q^{step}_{R,d}} + C_{R,d} \tag{10}$$

where $\{Q^{step}_{v,p} | v \in \{L,R\}, p \in \{t,d\}\}$ are the quantization step sizes for the corresponding videos, and $\{K_{v,p}, C_{v,p} | v \in \{L,R\}, p \in \{t,d\}\}$ are model parameters which are, respectively, characterized by the first encoded frame of the corresponding video. With the above set of quantization steps, $\{Q^{step}_{v,p}\}$, the initial QP guesses for the texture and depth videos of the left and right views can be established through the $QP - Q^{step}$ relational equation $Q^{step} = 2^{(QP-4)/6}$ [31].

After setting the target bitrates and the initial QPs, the left-view and right-view texture and depth videos can be encoded with these parameters. To compress the left-view and right-view videos with the specified coding parameters described above, the AMVDC combines each texture image with its corresponding depth image to yield a combined image, and the multi-view video encoder can be utilized to encode the video of combined images. To prevent mutual references between texture and depth data in one combined image, the texture and the depth data are encoded as two independent slices in one encoded frame [22]. In addition, it adopts a hierarchically coded B picture for compression, in which the left-view video is independently encoded while the right-view one uses the disparity compensated prediction technique.

Although the parameter QPs have been specified for coding videos, it does not imply that the encoded bitrate can precisely achieve its target value. To solve this problem, the AMVDC adopts the H.264/AVC [12] bitrate control strategy, which performs bitrate control by assigning appropriate QPs in the encoding process at the GOP layer, frame layer and macroblock layer. The bitrate allocated to one GOP is determined according to its target bitrate and the initial QP. For the frame layer rate control, it allocates bitrates according to frame complexity, buffer fullness and the remaining bitrates in the current GOP. The quantization step $Q^{step}(i,j)$ for the jth frame in the ith GOP can then be specified by the quadratic rate-quantization model [8]

$$\widehat{R}(i,j) = a_1 \times \frac{MAD(i,j)}{Q^{step}(i,j)} + a_2 \times \frac{MAD(i,j)}{Q^{step}(i,j)^2} + H(i,j), \tag{11}$$

where $\widehat{R}(i,j)$ denotes the allocated bitrate for the jth frame in the ith GOP, $MAD(i,j)$ is the mean absolute difference of graylevels between the original jth image frame and its predicted frame, $H(i,j)$ is the sum of header bits and motion bits, and a_1 and a_2 are the model parameters. Based on the above-mentioned relationship between the QP and the quantization step, the QP for encoding the jth frame in the ith GOP can be computed. Besides, since the texture and depth contents in one video are time-variant, before encoding the next GOP, the AMVDC updates the target bitrates specified in Eqs. (1)–(4) to fully utilize the available bitrate budget. After executing the above control steps, the 3D-MVD with left and right views is encoded with asymmetric quality, which will be compensated for and made balanced at the decoder.

As the chroma component of the right-view video is excluded from encoding, its reconstructed quality at the decoder would impact the AMVDC performance. To improve the reconstructed quality of the chroma component at the decoder, the 3D warping technique [1] is utilized by the AMVDC to reconstruct the discarded right-view chroma component based on the reconstructed chroma and depth information of the left view. For the artifacts of 3D warping, occlusion and mapping depletion region, a colorization method [10], which assumes that luma pixels with coherent neighbors will also demonstrate coherence in the chroma domain, is adopted by the AMVDC to inpaint for the artifacts of the chroma data.

3. Proposed fast and quality-efficient AMVDC for depth video under the bitrate constraint

The AMVDC proposed by Shao et al. is designed to enhance the quality of the left-view reconstructed depth video by utilizing the saved bitrate from not encoding the right-view chroma components. However, the special characteristics of a depth image and its correlation with the corresponding texture image are not well utilized in their scheme such that the quality enhancement is limited. In this section, we propose a novel fast and quality-efficient scheme, which investigates the special characteristics of the depth image and the correlation between the texture video and the depth video, for improving the depth image coding of the AMVDC proposed by Shao et al. In what follows, the two improved strategies proposed in our scheme, (1) texture image-assisted inter prediction and (2) adaptive weighting mode in-loop filter, are presented.

3.1. The proposed texture image-assisted inter prediction strategy

For scene background or object regions in one video, the corresponding depth images often demonstrate similar pixel values in adjacent video frames. As both depth and texture images are captured from the same real-world 3D scene, the consistencies between successive depth images and texture images are also highly correlated. Recently, Lee et al. [9] developed, based on the temporal and inter-view correlations of texture images, an efficient 3D-MVD compression scheme to skip selected blocks of the corresponding depth images. However, the check for their skip mode based on the inter-view correlation often requires substantial computational cost in view-to-view warping and its results heavily depend on the used hole-filling technique. Besides, the spatial and temporal correlations of depth images are not taken into consideration in Lee et al.'s scheme, so the wrong skip judgement of the selected depth blocks sometimes occurs when only the temporal correlation of texture images is used. For example, in Fig. 2, since the block of the current texture image is similar to the co-located block of the temporally-referenced texture image, Lee et al.'s scheme skips the normal encoding procedure and wrongly duplicates the co-located encoded block of the temporally-referenced depth image as the inter prediction result of the block of the current depth image.

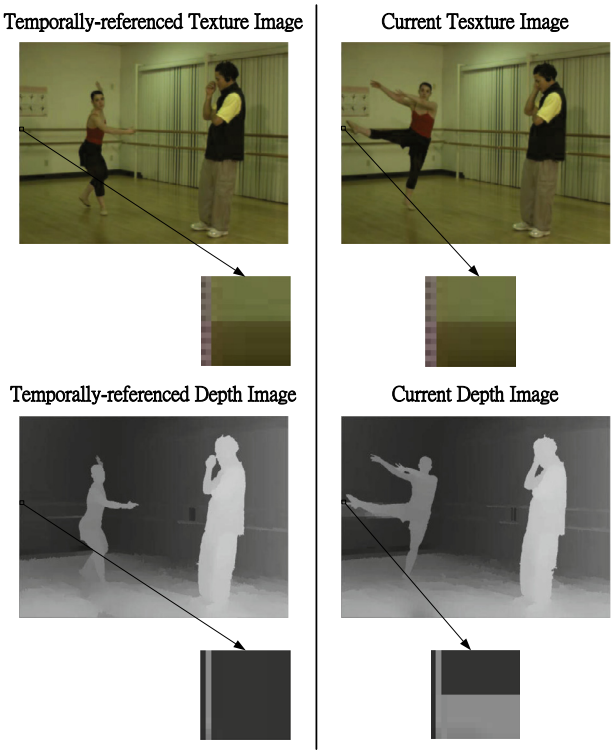

Fig. 2. An example of the wrong skip judgement for the current depth block by Lee et al.'s scheme.

Inspired by the above observations, the proposed texture image-assisted inter prediction (TIAIP) strategy is designed to utilize the temporal correlations between successive texture images and between the associated depth images to determine whether the current depth block can directly copy from its co-located block or not, e.g., inter prediction. Since the inter prediction is performed in the P and B frames when adopting the hierarchical B picture coding structure, how to encode the depth blocks in the P and B frames by the proposed TIAIP strategy is presented in the following paragraphs.

For both left and right views, when the current depth image to be encoded, denoted as D_c, is a P-frame, there exists a previously encoded one, denoted as $\widehat{D}_r$, to act as a reference frame. Meanwhile, the corresponding texture images of D_c and D_r have already been encoded, denoted as $\widehat{T}_c$ and $\widehat{T}_r$, before encoding D_c. For each 16×16 depth block in D_c, the proposed TIAIP strategy determines whether to directly copy the co-located block in $\widehat{D}_r$ according to the temporal correlations between $\widehat{T}_r$ and $\widehat{T}_c$ and between $\widehat{D}_r$ and the partially reconstructed $\widehat{D}_c$. The temporal correlation is measured by computing the mean square error (MSE) between two 16×16 blocks that are spatially or temporally associated with the current depth block to be encoded.

As shown in Fig. 3(a), for the current depth block, B_c, to be encoded, the co-located blocks in $\widehat{T}_r$ and $\widehat{T}_c$ are used to compute MSE_T, and its upper block in the partially reconstructed $\widehat{D}_c$ and the co-located upper block in $\widehat{D}_r$ are used to compute MSE_{DU}, while the counterpart of its left block yields MSE_{DL}. Since a smaller MSE indicates a higher temporal correlation, when all three MSE values are smaller than the pre-defined thresholds, the temporal correlation is strong and the current depth block utilizes the co-located block in the $\widehat{D}_r$ as its inter prediction in the proposed TIAIP strategy.

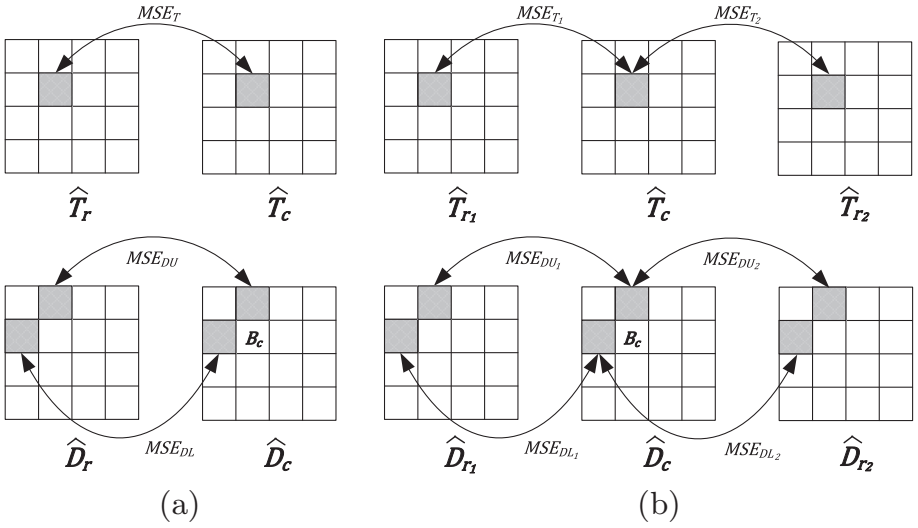

Fig. 3. The utilization of temporal correlations for coding the current depth block: (a) P-frame and (b) B-frame in the proposed TIAIP strategy.

Note that setting proper thresholds for the three MSE measures is crucial to the performance of the proposed TIAIP strategy. For the TIAIP strategy to perform best, experiments show that the optimal thresholds for MSE_T, MSE_{DU}, and MSE_{DL} increase linearly with QP values adopted in coding T_c and D_c, respectively. This is expected since larger QPs usually result in larger codec errors, e.g., larger MSE. In addition, it is found that the optimal thresholds for MSE_{DU} and MSE_{DL} are almost the same. Therefore, setting the three pre-defined thresholds can be formulated as follows:

$$Th_{MSE_T} = \alpha_T + \beta_T \times QP_T, \tag{12}$$

$$Th_{MSE_{DU}} = Th_{MSE_{DL}} = \alpha_D + \beta_D \times QP_D, \tag{13}$$

where QP_T and QP_D are the QP values used in coding T_c and D_c, respectively, and the parameters $(\alpha_T, \beta_T, \alpha_D, \beta_D)$ are set to $(-0.25, 26.25, -4, 277.33)$ empirically for helping the proposed TIAIP strategy to yield good performance.

To encode one depth image, D_c, as a B-frame, it exploits bi-directional temporal correlations with reference to frames $\widehat{D}_{r_1}$ and $\widehat{D}_{r_2}$ to yield two MSE measures. $\widehat{T}_{r_1}$ and $\widehat{T}_{r_2}$ are the counterparts for the encoded texture image $\widehat{T}_c$. Fig. 3(b) demonstrates how to measure temporal correlations in the proposed TIAIP strategy when encoding a B-frame depth image, D_c, where B_c is the current depth block to be encoded and $\{MSE_{T_i}, MSE_{DU_i}, MSE_{DL_i}\}_{i=1,2}$, are the two sets of MSE values computed from the blocks in the current texture and depth images and the corresponding reference frames. In the proposed TIAIP strategy, the two sets of MSE values are compared with the corresponding threshold set, i.e., $\{Th_{MSE_T}, Th_{MSE_{DU}}, Th_{MSE_{DL}}\}$, to determine whether the current depth block can be predicted by direct copy from the co-located block in $\widehat{D}_{r_1}$ or $\widehat{D}_{r_2}$. If there exists only one set of triple MSE values, i.e, $\{MSE_{T_i}, MSE_{DU_i}, MSE_{DL_i}\}$, less than the corresponding thresholds in $\{Th_{MSE_T}, Th_{MSE_{DU}}, Th_{MSE_{DL}}\}$, the proposed TIAIP strategy selects its corresponding co-located reference block as the inter prediction for the current depth block. When both sets of triple MSE values are less than the thresholds, an MSE ratio, Δ_R, is defined by

$$\Delta_R = \frac{MSE_{T_1}}{MSE_{T_2}} \times \frac{MSE_{DU_1}}{MSE_{DU_2}} \times \frac{MSE_{DL_1}}{MSE_{DL_2}}. \tag{14}$$

When $\Delta_R < 1$, the proposed TIAIP strategy selects the co-located block from the reconstructed $\widehat{D}_{r_1}$ for inter prediction or from $\widehat{D}_{r_2}$ when $\Delta_R \geqslant 1$. The complete flowcharts of the proposed TIAIP strategy for encoding the P-frame depth image and the B-frame depth image are shown in Fig. 4(a) and (b), respectively.

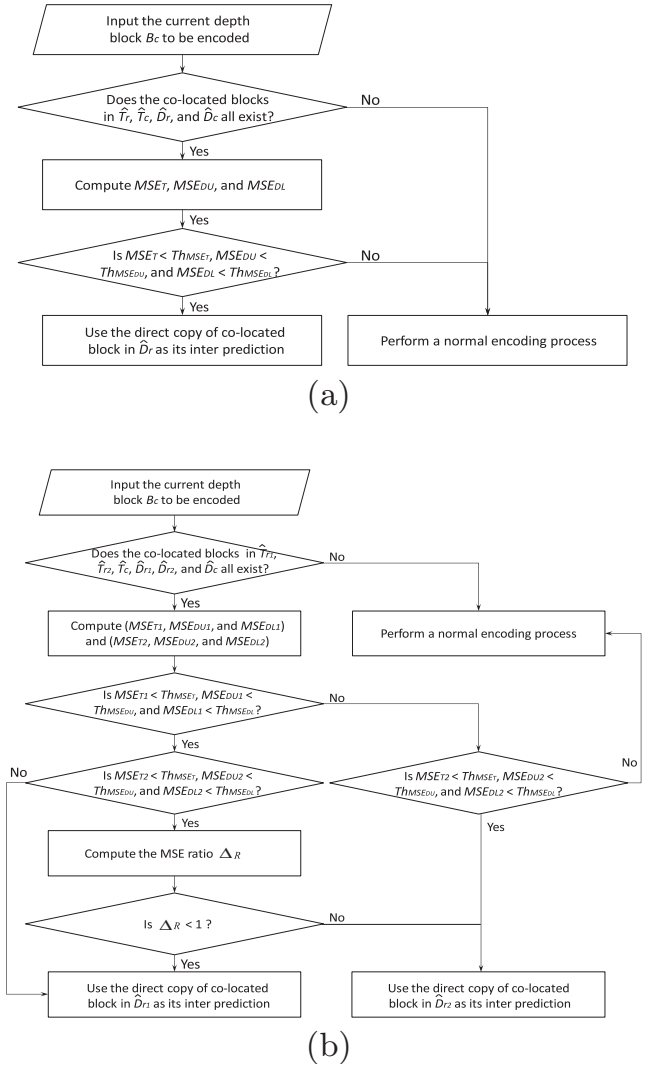

Fig. 4. Flowcharts for coding depth images: (a) P-frame and (b) B-frame in the proposed TIAIP strategy.

In the practical implementation, for the skipped depth blocks by the proposed TIAIP strategy, at the encoder side, we still set their coding modes to be the traditional SKIP mode and a small amount of the bitrate of the compressed 3D-MVD is used to store the parameter information associated with Th_{MSE_T} and $Th_{MSE_{DU}}$. Then, at the decoder side, for the depth blocks labeled as the SKIP mode, based on the thresholds Th_{MSE_T} and $Th_{MSE_{DU}}$ computed from the transmitted parameters, we re-perform the proposed TIAIP strategy on each of them so as to judge whether the current depth block uses the proposed coding mode or not. It is worth noting that in addition to speeding up the AMVDC encoding process to a large extent, implementing the block skipping in the proposed TIAIP strategy by the above manner can also save bitrates which can be further re-allocated to unencoded depth blocks in the edge regions for enhancing compression efficiency.

3.2. The proposed adaptive weighted mode in-loop filter strategy

Due to different pixel variation characteristics, the deblocking filter in H.264/AVC designed for texture images may not be suitable for dealing with the depth images. After presenting the proposed TIAIP strategy, the newly proposed adaptive weighted mode in-loop filter (AWMIF) strategy will be presented to smooth or sharpen the depth images according to content variation while suppressing the compression artifacts by utilizing local image features and joint correlations between the texture and depth images.

D_c and T_c have been defined to denote the current depth image to be filtered and its corresponding texture image, and $d(p)$ and $f(p)$ denote, respectively, the depth and intensity values for a pixel p in D_c and T_c. A localized histogram $H(p,d)$ of pixel p for all depth values, $d \in [0,255]$, is first established by referencing a set of p's neighboring pixels, $\mathbb{W}(p)$. The depth value that yields the maximum $H(p,d)$ is the filtered depth value for pixel p, $\hat{d}(p)$; that is,

$$H(p,d) = \sum_{q \in \mathbb{W}(p)} w(p,q) \cdot G_r(d - d(q))$$
$$\hat{d}(p) = \arg \max_{0 \leqslant d \leqslant 255} H(p,d), \tag{15}$$

where $G_r(\cdot)$ is a zero mean Gaussian function with standard deviation σ_r and $w(p,q)$ is a weighting function defined as follows:

$$w(p,q) = G_s(p-q) \cdot G_t(f(p) - f(q) + \zeta(p)) \cdot G_d(d(p) - d(q) + \epsilon(p)), \tag{16}$$

where $G_s(\cdot), G_t(\cdot)$, and $G_d(\cdot)$ are zero mean Gaussian functions with standard deviations, σ_s, σ_t, and σ_d, respectively, and are referred to as the domain filter, range filter and depth filter in our work, respectively.

The parameters of the proposed AWMIF strategy comprise four standard deviation values, $\{\sigma_r, \sigma_s, \sigma_t, \sigma_d\}$, and two offsets (ζ, ϵ). The value of σ_s is set to 1 when the window size of $\mathbb{W}(p)$ is 7×7 [27]. When both ζ and ϵ are zero, the proposed AWMIF is the conventional weighted mode filter (WMF) [16,19], indicating that the conventional WMF is one special case of the proposed AWMIF. The main target of the proposed AWMIF strategy is to refine the quality of the depth image by adaptively adjusting these parameters excluding σ_s. The determination of ζ and ϵ will be explained in what follows.

Fig. 5(a) and (b) show the texture and depth images from the test *Ballet* video. For demonstration, three local image blocks of Fig. 5(a) and (b) are enlarged and shown in Fig. 5(c), (e), and (g) as well as Fig. 5(d), (f), and (h), respectively. In general, when both texture and depth variations are similar, as shown in Fig. 5(c) and (d) as well as Fig. 5(e) and (f), the texture information can help to enhance the corresponding depth information. On the contrary, when the texture and depth variations are not coherent, as shown in Fig. 5(g) and (h), the texture information may not benefit the enhancement of the depth image.

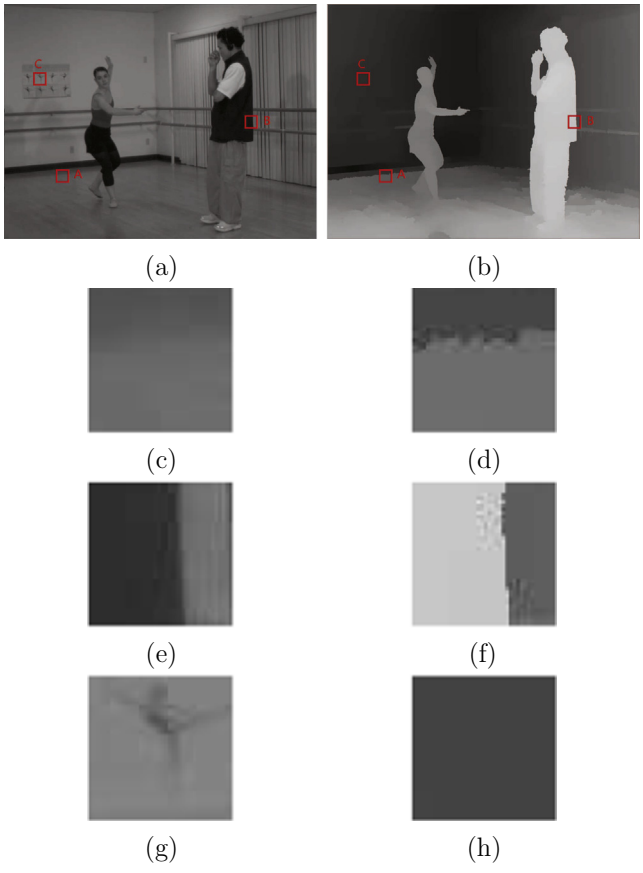

Fig. 5. Snapshots from the texture and depth images of the test *Ballet* video: (a) the texture image with luma component; (b) the depth image; (c)–(d), (e)–(f), and (g)–(h) enlarged image blocks of (A, B, C) in (a) and (b), respectively.

Based on the above observation, we adapt the control parameters of the proposed AWMIF to the local similarity between the texture and depth images for providing different degrees of quality enhancement for the depth image. To measure the local similarity between the texture and depth images while avoiding suffering from the affection of the compression noise, the proposed AWMIF performs a pixelwise Laplacian of Gaussian (LoG) operator with a 7×7 window and $\sigma_{LoG} = 1$ on both the texture and depth images to yield their edge maps with strength values, respectively. Let LoG_T and LoG_D denote, respectively, the edge maps with the strength values of the texture and depth images. Then, for each pixel p in the depth image, by comparing the edge strength values between its neighboring pixels and their co-located pixels in the texture image, the local similarity measurement for p can be represented as

$$\rho_{LoG}(p) = \max\left\{\frac{1}{|N(p)|}\sum_{q \in N(p)} exp\left\{-\frac{1}{2}\left[\frac{LoG_T(q) - LoG_D(q)}{\sigma_\rho}\right]^2\right\}, \times \frac{1}{|N(p)|}\sum_{q \in N(p)} exp\left\{-\frac{1}{2}\left[\frac{-LoG_T(q) - LoG_D(q)}{\sigma_\rho}\right]^2\right\}\right\}, \tag{17}$$

where $N(p)$ is the set of neighboring pixels inside a 7×7 window centered at p, $|N(p)|$ is the cardinality of $N(p)$, and σ_ρ is set to 0.35 empirically. Note that in Eq. (17), the latter term is necessary since if only the former term is used, similar local patches in the texture and depth images such as Fig. 5(e) and (f) would be judged as not being similar due to the sign of the LoG output.

Fig. 6. Correlation map of the first frame in the test *Ballet* video.

The correlation map of the first frame in the test *Ballet* video obtained through Eq. (17) is shown in Fig. 6, in which darker pixels denote less texture-depth image correlation at those positions. To reflect whether the texture and depth images match well with each other on the effect of the range and depth filters when enhancing each pixel p in the depth image, we thus adjust σ_t and σ_d according to $\rho_{LoG}(p)$ as follows:

$$\sigma_t = e \times \sigma_{t,Blk} \times (1 - \rho_{LoG}(p)), \tag{18}$$

$$\sigma_d = f \times \sigma_{d,Blk}, \tag{19}$$

Table 1
The two test 3D-MVDs information.

Sequence	Resolution	GOP	Views to be encoded (left view, right view)	View to be synthesized
Ballet	1024 × 768	15	(4, 6)	5
Breakdancers	1024 × 768	15	(4, 6)	5

Table 2
The five coding schemes in our experiments.

Scheme	Description
I	Traditional MVD-based video coding scheme with a fixed 5:1 ratio of bitrate allocation for texture and depth videos
II	The AMVDC scheme proposed by Shao et al.
III	The AMVDC scheme improved by the proposed TAIPS strategy
IV	The AMVDC scheme improved by the proposed AWMIFS strategy
V	The AMVDC scheme improved by the proposed TAIPS and AWMIFS strategies

where $\sigma_{t,Blk}$ and $\sigma_{d,Blk}$ represent, respectively, the pixel value variations over a 7×7 window centered at p for the texture and depth images, and parameters e and f are set to 10 and 1 empirically. In short, when adjusting σ_t and σ_d, the value of σ_t would be linearly proportional to $1 - \rho_{LoG}(p)$ and considers the local pixel value variation $\sigma_{t,Blk}$; the value of σ_d considers the local pixel value variation $\sigma_{d,Blk}$.

Since adjusting ζ and ϵ are basically the same, only the former is described. For one pixel p in the depth image, to determine whether to perform sharpening or not for the range filter, the absolute value of its LoG output, $\|LoG_T(p)\|$, of the texture image is compared with a threshold defined as

$$Th_t = \mathrm{E}[|LoG_T(p)|] + 0.5 \times \sqrt{\mathrm{Var}[|LoG_T(p)|]}, \tag{20}$$

where $\mathrm{E}[|LoG_T(p)|]$ and $\mathrm{Var}[|LoG_T(p)|]$ are the mean and variance of the absolute values of the LoG output over the whole texture image, respectively. When $|LoG_T(p)| > Th_t$, it indicates that p is likely to be an edge pixel in the texture image and a pixel-adaptive difference, $\triangle_{f(p)}$, is calculated as follows for setting ζ

$$\triangle_{f(p)} = \begin{cases} \mathrm{E}_H(p) - f(p), & \text{if } LoG_T(p) > 0 \\ f(p) - \mathrm{E}_L(p), & \text{if } LoG_T(p) < 0 \\ 0, & \text{otherwise,} \end{cases} \tag{21}$$

where

$$\mathrm{E}_H(p) = \frac{\sum_{q \in N(p), f(q) \geqslant f(p)} f(q)}{\sum_{q \in N(p), f(q) \geqslant f(p)} 1}, \tag{22}$$

$$\mathrm{E}_L(p) = \frac{\sum_{q \in N(p), f(q) \leqslant f(p)} f(q)}{\sum_{q \in N(p), f(q) \leqslant f(p)} 1}. \tag{23}$$

Since the value of ζ depends on $\triangle_{f(p)}$ and should be constrained by the local similarity, we set it as

$$\zeta(p) = g \times \triangle_{f(p)} \times (1 - \rho_{LoG}(p)), \tag{24}$$

where the parameter g is set to 0.29 from the experiments. Note that, slightly different from Eq. (24), the setting for offset ϵ in the depth filter is only controlled by the associated pixel-adaptive difference $\triangle_{d(p)}$, and the parameter g is also set to 0.29.

As for setting the value of σ_r, works in [16] showed that: (1) when $\sigma_r \to \infty$, the conventional weighted mode filtering acts as a smoothing filter, cf. the joint bilateral filtering [6] and (2) σ_r should be proportional to the amount of noise in the depth image to be filtered. Since the amount of noise increases with QP, we propose to take the QP value into consideration when setting the value of σ_r for the proposed AWMIF. Furthermore, by considering the pixel value variation over the whole depth image for adjusting the σ_r, the final σ_r setting is formulated as

$$\sigma_r = h \times QP_d + i \times \mathrm{Var}(d(p)) + j, \tag{25}$$

where QP_d is the QP adopted for the depth image to be filtered, $\mathrm{Var}(d(p))$ is the variance of pixel values over the whole depth image,

Table 3
Bitrate allocations (in kbps) of the texture and depth videos for the traditional Scheme I, Scheme II by Shao et al., and the proposed Schemes III–V.

3D-MVD	Target bitrate	Scheme I		Scheme II		Scheme III		Scheme IV		Scheme V	
		Texture	Depth	Texture	Depth	Texture	Depth	Texture	Depth	Texture	Depth
Ballet	2400	2136.0	846.3	1710.3	789.2	1877.7	620.6	1665.7	827.9	1876.1	621.6
	5200	4716.6	1208.9	4053.9	1360.9	4154.7	1280.8	4048.8	1359.2	4154.9	1279.6
	8400	7716.4	1724.6	6626.9	2173.1	6705.8	2068.0	6592.9	2176.9	6706.7	2064.0
	12,000	11049.4	2359.6	9585.5	3043.3	9614.5	2912.2	9586.7	3043.1	9617.2	2905.8
Breakdancers	2400	2121.9	599.8	1863.1	620.0	1905.6	582.8	1846.5	636.4	1903.8	582.5
	5200	4665.7	991.1	4173.9	1264.4	4183.1	1253.6	4171.7	1263.9	4182.6	1252.6
	8400	7636.5	1523.7	6748.7	2076.8	6748.4	2030.9	6748.9	2076.9	6748.7	2027.7
	12,000	10923.0	2154.4	9586.4	3062.7	9580.5	2958.6	9586.8	3062.4	9580.4	2954.4

and the parameters (h,i,j) are set to $(0.84, 0.0043, -24)$ from the experiments.

For the conventional WMF, the Gaussian functions adopted by the range and depth filters are, respectively, located at the current pixel values, $f(p)$ and $d(p)$, and their functional values decrease as the pixel values fall farther away from the current pixel values. The proposed AWMIF, by adding a pixel-adaptive offset ζ to the range filter and a ϵ to the depth filter, can improve the conventional WMF to be a more powerful one capable of sharpening the edges in a depth image. When compared with the H.264/AVC deblocking filter, the proposed AWMIF can smooth or sharpen the depth image adaptively according to the individual characteristics of the texture and depth images together with their correlation, which can suppress the compression artifacts and improve the depth image quality. The proposed AWMIF can further reduce the coded bitrate for the next depth image because a better reconstructed current depth image frame also acts as a better reference frame for the next one. As expected, the proposed AWMIF helps to improve the depth image compression efficiency in AMVDC and the quality of the synthesized view images.

3.3. Remarks on experimental parameters

Since the parameters in both the proposed TIAIP and AWMIF strategies play a crucial role in whether to deliver a robust compression performance or not, these parameters need to be carefully determined. In what follows, how to determine these parameters in our experiments will be described.

In the proposed TIAIP strategy, we need to determine the values of parameters (α_T, β_T) and (α_D, β_D) for Th_{MSE_T} and $Th_{MSE_{DU}}$, respectively. To this end, we use the following procedure to encode the first two GOPs of the concerned 3D-MVD with QP $= 5, 10, \ldots, 50$. Under each considered QP circumstance, for each 16×16 depth block to be encoded in the P-frame and B-frame, we first check whether directly inheriting its co-located reconstructed reference depth block as its inter prediction incurs a synthesis distortion or not. If no synthesis distortion occurs, the current depth block skips the normal encoding process and the values of the associated MSE_T, MSE_{DU}, and MSE_{DL} are collected; otherwise, the normal encoding process is performed on the current depth block. Next, for each considered QP circumstance, by averaging the collected MSE_T values, the empirical threshold Th_{MSE_T} is delivered. Similar operation is also applied to the collected MSE_{DU} and MSE_{DL} values so as to compute the empirical threshold $Th_{MSE_{DU}}$ for each considered QP circumstance. Once the empirical thresholds Th_{MSE_T} and $Th_{MSE_{DU}}$ at all the considered QPs are obtained, we approximate the values by the polynomial functions as Eqs. (12) and (13), respectively. As a result, the values of parameters (α_T, β_T) and (α_D, β_D) in Eqs. (12) and (13) can be delivered.

As for the proposed AWMIF strategy, since it involves more parameters, we adopt a sequential way to determine these parameters. When finishing the above encoding procedure for determining the parameters required in the proposed TIAIP strategy, we have the compression result of the first two GOPs of the concerned 3D-MVD for each considered QP. Under all the

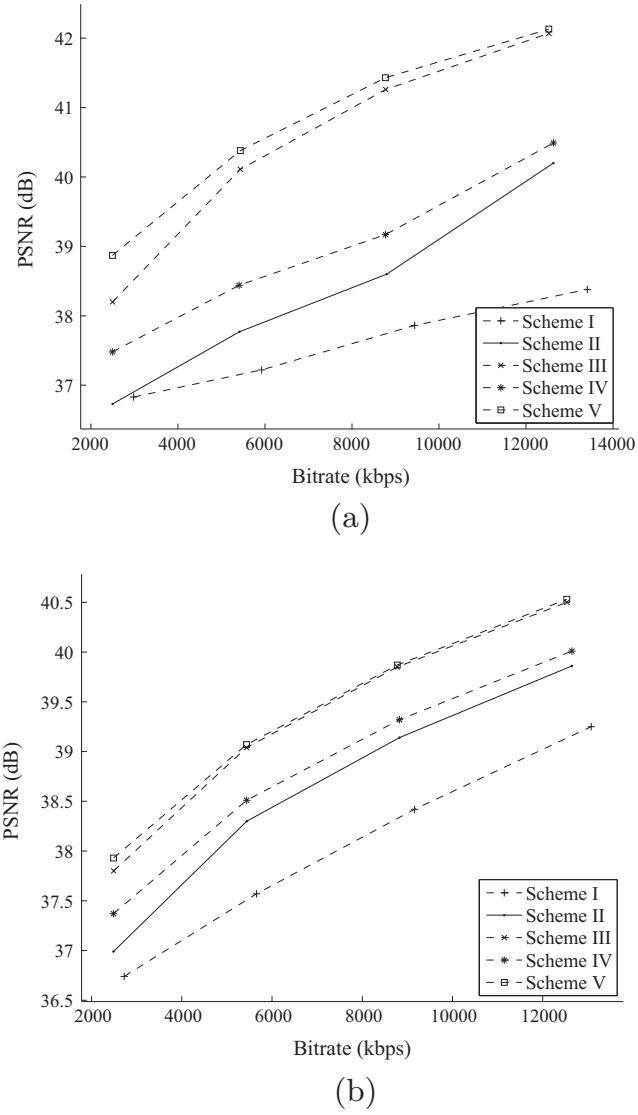

Fig. 7. RD performance comparison for the two test 3D-MVDs: (a) *Ballet* and (b) *Breakdancers*.

Table 4
Quality in terms of PSNRs of the reconstructed depth and synthesized view videos for the traditional Scheme I, Scheme II by Shao et al., and the proposed Schemes III–V.

3D-MVD	Target bitrate (kbps)	Reconstructed depth video (dB) Scheme					Synthesized view video (dB) Scheme				
		I	II	III	IV	V	I	II	III	IV	V
Ballet	2400	38.37	38.26	43.04	39.78	44.00	36.83	36.73	38.20	37.48	38.87
	5200	39.64	41.81	50.99	43.58	51.86	37.22	37.77	40.11	38.44	40.38
	8400	41.57	46.37	56.04	47.50	55.65	37.86	38.60	41.26	39.17	41.43
	12,000	42.86	51.94	59.80	52.23	57.43	38.38	40.20	42.07	40.49	42.13
Breakdancers	2400	39.27	40.04	43.14	40.41	43.31	36.74	36.99	37.80	37.37	37.93
	5200	41.40	43.71	47.97	44.20	48.02	37.57	38.30	39.04	38.51	39.07
	8400	43.42	47.22	51.63	47.35	51.17	38.42	39.14	39.85	39.32	39.87
	12,000	46.51	49.85	54.69	49.64	53.32	39.25	39.86	40.50	40.01	40.53
Average		41.63	44.90	50.91	45.59	50.60	37.78	38.45	39.85	38.85	40.03

considered QP circumstances, for each pixel in the image frames of the first two GOPs of the reconstructed depth videos, according to its and neighboring pixels' LoG responses, we first classify it into either the set that contains the pixels in the smooth regions and the pixels at the midpoint of the edge or the set that contains the pixels above and below the midpoint of the edge. For convenience, the former set is denoted as S_1 and the latter set is denoted as S_2. Since the pixels in S_1 do not need to be sharpened, which means that their ζ and ϵ values must be zero, this fact enables us to determine the parameters σ_ρ, e, f, h, i, and j first. The determination for the six parameters can be formulated as the following minimization problem:

$$\{\sigma_\rho^*, e^*, f^*, h^*, i^*, j^*\} = \arg \min_{\sigma_\rho, e, f, h, i, j} \sum_{p \in S_1} \|\hat{d}(p) - d(p)\|^2, \tag{26}$$

where $d(p)$ is the depth value of pixel p in the original depth video. In considerations of the suggestion for adjusting the standard deviation of the Gaussian function in [27,19] and the magnitude of each factor in Eqs. 17,18,19 and (25), we perform an exhaustive search in the parameter space, where $0.1 \leqslant \sigma_\rho \leqslant 0.5$ with step size 0.05, $5 \leqslant e \leqslant 20$ with step size 1, $0.5 \leqslant f \leqslant 1.5$ with step size 0.1, $0.5 \leqslant h \leqslant 1$ with step size 0.05, $0.002 \leqslant i \leqslant 0.005$ with step size 0.0001, and $-30 \leqslant j \leqslant 30$ with step size 2, so as to obtain the parameters which minimize the above sum of square errors. After obtaining the values of parameters σ_ρ, e, f, h, i, and j, the determination for the two parameters in the sharpening functions ζ and ϵ follows. Given the values of parameters σ_ρ, e, f, h, i, and j, by using the pixels in S_2 as the observation samples, the determination for the two parameters in ζ and ϵ can also be formulated as a minimization problem like Eq. (26). Through an exhaustive search over the interval [0.15, 0.45] with step size 0.02 for the two parameters, the final two parameters required in the proposed AWMIF strategy can be obtained.

Note that the above-mentioned parameter determination process may not yield the globally optimal values for the parameters required in the proposed scheme, but it can effectively reduce the computational complexity of the parameter determination while delivering the satisfactory results. In addition, the

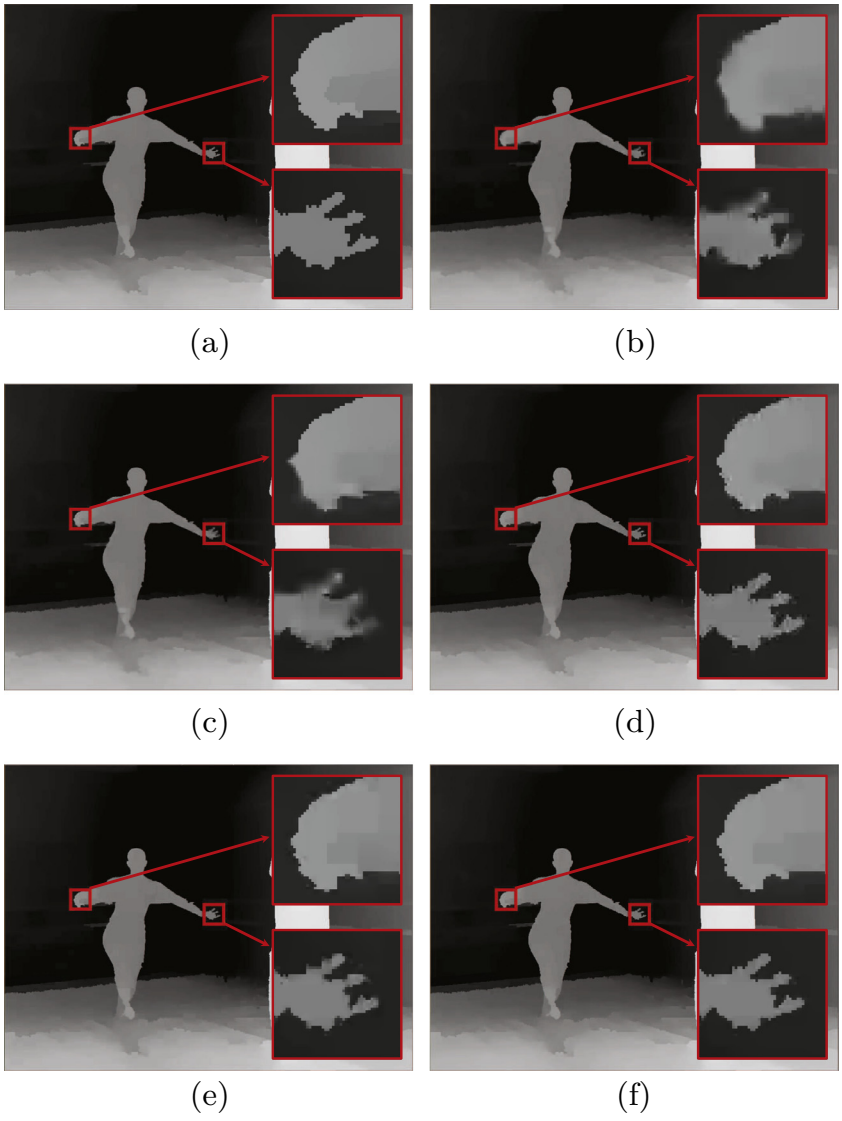

Fig. 8. Magnified image sub-regions from the reconstructed depth videos of *Ballet* for subjective quality comparison: (a) original depth image; (b) reconstructed depth image using the traditional Scheme I; (c) reconstructed depth image using Scheme II by Shao et al.; (d) reconstructed depth image using the proposed Scheme III; (e) reconstructed depth image using the proposed Scheme IV; and (f) reconstructed depth image using the proposed Scheme V.

experiments show that the two considered 3D-MVDs, *Ballet* and *Breakdancers*, have similar determined values for these parameters, so their average value for each parameter is taken as the experimental parameter setting of the proposed scheme.

4. Experimental results

To demonstrate the low computational cost and quality-efficient merits of the proposed depth video coding scheme, several experiments have been conducted with the reference software JMVM 7.0 [29] on the two test 3D-MVDs, *Ballet* and *Breakdancers*, provided by Microsoft Research [33]. In addition to being used to display 3D video, the depth images are mainly used for synthesizing virtual view images. The quality of both the reconstructed depth images and the synthesized view images, rendered using the reconstructed texture and depth images, is evaluated by PSNR. In addition to PSNR performance, the encoding time complexity is also used for compression performance measurement. For each test 3D-MVD, we select two different views as the left and right views to encode their first 100 frames. The synthesized view video rendered from the two views via the view synthesis reference software (VSRS) 3.5 [34] is used for quality evaluation. The experiments adopt the typical hierarchical B picture prediction structure. The test 3D-MVD information comprising name, resolution, GOP size, and the number of views to be encoded and synthesized is listed in Table 1.

Since the proposed scheme is designed to improve the depth video coding of the AMVDC by Shao et al. comparisons with the AMVDC scheme are required. The five concerned 3D-MVD coding schemes listed in Table 2, which comprise the traditional scheme with a fixed 5:1 ratio of bitrate allocation, the AMVDC scheme, and three variants of the proposed scheme for texture and depth videos, are carried out in our experiments for performance evaluation. For objective evaluation, the bitrate constraints are set to 2400, 5200, 8400, and 12,000 kbps when encoding the two test 3D-MVDs. All these coding schemes and the compression system JMVM 7.0 are realized by Visual C++ 2008, and the hardware platform is an IBM compatible computer with Intel i7-3370 CPU 3.4 GHz and 16 GB RAM under the Microsoft Windows 7 64-bit operating system.

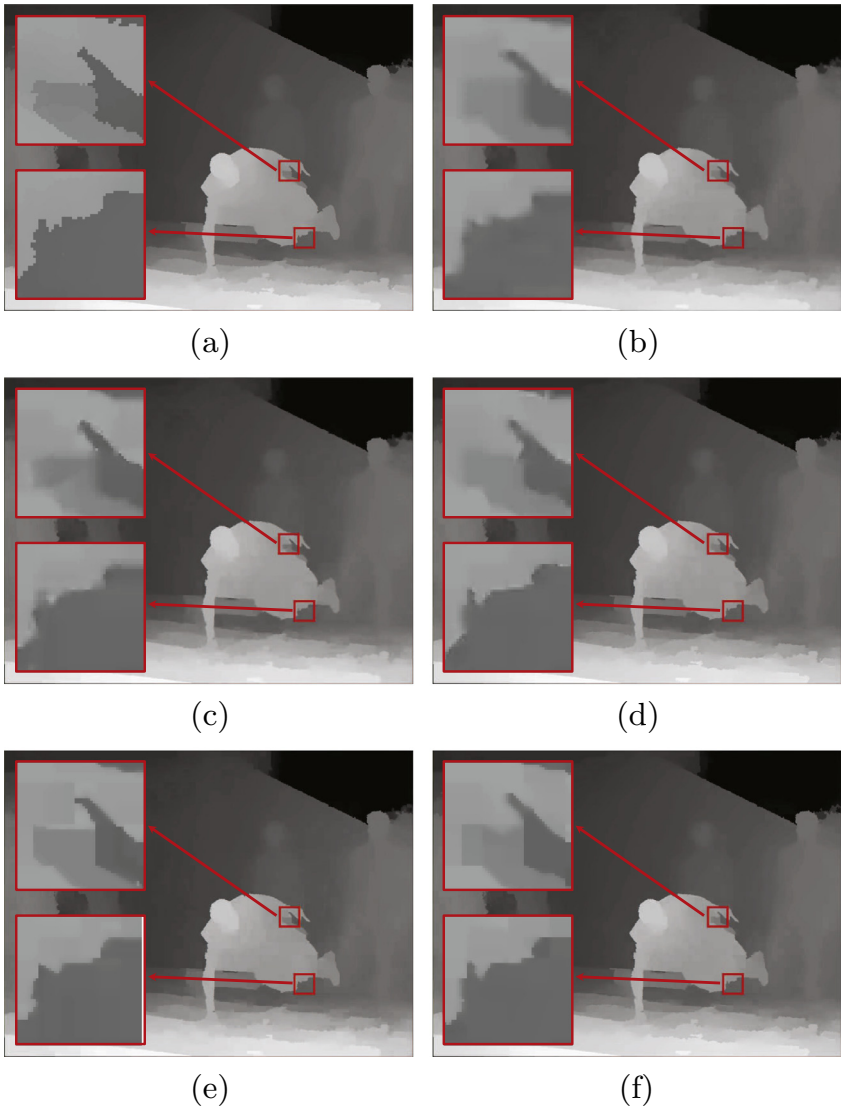

Fig. 9. Magnified image sub-regions from the reconstructed depth videos of *Breakdancers* for subjective quality comparison: (a) original depth image; (b) reconstructed depth image using the traditional Scheme I; (c) reconstructed depth image using Scheme II by Shao et al.; (d) reconstructed depth image using the proposed Scheme III; (e) reconstructed depth image using the proposed Scheme IV; and (f) reconstructed depth image using the proposed Scheme V.

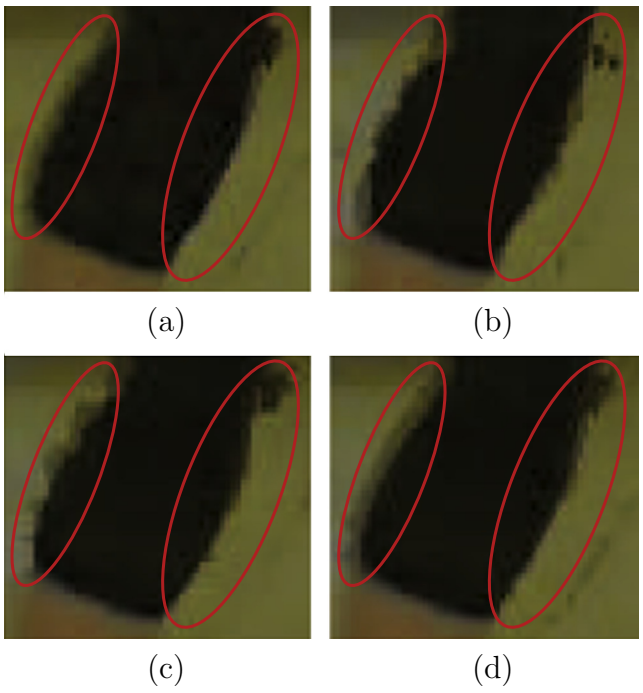

Fig. 10. Magnified image sub-regions from the synthesized view videos of *Ballet* for subjective quality comparison: (a) original synthesized image; (b) synthesized image using the traditional Scheme I; (c) synthesized image using Scheme II by Shao et al.; and (d) synthesized image using the proposed Scheme V.

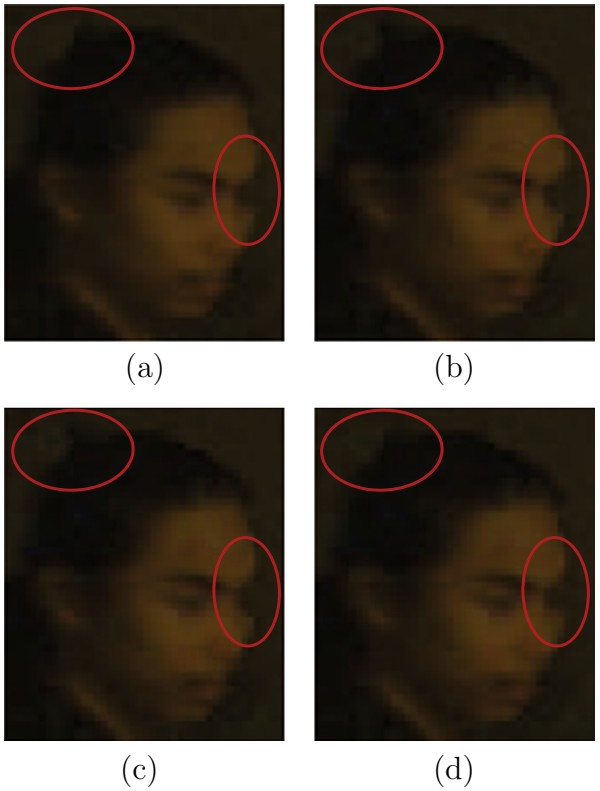

Fig. 11. Magnified image sub-regions from the synthesized view videos of *Breakdancers* for subjective quality comparison: (a) original synthesized image; (b) synthesized image using the traditional Scheme I; (c) synthesized image using Scheme II by Shao et al.; and (d) synthesized image using the proposed Scheme V.

4.1. Compression and quality performance comparison evaluation

The resultant bitrate allocations for texture and depth videos and the PSNRs for reconstructed depth and synthesized view videos using the five concerned 3D-MVD coding schemes are listed in Tables 3 and 4, respectively. The PSNR of the synthesized view video rendered from the reconstructed texture and depth videos is a weighted average with weights (0.8, 0.1, 0.1) for YUV channels, and is computed with respect to that rendered using the original ones. The PSNR of the reconstructed depth videos is the average PSNR of the left-view and right-view reconstructed depth videos. Fig. 7 shows the rate-distortion (RD) plots of the five coding schemes, which demonstrates the tradeoff between the quality of the synthesized view and the bitrate, for the two test 3D-MVDs.

The results in Table 4 show that the proposed Schemes III–V, always achieve better PSNR performance than the other two. For Scheme I, as the fixed bitrate allocation ratio of 5:1 for the texture and depth videos is set without considering the different image characteristics of the texture and depth videos, the quality of the reconstructed depth and synthesized view videos is limited as expected. Scheme II outperforms Scheme I in terms of PSNR performance, while the proposed three schemes outperform Scheme II. Although Scheme II allocates the bitrate saved from the unencoded

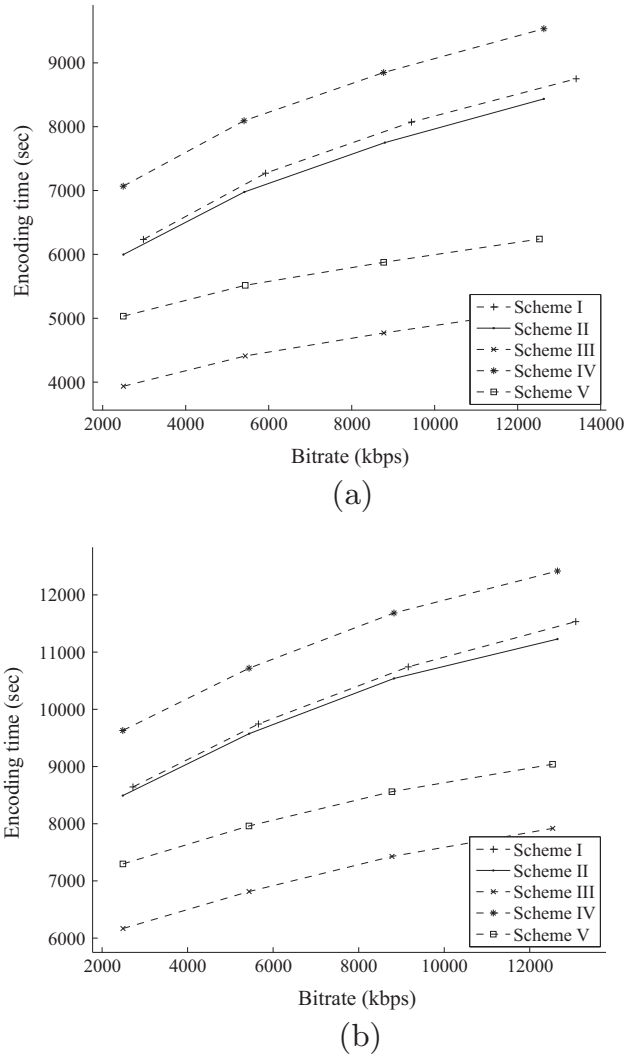

Fig. 12. Performance comparison on bitrate vs. encoding time for the two test 3D-MVDs: (a) *Ballet* and (b) *Breakdancers*.

Table 5
Computational complexity comparison of Schemes I–V in terms of the encoding time.

3D-MVD	Target bitrate (kbps)	Actual encoding time (DPET)				
		Scheme I	Scheme II	Scheme III	Scheme IV	Scheme V
Ballet	2400	6234.75 (0%)	5997.98 (−3.80%)	3935.29 (−36.88%)	7068.50 (13.37%)	5032.04 (−19.29%)
	5200	7271.95 (0%)	6977.35 (−4.05%)	4410.08 (−39.35%)	8092.51 (11.28%)	5515.50 (−24.15%)
	8400	8071.92 (0%)	7751.54 (−3.97%)	4769.60 (−40.91%)	8847.84 (9.61%)	5876.40 (−27.20%)
	12,000	8750.93 (0%)	8435.78 (−3.60%)	5127.21 (−41.41%)	9533.85 (8.95%)	6240.38 (−28.69%)
Breakdancers	2400	8642.54 (0%)	8492.33 (−1.74%)	6168.17 (−28.63%)	9630.00 (11.43%)	7297.41 (−15.56%)
	5200	9742.75 (0%)	9574.35 (−1.73%)	6815.04 (−30.05%)	10719.12 (10.02%)	7960.49 (−18.29%)
	8400	10741.71 (0%)	10539.32 (−1.88%)	7428.30 (−30.85%)	11682.93 (8.76%)	8560.05 (−20.31%)
	12,000	11532.13 (0%)	11228.31 (−2.63%)	7919.35 (−31.33%)	12417.22 (7.68%)	9039.28 (−21.62%)
Average		8873.58 (0%)	8624.62 (−2.81%)	5821.63 (−34.39%)	9748.99 (9.87%)	6940.19 (−21.79%)

right-view chroma component to encode the depth video, it does not utilize the special characteristics of the depth videos or the correlation between the texture and depth videos in the same view to improve the coding performance.

In the proposed Scheme III, although the reconstructed depth videos may suffer from slight quality degradation since the proposed TIAIP strategy directly inherits the co-located reference depth blocks without the normal encoding process, the TIAIP strategy reallocates the saved bitrate from the skipped depth blocks to enhance the compression of the subsequent unencoded blocks in the color and depth images, which helps to yield high-quality reconstructed depth and synthesized view videos. The proposed Scheme IV outperforms Scheme II by Shao et al. in terms of quality of both the reconstructed depth and synthesized view videos by 0.69 dB and 0.4 dB on average, respectively, as the proposed AWMIF strategy effectively suppresses the artifacts, and the refined reconstructed depth video can provide more accurate pixel correspondence for view synthesis. By utilizing both the proposed TIAIP and AWMIF strategies, the proposed Scheme V achieves the best PSNR performance, which outperforms the existing Schemes I and II in terms of the quality of the synthesized view video by 2.25 dB and 1.58 dB on average, respectively.

In addition to the objective performance evaluation, subjective evaluation is also carried out to demonstrate the superiority of the proposed coding schemes in terms of improving the perception quality. Figs. 8 and 9 demonstrate the enlarged subimages from the original depth videos and the reconstructed depth videos by the five coding schemes on the test 3D-MVDs. In Schemes I–III and IV–V, the H.264/AVC deblocking filter and the proposed AWMIF are applied, respectively. As shown, the images coded by the proposed AWMIF demonstrate fewer compression artifacts and blurring effects along the object boundaries, as compared with those adopting the H.264/AVC deblocking filter. This observation justifies the capability of the proposed AWMIF of suppressing compression artifacts and enhancing the synthesized image quality. Figs. 10 and 11 show the enlarged synthesized subimages rendered from the original and reconstructed texture and depth videos by the previous Schemes I, II, and the proposed V on the two test 3D-MVDs. As shown, the proposed Scheme V can yield images with fewer visual artifacts and better perception quality, as compared with those using the existing Schemes I and II.

4.2. Computational complexity analysis

The computational complexity is measured by recording the practical encoding time of the five concerned 3D-MVD coding schemes. Fig. 12 demonstrates the computational complexity performance by plots of bitrate vs. encoding time for the five 3D-MVD coding schemes. In addition, the actual encoding time required in Schemes I–V and the corresponding decreased percentages in the encoding time (DPET) for Schemes II–V are tabulated in Table 5, and are defined as

$$\mathrm{DPET}(i) = \frac{\mathrm{ET}(i) - \mathrm{ET(I)}}{\mathrm{ET(I)}} \times 100\%, \text{for } i = \mathrm{II}, \ldots, \mathrm{V},$$

where $\mathrm{ET}(x), x = \mathrm{I}, \mathrm{II}, \ldots, \mathrm{V}$, represents the encoding time required for Scheme x to encode the test 3D-MVD.

Table 5 shows that the proposed Schemes III and V can reduce the encoding time ranging from 28.63% to 41.41% and 15.56% to 28.69%, respectively, whereas the proposed Scheme IV requires the longest encoding time. Further investigations reveal that the proposed TIAIP strategy adopted in the proposed Schemes III and V directly inherits the co-located reference depth block as the inter prediction result for most depth blocks and then skipping the normal encoding procedures for these blocks helps to largely reduce the encoding time, whereas the proposed AWMIF strategy adopted in the proposed Schemes IV and V requires more operations to adaptively adjust the standard deviations and offsets for each local depth block as compared with the H.264/AVC deblocking filter. Since severe compression artifacts are usually found to reside on object boundaries in the foreground, applying the proposed AWMIF strategy in the proposed Schemes IV and V only to boundary regions may help to speed up the encoding process while reducing the artifact suppression performance by only a little on smooth regions.

In total, the experiments justify the effectiveness of the proposed fast and quality-effective coding scheme for depth videos under various constrained bitrates, which can provide better reconstructed texture and depth videos and synthesized view videos with less computational complexity as compared with the AMVDC scheme by Shao et al. and the traditional 3D-MVD coding scheme.

5. Conclusion

A fast and quality-efficient scheme for AMVDC with bitrate constraints has been presented in this paper. In the proposed scheme, two depth image coding strategies, the texture image-assisted inter prediction (TIAIP) strategy and the adaptive weighted mode in-loop filter (AWMIF) strategy, are delivered to significantly speed up the encoding process while improving the quality of the reconstructed texture and depth videos and of the synthesized view videos. By analyzing the temporal correlations between depth images and their associated texture images, the proposed TIAIP strategy determines whether or not the current depth block can directly copy the co-located reference depth block and skip the routine coding procedures so as to speed up the encoding time. The proposed AWMIF strategy then follows to refine the quality of the reference depth image by measuring the coherence between

the adjacent image blocks of the encoded texture and depth images. The experimental results on two test 3D-MVDs demonstrate that the proposed asymmetric coding scheme for 3D-MVDs can perform better than the state-of-the-art scheme by Shao et al., and the traditional 5:1 bitrate allocation scheme in terms of computational complexity and quality of reconstructed depth and synthesized view videos.

Acknowledgements

The work of C.-H. Lin and K.-L. Chung was supported by the Ministry of Science and Technology of Taiwan, under the Contracts MOST 101-2221-E-011-139-MY3 and MOST 102-2221-E-011-055-MY3. The work of J.-J. Chen was supported by the Ministry of Science and Technology of Taiwan, under the Contract NSC 102-2221-E-011-036.

References

[1] C. Fehn, Depth-image-based rendering (DIBR), compression and transmission for a new approach on 3-D-TV, in: Proc. of SPIE, Stereoscopic Image Processing and Rendering, 2004, pp. 93–104.
[2] C. Fehn, R. de la Barre, S. Pastoor, Interactive 3-DTV-Concepts and key technologies, in: Proc. of the IEEE, 2006, pp. 524–538.
[3] S. Grewatsch, E. Muller, Sharing of motion vectors in 3D video coding, in: Proc. of IEEE International Conference on Image Processing, 2004, pp. 3271–3274.
[4] X.X. Huang, M.J. Chen, C.H. Yeh, H.W. Chi, C.Y. Chen, Efficient multi-view video coding using inter-view information, Signal Process.: Image Commun. 29 (6) (2014) 667–677.
[5] B. Julesz, Foundations of Cyclopean Perception, University of Chicago Press, Chicago, IL, USA, 1971.
[6] J. Kopf, M. Cohen, D. Lischinski, M. Uyttendaele, Joint bilateral upsampling, ACM Trans. Graph. 26 (3) (2007) 1–5.
[7] P. Lai, A. Ortega, C.C. Dorea, P. Yin, C. Gomila, Improving view rendering quality and coding efficiency by suppressing compression artifacts in depth-image coding, in: Proc. of SPIE, Visual Communication and Image Processing, 2009, pp. 725700-1–725700-10.
[8] H.J. Lee, T. Chiang, Y.Q. Zhang, Scalable rate control for MPEG-4 video, IEEE Trans. Circ. Syst. Video Technol. 10 (6) (2000) 878–894.
[9] J.Y. Lee, H.C. Wey, D.S. Park, A fast and efficient multi-view depth image coding method based on temporal and inter-view correlations of texture images, IEEE Trans. Circ. Syst. Video Technol. 21 (12) (2011) 1859–1868.
[10] A. Levin, D. Lischinski, Y. Weiss, Colorization using optimization, ACM Trans. Graph. 23 (3) (2004) 689–694.
[11] Y. Liu, Q. Huang, S. Ma, D. Zhao, W. Gao, Joint video/depth rate allocation for 3D video coding based on view synthesis distortion model, Signal Process.: Image Commun. 24 (8) (2009) 666–681.
[12] S. Ma, W. Gao, Y. Lu, Rate-distortion analysis for H.264/AVC video coding and its application to rate control, IEEE Trans. Circ. Syst. Video Technol. 15 (12) (2005) 1533–1544.
[13] M. Magnor, P. Ramanathan, B. Girod, Multi-view coding for image based rendering using 3-D scene geometry, IEEE Trans. Circ. Syst. Video Technol. 13 (11) (2003) 1092–1106.
[14] P. Merkle, Y. Morvan, A. Smolic, D. Farin, K. Muller, P.H.N. de With, T. Wiegand, The effect of multiview depth video compression on multiview rendering, Signal Process.: Image Commun. 24 (1–2) (2009) 73–88.
[15] P. Merkle, A. Smolic, K. Muller, T. Wiegand, Efficient prediction structures for multiview video coding, IEEE Trans. Circ. Syst. Video Technol. 17 (11) (2007) 1461–1473.
[16] D. Min, J. Lu, M.N. Do, Depth video enhancement based on weighted mode filtering, IEEE Trans. Image Process. 21 (3) (2012) 1176–1190.
[17] Y. Morvan, D. Farin, P.H.N. deWith, Joint depth/texture bit-allocation for multi-view video compression, in: Proc. of Picture Coding Symposium (PCS), 2007, pp. 43–49.
[18] Y. Morvan, D. Farin, P.H.N. deWith, System architecture for freeviewpoint video and 3D-TV, IEEE Trans. Consum. Electron. 54 (2) (2008) 925–932.
[19] V.A. Nguyen, D. Min, M.N. Do, Efficient techniques for depth video compression using weighted mode filtering, IEEE Trans. Circ. Syst. Video Technol. 23 (2) (2013) 189–202.
[20] H. Oh, Y.S. Ho, H.264-based depth map sequence coding using motion information of corresponding texture video, in: Proc. of Pacific Rim Symposium on Advances in Image and Video Technology, 2006, pp. 898–907.
[21] P. Ramanathan, B. Girod, Rate-distortion analysis for light field coding and streaming, Signal Process.: Image Commun. 21 (6) (2006) 462–475.
[22] F. Shao, G. Jiang, M. Yu, K. Chen, Y.S. Ho, Asymmetric coding of multi-view video plus depth based 3-D video for view rendering, IEEE Trans. Multimedia 14 (1) (2012) 157–167.
[23] L.Q. Shen, Z. Liu, S.X. Liu, Z.Y. Zhang, P. An, Selective disparity estimation and variable size motion estimation based on motion homogeneity for multi-view coding, IEEE Trans. Broadcast. 55 (4) (2009) 761–766.
[24] L.Q. Shen, Z.Y. Zhang, Z. Liu, Inter mode selection for depth map coding in 3D video, IEEE Trans. Consum. Electron. 58 (3) (2012) 926–931.
[25] M. Tanimoto, Overview of free viewpoint television, Signal Process.: Image Commun. 21 (6) (2006) 454–461.
[26] C.H. Yeh, M.F. Li, M.J. Chen, M.C. Chi, X.X. Huang, H.W. Chi, Fast mode decision algorithm through inter-view rate-distortion prediction for multiview video coding system, IEEE Trans. Ind. Inform. 10 (1) (2014) 594–603.
[27] B.Y. Zhang, J.P. Allebach, Adaptive bilateral filter for sharpness enhancement and noise removal, IEEE Trans. Image Process. 17 (5) (2008) 664–678.
[28] Draft ITU-T recommendation and final draft international standard of joint video specification (ITU-T Rec. H.264/ISO/IEC 14496-10 AVC), JVT of ISO/IEC and ITU-T, 2003.
[29] ISO/IEC JTC1/SC29/WG11, WD 3 Reference Software for MVC, Doc. JVT-AC207., 2008.
[30] ISO/IEC JTC1/SC29/WG11, Distortion model of virtual views based on texture and depth compression, Doc. M171880., 2010.
[31] ISO/IEC JTC1/SC29/WG11 and ITU-T SG16/Q.6, Adaptive Basic Unit Layer Rate Control for JVT, Doc. JVT-G012., 2003.
[32] ISO/IEC MPEG and ITU-T VCEG, Multi-view video plus depth (MVD) format for advanced 3D video systems, SDoc. JVT-W100., 2007.
[33] MSR 3-D Video Sequences. <http://www.research.microsoft.com/vision/ImageBasedRealitites/3DVideoDownload/>.
[34] View synthesis reference software (VSRS) 3.5, Tech. Rep., ISO/IEC JTC1/SC29/WG11, 2010.