Optical Engineering

SPIEDigitalLibrary.org/oe

Object-based depth image–based rendering for a three-dimensional video system by color-correction optimization

Feng Shao
Gang-yi Jiang
Mei Yu
Yun Zhang

Downloaded From: http://opticalengineering.spiedigitallibrary.org/ on 06/16/2015 Terms of Use: http://spiedl.org/terms

Object-based depth image–based rendering for a three-dimensional video system by color-correction optimization

Feng Shao
Gang-yi Jiang
Mei Yu
Ningbo University
Faculty of Information Science and Engineering
No. 818 Fenghua Road
Ningbo, Zhejiang 315211, China
E-mail: jianggangyi@nbu.edu.cn

Yun Zhang
Chinese Academy of Sciences
Shenzhen Institutes of Advanced Technology
Shenzhen, 518055, China

Abstract. Three-dimensional (3-D) video technologies are becoming increasingly popular because they can provide high quality and immersive experience to end users. Depth image-based rendering (DIBR) is a key technology in 3-D video systems due to its low bandwidth cost as well as the arbitrary rendering viewpoint. We propose an object-based DIBR method by color-correction optimization. The proposed method first performs temporal consistent rendering to reduce the rendering complexity. Then, by segmenting the depth map into foreground and background, the object-based scalable rendering is performed to improve the rendering quality and reduce the rendering complexity. Finally, the rendered virtual view is further optimized by color-correction operation. Experimental results show that, compared to the results without the above optimization operations, the proposed method can reduce >40% computational complexity while maintaining high rendering quality. *© 2011 Society of Photo-Optical Instrumentation Engineers (SPIE).* [DOI: 10.1117/1.3569627]

Subject terms: depth image-based renderings; three-dimensional video systems; color correction optimization; depth maps; multiview videos.

Paper 100441RR received Jun. 1, 2010; revised manuscript received Feb. 27, 2011; accepted for publication Mar. 2, 2011; published online Apr. 25, 2011.

1 Introduction

Three-dimensional (3-D) video has gained popularity with its ability to give viewers an enhanced experience of multimedia in comparison to traditional two-dimensional (2-D) video. With these features, 3-D video will revolutionize visual media by enabling three-dimensional television (3-DTV) and free viewpoint video (FVV) applications.[1,2] In these applications, 3-DTV offers depth perception of the scene, while FVV allows the user to freely select the viewpoint of the scene. This opens up many new and interesting research topics as well as applications, such as multiview imaging, multiview video coding, virtual view rendering for 3-DTV or FVV, etc.

In order to represent the 3-D scene, many 3-D video formats were proposed. Conventional multiview video format was the simplest representation that only color video data were involved. The next more complex format was a monoscopic color video plus associated per-pixel depth information representation, which had been investigated by the European information society technologies project "advanced three-dimensional television system technologies."[3] However, only a very limited continuum around the available original viewpoint was supported for such representation. Therefore, Moving Pictures Experts Group (MPEG) specified a multiview video plus depth (MVD) format for 3-D video representation,[4] which had been widely used in research and standardization activities due to its flexible representation and compatibility with existing coding and transmission technologies. Recently, MPEG had started the exploration works on depth estimation and view synthesis for developing the 3-D video standard.[5,6]

In MVD representation, virtual view can be synthesized from MVD data by using depth image-based rendering (DIBR) technique.[7] The main advantage of the DIBR technique compared to traditional image-based rendering[8] (IBR) is that it can provide high-quality 3-D video with large viewing angle, which has emerged as a very promising rendering method for 3-D video display. Because accurate depth map acquisition is still an unsolved problem, DIBR requires solving high rendering quality due to the inaccurate depth information. Many works were carried out on this aspect.[9–11] Oh et al. proposed a depth-based in-painting method to fill the holes caused by disocclusion regions and inaccurate depth values.[9] Chen et al. proposed an edge-dependent Gaussian filter to solve the hole-filling problem in DIBR.[10] Daribo et al. proposed a distance-dependent filtering method to smooth sharp depth changes near object boundaries.[11] However, even though these methods can eliminate the influence of inaccurate depth information to a certian extent, how to improve the rendering quality in DIBR is still a very interesting problem in 3-D video research.

From another perspective, color inconsistency between views will greatly deteriorate the performance of the rendered virtual view. Color-correction preprocessing is an effective way to compensate the color inconsistency. Many color-correction preprocessing methods were proposed. Yamamoto *et al*. corrected luminance and chrominance of views by using lookup tables, the correspondences of which were detected by scale invariant feature transform.[12] Fecker modified the luminance and chrominance variations by calculating lookup tables from histograms of two views.[13] In our previous method, color correction was treated as an optimization problem, and the effective color mapping relationships were calculated by using dynamic programming technique.[14] However, even though these preprocessing methods can achieve good

0091-3286/2011/$25.00 © 2011 SPIE

Downloaded From: http://opticalengineering.spiedigitallibrary.org/ on 06/16/2015 Terms of Use: http://spiedl.org/terms

performance, they can not completely resolve color inconsistency in view rendering, and current DIBR algorithms completely ignored the color inconsistency problem.

To date, how to improve the rendering quality and reduce the rendering complexity simultaneously remains an important problem worthy of study. In model-based rendering, the depth map was triangulated into meshes so as to make use of high-speed graphical hardware.[15] To avoid real-time triangulation, the current DIBR algorithm in 3-D video adopted per-pixel depth information.[16] However, the rendering complexity is comparatively high for pixelwise rendering because the warping operation should be repeated for each pixel. This paper proposes an object-based rendering method for 3-D video systems in order to promote their rendering performances. The main characteristics of the proposed method are as follows:

1. By properly segmenting depth map into foreground and background, the rendering quality can be significantly improved.
2. By implementing temporal consistent rendering and block-based scalable rendering, large rendering complexity reduction can be achieved.
3. By conducting appropriate color-correction optimization, the perceptual quality of the rendered view can be improved.

This remainder of this paper is organized as follows. In Sec. 2, a problem in 3-D video system is discussed. Section 3 presents the proposed object-based DIBR method. The experimental results are analyzed in Sec. 4, and conclusions are drawn in Sec. 5.

2 Problem Description in Three-Dimensional Video System

The system framework of 3-D video systems is illustrated in Fig. 1. In the system, the MVD format data consist of several camera sequences of color images and associated depth maps, as illustrated in Fig. 2. The 3-D video system includes a number of sophisticated processing steps that are partially unresolved and still require research. At the sender side, multiview color images are captured with multiple cameras. The captured images may contain the misalignment and color differences of the cameras. Then image correction, including geometric calibration[17] and color correction[18] as well as depth estimation,[19] are the key technologies to be implemented. Then, depth and color video data are compressed in a joint encoding or independent encoding mode. At the client side, after receiving both depth and color video bit streams, virtual views are synthesized by DIBR technique. Therefore, MVD data coding as well as DIBR are crucial processes in the 3-D video system because they will affect the transmission capacity and rendering quality in the system.

However, most existing multiview video coding (MVC) schemes[20,21] have not taken regional selective attention and 3-D depth perception of human visual system (HVS) into consideration. It has been revealed that HVS is more sensitive to the distortion in region of interest (ROI) than the background region.[22] This property has been successfully applied to depth perceptual ROI-based MVC.[23] For depth map coding, the impact on the quality of the rendered view is an important factor to evaluate the performance of depth map coding algorithms. According to the analysis in Ref. 24, because the rendering position error increases monotonically as the coding error increases, the importance of different regions for rendering should be different. As shown in Fig. 3, by analyzing the just-noticeable difference of depth for view rendering,[25] four types of blocks are used to classify the depth map based on their importance for view rendering. It is obvious that depth map distortion in boundaries and foreground regions will have a great impact on the rendering quality against other regions.

Therefore, an object-based approach should be an important inspiration to yield better rendering results. Besides, the object-based approach has other outstanding advantages. First, it is forward compatible with the ROI-based coding framework.[23] Second, desirable functionalities such as scalability of contents and interactivity with individual objects can be further enhanced. In this paper, we only focus on the rendering problem in 3-D video system. In order to promote the rendering performance in quality and complexity, the proposed object-based rendering approach aims to solve the following two issues. One is how to reduce the rendering complexity by fully exploiting the redundant information from adjacent pixels in depth map or successive pixels in depth sequence. The other is how to improve the rendering quality on the premise of low complexity by object-based project methodology. The key novelty of our work is that we take full advantage of available information in a 3-D video system in order to achieve overall lower rendering complexity as well as higher rendering quality.

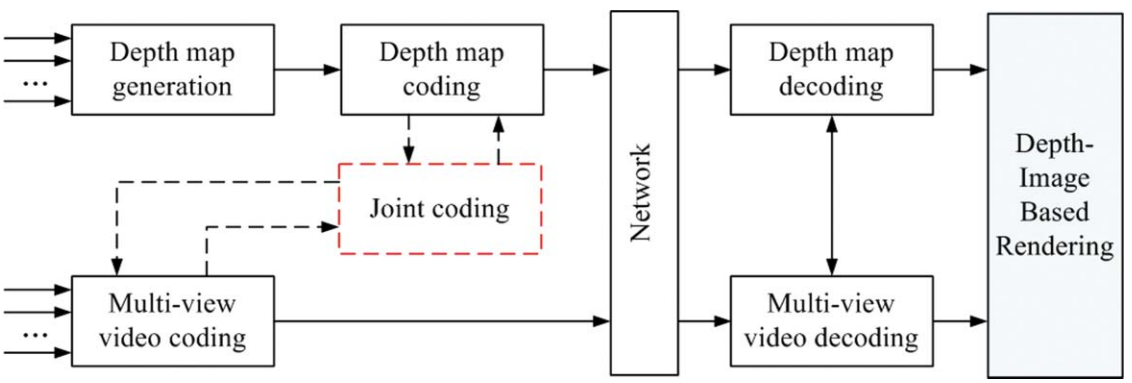
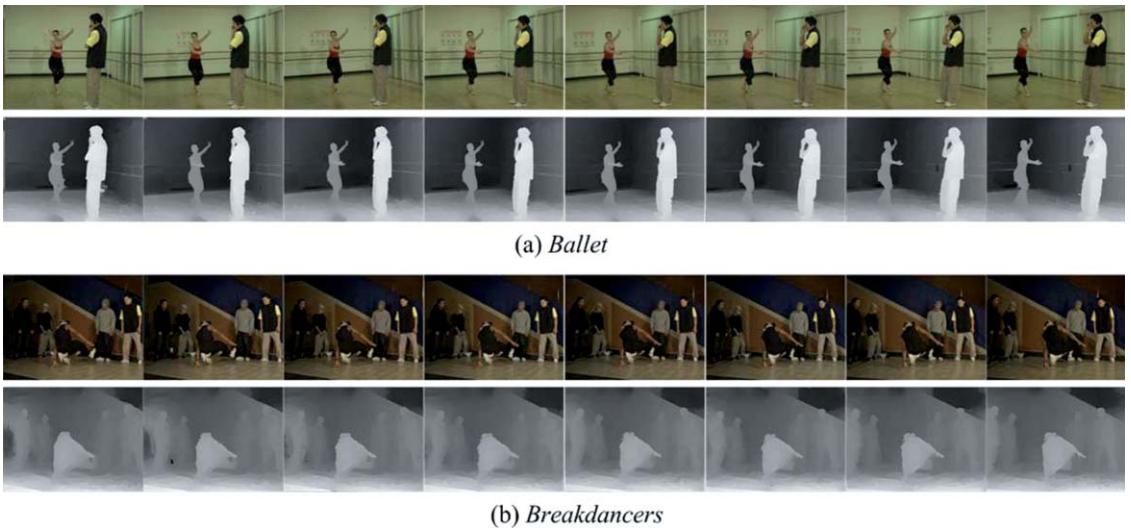


Fig. 1 System framework of 3-D video system.

Downloaded From: http://opticalengineering.spiedigitallibrary.org/ on 06/16/2015 Terms of Use: http://spiedl.org/terms



(a) *Ballet*

(b) *Breakdancers*

Fig. 2 Example for the MVD format with color image and the corresponding depth map: (a) "Ballet" and (b) "Breakdancers."

3 Proposed Object-Based Depth Image-Based-Rendering Method by Color-Correction Optimization

DIBR is a technique to render a virtual view image by warping the pixels of a reference view image to a target view image using the per-pixel depth value.[26] 3-D warping projects an image to another image plane. It can be decomposed into two fundamental steps. First, the pixel position (x,y) at the frame coordinate system of camera at p, is projected into a world coordinate $[u, v, w]^T$. With intrinsic parameter matrix $\mathbf{A}_p$ (3×3) of the camera at p, rotation matrix $\mathbf{R}_p$ (3×3), and the translation vector $\mathbf{T}_p$ (3×1), this projection can be represented as

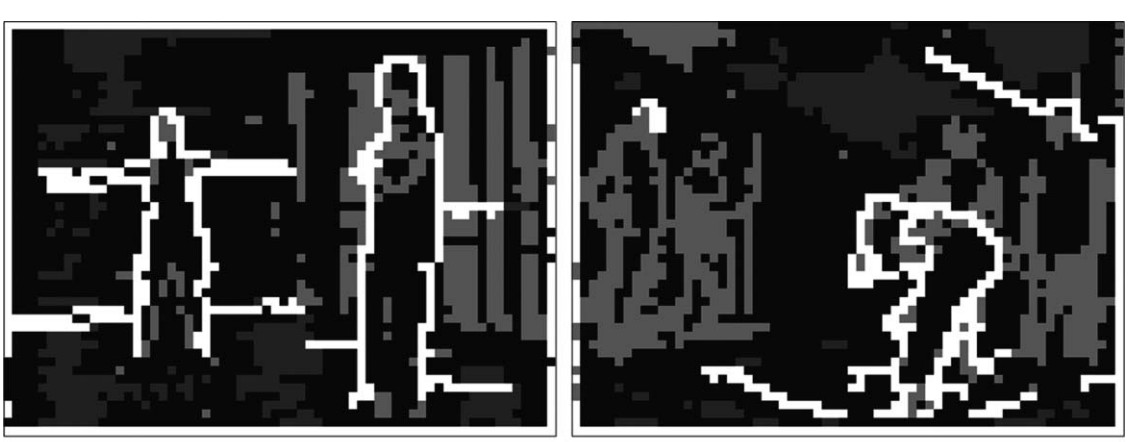
$$[u, v, w]^T = \mathbf{R}_p \cdot \mathbf{A}_p^{-1} \cdot [x, y, 1]^T \cdot Z_p(x, y) + \mathbf{T}_p. \quad (1)$$

Then, $[u, v, w]^T$ is mapped to coordinate $[x', y', z']^T$ of the virtual camera at p' with $\mathbf{A}_{p'}$, $\mathbf{R}_{p'}$, and $\mathbf{T}_{p'}$

$$[x', y', z']^T = \mathbf{A}_{p'} \cdot \mathbf{R}_{p'}^{-1} \cdot \{[u, v, w]^T - \mathbf{T}_{p'}\}. \quad (2)$$

We can find that coordinate transformation in the above projection equations will spend high computational cost. Ideally, when one pixel is projected from one camera to another camera, 13 floating-point multiplications and 12 floating-point additions are needed. Because of the extremely high computational complexity in 3-D warping, the capability of DIBR technique will be limited for low complexity applications. In what follows, we are interested in reducing the complexity of 3-D warping under the object-based framework. In principle, adjacent pixels may have a similar warping relation.

According to the above analyses, we propose an object-based DIBR method for 3-D video. General framework of the proposed method is given in Fig. 4. In the framework, the MVD data and object mask are inputted from encoder. First, temporal-consistent regions are detected and temporal-consistent rendering is performed to reduce the rendering complexity. Then, based on object-mask information, the depth map is segmented into foreground and background, and the object-based scalable rendering is performed to improve the rendering quality and reduce the rendering complexity. Finally, color correction is implemented to optimize the rendered results. In the framework, three key technologies are performed that are essentially correlative.



(a) (b)

Fig. 3 Example of the classification of different types of blocks for view rendering: (a) "Ballet" and (b) "Breakdancers."

Downloaded From: http://opticalengineering.spiedigitallibrary.org/ on 06/16/2015 Terms of Use: http://spiedl.org/terms

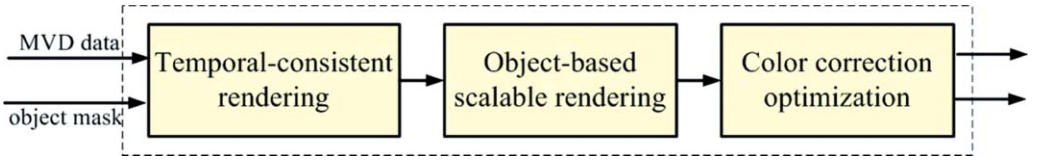


Fig. 4 Framework of the proposed DIBR method.

3.1 *Temporal Consistent Rendering*

According to the analysis, when a macroblock (MB) at current frame has the SKIP mode, the corresponding MB in the previous frame tends to have the same depth value as well. On the basis of this statistical tendency, the warping relations of these MBs in the current frame can be directly decided by referring to the previous frame without additional calculation. In order to identify the relation, a MB temporal consistency (MTC) parameter at time t, denoted as MTC_t, is defined. The decision rule for obtaining $\mathrm{MTC_t}$ can be expressed as

$$\mathrm{MTC}_t = \begin{cases} 1, & \text{if } \mathrm{MODE}_d = \{\mathrm{SKIP}\} \\ 0, & \text{otherwise} \end{cases}. \tag{3}$$

Here, if $\mathrm{MTC}_t = 1$, then the warping relation in the previous frame is directly adopted for the current MB. Otherwise, the warping relation is calculated for each pixel in the MB. Figure 5 shows the flowchart of the proposed warping algorithm. In the algorithm, in order to reduce the warping complexity for these pixels with $\mathrm{MTC}_t(x, y) = 0$, the warping operation is further optimized by object-based scalable rendering approach. The detailed approach is described in Sec. 3.2.

3.2 *Object-Based Scalable Rendering*

However, scenes with large depth variations will lead to serious rendering artifacts, such as ghosting and blurring around depth discontinuities. An effective approach to yield better rendering results is to segment the scene into different depth layers so that the adverse effect of depth discontinuities can be mitigated by matting and inpainting techniques.[27] This idea has been demonstrated in the layered depth image, which is possible to detect occlusion and interpolate the image pixels in rendering.[28] An object-based scalable rendering method with much less computational complexity is proposed in this paper.

An important step in the object-based approach is to segment objects into different depth layers. It is assumed that object mask information is transmitted from the encoder, because it is very time consuming to extract the object information at the decoder side. In the proposed method, the extracted object mask in Ref. 23 is directly used, which can be viewed as a generalization of the depth perceptual ROI for MVC. The ROI object is modeled by combining the features from motion, depth, and texture, and it is defined as

$$\mathrm{M_{ROI}} = \{\mathrm{M_m}, \mathrm{M_c}, \mathrm{M_f}, \mathrm{M_d}\}, \tag{4}$$

where $\mathrm{M_m}$, $\mathrm{M_c}$, $\mathrm{M_f}$, and $\mathrm{M_d}$ denote motion mask, contour mask, foreground mask, and depth mask, respectively. In order to clearly represent the object distribution information for each pixel, if a pixel belongs to the ROI object, $\mathrm{M_{ROI}}(x, y) = 1$; otherwise, $\mathrm{M_{ROI}}(x, y) = 0$. The corresponding detailed equations can be referred in Ref. 23.

Then, after excluding those pixels with $\mathrm{MTC_t}(x, y) = 1$ from the extracted ROI object, the object mask for rendering can be expressed as

$$\mathrm{M_{obj}}(x, y) = \begin{cases} 1, & \text{if } \mathrm{MTC}_t(x, y) = 0 \text{ and } \mathrm{M_{ROI}}(x, y) = 1 \\ 0, & \text{if } \mathrm{MTC}_t(x, y) = 0 \text{ and } \mathrm{M_{ROI}}(x, y) = 0 \end{cases}. \tag{5}$$

Similarly, if $\mathrm{M_{obj}}(x, y) = 1$, the pixel belongs to the foreground object; otherwise, it belongs to the background object. As shown in Fig. 6, by applying the object mask $\mathrm{M_{obj}}$ on "Ballet" and "Breakdancers" sequences, the foreground object is detected as marked in black.

From another perspective, according to the analysis, if a pixel in the reference image at location (x, y) in the reference image is projected to (x', y') in the virtual image, its adjacent pixel $(x + 1, y)$ in the reference image will be approximately projected to $(x' + 1, y')$ in the virtual image if their depth values are similar. This approximation will hold true for most regions with a flat depth map. Therefore, we define eight MB partition types according to their block sizes, such as 8×8, 8×4, 4×8, 4×4, 4×2, 2×4, 2×2, and single pixel, as shown in Fig. 7. For block-based scalable rendering, the projection relation is approximately the same for all pixels within the block if the block is proved to be flat. Thus, only one projection is needed for all pixels within the block. We use the following first-order form variance information to determine the flatness of the block:

$$v_{\mathrm{B}} = \frac{1}{p \times q} \sum_{(x,y)\in\Omega} |d(x, y) - m_{\mathrm{B}}|, \tag{6}$$

$$m_{\mathrm{B}} = \frac{1}{p \times q} \sum_{(x,y)\in\Omega} d(x, y), \tag{7}$$

where $d(x, y)$ denotes depth value in the pixel position (x, y), $p \times q$ denotes the block size, and Ω is the set of all pixels within the block. If the current block satisfies $v_{\mathrm{B}} \leq T_{\mathrm{f}}$, then

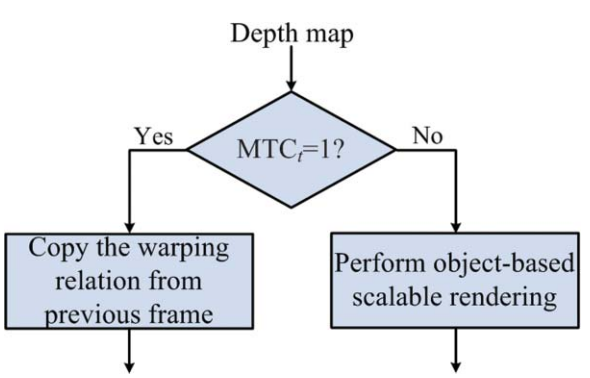


Fig. 5 Flowchart of the proposed 3-D warping algorithm.

Downloaded From: http://opticalengineering.spiedigitallibrary.org/ on 06/16/2015 Terms of Use: http://spiedl.org/terms

Fig. 6 Example of the extracted foreground object for rendering: (a) "Ballet" (view 4, 40th frame), (b) "Ballet" (view 6, 40th frame), (c) "Breakdancers" (view 4, 40-th frame), (d) "Breakdancers" (view 6, 40th frame).

it is regarded as flat. Otherwise, the block will continue to be split in a scalable manner. Here, $T_f = 1$ in the experiment.

Then, by combining the above object- and block-based scalable rendering schemes, the detail algorithm is described as follows:

Step 1. In order to avoid ghosting artifacts at the boundaries, the boundary pixels are first projected from source view to the virtual view by using pixelwise warping.

Step 2. Then, in order to avoid the problem that the pixels with large depth values are projected to the location of the pixels with small depth values, the foreground object is first projected. Small block scalability from 4×4, 4×2, 2×4, 2×2 to single pixel is used to achieve the projection from the source view to the virtual view, and small holes in the warped image are interpolated from neighboring available pixels.

Step 3. Finally, large block scalability from 8×8, 8×4, 4×8, 4×4 to single pixel is used to project the background object from the source view to the virtual view. If the pixel has been projected in the warped image, then it is no longer necessary to project again in this step.

3.3 *Proposed Color-Correction Optimization Method*

After generating the corresponding virtual views from left and right source views, the next step is view blending to combine the warped virtual views. The general intermediate-view interpolation method is a weighted sum of two images based on their baseline distance.[29] In the proposed method, in order to simplify the blending operation, the supposed left view as the base view and the right view as the assistant view, the virtual view synthesis is expressed as

$$I_v(x_v, y_v) = \begin{cases} I_l(x_l, y_l) & \text{if } (x_l, y_l) \text{ is visible in left view} \\ I_r(x_r, y_r) & \text{if } (x_r, y_r) \text{ is only visible in right view,} \\ 0 & \text{otherwise} \end{cases} \tag{8}$$

where $I_v(x_v, y_v)$, $I_l(x_l, y_l)$, and $I_r(x_r, y_r)$ are the pixel values of the matching points in virtual view, left view, and right view, respectively, and $I_v(x_v, y_v) = 0$ denotes a hole point. It should be noted that in the proposed DIBR method, in order to solve the following problems, such as hole filling, visibility, and resampling, the similar means with view synthesis reference software[16] is used.

However, due to the difficulty in acquiring accurately scene color in different camera viewpoints, there may be obvious color differences between newly added regions and

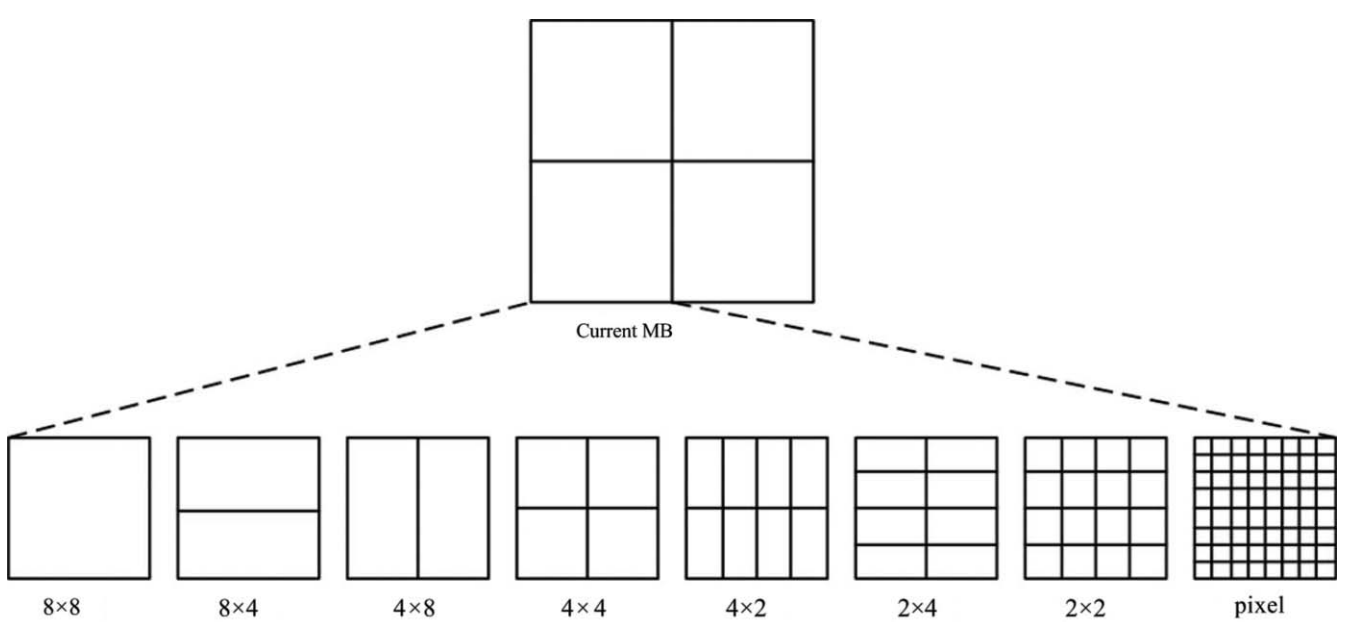


Fig. 7 MB partition rule defined in the proposed method.

Downloaded From: http://opticalengineering.spiedigitallibrary.org/ on 06/16/2015 Terms of Use: http://spiedl.org/terms

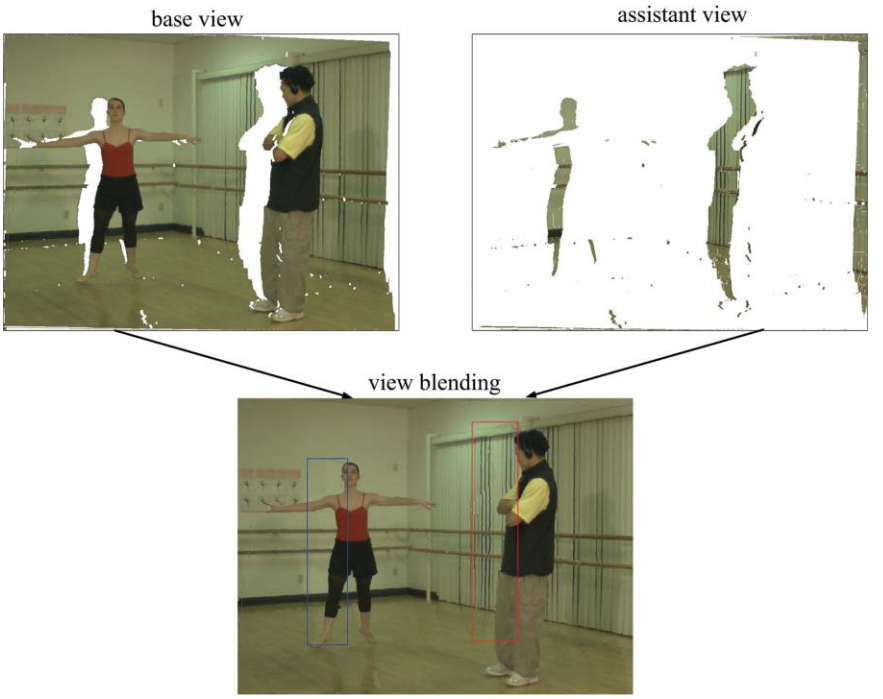


Fig. 8 View blending operation in DIBR.

its surrounding regions after view blending. As depicted in Fig. 8, these inconsistencies might lead to some noticeable artifacts in the virtual view. Therefore, additional color-correction operation must be designed to ensure color consistency in the virtual view. Here, we propose a color-correction algorithm embedded in the rendering framework. The detailed color-correction algorithm is described as follows:

Step 1. For the warped pixels from base view, we compute the means and standard deviations of all the warped pixels in YUV color space, denoted by μ_1^{Y}, μ_1^{U}, μ_1^{V}, δ_1^{Y},δ_1^{U}, δ_1^{V}, and the number of the warped pixels denoted by num1.

Step 2. For the warped pixels from assistant view, we compute the means and standard deviations of all the warped pixels in YUV color space, denoted by μ_2^{Y}, μ_2^{U}, μ_2^{V}, δ_2^{Y},δ_2^{U}, δ_2^{V}, and the number of the warped pixels denoted by num2.

Step 3. We weight the means and standard deviations for base view and assistant view, and the weighted the mean and standard deviation for the Y component are expressed as

$$\mu^{\mathrm{Y}} = \frac{\mu_1^{\mathrm{Y}} \times \mathrm{num1} + \mu_2^{\mathrm{Y}} \times \mathrm{num2}}{\mathrm{num1} + \mathrm{num2}}, \tag{9}$$

$$\delta^{Y} = \frac{\delta_1^{Y} \times \mathrm{num1} + \delta_2^{Y} \times \mathrm{num2}}{\mathrm{num1} + \mathrm{num2}}. \tag{10}$$

Step 4. We conduct color correction for the pixels in the synthesized view based on their classification results, and the correction equation for Y component is expressed as

$$Y'(x,y) = \begin{cases} \frac{\delta^Y}{\delta_1^Y}(Y(x,y) - \mu_1^Y) + \mu^Y, & \text{if}(x,y) \in \text{base view} \\ \frac{\delta^Y}{\delta_2^Y}(Y(x,y) - \mu_2^Y) + \mu^Y, & \text{if}(x,y) \in \text{assistant view} \end{cases}, \tag{11}$$

where $Y'(x, y)$ denotes the corrected value of the pixel. For UV components, similar operations with steps 3 and 4 are performed.

However, color incoherence will be still obvious in the blended boundaries even after color correction. In other words, color correction alone cannot remove all the color difference in the rendered view. In order to eliminate such noises in the rendered view, a filtering operation is designed. As shown in Fig. 9, the ghost contour is first detected from the rendered view, and it is then expanded to cover the boundaries between foreground and background. Then, a three-tap low-pass filter is applied to the contour in both horizontal and vertical directions to reduce the ghost contour and provide a more natural appearance.

4 Experimental Results

In order to evaluate the performance of the proposed DIBR method, several experiments are conducted with test sequences of "Ballet" and "Breakdancers." The sequences are captured by Microsoft Research,[30] and the depth maps of these sequences had been computed by using a stereo-matching algorithm. In the rendering experiments, the fourth and sixth views are regarded as base and assistant views, and the fifth view is regarded as virtual view. In order to

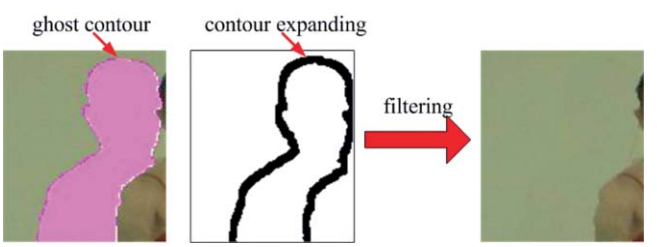


Fig. 9 Example of filtering operation for object boundaries.

Downloaded From: http://opticalengineering.spiedigitallibrary.org/ on 06/16/2015 Terms of Use: http://spiedl.org/terms

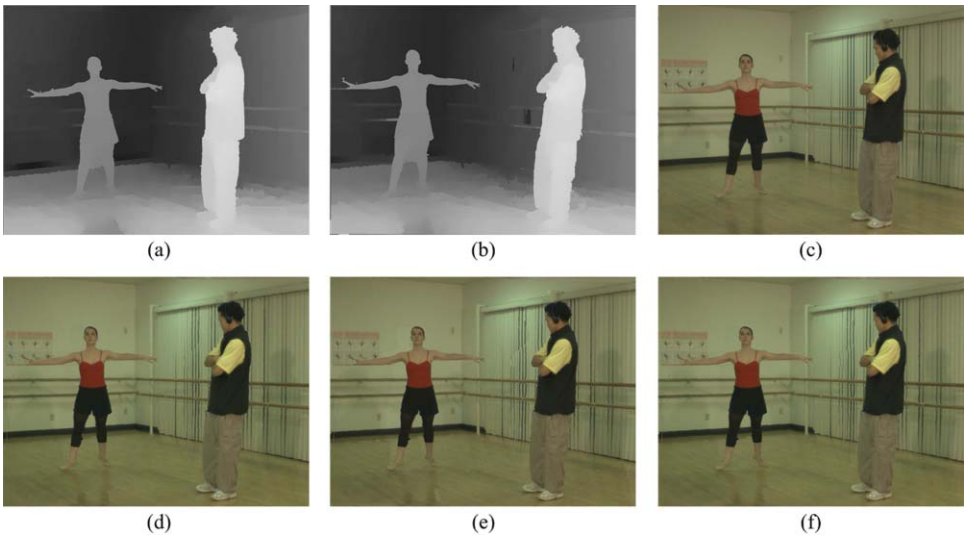


Fig. 10 View rendering results of "Ballet:" (a) Depth map (view 4, 40th frame), (b) depth map (view 6, 40-th frame), (c) original virtual view image (view 5, 40th frame), (d) the rendered virtual view image by scheme 1, (e) the rendered virtual view image by scheme 2, and (f) the rendered virtual view image by scheme 3.

objectively measure the performances of the proposed method, we design the following three schemes:

Scheme 1. Traditional pixel-wise DIBR method with color correction optimization

Scheme 2. The proposed DIBR method without color correction optimization

Scheme 3. The proposed DIBR method

Noted that, for scheme 1, pixel-based warping is performed for all pixels, and color-correction optimization in the framework is still preserved.

4.1 *Subjective Rendering Results*

Because different bit-allocation strategies between the color video and depth map will affect the rendering quality,[31] in order to eliminate these influences, original color video and depth map are used to synthesize the virtual view. Figures 10(a)–10(c) and 11(a)–11(b) show the original depth maps in views 4 and 6. Compared to the original virtual view images in Figs. 10(c)–11(c), the rendered virtual view images using the three schemes in Figs. 10(d)–10(f) and 11(d)–11(f) have much better subjective results. However, the rendered virtual view images generated by scheme 2 have noticeable artifacts in the disoccluded boundaries, as enlarged in Figs. 12 and 13. Obviously, the proposed color-correction optimization can eliminate these artifacts in the rendered virtual view images. Besides, the differences between the rendered virtual view images using scheme 1 and the proposed scheme are insignificant, but the computational complexity is vastly different.

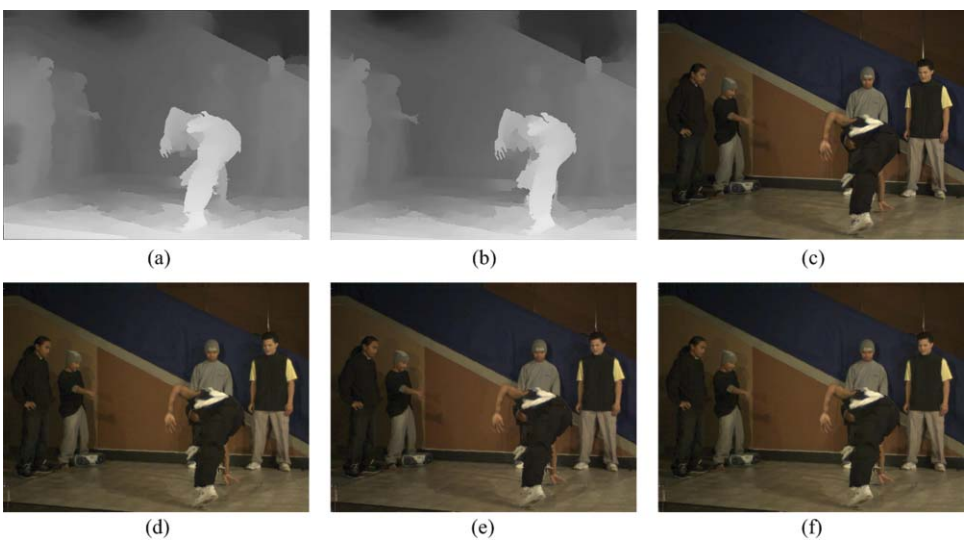


Fig. 11 View rendering results of "Breakdancers." (a) depth map (view 4, 40th frame), (b) depth map (view 6, 40th frame), (c) original virtual view image (view 5, 40th frame), (d) the rendered virtual view image by scheme 1, (e) the rendered virtual view image by scheme 2, and (f) the rendered virtual view image by scheme 3.

Downloaded From: http://opticalengineering.spiedigitallibrary.org/ on 06/16/2015 Terms of Use: http://spiedl.org/terms

Fig. 12 Detail examples for rendering results of "Ballet:" (a) Original virtual view image, (b) the rendered virtual view image by scheme 1, (c) the rendered virtual view image by scheme 2, and (d) the rendered virtual view image by scheme 3.

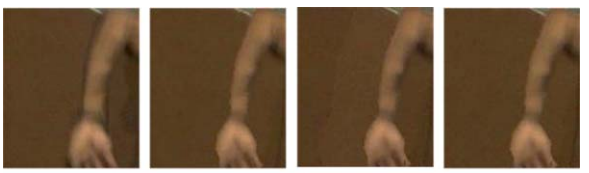


Fig. 13 Detail examples for rendering results of "Breakdancers:" (a) Original virtual view image, (b) the rendered virtual view image by scheme 1, (c) the rendered virtual view image by scheme 2, and (d) the rendered virtual view image by scheme 3.

Table 1 Objective performance comparison of different schemes.

Sequences\ Schemes	Ballet		Breakdancers	
	PSNR (dB)	SSIM	PSNR (dB)	SSIM
1	32.07	0.8615	31.90	0.8146
2	31.66	0.8492	31.51	0.8138
3	31.83	0.8526	31.81	0.8130

Table 2 Speedup performance comparison of different schemes.

Sequence	QP	Proposed (ms)	Saving (%)
Ballet	20	22.56	44.98
	28	19.91	51.44
	32	18.71	54.37
	36	17.32	57.77
Breakdancers	20	23.05	43.79
	28	22.05	46.23
	32	21.13	48.46
	36	19.86	51.57

4.2 *Objective Rendering Results*

In order to objectively evaluate the quality of the rendered view, the original virtual view is regard as reference, and the average peak signal-to-noise ratio (PSNR) and structural similarity (SSIM) evaluation values between the rendered and the reference views are measured. Table 1 shows the average PSNR and SSIM comparison results of the three schemes. Figures 14 and 15 show the PSNR and SSIM comparison results for each frame. From the results, it can be seen that scheme 1 gives the best rendering quality, and the proposed scheme follows closely to scheme 1, while scheme 2 gives the worst performance against other schemes. That is to say, the proposed color-correction optimization operation can largely promote the rendering quality, and the proposed object-based scalable rendering operation will slightly degrade the rendering quality. The overall rendering quality of the proposed scheme is very close to the results in Ref. 9, and in some cases, the proposed scheme will be better than Ref. 9.

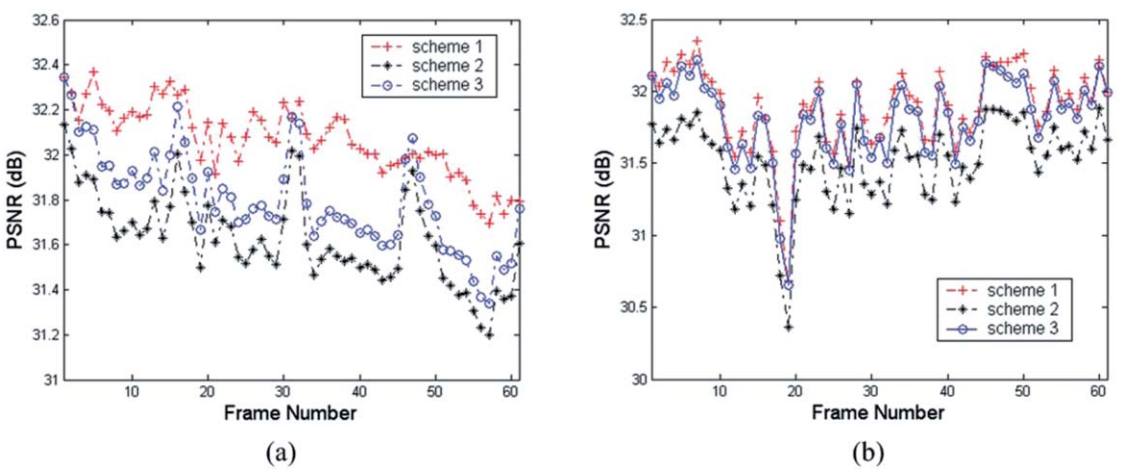


Fig. 14 Comparison of PSNR for each frame: (a) "Ballet" and (b) "Breakdancers."

Downloaded From: http://opticalengineering.spiedigitallibrary.org/ on 06/16/2015 Terms of Use: http://spiedl.org/terms

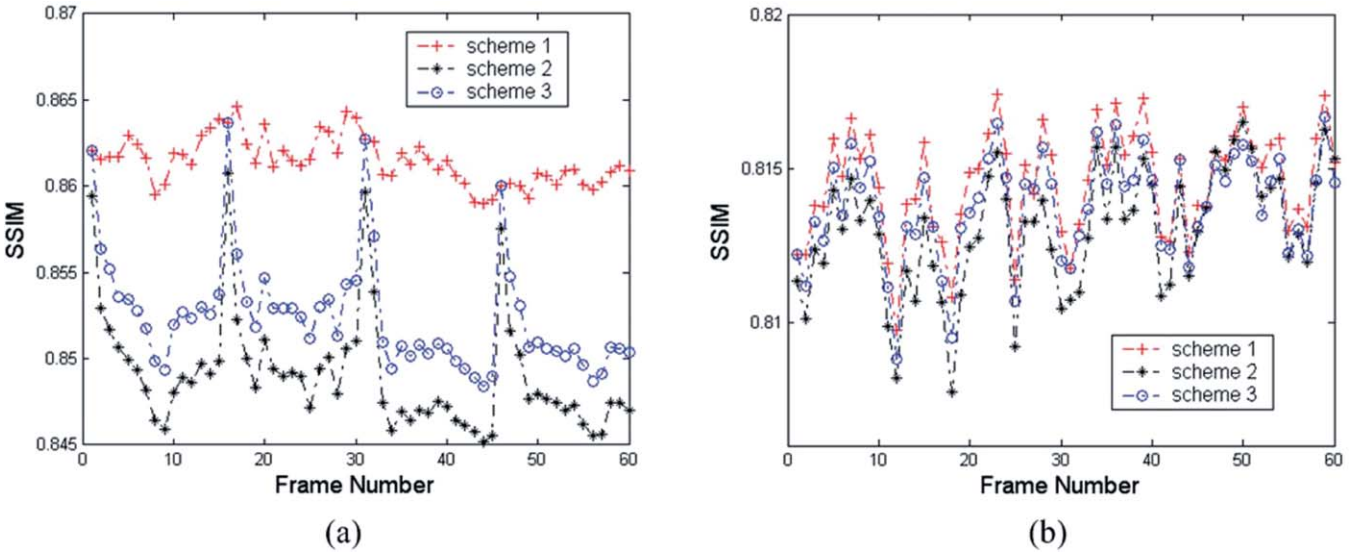


Fig. 15 Comparison of SSIM for each frame: (a) "Ballet" and (b) "Breakdancers."

4.3 *Computational Complexity*

In order to objectively evaluate the computational complexity of the proposed method, we measure the software execution time of pixelwise rendering and object-based scalable rendering schemes in the same test condition. The algorithm is executed through a PC in Visual C + + .NET version 2005. The test PC was equipped with a 2.1-GHZ Intel Core2 Duo CUP and 1 GB DDR SDRAM memory. Because 3-D warping will occupy most of the computational complexity in view rendering, we only calculate the time of 3-D warping in the rendering process. The comparison results are shown in Table 2. It is obvious that compared to the pixelwise rendering scheme, the proposed object-based scalable rendering scheme can significantly reduces the computational complexity, ranging from 43.79 to 57.77%. Moreover, the speed of warping will be improved with the increasing quantization parameter (QP) value. This is because more MBs are encoded by the SKIP mode with a larger QP value, and the warping relations of more MBs are directly copied from the previous frame. Thus, large savings of the computational complexity can be achieved.

5 Conclusions

DIBR is a key technology in 3-D video system. In this paper, an object-based DIBR method is proposed by color-correction optimization. The proposed method takes the redundant information for rendering into account, providing significant complexity reduction. Moreover, the proposed method segments the depth map into foreground and background objects to improve the rendering quality. In the future, we will undertake further research on three aspects. First, depth-map enhancement method can be used to improve the rendering quality. Second, object boundaries processing can be used to eliminate ghost contour. Third, the rendering performance can be further promoted by using some quality improving techniques.

Acknowledgments

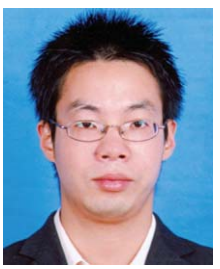
This work was supported by the Natural Science Foundation of China (Grant Nos. 60872094, 60832003, 60902096, and 61071120), the Specialized Research Fund for the Doctoral Program of Higher Education of China (Grant Nos. 20093305120002 and 200816460003). It was also sponsored by the K.C.Wong Magna Fund in Ningbo University.

References

1. A. Kubota, A. Smolic, M. Magnor, M. Tanimoto, T. Chen, and C. Zhang, "Multiview imaging and 3DTV," *IEEE Signal Process. Mag.* **24**(6), 10–21 (2007).
2. Y. Morvan, D. Farin, and P.H. N. de With, "System architecture for free-viewpoint video and 3D-TV," *IEEE Trans. Consum. Electron.* **54**(2), 925–932 (2008).
3. C. Fehn, R. de la Barré, and S. Pastoor, "Interactive 3-DTV-Concepts and Key Technologies," *Proc IEEE* **94**(3), 524–538 (March 2006).
4. ISO/IEC JTC1/SC29/WG11, "Multi-view video plus depth (MVD) format for advanced 3D video systems," Doc. W100, Marrakech, Morocco (April 2007).
5. ISO/IEC JTC1/SC29/WG11, "Depth estimation reference software (DERS) with image segmentation and block matching," Doc. M16092, Lausanne, Switzerland, (February 2009).
6. ISO/IEC JTC1/SC29/WG11, "Reference software of depth estimation and view synthesis for FTV/3DTV," Doc. M15836, Busan, Korea (October 2008).
7. C. Fehn, "Depth-image-based rendering (DIBR), compression and transmission for a new approach on 3D-TV," *Proc. SPIE* **5291**, 93–104 (2004).
8. C. Zhang and T. Chen, "A survey on image-based rendering—representation, sampling and compression," *Signal Process. Image Commun.* **19**(1), 1–28 (2004).
9. K. J Oh, S. Yea, A. Vetro, and Y.S. Ho, "Virtual view synthesis method and self-evaluation metrics for free viewpoint television and 3D video," *Int. J. Image Syst. Technol.* **20**(3), 378–390 (2010).
10. W. Y. Chen, Y. L. Chang, S. F. Lin, L. F. Ding, and L. G. Chen, "Efficient depth image based rendering with edge-dependent depth filter and interpolation," in *Proc. of IEEE Int. Conf. on Multimedia and Expo*, Amsterdam, pp. 1314–1317 (2005).
11. I. Daribo, C. Tillier, and B. Pesquet-Popescu, "Distance dependent depth filtering in 3D warping for 3DTV," in *Proc. of IEEE Workshop on Multimedia Signal Processing*, Thessaloniki, pp. 312–315 (2007).
12. K. Yamamoto, M. Kitahara, H. Kimata, T. Yendo, T. Fujii, T. Tanimoto, S. Shimizu, K. Kamikura, and Y. Yashima, "Multiview video coding using view interpolation and color correction," *IEEE Trans. Circuits Syst. Video Technol.* **17**(11), 1436–1449 (2007).
13. U. Fecker, M. Markowsky, and A. Kaup, "Histogram-based prefiltering for luminance and chrominance compensation of multi-view video," *IEEE Trans. Circuits Syst. Video Technol.* **18**(9), 1258–1267 (2008).
14. F. Shao, G. Y. Jiang, and M. Yu, "Multi-view video color correction using dynamic programming," *J. Syst. Eng. Electron.* **19**(6), 1115–1120 (2008).
15. P. Verlani, A. Goswami, P.J. Narayanan, S. Dwivedi, and S.K. Penta, "Depth image: representations and real-time rendering," in *Proc. of Int.*

Downloaded From: http://opticalengineering.spiedigitallibrary.org/ on 06/16/2015 Terms of Use: http://spiedl.org/terms

Symp. on 3D Data Processing, Visualization, and Transmission, Chapel Hill, NC, pp. 962–969 (2006).
16. ISO/IEC JTC1/SC29/WG11, "Virtual synthesis algorithm in view synthesis reference software 2.0," Doc. M16090, Lausanne, Switzerland (February 2009).
17. Z. Y. Zhang, "Camera calibration with one-dimensional objects," *IEEE Trans. Pattern Anal. Mach. Intell.* **26**(7), 892–899 (2004).
18. F. Shao, G.Y. Jiang, M. Yu, and Y.S. Ho, "Fast color correction for multiview video by modeling spatio-temporal variation," *J. Vis. Commun. Image Represent.* **21**(5-6), 392–403 (2010).
19. ISO/IEC JTC1/SC29/WG11, "Depth estimation reference software (DERS) with image segmentation and block matching," Doc. M16092, Lausanne, Switzerland (February 2009).
20. P. Merkle, A. Smolic, K. Müller, and T. Wiegand, "Efficient prediction structures for multiview video coding," *IEEE Trans. Circuits Syst. Video Technol.* **17**(11), 1461–1473 (2007).
21. S. Yea and A. Vetro, "View synthesis prediction for multiview video coding," *Signal Process. Image Commun.* **24**(1–2), 89–100 (January 2009).
22. Z. Lu, W. Lin, X. Yang, E. Ong, and S. Yao, "Modelling visual attention's modulatory aftereffects on visual sensitivity and quality evaluation," *IEEE Trans. Image Process.* **14**(11), 1928–1942 (2005).
23. Y. Zhang, G. Y. Jiang, M. Yu, Y. Yang, Z. J. Peng, and K. Chen, "Depth perceptual region-of-interest based multi-view video coding," *J. Vis. Commun. Image Represent.* **21**(5–6), 498–512 (2010).
24. P. Lai, A. Ortega, C. C. Dorea, P. Yin, and C. Gomila, "Improving view rendering quality and coding efficiency by suppressing compression artifacts in depth-image coding," *Proc. SPIE* **7257**, 72570O (2009).
25. D. V. S. X. De Silva, W. A. C. Fernando, G. Nur, E. Ekmekcioglu, and S. T. Worrall, "3D video assessment with just noticeable difference in depth evaluation," in *Proc. of Int. Conf. on Image Processing*, Hong Kong, pp. 4013–4016 (September 2010).
26. Y. Mori, N. Fukushima, T. Yendo, T. Fujii, and M. Tanimoto, "View generation with 3D warping using depth information for FTV," *Signal Process. Image Commun.* **24**(1–2), 65–72 (January 2009).
27. S. C. Chan, Z. F. Gan, K. T. Ng, K. L. Ho, and H. Y. Shum, "An object-based approach to image/video-based synthesis and processing for 3-D and multiview televisions," *IEEE Trans. Circuits Syst. Video Technol.* **19**(6), 821–831 (June 2009).
28. S. U. Yoon and Y. S. Ho, "Multiple color and depth video coding using a hierarchical representation," *IEEE Trans. Circuits Syst. Video Technol.* **17**(11), 1450–1460 (November 2009).
29. K. Müller, A. Smolic, K. Dix, P. Merkle, P. Kauff, and T. Wiegand, "View synthesis for advanced 3D video systems," *EURASIP J. Image Video Process.* **2008**, 438148 (2008).
30. C. L. Zitnick, S. B. Kang, M. Uyttendaele, S. Winder, and R. Szeliski, "High-quality video view interpolation using a layered representation," *ACM Trans. Graphics* **23**(3), 600–608 (2004).
31. Y. W. Liu, Q. M. Huang, S. W. Ma, D. B. Zhao, and W. Gao, "Joint video/depth rate allocation for 3D video coding based on view synthesis distortion model," *Signal Process. Image Commun.* **24**(8), 666–681 (September 2009).



Feng Shao received his BS and PhD from Zhejiang University, Hangzhou, China, in 2002 and 2007, respectively, all in Electronic Science and Technology. He is currently an associate professor in Faculty of Information Science and Engineering, Ningbo University, China. His research interests include image/video coding, image processing, and image perception.

Gang-yi Jiang received his MS from Hangzhou University in 1992 and his PhD from Ajou University, Korea, in 2000. He is now a professor in Faculty of Information Science and Engineering, Ningbo University, China. His research interests mainly include digital video compression and communications, multiview video coding and image processing.

Mei Yu received her MS from Hangzhou Institute of Electronics Engineering, China, in 1993, and PhD form Ajou University, Korea, in 2000. She is now a professor in Faculty of Information Science and Engineering, Ningbo University, China. Her research interests include image/video coding and video perception.

Yun Zhang received his BS and MS in electrical engineering from Ningbo University, Ningbo, China, in 2004 and 2007, respectively, and PhD in computer science from Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China, in 2010. From 2009 to 2010, he was a visiting scholar with the Department of Computer Science, City University of Hong Kong, Kowloon, Hong Kong. In 2010, he joined the Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, as an assistant researcher. His research interests are multiview video coding, video object segmentation, and content-based video processing.

Downloaded From: http://opticalengineering.spiedigitallibrary.org/ on 06/16/2015 Terms of Use: http://spiedl.org/terms