Efficient Synthesis-Based Depth Map Coding in AVC-Compatible 3D Video Coding

Jin Young Lee and Hyun Wook Park, *Senior Member, IEEE*

***Abstract*— A multiview video plus depth format was introduced to support 3D depth perception and arbitrary view generation. To achieve high coding performance in AVC-compatible 3D video coding (3D-AVC), we herein propose an efficient depth map coding method that includes synthesis-based depth residual coding and filtering. The synthesis-based depth residual coding method adaptively encodes a residual by predicting synthesis distortion from spatial complexity of the corresponding texture, in a nonnormative way. In order to compensate for possible error from the proposed coding method, synthesis-based depth filtering is additionally conducted on the reconstructed depth map. Experimental results demonstrate that the proposed method reduces the bit rates by 4.4% and 4.8%, compared with the original method for the decoded and synthesized PSNRs, respectively, while satisfying the compatibility with the current 3D-AVC.**

***Index Terms*— AVC-compatible 3D video coding (3D-AVC), depth filtering, depth map, residual coding, synthesis distortion.**

I. Introduction

RECENT developments of multiview video technologies have enabled the next generation broadcasting service for 3D video [1]. However, 3D video service requires a large transmission bandwidth and storage space for multiview data. One promising solution to reduce the amount of data is to use the multiview video plus depth (MVD) format as the 3D video format. The MVD format consists of a texture image and its corresponding depth map. Fig. 1 shows the MVD format of a *Newspaper* sequence. The depth map represents the distance between an object and a camera, which is necessary for rendering the virtual views generated by a depth image-based rendering technique [2].

Many 3D video coding methods based on the MVD format have recently been introduced. For instance, texture motion vectors were used as predictors for accurate motion vector prediction in depth map coding [3]. A synthesis-based depth mode decision method was employed to improve the virtual-view image instead of using the reconstructed depth map, by which a view synthesis distortion (VSD) metric was introduced [4].

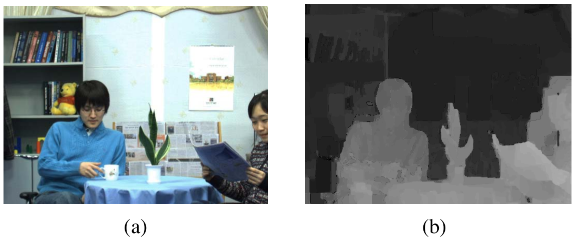


Fig. 1. Example of MVD format including (a) texture and (b) depth images.

Through the analysis of an impact of depth map coding error on the synthesized virtual views, a new Lagrangian multiplier for mode decision [5] and a model-based joint bit allocation method between texture and depth images [6] were introduced. A depth block skip (DBS) method was proposed, which utilized temporal and inter-view correlations of already coded texture images [7]. In addition, many techniques have been developed to improve the coding performance, such as depth intra-skip mode [8] and view synthesis prediction (VSP) [9]–[11]. Since a sharp edge in a depth image plays an important role in view synthesis, in-loop depth filtering was proposed to preserve the edge [12]. An adaptive loop filtering (ALF) method turned on or off the filtering according to edge information from the reconstructed depth map, and found the optimum parameters through a quick search [13]. A Wiener filter was performed on virtual views to improve the quality of the synthesized virtual views [14].

The Joint Collaborative Team on 3D Video Coding Extension Development (JCT-3V) was established by the ITU-T Video Coding Experts Group and ISO/IEC Moving Picture Experts Group to develop a 3D video coding standard. JCT-3V has developed several 3D video coding standards based on Multiview Video Coding (MVC) [15], [16], H.264/AVC [16], and High Efficiency Video Coding (HEVC) [17]. First, MVC-compatible extension including depth (MVC + D) enables 3D enhancements while maintaining MVC stereo compatibility [16], [18]. Next, multiview extension of HEVC is allowed only to modify the high-level syntax of HEVC [19], similar to the MVC extension of H.264/AVC. Finally, AVC-compatible 3D video coding (3D-AVC) and HEVC-compatible 3D video coding (3D-HEVC) provide high coding efficiency by allowing block-level changes in H.264/AVC and HEVC, respectively.

3D-AVC achieves a bit rate reduction of about 20%, compared with MVC + D, by developing texture image coding tools [20], [21], such as depth-based motion vector prediction,

Manuscript received December 22, 2014; revised April 2, 2015 and May 8, 2015; accepted May 26, 2015. Date of publication June 4, 2015; date of current version June 2, 2016. This paper was recommended by Associate Editor M. Hannuksela.

J. Y. Lee is with Samsung Electronics, Suwon 442-742, Korea, and also with the Department of Electrical Engineering, Korea Advanced Institute of Science and Technology, Daejeon 305-701, Korea (e-mail: jinyoung79.lee@gmail.com).

H. W. Park is with the Department of Electrical Engineering, Korea Advanced Institute of Science and Technology, Daejeon 305-701, Korea (e-mail: hwpark@kaist.ac.kr).

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

Digital Object Identifier 10.1109/TCSVT.2015.2441491

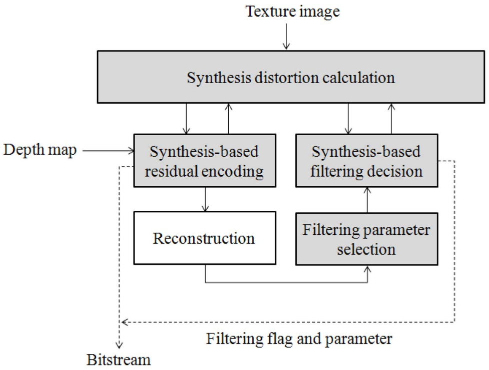

Texture image
Synthesis distortion calculation
Depth map
Synthesis-based residual encoding
Synthesis-based filtering decision
Reconstruction
Filtering parameter selection
Filtering flag and parameter
Bitstream

Fig. 2. Proposed synthesis-based depth map coding method.

block-based VSP, and adaptive luminance compensation, all of which are normative and require block-level changes. For the depth map coding, only the nonnormative tool such as the VSD-based mode decision [4] and the slice-level tools such as depth-range-based weighted prediction [21], nonlinear depth representation [22], and slice header prediction [23] were adopted into 3D-AVC, so the coding performance of the depth map was relatively less improved in comparison with that of the texture image. In this paper, we propose a synthesis-based depth map coding method. The main contribution of this paper is to improve further the depth coding efficiency without block-level changes at a decoder, for compatibility with the current 3D-AVC. Since the 3D-AVC decoder has recently been standardized and is ready to be used in the 3D industry, it is very meaningful to improve the coding performance in a nonnormative way.

The proposed method consists of synthesis-based depth residual coding and filtering. The synthesis-based depth residual coding method adaptively compresses a residual signal generated from a difference between original and predicted depth values at an encoder. The residual signal is analyzed and classified to determine whether or not it is important for the improvement of the multiview image quality. The synthesis-based depth filtering method is proposed for use in post-processing in order to compensate for possible error from the proposed depth residual coding method. Experimental results demonstrate that the proposed method obtains higher coding performance and better subjective 3D quality than does the original 3D-AVC.

This paper is organized as follows. In Section II, we introduce the synthesis-based depth residual coding method using the spatial complexity of the corresponding texture and the synthesis-based depth filtering method. In Section III, the experimental results are provided for the coding performance and the 3D subjective quality assessment of the synthesized stereoscopic view pairs. Finally, we conclude this paper in Section IV.

II. Synthesis-Based Depth Map Coding

Fig. 2 shows a flowchart of the proposed synthesis-based depth map coding method. A depth residual from motion compensation is encoded in consideration of the distortion in the synthesized virtual views. The depth map reconstructed using the depth residual is refined by adaptive filtering. In order to indicate whether the filtering is to be applied, the proposed method signals a filtering flag and a related parameter, which are included in a supplemental enhancement information (SEI) message in a coded bit stream.

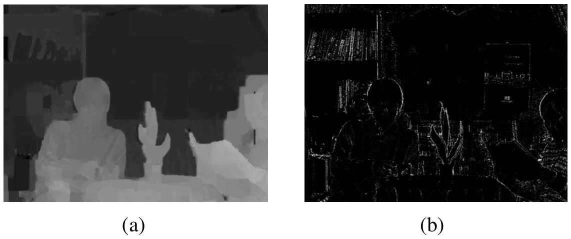


Fig. 3. (a) Distorted depth map and (b) difference between the synthesized virtual-view images generated from the original and distorted depth maps.

A. Synthesis-Based Depth Residual Coding

Since the decoded depth image is not directly displayed to viewers, and is only used for the synthesis of a virtual-view image, the distortion of the virtual-view images is much more important than the depth distortion itself. Sometimes, some amount of depth distortion may not affect the synthesized multiview image quality. As an example, a distorted depth map was generated by adding random distortion, ranging from -3 to $+3$, to an original depth map in a *Newspaper* sequence. Fig. 3 shows the distorted depth map and absolute difference between the virtual-view images synthesized from the original and the distorted depth maps, respectively. The results demonstrate that the differences are hard to observe in homogeneous areas. On the other hand, the distortions in object boundaries and complex regions are significant. This means that depth distortion does not directly represent the synthesis distortion, that is, the distortion in the synthesized virtual-view images. The synthesis distortion is strongly related to both the spatial complexity and the depth distortion. Based on these observations, the proposed method neglects depth residuals that do not significantly influence the virtual-view image by measuring the synthesis-based prediction distortion, and only encodes the important residual that significantly influences the virtual-view image.

At a pixel position (x, y) within a certain macroblock, the depth residual $R(x, y)$ is computed as

$$R(x, y) = D(x, y) - D'(x, y) \qquad (1)$$

where $D(x, y)$ and $D'(x, y)$ are the original depth map and the predicted one, respectively. The depth residual is the depth prediction error that may distort the synthesized virtual-view image (called synthesis distortion). For example, when the virtual view corresponds to the left side of the current view, we define the synthesis-based prediction distortion, $\mathrm{SPD}_L(x, y)$, to measure the prediction error of the synthesized virtual-view

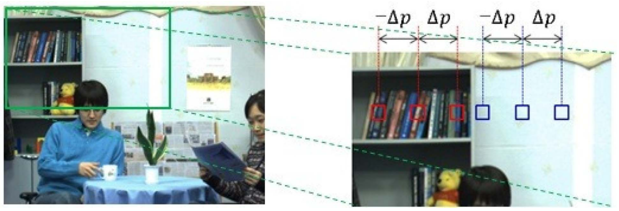


Fig. 4. Influence of the translational position error on synthesis distortion in the complex (red squares) and homogeneous regions (blue squares).

image as

$$\begin{aligned}\mathrm{SPD}_L(x, y) &\\ &= |V(x, y) - V'(x, y)| \\ &= |F_w(T(x, y), D(x, y)) - F_w(T(x, y), D'(x, y))| \\ &= |T(x, y) - T(x - \Delta p(x, y), y)| \qquad (2)\end{aligned}$$

where $V(x, y)$ represents a texture image of a virtual view synthesized from an original texture image $T(x, y)$ and the original depth map $D(x, y)$ through a warping function F_w, and $V'(x, y)$ is a synthesized image from the predicted depth map $D'(x, y)$. Here, Δp represents a translational position error generated from the depth error between $D(x, y)$ and $D'(x, y)$. Under the assumption of a 1D parallel camera setting, all captured views have the same scenes with horizontal disparities, so Δp has only horizontal translation. The translational position error can be simply represented as follows [24]:

$$\Delta p(x, y) = \alpha \cdot R(x, y) \qquad (3)$$

where α is a proportional coefficient determined from the camera parameters as

$$\alpha = \frac{f \cdot L}{255} \cdot \left(\frac{1}{Z_{\mathrm{Near}}} - \frac{1}{Z_{\mathrm{Far}}}\right) \qquad (4)$$

where f and L denote the focal length and the baseline distance between two horizontally adjacent cameras, respectively, and Z_{Near} and Z_{Far} denote the nearest and farthest depth values, respectively. If a virtual view of the right side of the current view is synthesized, the synthesis distortion can be represented as

$$\mathrm{SPD}_R(x, y) = |T(x, y) - T(x + \Delta p(x, y), y)| \qquad (5)$$

Fig. 4 shows the influence of the translational position error in the complex and homogeneous regions of a *Newspaper* sequence, respectively. The complex and homogeneous regions are marked with red and blue squares in Fig. 4, respectively. If a pixel belongs to a complex region such as the object boundaries, the synthesis distortion will become large in proportion to the position error. However, if a pixel is located in the homogeneous region, the synthesis distortion will be very small regardless of the position error and it might even be unnoticeable. Therefore, both the position error and the spatial complexity of the texture image should be considered together for the estimation of the synthesis distortion.

The spatial complexity, which indicates whether the associated region to be synthesized is complex or homogenous, is determined by comparison of the pixels ranging from $(x - \Delta p, y)$ to $(x + \Delta p, y)$ in the proposed method. Therefore, (2) and (5) are further extended to consider the spatial complexity as

$$\begin{aligned}\mathrm{SPD}(x, y) = \frac{1}{\Delta p(x, y)} \sum_{k=1}^{\Delta p(x,y)} (&|T(x, y) - T(x + k, y)| \\ &+ |T(x, y) - T(x - k, y)|)\end{aligned} \qquad (6)$$

A decoder synthesizes the virtual views from the reconstructed texture image and depth map. Therefore, an encoder can use the reconstructed texture image $T'(x, y)$ to estimate the synthesis distortion, if the texture image is encoded before the corresponding depth map. Then, $T(x, y)$ in (6) can be replaced with $T'(x, y)$, when the reconstructed texture image is available.

According to the synthesis distortion, the proposed method adaptively encodes the depth residual. If the synthesis distortion calculated from (6) is equal to or less than a threshold, the proposed method expects that the influence of the depth error is not critical. Under this condition, the depth residual is not encoded [i.e., $R(x, y) = 0$]. On the other hand, if the synthesis distortion is greater than the threshold, it means that it is highly probable that the region is complex, so the residual is encoded. To consider both the coding performance and the subjective 3D quality, the proposed method defines the threshold value using a just-noticeable-difference (JND) model [25]. JND represents the maximum luminance change that can be perceived by the human visual system. For example, subjective evaluation conducted in [25] revealed that human eyes were relatively more sensitive to change at a medium level of luminance than to change of luminance in dark or bright regions. Based on the subjective evaluation, the visibility threshold of the JND model was defined as

$$\mathrm{JND}(k) = \begin{cases} 17 \cdot \left(1 - \left(\frac{k}{127}\right)^{0.5}\right) + 3, & 0 \le k < 128 \\ \frac{3}{128} \cdot (k - 127) + 3, & 128 \le k < 256 \end{cases} \qquad (7)$$

where k is a luminance value between 0 and 255. The JND model was extensively studied and demonstrated its efficiency in the perceptual video coding [26]–[29]. For example, it was used to suppress the residuals in a pixel domain [26]–[28] and coefficients in a transform domain [29]. Recently, the study of the JND model has been extended into 3D video coding to suppress the depth details that were not perceivable by viewers [30], [31]. Based on the analysis of these prior works, the proposed method can suppress the depth residual during encoding, through the following process:

$$R(x, y) = 0, \quad \text{if } \mathrm{SPD}(x, y) \le \mathrm{JND}(T'(x, y)) \qquad (8)$$

In (6), all the texture pixels between $(x - \Delta p, y)$ and $(x + \Delta p, y)$ need to be processed to calculate the synthesis distortion. This results in a significant increase in the computational complexity, especially for a large translational position error Δp. In order to minimize the computational complexity, the proposed method sets the maximum allowable position error to one. Therefore, the proposed suppression of the depth residual is performed only when the position error is equal to or less than one, whereas the depth residual is encoded without suppression when the position error is greater than one. Finally, the depth residual can be discarded, when any of the following conditions are satisfied:

1) $\Delta p = 0$;
2) $\Delta p = 1$ and $|T'(x, y) - T'(x + 1, y)| + |T'(x, y) - T'(x - 1, y)| \leq \mathrm{JND}(T'(x, y))$.

In addition, the depth residual is always encoded without suppression for the small block size modes of P8×8, Intra8×8, and Intra4×4, because the small block size modes are generally selected for complex regions rather than homogenous ones.

After the depth residuals are investigated by the proposed method, the remaining residuals are transformed into a Discrete Cosine Transform (DCT) domain and then quantized. Finally, these quantized DCT coefficients are encoded by a context-adaptive binary arithmetic coding. The decoding process is performed in the reverse order of the encoding process above.

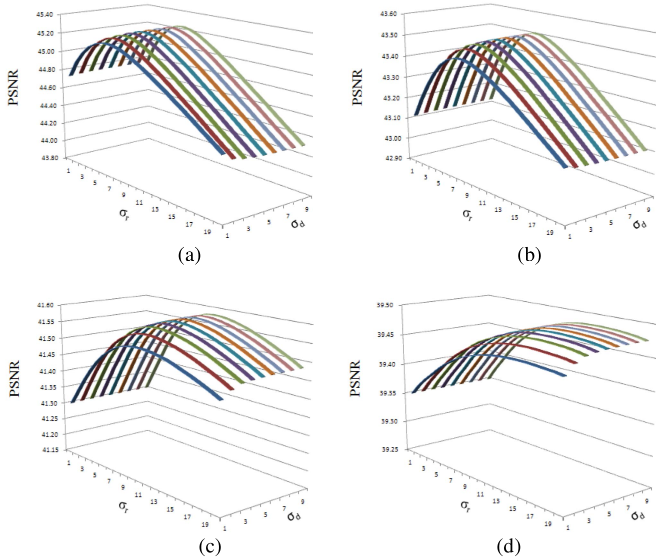
B. Synthesis-Based Depth Filtering

When an image is encoded, block and ringing artifacts are usually generated due to data quantization. Although a deblocking filtering is included in H.264/AVC to reduce these artifacts, it is not appropriate to maintain a strong edge in the depth map. Since depth distortion influences the translational position error in (3), it can leave viewers of 3D displays in severely annoying discomfort. To reduce quantization artifacts in the depth map, and to compensate for the depth error from the proposed method, a synthesis-based bilateral filtering is applied to the reconstructed depth map.

Bilateral filtering, which preserves edges and reduces noise, is very popular in the denoising field [32]. It provides a weighted average of pixels within a local neighborhood. The weights depend on both spatial closeness and intensity difference, and a filtered depth map $\tilde{D}(x, y)$ is calculated as

$$\tilde{D}(x, y) = \frac{\sum_{(m,n)\in N(x,y)} \hat{D}(m, n) \cdot e^{\frac{-\|(m,n)-(x,y)\|^2}{2\cdot\sigma_d^2}} \cdot e^{\frac{-\|\hat{D}(m,n)-\hat{D}(x,y)\|^2}{2\cdot\sigma_r^2}}}{\sum_{(m,n)\in N(x,y)} e^{\frac{-\|(m,n)-(x,y)\|^2}{2\cdot\sigma_d^2}} \cdot e^{\frac{-\|\hat{D}(m,n)-\hat{D}(x,y)\|^2}{2\cdot\sigma_r^2}}} \tag{9}$$

where $\hat{D}(x, y)$ is the reconstructed depth map, and σ_d and σ_r denote the domain and range parameters, respectively, controlling the falloff of the weights. Here, (m, n) belongs to a neighboring pixel set $N(x, y)$. In the bilateral filter, the domain and range parameters are important, because they specify the behavior of the filter. An empirical study in the denoising field optimized the parameter values [33]. Recently, there have been some studies of depth map filtering in 3D-AVC [12], [13]. The optimum parameters are obtained through exhaustive search. In order to avoid excessive search, the proposed method determines the optimum parameters as a function of the depth distortion, based on an analysis of the relation between the parameters, depth distortion, and synthesis performance.



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

Fig. 5. PSNR of the synthesized virtual-view image with respect to the parameter values when the bilateral filter is applied to the reconstructed depth map at four different QPs. (a) 26, (b) 31, (c) 36, and (d) 41.

TABLE I
MAXIMUM AND MINIMUM PSNRS FOR THE DOMAIN PARAMETER RANGING FROM 1 TO 9, WHEN THE RANGE PARAMETER IS EQUAL TO 7 AND 14, RESPECTIVELY

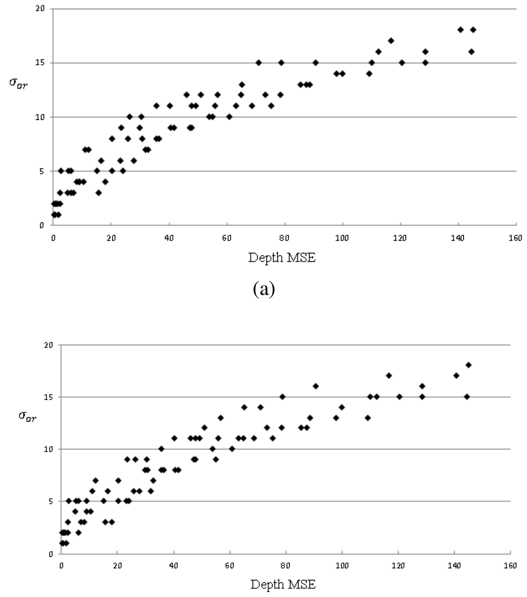
σ_r	7				14			
QP	26	31	36	41	26	31	36	41
Max PSNR	45.22	43.51	41.55	39.45	44.83	43.33	41.56	39.48
Min PSNR	45.18	43.44	41.50	39.43	44.77	43.32	41.51	39.45
Difference	0.04	0.07	0.05	0.02	0.06	0.01	0.05	0.03

To investigate how much the domain and range parameters affect the quality of the synthesized virtual-view image, bilateral filtering with various parameter values is conducted. Since the quality of the synthesized virtual view is more important than the depth map quality itself, the PSNR of the virtual-view image synthesized with the filtered depth map is analyzed. Fig. 5 shows the PSNR of the synthesized virtual-view image when the bilateral filter is applied to the reconstructed depth map, with the four different quantization parameters (QPs) of 26, 31, 36, and 41, for a *GT_Fly* sequence. The results reveal that the PSNR is very sensitive to the range parameter. Table I shows that the PSNR difference between the maximum and minimum PSNRs obtained from various values of the domain parameter is less than 0.07 dB. Table II shows that the PSNR difference from various values of the range parameter is much larger than that of the domain parameter. Based on these observations, the term controlled by the domain

TABLE II

MAXIMUM AND MINIMUM PSNRs FOR THE RANGE PARAMETER RANGING FROM 1 TO 20, WHEN THE DOMAIN PARAMETER IS EQUAL TO 3 AND 8, RESPECTIVELY

σ_d	3				8			
QP	26	31	36	41	26	31	36	41
Max PSNR	45.23	43.49	41.57	39.48	45.24	43.50	41.58	39.48
Min PSNR	44.34	43.11	41.30	39.35	44.31	43.10	41.30	39.35
Difference	0.89	0.38	0.27	0.13	0.93	0.40	0.28	0.13



(a)

(b)

Fig. 6. Relation between the optimum range parameter and depth MSE when the domain parameter is equal to (a) 3 and (b) 8.

parameter can be ignored in (9), because the PSNR is not sensitive to the domain parameter.

As shown in Fig. 5, the optimum value of the range parameter is different according to QP. As QP increases, the optimum value tends to have a large value. In general, when QP is high, the mean squared error (MSE) between the original and reconstructed depth maps is large. In order to analyze the relation between the parameter and the depth MSE, the optimum range parameter (σ_{or}) that minimizes synthesis distortion is estimated when the domain parameter is equal to 3 and 8, as shown in Fig. 6. From the relationship indicated in Fig. 6, the optimum value of the range filter parameter can be approximated to a function of the depth MSE as

$$\sigma_{\text{or}} = \alpha \cdot \text{MSE}^{\beta} \tag{10}$$

For the domain parameter values of 3 and 8, α and β are approximated to 1.5 and 0.5, respectively. Since the domain filter parameter negligibly affects the quality of the synthesized virtual-view image, (9) can be simplified as

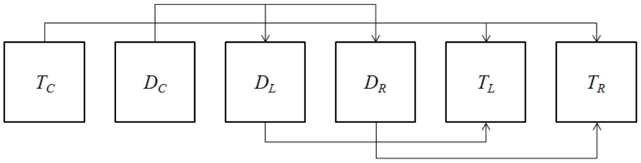


Fig. 7. P-I-P configuration in the depth-first coding order.

$$\tilde{D}(x,y) = \frac{\displaystyle\sum_{(m,n)\in N(x,y)} \hat{D}(m,n) \cdot e^{\frac{-\|\hat{D}(m,n)-\hat{D}(x,y)\|^2}{2\cdot\sigma_{\text{or}}^2}}}{\displaystyle\sum_{(m,n)\in N(x,y)} e^{\frac{-\|\hat{D}(m,n)-\hat{D}(x,y)\|^2}{2\cdot\sigma_{\text{or}}^2}}}. \tag{11}$$

Finally, if the synthesis distortion from the filtered depth map is less than that from the reconstructed depth map, the filtering flag is set to one. Otherwise, the flag is set to zero. When the flag is equal to one, an SEI message including the fixed-length coded range parameter of (10) is transmitted to a decoder.

III. EXPERIMENTAL RESULTS

Experiments were performed using the common test condition (CTC) [34] used in JCT-3V. Various sequences with a resolution of 1024×768, or of 1920×1088, were used for the experiments. As specified in CTC [34], both the full and half depth resolutions were considered in the experiment. The VSD model [4] was used for the depth mode decision, instead of the rate-distortion optimization model [35]. The Bjontegaard delta rate (BD-rate) [36] was measured for the performance evaluation at four different QPs (26, 31, 36, and 41). The software 3D-ATM 10.0 [37] and VSRS-1D-Fast [38] were used as references for the coding and rendering, respectively. For an objective comparison of the 3D video coding, we measured the total bit rate of the texture images and depth maps. Two types of PSNR were measured: 1) a decoded PSNR, which was an average PSNR of the decoded texture image and 2) a synthesized PSNR, which was an average PSNR between the synthesized virtual-view images from the original and decoded depth maps. As specified in CTC [34], the view prediction structure was P-I-P and the coding order was depth-first, as shown in Fig. 7. For example, let us assume that T_C, D_C, T_L, D_L, T_R, and D_R represent the texture images and depth maps at the center, left, and right views, respectively. First, T_C and D_C were encoded as the I view. Next, D_L and D_R were encoded as the P view, where the already coded D_C was used as the reference in the inter-view prediction. Finally, T_L and T_R were also encoded as the P view. They could be encoded using the already coded T_C as the reference in the inter-view prediction. The depth-based texture coding tools encoded T_L and T_R using the coded D_L and D_R [20], [21]. The arrow in Fig. 7 represents the prediction direction from the reference to the target to be coded. In addition, the subjective quality assessment was performed using the double-stimulus continuous quality-scale (DSCQS) test method in ITU-R BT.500-11 [39].

TABLE III
BD-Rate Saving Compared With the Original Method From (a) DBS, (b) ALF, (c) Combination of DBS and ALF, (d) Synthesis-Based Depth Residual Coding, (e) Synthesis-Based Depth Filtering, and (f) Combination of the Synthesis-Based Depth Residual Coding and Filtering in the Full Depth Resolution

Sequence	Type	(a)	(b)	(c)	(d)	(e)	(f)
Balloons	Dec.	1.0%	-0.1%	0.9%	8.1%	0.0%	8.1%
	Syn.	0.7%	-0.6%	0.0%	7.3%	0.1%	7.4%
Kendo	Dec.	2.3%	0.0%	2.2%	14.3%	0.0%	14.3%
	Syn.	1.5%	-0.7%	0.6%	12.7%	0.3%	13.2%
Newspaper	Dec.	1.3%	-0.4%	0.9%	4.4%	0.0%	4.4%
	Syn.	0.8%	-1.2%	-0.3%	3.2%	1.0%	4.1%
GT_Fly	Dec.	0.0%	0.1%	0.1%	2.6%	0.0%	2.6%
	Syn.	0.0%	0.6%	0.6%	2.4%	1.0%	3.2%
Poznan_Hall2	Dec.	2.2%	0.0%	2.1%	1.5%	0.0%	1.5%
	Syn.	1.5%	-0.3%	1.2%	0.9%	0.2%	1.1%
Poznan_Street	Dec.	1.7%	-0.3%	1.4%	1.7%	0.0%	1.7%
	Syn.	1.4%	-0.5%	1.0%	1.4%	0.2%	1.7%
Undo_Dancer	Dec.	0.4%	0.9%	1.1%	0.1%	0.0%	0.1%
	Syn.	-0.1%	2.5%	1.9%	-0.1%	1.5%	1.4%
Shark	Dec.	0.0%	0.6%	0.6%	2.2%	0.0%	2.2%
	Syn.	0.0%	4.9%	4.9%	2.0%	4.3%	6.2%
Average	Dec.	1.1%	0.1%	1.2%	4.4%	0.0%	4.4%
	Syn.	0.7%	0.6%	1.2%	3.7%	1.1%	4.8%

TABLE IV
BD-Rate Saving Compared With the Original Method From (a) DBS, (b) ALF, (c) Combination of DBS and ALF, (d) Synthesis-Based Depth Residual Coding, (e) Synthesis-Based Depth Filtering, and (f) Combination of the Synthesis-Based Depth Residual Coding and Filtering in the Half Depth Resolution

Sequence	Type	(a)	(b)	(c)	(d)	(e)	(f)
Balloons	Dec.	0.3%	-0.1%	0.2%	2.3%	0.0%	2.3%
	Syn.	0.1%	0.0%	0.2%	1.6%	0.6%	2.2%
Kendo	Dec.	0.8%	-0.2%	0.6%	4.9%	0.0%	4.9%
	Syn.	0.7%	-0.3%	0.2%	3.7%	0.7%	4.4%
Newspaper	Dec.	0.3%	-0.1%	0.2%	1.2%	0.0%	1.2%
	Syn.	0.3%	0.0%	0.3%	0.0%	1.8%	1.8%
GT_Fly	Dec.	0.0%	-0.1%	-0.1%	0.5%	0.0%	0.5%
	Syn.	0.0%	-0.4%	-0.4%	0.1%	0.5%	0.6%
Poznan_Hall2	Dec.	0.5%	-0.2%	0.4%	0.5%	0.0%	0.6%
	Syn.	0.2%	0.1%	0.3%	0.0%	0.7%	0.7%
Poznan_Street	Dec.	0.4%	-0.2%	0.3%	0.3%	0.0%	0.3%
	Syn.	0.3%	0.0%	0.4%	0.2%	0.5%	0.7%
Undo_Dancer	Dec.	0.1%	0.0%	0.1%	0.0%	0.0%	0.0%
	Syn.	-0.1%	4.4%	4.2%	-0.1%	3.6%	3.5%
Shark	Dec.	0.0%	0.1%	0.1%	0.8%	0.0%	0.8%
	Syn.	0.0%	2.4%	2.4%	0.4%	3.0%	3.3%
Average	Dec.	0.3%	-0.1%	0.2%	1.3%	0.0%	1.3%
	Syn.	0.2%	0.8%	1.0%	0.7%	1.4%	2.2%

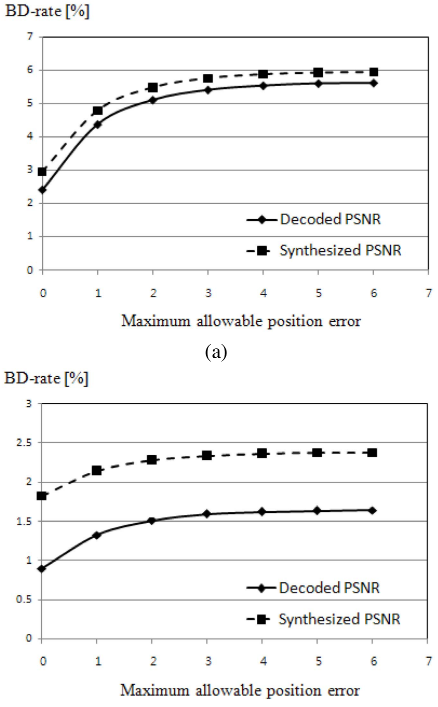
A. Coding Performance Evaluation

In order to clarify the coding performance of the proposed synthesis-based depth coding method, we implemented DBS [7] and ALF [13]. Tables III and IV show the overall coding performance improvements of DBS, ALF, the combination of DBS and ALF, the synthesis-based depth residual coding, the synthesis-based depth filtering, and the combination of the synthesis-based depth residual coding and filtering. These were compared with 3D-AVC (the original method in this paper), when the depth resolution was full and half, respectively. For a fair comparison, both ALF and the proposed synthesis-based depth filtering were performed with a 3×3 window size. In Tables III and IV, a positive BD-rate represents a coding gain and a negative BD-rate indicates a coding loss. Since DBS skips the depth block adaptively according to the correlation of the texture images and ALF is the in-loop filter compensating for the coding error in the reconstructed depth map, the concept and the objective of DBS and ALF are very similar to those of the proposed synthesis-based depth residual coding and filtering method. As shown in Table III, when the depth resolution is full, the combination of the proposed methods saves BD-rates of 4.4% and 4.8%, compared with the original method for the decoded PSNR (Dec.) and the synthesized PSNR (Syn.) on average, respectively. When the depth resolution is half in Table IV, it saves BD-rates of 1.3% and 2.2%, compared with the original method for the decoded and synthesized PSNRs, respectively, on average. It should be noted that the proposed synthesis-based depth filtering does not affect the decoded PSNR because it is performed out of the coding loop. The combination of the proposed methods achieves a higher coding gain than that of DBS and ALF, because DBS focuses on the difference between the consecutive texture images, and the parameters of ALF are determined considering only the depth distortion. However, the proposed method focuses on the minimization of synthesis distortion, which is the most important goal of the depth map coding. In addition, the proposed method does not require block-level changes at a decoder, so it is compatible with 3D-AVC, whereas DBS and ALF are not compatible with 3D-AVC. As mentioned in Section I, the 3D-AVC decoder has recently been finalized and is ready to be used in the 3D application. Therefore, the compatibility with 3D-AVC is one of the main advantages of the proposed method.

The coding performance is higher in the depth map of full resolution than in that of half resolution. Since the data size in the half depth map is reduced four times, compared with the full depth, the depth bit amount can be a small portion of the total coding bit amount. This results in a relatively small coding improvement. In addition, for sequences of *Poznan_Hall2*, *Poznan_Street*, and *Undo_Dancer*, the proposed method achieves a smaller coding gain than for other sequences, as shown in Table III. Table V shows the average depth distortion between the original and

TABLE V
AVERAGE DEPTH DISTORTION

Sequence	QP			
	26	31	36	41
Balloons	49.2	70.0	97.6	152.1
Kendo	139.8	204.0	281.4	351.6
Newspaper	30.6	50.0	93.0	193.7
GT_Fly	26.6	46.0	74.5	126.3
Poznan_Hall2	6.3	6.9	16.0	37.7
Poznan_Street	4.2	6.8	13.2	31.3
Undo_Dancer	2.6	4.9	12.0	30.6
Shark	29.4	43.8	72.2	132.0



(a)

(b)

Fig. 8. Coding performance according to the maximum allowable position error when the depth resolution is (a) full and (b) half.

reconstructed depth maps in 3D-AVC. These sequences have relatively low depth distortion, which means that the prediction is well performed, and that the amount of residual information is small. Therefore, the proposed method cannot achieve much improvement for sequences having little depth distortion. However, for a *Kendo* sequence, the proposed method achieves the highest BD-rate reduction (14.3%), compared with the original method for the decoded PSNR, because the sequence has a large depth residual that could be efficiently compressed by the proposed method.

As mentioned in Section II-A, in order to minimize computational complexity, the proposed synthesis-based residual coding is applied only when the position error is equal to or less than one (i.e., the maximum allowable position error is set to one). Fig. 8 shows the coding performance according to the maximum allowable position error. In Fig. 8, solid and dashed lines indicate the BD-rate improvements compared with the original method for the decoded and synthesized PSNRs, respectively. As the maximum allowable position error becomes large, the coding performance also increases, and it is saturated in the end. The coding gain, in particular, rapidly increases when the maximum allowable position error increases from zero to one. Fig. 9 shows the maximum number of addition and subtraction operations (M_O) in (6), according to the maximum allowable position error. This illustrates that the number of operations is proportional to the maximum allowable position error, so there is a tradeoff between coding performance and computational complexity. If an application does not need real-time computation, it is recommended not to restrict the maximum allowable error to achieve higher coding performance. For example, when an allowed maximum error is large, additional BD-rate savings of about 1% can be achieved in the full depth resolution.

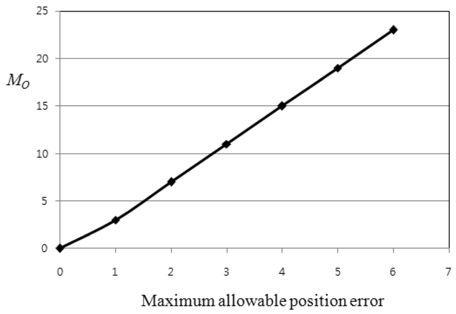


Fig. 9. Maximum number of addition and subtraction operations in (6) according to the maximum allowable position error.

TABLE VI
DEPTH CODING PERFORMANCE OF THE PROPOSED METHOD

Sequence	Resolution	
	Full	Half
Balloons	27.1%	17.9%
Kendo	36.3%	24.8%
Newspaper	15.3%	10.4%
GT_Fly	11.0%	10.7%
Poznan_Hall2	6.8%	4.9%
Poznan_Street	7.7%	4.8%
Undo_Dancer	2.1%	4.1%
Shark	10.1%	10.0%
Average	14.6%	11.0%

Table VI shows the depth coding performance measured using only the depth bit rate and the synthesized PSNR. In general, the overall performance of 3D video coding is evaluated by considering the decoded and synthesized PSNRs over the total bit rate of the texture and depth [34], as shown in Tables III and IV. However, since the proposed method

TABLE VII
PERCENTAGES OF BLOCKS THAT INCLUDE THE NONZERO RESIDUAL

Sequence	QP	Original	Proposed
Balloons	31 41	10.6% 3.3%	7.6% 3.1%
Kendo	31 41	7.2% 2.0%	5.9% 1.9%
Newspaper	31 41	9.8% 4.0%	8.3% 3.2%
GT_Fly	31 41	6.0% 2.9%	5.5% 2.5%
Poznan_Hall2	31 41	1.9% 0.7%	1.4% 0.4%
Poznan_Street	31 41	11.1% 3.2%	9.7% 2.9%
Undo_Dancer	31 41	5.4% 3.4%	4.6% 2.8%
Shark	31 41	11.7% 5.4%	10.8% 4.8%

is applied to the depth coding, and the depth map is only employed to synthesize the intermediate views, it is worth investigating the depth coding performance as additional information. The result shows that the proposed method saves the depth bit rates of 14.6% and 11.0% on average in the full and half resolutions, respectively. The depth coding performance is significantly higher in a *Kendo* sequence, but relatively lower in *Poznan_Hall2*, *Poznan_Street*, and *Undo_Dancer* sequences.

Table VII shows percentages of blocks that include nonzero residual, when QP is equal to 31 and 41 for the full depth map coding. This indicates that the proposed method encodes fewer nonzero residual blocks than does the original method, meaning that many residuals could be changed to zero by the proposed synthesis-based depth residual coding. Fig. 10 shows the original depth maps and their binary maps, indicating the residual coding blocks, where white blocks include the nonzero residual and black blocks have no residual (i.e., all-zero residual).

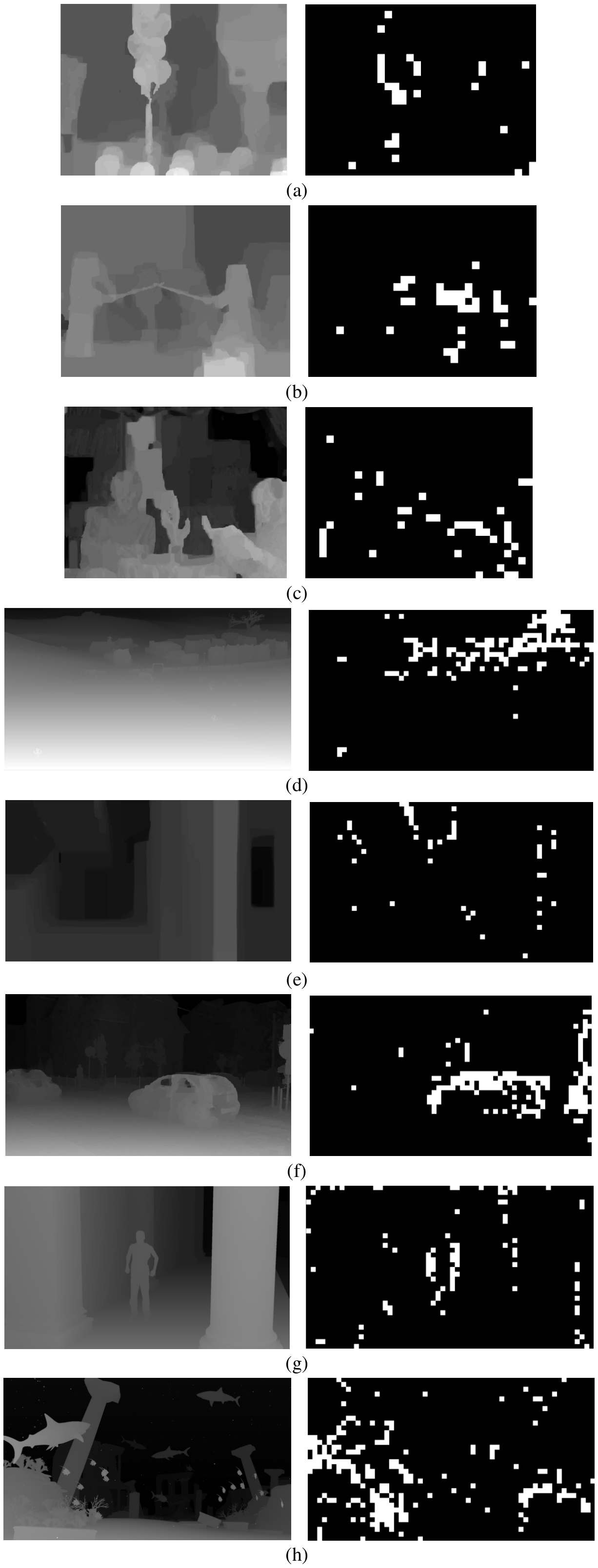


Fig. 10. Original depth maps and their binary maps indicating the blocks that include the nonzero residual (white) and all-zero residual (black) in (a) *Balloons*, (b) *Kendo*, (c) *Newspaper*, (d) *GT_Fly*, (e) *Poznan_Hall2*, (f) *Poznan_Street*, (g) *Undo_Dancer*, and (h) *Shark*.

B. Subjective Quality Assessment

Based on the DSCQS method in ITU-R BT.500-11 [39], a subjective quality evaluation was conducted for the synthesized intermediate views, in order to evaluate the coding performance [34]. Two intermediate views, known as a stereoscopic view pair, were displayed on a Samsung 60-in stereoscopic LED TV. Eighteen professional subjects participated in the experiment. The subjects were consecutively presented with the two stereoscopic view pairs, which were, respectively, rendered using the depth images reconstructed by the original 3D-AVC and the proposed method, to establish their opinion of each. Then, they were asked to rate their overall perceived video quality from 0 (lowest quality) to 100 (highest quality). The presentation order of the videos was randomized. Fig. 11 shows the subjective quality test results for the synthesized stereoscopic view pairs of a *Kendo* sequence, at target bit rates of 0.5 and 1.0 Mbps, and of

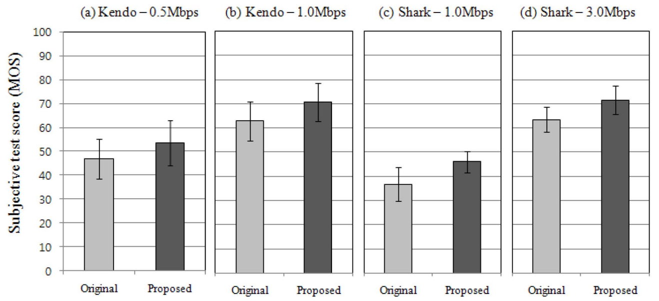
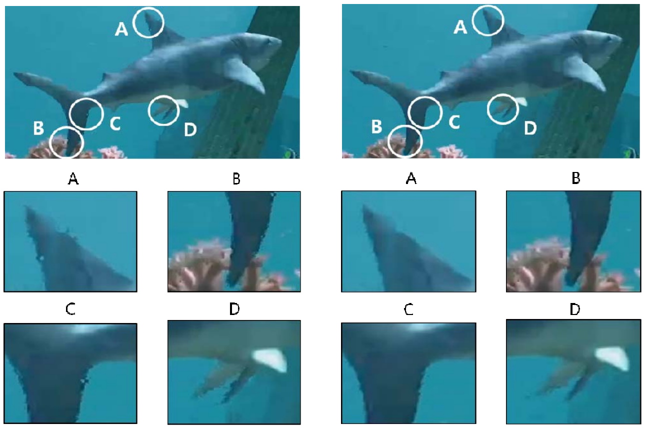


Fig. 11. Subjective quality test results (MOS) at target bit rates of (a) 0.5 and (b) 1.0 Mbps in a *Kendo* sequence, and (c) 1.0 and (d) 3.0 Mbps in a *Shark* sequence. The error bars show 95% confidence interval of the mean.



(a) (b)

Fig. 12. Examples of synthesized images from the (a) original 3D-AVC and (b) proposed method for a *Shark* sequence.

a *Shark* sequence at target bit rates of 1.0 and 3.0 Mbps, respectively. The y-axis indicates the average mean opinion scores (MOS) of the original and proposed methods. Because all the scores of the proposed method were higher than those of the original method, we can conclude that the proposed method achieves better subjective quality than the original 3D-AVC. Fig. 12 shows examples of the synthesized images from the original and proposed methods in a *Shark* sequence. As shown in the white circles, the proposed method provides better quality than the original 3D-AVC.

IV. Conclusion

This paper introduced a synthesis-based depth map coding method to improve the depth coding performance. The synthesis-based depth residual coding method determined how much the depth residual influenced the synthesized virtual-view image quality. In consideration of the synthesis distortion based on the spatial complexity of the corresponding texture, the proposed method discarded the depth residual when its influence was small. In addition, the synthesis-based depth filtering was proposed to improve the synthesized image quality. The filter parameter was directly determined from the error between the original and reconstructed depth maps. Depth filtering was applied to the reconstructed depth map according to synthesis distortion. The experimental results demonstrated that the proposed method was significantly more efficient in terms of coding performance and subjective 3D quality. Furthermore, the proposed method is compatible with the current 3D-AVC.

The 3D-HEVC was recently finalized in JCT-3V. The concept of the proposed method could also be utilized using 3D-HEVC. However, 3D-AVC and 3D-HEVC are based on different video codecs. In fact, the residual coding in 3D-HEVC is very different from that in 3D-AVC. For instance, 3D-HEVC uses residual quad-tree coding and transform skip, which are not used in 3D-AVC. Therefore, in future work, we will extend the proposed method for use with 3D-HEVC.

References

[1] A. Smolic *et al.*, "3D video and free viewpoint video—Technologies, applications and MPEG standards," in *Proc. IEEE Int. Conf. Multimedia Expo.*, Jul. 2006, pp. 2161–2164.

[2] C. Fehn, "Depth-image-based rendering (DIBR), compression, and transmission for a new approach on 3D-TV," *Proc. SPIE, Stereosc. Displays Virt. Reality Syst. XI*, vol. 5291, p. 93, May 2004.

[3] H. Oh and Y.-S. Ho, "H.264-based depth map sequence coding using motion information of corresponding texture video," in *Advances in Image and Video Technology* (Lecture Notes in Computer Science), vol. 4319. Berlin, Germany: Springer-Verlag, Dec. 2006.

[4] B. T. Oh, J. Lee, and D.-S. Park, "Depth map coding based on synthesized view distortion function," *IEEE J. Sel. Topics Signal Process.*, vol. 5, no. 7, pp. 1344–1352, Nov. 2011.

[5] H. Yuan, S. Kwong, J. Liu, and J. Sun, "A novel distortion model and Lagrangian multiplier for depth maps coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 24, no. 3, pp. 443–451, Mar. 2014.

[6] H. Yuan, Y. Chang, J. Huo, F. Yang, and Z. Lu, "Model-based joint bit allocation between texture videos and depth maps for 3-D video coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 21, no. 4, pp. 485–497, Apr. 2011.

[7] J. Y. Lee, H.-C. Wey, and 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. Circuits Syst. Video Technol.*, vol. 21, no. 12, pp. 1859–1868, Dec. 2011.

[8] K.-J. Oh, J. Lee, and D.-S. Park, "High priority intra coding method for depth video coding," in *Proc. Picture Coding Symp.*, May 2012, pp. 45–48.

[9] S. Shimizu, M. Kitahara, H. Kimata, K. Kamikura, and Y. Yashima, "View scalable multiview video coding using 3-D warping with depth map," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 11, pp. 1485–1495, Nov. 2007.

[10] S.-T. Na, K.-J. Oh, C. Lee, and Y.-S. Ho, "Multi-view depth video coding using depth view synthesis," in *Proc. IEEE Int. Symp. Circuits Syst.*, May 2008, pp. 1400–1403.

[11] C. Lee and Y.-S. Ho, "A framework of 3D video coding using view synthesis prediction," in *Proc. Picture Coding Symp.*, May 2012, pp. 9–12.

[12] K.-J. Oh, A. Vetro, and Y.-S. Ho, "Depth coding using a boundary reconstruction filter for 3-D video systems," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 21, no. 3, pp. 350–359, Mar. 2011.

[13] I. Lim, H. Wey, and J. Lee, "Region-based adaptive bilateral filter in depth map coding," in *Proc. 18th IEEE Int. Conf. Image Process.*, Sep. 2011, pp. 121–124.

[14] H. Yuan, J. Liu, H. Xu, Z. Li, and W. Liu, "Coding distortion elimination of virtual view synthesis for 3D video system: Theoretical analyses and implementation," *IEEE Trans. Broadcast.*, vol. 58, no. 4, pp. 558–568, Dec. 2012.

[15] A. Vetro, T. Wiegand, and G. J. Sullivan, "Overview of the stereo and multiview video coding extensions of the H.264/MPEG-4 AVC standard," *Proc. IEEE*, vol. 99, no. 4, pp. 626–642, Apr. 2011.

[16] *Advanced Video Coding for Generic Audiovisual Services*, document Rec. ITU-T H.264, Feb. 2014.

[17] *High Efficiency Video Coding*, document Rec. ITU-T H.265, Apr. 2013.

[18] Y. Chen, M. M. Hannuksela, T. Suzuki, and S. Hattori, "Overview of the MVC + D 3D video coding standard," *J. Vis. Commun. Image Represent.*, vol. 25, no. 4, pp. 679–688, May 2014.

[19] G. J. Sullivan, J. M. Boyce, Y. Chen, J.-R. Ohm, C. A. Segall, and A. Vetro, "Standardized extensions of High Efficiency Video Coding (HEVC)," *IEEE J. Sel. Topics Signal Process.*, vol. 7, no. 6, pp. 1001–1016, Dec. 2013.

[20] J. Y. Lee *et al.*, "Depth-based texture coding in AVC-compatible 3D video coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 25, no. 8, pp. 1347–1361, Aug. 2015.

[21] M. M. Hannuksela *et al.*, "Multiview-video-plus-depth coding based on the advanced video coding standard," *IEEE Trans. Image Process.*, vol. 22, no. 9, pp. 3449–3458, Sep. 2013.

[22] O. Stankiewicz, K. Wegner, and M. Domanski, "Nonlinear depth representation for 3D video coding," in *Proc. 20th IEEE Int. Conf. Image Process.*, Sep. 2013, pp. 1752–1756.

[23] W. Yang, J. Yanhuo, Y. Chang, Y. Chen, and L. Zhang, "Slice header prediction for depth maps bit reduction," in *Proc. IEEE 5th Int. Congr. Image Signal Process.*, Oct. 2012, pp. 58–62.

[24] W.-S. Kim, A. Ortega, P. Lai, D. Tian, and C. Gomila, "Depth map coding with distortion estimation of rendered view," *Proc. SPIE, Vis. Inf. Process. Commun.*, vol. 7543, p. 75430B, Jan. 2010.

[25] C.-H. Chou and Y.-C. Li, "A perceptually tuned subband image coder based on the measure of just-noticeable-distortion profile," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 5, no. 6, pp. 467–476, Dec. 1995.

[26] X. Yang, W. Lin, Z. Lu, E. Ong, and S. Yao, "Motion-compensated residue preprocessing in video coding based on just-noticeable-distortion profile," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 15, no. 6, pp. 742–752, Jun. 2005.

[27] X. K. Yang, W. S. Ling, Z. K. Lu, E. P. Ong, and S. S. Yao, "Just noticeable distortion model and its applications in video coding," *Signal Process., Image Commun.*, vol. 20, no. 7, pp. 662–680, Aug. 2005.

[28] L. Zhang, Q. Peng, Q.-H. Wang, and X. Wu, "Stereoscopic perceptual video coding based on just-noticeable-distortion profile," *IEEE Trans. Broadcast.*, vol. 57, no. 2, pp. 572–581, Jun. 2011.

[29] Z. Chen and C. Guillemot, "Perceptually-friendly H.264/AVC video coding based on foveated just-noticeable-distortion model," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 20, no. 6, pp. 806–819, Jun. 2010.

[30] D. V. S. X. De Silva, E. Ekmekcioglu, W. A. C. Fernando, and S. T. Worrall, "Display dependent preprocessing of depth maps based on just noticeable depth difference modeling," *IEEE J. Sel. Topics Signal Process.*, vol. 5, no. 2, pp. 335–351, Apr. 2011.

[31] 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. 17th IEEE Int. Conf. Image Process.*, Sep. 2010, pp. 4013–4016.

[32] C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images," in *Proc. IEEE 6th Int. Conf. Comput. Vis.*, Jan. 1998, pp. 839–846.

[33] M. Zhang and B. K. Gunturk, "Multiresolution bilateral filtering for image denoising," *IEEE Trans. Image Process.*, vol. 17, no. 12, pp. 2324–2333, Dec. 2008.

[34] D. Rusanovskyy, K. Müller, and A. Vetro, *Common Test Conditions of 3DV Core Experiments*, document Rec. JCT3V-E1F100, Aug. 2013.

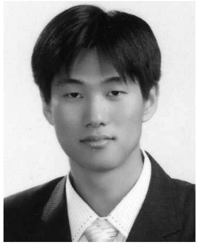
[35] A. Ortega and K. Ramchandran, "Rate-distortion methods for image and video compression," *IEEE Signal Process. Mag.*, vol. 15, no. 6, pp. 23–50, Nov. 1998.

[36] G. Bjontegaard, *Calculation of Average PSNR Differences Between RD-Curves*, document Rec. VCEG-M33, Apr. 2001.

[37] *3D-ATM Software*. [Online]. Available: http://mpeg3dv.nokiaresearch.com/svn/mpeg3dv/tags/

[38] *VSRS-1D-Fast Software*. [Online]. Available: https://hevc.hhi.fraunhofer.de/svn/svn_3DVCSoftware/

[39] *Methodology for the Subjective Assessment of the Quality of Television Pictures*, document Rec. ITR-R BT.500-11, 2002.



Jin Young Lee received the B.S. degree in information and communication engineering from Sungkyunkwan University, Suwon, Korea, in 2006, and the M.S. degree in electrical engineering from Korea Advanced Institute of Science and Technology, Daejeon, Korea, in 2008, where he is currently working toward the Ph.D. degree with the Department of Electrical Engineering with an academic support program from Samsung Electronics, Suwon.

He has been with Samsung Electronics since 2008. Since 2011, he has actively contributed to the development of 3D video coding standards (3D-AVC and 3D-HEVC), and served as a Software Coordinator in 3D-AVC. His current research interests include image processing, video coding, and 3D video communication systems.

Hyun Wook Park (SM'99) received the B.S. degree in electrical engineering from Seoul National University, Seoul, Korea, in 1981, and the M.S. and Ph.D. degrees in electrical engineering from Korea Advanced Institute of Science and Technology (KAIST), Daejeon, Korea, in 1983 and 1988, respectively.

He was a Research Associate with University of Washington, Seattle, WA, USA, from 1989 to 1992, and a Senior Executive Researcher with Samsung Electronics Company, Ltd., Suwon, Korea, from 1992 to 1993. He served as the Department Head of Electrical Engineering with KAIST from 2005 to 2011. He has been a Professor with the Department of Electrical Engineering, KAIST, since 1993. His research interests include image computing system, image compression, medical imaging, and multimedia system.

Dr. Park has served as an Associate Editor of *International Journal of Imaging Systems and Technology*.