Optical Engineering

SPIEDigitalLibrary.org/oe

Depth error compensation for camera fusion system

Cheon Lee
Sung-Yeol Kim
Byeongho Choi
Yong-Moo Kwon
Yo-Sung Ho

SPIE

Depth error compensation for camera fusion system

Cheon Lee
Sung-Yeol Kim
School of Information and Communications
Gwangju Institute of Science and Technology
123 Cheomdan-gwagiro, Buk-ku
Gwangju 500-712, Republic of Korea

Byeongho Choi
Multimedia IP Research Center
Korea Electronics Technology Institute
Seongnam-si, Kyeonggi-do 463-816, Republic of Korea

Yong-Moo Kwon
Imaging Media Research Center
Korea Institute of Science and Technology
Hwarangno 14-gil 5, Seongbuk-gu
Seoul 136-791, Republic of Korea

Yo-Sung Ho
School of Information and Communications
Gwangju Institute of Science and Technology
123 Cheomdan-gwagiro, Buk-ku
Gwangju 500-712, Republic of Korea
E-mail: hoyo@gist.ac.kr

Abstract. When the three-dimensional (3-D) video system includes a multiview video generation technique using depth data to provide more realistic 3-D viewing experiences, accurate depth map acquisition is an important task. In order to generate the precise depth map in real time, we can build a camera fusion system with multiple color cameras and one time-of-flight (TOF) camera; however, this method is associated with depth errors, such as depth flickering, empty holes in the warped depth map, and mixed pixels around object boundaries. In this paper, we propose three different methods for depth error reduction to minimize such depth errors. In order to reduce depth flickering in the temporal domain, we propose a temporal enhancement method using a modified joint bilateral filtering at the TOF camera side. Then, we fill the empty holes in the warped depth map by selecting a virtual depth and applying a weighted depth filtering method. After hole filling, we remove mixed pixels and replace them with new depth values using an adaptive joint multilateral filter. Experimental results show that the proposed method reduces depth errors significantly in near real time. [DOI: 10.1117/1.OE.52.7.073103]

Subject terms: camera fusion system; time-of-flight camera; depth camera; depth map generation; depth error reduction.

Paper 130293 received Feb. 24, 2013; revised manuscript received Jun. 2, 2013; accepted for publication Jun. 4, 2013; published online Jul. 9, 2013.

1 Introduction

Three-dimensional (3-D) video is a promising technology that can lead the next generation of multimedia services and applications. Recently, the research on 3-D video has become a hot issue due to growing demands of 3-D applications. This tendency will be continued until related technologies, i.e., computer graphics, computer vision, video compression, high-speed processing units, high-resolution displays and cameras, are converged all together.[1]

The main issue concerning 3-D video producing is how to provide the depth impression with minimal visual fatigues. One of the popular approaches is the use of virtual views generated by a depth-based-image rendering technique.[2] Since the depth data describes the distance between the camera and objects in a scene, multiview image generation is achievable. With the generated multiview images, the 3-D displays such as the stereoscopic display or autostereoscopic display can provide better depth impression of 3-D viewing.

In order to support such comfortable 3-D viewing, the moving picture experts group (MPEG) has investigated the 3-D video coding technologies which compress the video-plus-depth data.[3] Through intensive investigation works, experts in MPEG have developed both the depth estimation and the view synthesis methods.[4] Continuously, they called for proposals on 3-D video coding techniques.[5] As responses for the call, many coding tools were proposed for 3-D video coding.[6]

One of the important problems in producing of 3-D video that current researchers face is the generation of highly accurate depth data.[7] Although software-based depth estimation algorithms have been investigated for decades, obtaining precise depth information on texture-less or disoccluded regions indirectly is still an ill-posed problem.[8] In order to resolve this problem, a variety of direct depth-sensing devices such as structured light pattern sensors[9] and depth cameras[10] have been developed to generate accurate depth data in real time. However, due to the expensive cost of equipment, these active sensors are still far from manufacturing 3-D applications. Fortunately, since relatively cheap and compact time-of-flight (TOF) cameras were released, many depth capturing methodologies are being introduced using the TOF camera.

Several types of the camera fusion system have been proposed employing the TOF camera to capture depth data in real time. Lindner et al.[11] and Huhle et al.[12] developed a fusion camera system configured with one RGB camera and one TOF camera to reconstruct a 3-D model. Kim et al.[13] and Hahne et al.[14] employed high-resolution stereo cameras and one TOF camera to improve depth quality. Zhu et al.[15] presented a depth calibration method to improve depth accuracy. Lee et al.[16] improved the depth quality using a combination of the image segmentation and the depth estimation. In our previous work, the framework of 3-D scene capturing using the camera fusion system from capturing to multiview rendering was introduced.[17]

Although the TOF camera in the camera fusion system provides real-time depth measuring, there exist three depth errors; depth flickering in the temporal domain, holes in the warped depth map, and the mixed pixel problem. First, temporal depth flickering is induced by nonLambertian surfaces of objects; the captured depth values for a static object vary in time. Second, holes in the warped depth

map are generated by the viewpoint shifting. The camera fusion system usually refers warped depth data generated by projecting depth information obtained from the TOF camera onto the image plane of the RGB camera; hence, the newly revealed area has no measured depth values. Third, the mixed pixels are generated by false measuring of depth information around object boundaries.[18] If the infrared (IR) ray emitted from the TOF camera hits object boundary regions, part of the ray is reflected by front objects and the rest by background objects. Both reflections are received by the TOF camera, resulting in mixed measurement; these mixed pixels seriously degrade the quality of captured depth maps. In this paper, we introduce three depth error reduction methods to improve the accuracy of the captured depth maps.

The rest of this paper is organized as follows. In Sec. 2, the camera fusion system and its depth errors are introduced. In Sec. 3, three error reduction methods are proposed in detail. In Sec. 4, the experimental results are presented. Concluding remarks are given in Sec. 5.

2 Camera Fusion System and Depth Errors

2.1 *Camera Fusion System with Depth Camera*

As an extension to our previous work in Ref. 17, we configured a camera fusion system using two RGB cameras and one TOF camera, as shown in Fig. 1. The TOF camera is located in the center and two RGB cameras at either side. The center TOF camera captures depth video in real time with relatively low resolution, e.g., 200×200, and two RGB cameras capture high-resolution color images up to 1280×960 in real time. The first objective of this camera fusion system is to capture two RGB videos and their corresponding depth videos simultaneously. The second objective is to generate multiview images using captured data. In this work, depth error reduction methods are devised and applied since the quality of the generated multiview images is highly dependent on the accuracy of depth data.

Figure 2 describes the overall framework of the camera fusion system. Building up on the previous work in Ref. 17, three proposed depth error reduction methods are added. As a preprocessing, the camera calibration and image rectification are finished beforehand. The first depth error reduction method is a temporal refinement at depth capturing via the TOF camera. The refined depth data are warped to the RGB camera positions using a 3-D warping. Since the 3-D warping generates hole regions, the proposed

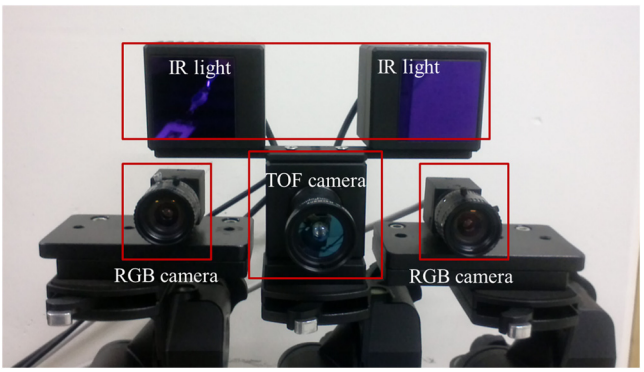


Fig. 1 Camera fusion system with time-of-flight camera.

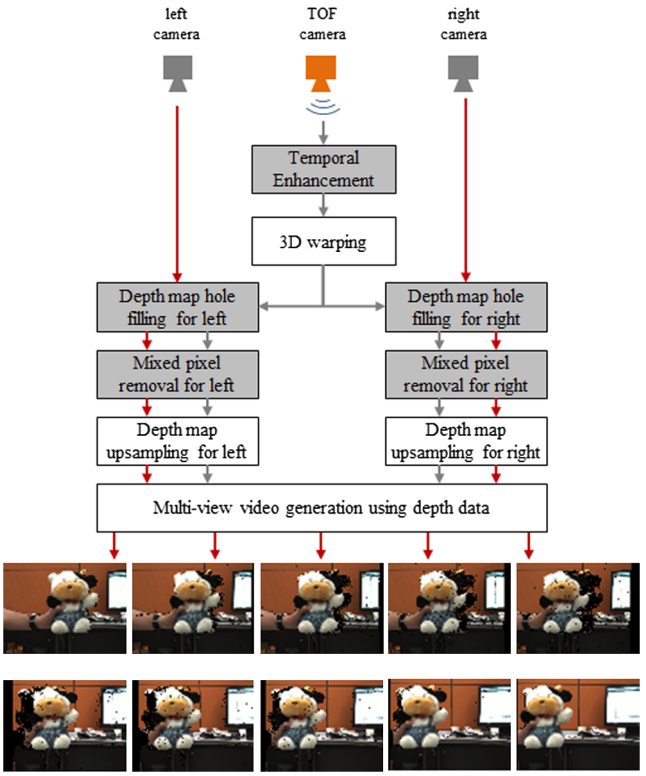


Fig. 2 Framework of multiview generation using camera fusion system.

hole filling method is applied to the warped depth map. Then, the proposed mixed pixel removal compensates for the depth discontinuity regions. After upsampling on the refined depth map,[19] the multiview generation part generates 36 views.

2.2 *Temporal Flickering on Captured Depth Data*

The principle of depth sensing with the TOF camera is measuring the receiving time of light reflected by an object in a scene.[20] In detail, a light pulse is transmitted by an IR light source and the range information is determined through the turn-around time with the knowledge of the speed of light. Therefore, the accuracy of the depth measuring is highly dependent on the Lambertian reflectance of object's surface. If an object has a nonLambertian surface, the reflected light pulse can vary over time. This is the main cause of the temporal flickering.

Figure 3 depicts the temporal flickering of the captured depth data. When the TOF camera captured a scene like Fig. 3(a), we traced the depth values of three points as shown in Fig. 3(b). The first check point was set on the middle of the screen, which is very reflective. The second check point was set on the plastic shoe rack. The third check point was set on the closet which is the most stable point. There was no movement of camera and objects during capturing. Figure 3(c) is the variations of three points over 100 frames. The first check point showed the most flickering effect, whereas the third check point showed minor depth variation. This flickering in captured depth video induces severe visual artifacts in multiview video generation.

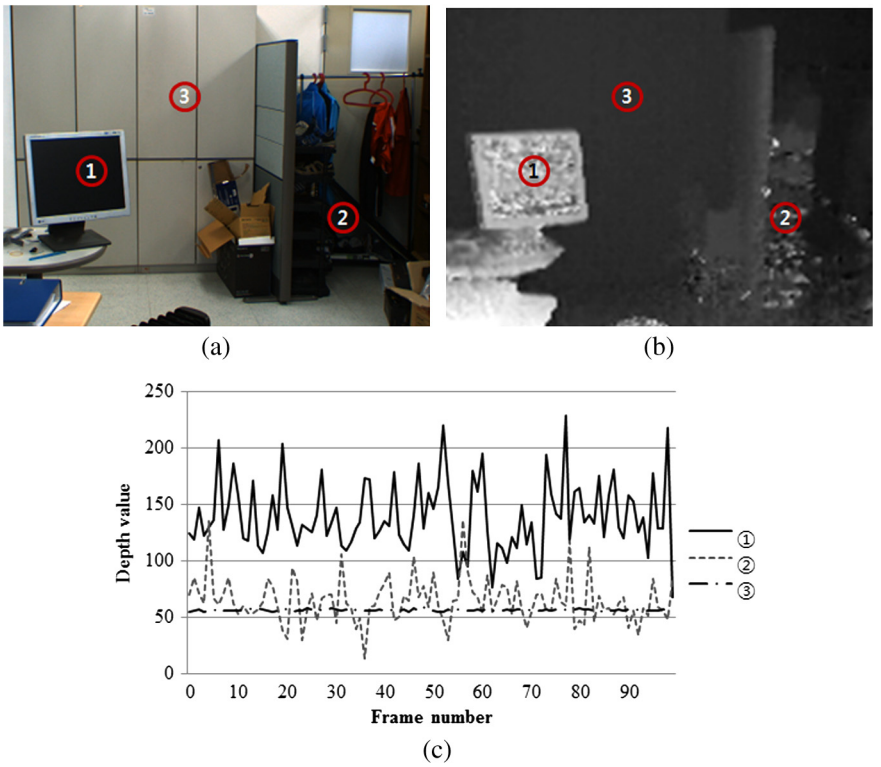


Fig. 3 Temporal flickering in depth map: (a) captured color image, (b) captured depth map, and (c) variation of depth values.

2.3 *Holes in Warped Depth Maps*

The depth maps warped from center to left and from center to right views have holes due to the viewpoint shifting, as shown in Fig. 4. The holes consist of one-pixel-width holes and wide holes; the former is generated by rounding errors in pixel mapping and the latter is induced by disoccluded regions at the target viewpoint. Figure 4(b) and 4(c) shows wide holes (black regions) around foreground objects. The one-pixel-width holes can be filled by means of a median filter (MF). However, wide holes can be filled by referring to neighboring pixels since there are no referable depth values in the original depth map at the center.

Typical hole filling method is to use an image inpainting,[21] which uses neighboring pixels and their gradient

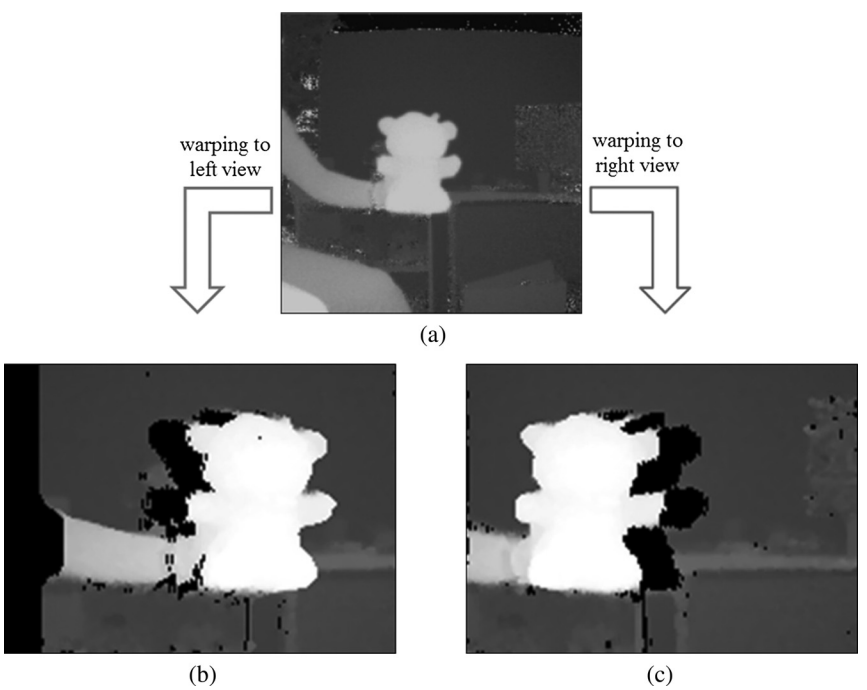
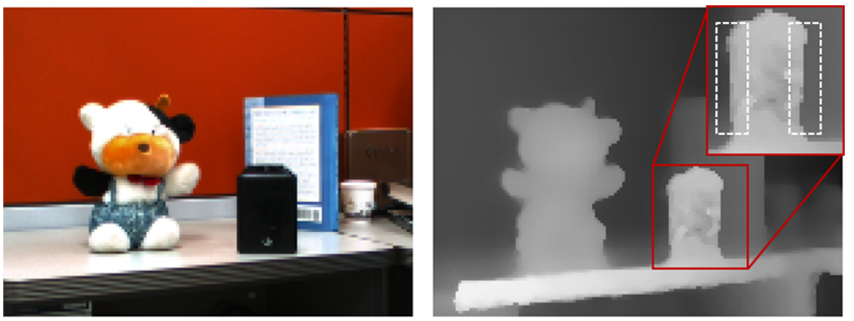


Fig. 4 Holes in warped depth maps: (a) captured depth map at center, (b) warped depth map to left, and (c) warped depth map to right.



(a) (b)

Fig. 5 Depth map errors due to mixed pixels: (a) color image and (b) warped depth map.

values. However, it does not consider geometrical positions of objects; hence, it generates mixed depth values from both foreground and background objects. In the previous work in Ref. 17, a smaller one among the valid depth values in horizontal direction was used to fill holes; we call this horizontal-direction hole filling. However, this method generates inconsistent depth values.

2.4 *Mixed Pixels around Object Boundary*

As mentioned in Sec. 1, mixed pixels in a depth map often arise around object boundaries when the TOF camera is used. Due to false depth sensing, the captured depth map induces incorrect pixel mapping around object boundaries. Consequently, edges of the depth map are not aligned with object boundaries of the color image. Figure 5 shows an example of mixed pixel problem. The black object in the image has a reflective surface; hence, the captured depth values of object boundaries are not consistent and are unstable, as shown in Fig. 5(b). Therefore, the mixed pixels in the warped depth map should be eliminated before conducting multiview video generation.

3 Depth Error Reduction Methods

3.1 *Temporal Enhancement for Depth Flickering*

Temporal flickering is inevitable since the TOF camera can neither distinguish reflectivity of objects nor compensate for depth errors during capturing automatically. Therefore, the goal of the proposed temporal enhancement is to reduce temporal flickering for static objects. For this, intensity image provided by the TOF camera is utilized. Since the intensity image has nothing to do with depth measuring, there is no flickering artifact; a static object with constant illumination has consistent intensity values in temporal domain. Using this property, the proposed method detects the flickering pixels and refines the depth data using modified joint bilateral filter (M-JBF).

The input data of the proposed temporal enhancement consists of the current intensity image (I_t), the current depth map (D_t) to be enhanced, and the previously refined depth map (D''_{t-1}), as shown in Fig. 6. First, the flickering depth values are detected by comparing two colocated depth values. In detail, for a pixel $p = (x, y)$, the detector compares two adjacent pixels in both D_t (p) and D''_{t-1} (p). If the difference is greater than a threshold value th, the depth value of p becomes zero, otherwise it has the same depth value of D_t (p) as

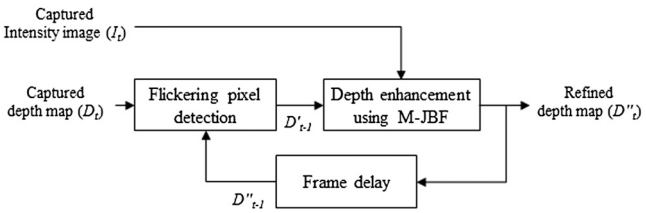


Fig. 6 Flows of temporal enhancement.

$$\begin{cases} D'_t(p_t) = D_t(p) & \text{if } |D(p_t) - D''_{t-1}(p)| < \text{th} \\ D'_t(p_t) = 0 & \text{otherwise} \end{cases}. \tag{1}$$

During this process, the detector generates an alpha map indicating the flickering pixels; $\alpha(p) = 1$ for the temporally stable pixels and $\alpha(p) = 0$ for the flickering pixels.

Next, the enhancement method assigns newly defined depth values to the flickering pixels using M-JBF. Since the flickering depth values are deleted from the captured depth map, M-JBF determines new depth values for the flickering pixels by referring to neighboring depth values. Let p and q be the target pixels to be determined and the referable neighboring pixels, respectively, and both pixels belong to a kernel Ω. Then M-JBF defines a new depth value as

$$D''_t(p) = \frac{\sum_{q\in\Omega}\omega_{\text{TE}}(p,q)\alpha(q)D'_t(q)}{\sum_{q\in\Omega}\omega_{\text{TE}}(p,q)}, \tag{2}$$

where ω_{TE} assigns weight to the reference depth pixel D'_{t-1} (q) as

$$\omega_{\text{TE}}(p,q) = \exp\left(-\frac{\|I_t(p) - I_t(q)\|^2}{2\sigma_c^2}\right)\exp\left(-\frac{\|p-q\|^2}{2\sigma_r^2}\right). \tag{3}$$

The constant variables σ_c and σ_r are the standard deviations for the intensity term and range term, respectively, and they control the similarity of intensity values and the range of neighboring pixels. In Eq. (2), the flickering pixels are removed in assigning weights since their alpha values are zero. Consequently, the proposed M-JBF determines a new depth value for the pixel p which is similar to the neighboring depth values.

3.2 *Hole Filling for Warped Depth Data*

The hole regions at the warped depth map are invisible in the depth camera view but visible in the RGB camera view. Therefore, the revealed hole regions at the RGB camera view arise around the foreground objects. With this property, it is reasonable to use the background depth values for the hole region's depth values. Proposed hole filling method determines a depth value using surrounding background depth values by minimizing noise depth values. One constraint is that the background depth value should be the closest one from the foreground object. Figure 7 describes the flows of the proposed depth hole filling method.

The proposed depth hole filling performs from the left-top pixel to right-bottom in raster scan order. Let D_w and α_w be the warped depth map with holes and the alpha map indicating holes, respectively. If the current pixel p is a hole, its alpha value is set to 0; otherwise its alpha value is set to 1. For the hole pixel p, the hole filling method determines a virtual depth value by selecting a minimum value among neighboring depth values in a certain window Ω as

$$\hat{d} = \min_{q\in\Omega} D_w(q). \tag{4}$$

This virtual depth value distinguishes the foreground's depth values among referable depth pixels.

Next, using the virtual depth value and neighboring depth values, the proposed modified-bilateral filter (M-BF) determines a new depth value for the hole pixel. M-BF determines the depth value for a hole p centered in Ω as

$$D_w(p,\hat{d}) = \frac{\sum_{q?\Omega}\omega_{\mathrm{HF}}(p,q,\hat{d})\alpha_w(q)D_w(q)}{\sum\limits_{q?\Omega}\omega_{\mathrm{HF}}(p,q,\hat{d})}, \tag{5}$$

where the weighting function ω_{HF} is defined by

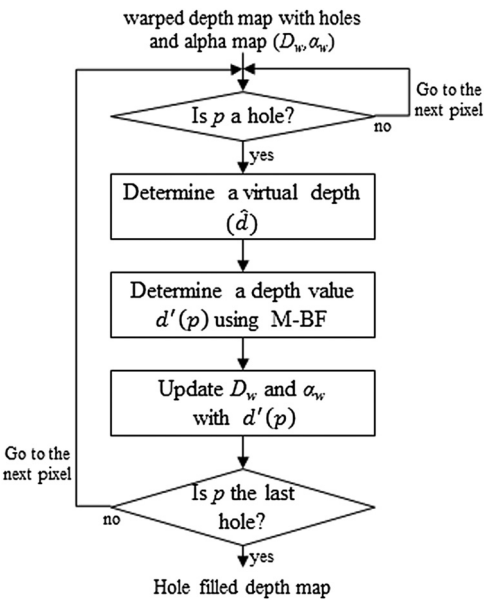


Fig. 7 Flows of depth map hole filling.

$$\omega_{\mathrm{HF}}(p,q,\hat{d}) = \exp\left(-\frac{\|\hat{d}-D(q)\|^2}{2\sigma_d^2}\right)\cdot\exp\left(-\frac{\|p-q\|^2}{2\sigma_r^2}\right), \tag{6}$$

where σ_d and σ_r are predetermined standard deviations for the depth term and the range term, respectively. The weighting function assigns high weight to the depth value similar to the virtual depth value, as presented in the first term of Eq. (6). The second term of Eq. (6) is the range term which considers the distance between the current pixel and the neighboring pixels.

Note that the proposed M-BF determines the hole's depth value using the weighted averaging via the background depth value. The hole filled depth values are similar to the background depth value without abrupt depth change.

3.3 *Mixed Pixel Removal*

As mentioned in Sec. 1, the mixed pixel problem induces miss-aligned depth discontinuities with object boundaries in color images. In the multiview video generation, this problem generates boundary noises.[22] As shown in Fig. 8, the proposed mixed pixel removal consists of three main steps: edge extraction, mixed pixel detection, and adaptive joint multilateral filtering.

In edge extraction, two edge maps, E_d and E_c, are obtained from the warped and hole filled depth map D and its corresponding color image C via Canny edge detection. Subsequently, mixed pixels are defined as follows: (1) for an edge pixel p given in E_c, an edge pixel q in E_d is chosen if q belongs to the kernel Ω having p at the center position, (2) a mixed pixel m is defined by the pixel between p and q. Figure 9 illustrates the procedure of mixed pixel detection.

In order to determine a new depth value for the mixed pixels, an adaptive joint multilateral filtering (A-JMF) is proposed. The mixed pixel m is replaced by the depth value calculated by weighted averaging with its neighboring pixels in Ω.

Formally, the new depth value D^{new} at p via A-JMF is computed by

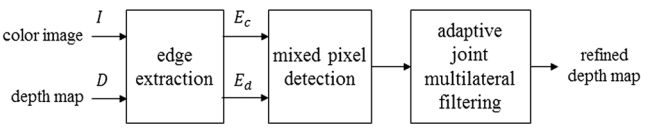


Fig. 8 Flows of mixed pixel removal.

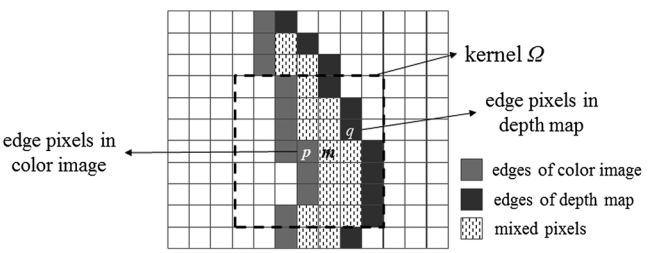


Fig. 9 Mixed pixel detection.

$$D^{\text{new}}(p)=\begin{cases}\sum_{q\in\Omega}\omega_{\text{MP}}(p,q)\cdot D(q)/\sum_{q\in\Omega}\omega_{\text{MP}}(p,q) & \text{if } p=m\\ D(p) & \text{otherwise}\end{cases}, \tag{7}$$

where ω_{MP} is a kernel weighting function of A-JMF. In Eq. (7), if and only if p is equal to m, A-JMF is carried out. Otherwise, $D^{\text{new}}(p)$ is directly assigned by $D(p)$.

ω_{MP} of A-JMF in Eq. (7) is defined by

$$\omega_{\text{MP}}(p,q)=f(\|p-q\|)g[\|I(p)-I(q)\|]h[D(p),D(q)], \tag{8}$$

where f, g, and h are spatial, color, and depth weighting functions, respectively.

By modeling via the Gaussian function, f, g, and h are represented by

$$\begin{aligned} f(\|p-q\|)&=\exp\left(-\frac{\|p-q\|^2}{2\sigma_f^2}\right)\\ g(\|I_p-I_q\|)&=\exp\left(-\frac{\|I(p)-I(q)\|^2}{2\sigma_g^2}\right)\\ h(D(p),D(q))&=\mu_{p,q}\exp\left(-(1-\mu_{p,q})\frac{\|D(p)-D(q)\|^2}{2\sigma_h^2}\right),\end{aligned} \tag{9}$$

where σ_f, σ_g, and σ_h are smoothing parameters of f, g, and h.

In particular, the proposed A-JMF excludes mixed pixels when calculating $D^{\text{new}}(p)$. For this, the scaling factor $\mu_{p,q}$ $(0\le\mu_{p,q}\le 1)$ in h controls the degree of reliability of $D(q)$; the closer q is to m, the lower the degree of reliability at q is. $\mu_{p,q}$ is represented by

$$\mu_{p,q}=\begin{cases}0 & \text{if } q=m\\ \frac{\|p-q\|}{K} & \text{otherwise}\end{cases}, \tag{10}$$

where K is the maximum distance between p and q. In Eq. (10), when q is equal to m, the degree of reliability at q is the lowest. Hence, setting $\mu_{p,q}$ to zero leads to $\omega_{\text{MP}}(p,q)$ being zero. Therefore, the mixed pixel value $D(m)$ is not used in calculation of $D^{\text{new}}(p)$. When q is not equal to m, the degree of reliability at q is determined by the distance between p and q.

4 Experimental Results and Discussion

The proposed depth error compensation methods are designed for capturing high quality depth maps using the camera fusion system, as shown in Fig. 1. Therefore, each method is applied to the captured depth maps and compared to the results with the conventional methods. After demonstrating the refined depth maps, we conducted additional experiments with the multiview video sequences that were enclosed with corresponding depth data; we compared the refined depth data to the original one. In addition, we measured the processing time of each method to evaluate complexity. The simulation system consists of Intel Core i7@2.30GHz processor, 8MB DDR2 RAM, and Windows7 64-bit. The window size for all filters was set to 11×11. The standard deviations, σ_d, σ_c and σ_r, were set to 0.1, 0.1, and 0.5 for the filters.

4.1 *Results on Temporal Enhancement*

Since the temporal flickering pixels in the captured depth map have similar characteristics with noise, we employed typical noise filters to the depth map for evaluation. First, an MF and bilateral filter (BF)[23] were applied. Second, the joint bilateral filter (JBF)[24] that uses the color image to enhance depth values around object boundaries was used. Third, the temporal depth filtering (TDF) with structural similarity (SSIM; TDF + SSIM)[25] was performed, which uses an SSIM measure to suppress transition depth errors by considering color variation. The fourth filter is the combination of JBF and filter (KF; JBF + KF).[26] Since KF traces the previous status of objects, the temporal consistency can be improved.

In order to compare the performance of each method, the above six methods and the proposed method were applied to the captured depth data via the depth camera, as shown in Fig. 3. We traced the variation of depth values for 100 frames at the point which is the most flickering point in Fig. 3(b). In addition, the processing time of each method was compared since the capturing complexity is important in the camera fusion system. The resolution of the depth map was 200×200.

Figure 10 shows the results of the depth values. As can be seen, the proposed method stabilized the depth values in time. Table 1 depicts the average depth values and standard deviations of each method for the point. The proposed method showed the closest average and the smallest standard

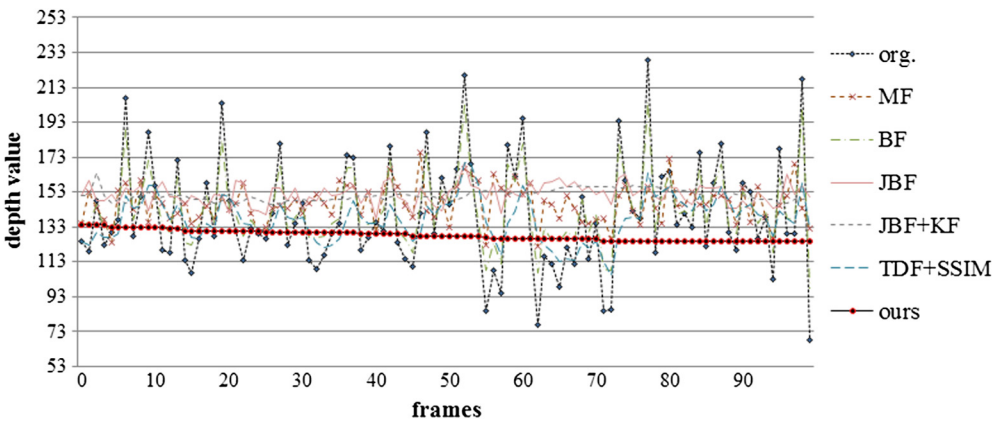


Fig. 10 Variation of temporal depth values.

Table 1 Analysis on depth value and processing speed.

	Orginal	MF	BF	JBF	JBF + KF	TDF + SSIM	Proposed
Average	139.7	146.3	143.2	151.8	151.2	137.7	128.2
Standard deviation	31.4	10.8	20.8	6.0	3.1	12.6	2.9
Speed (ms)	—	16.13	22.99	26.46	454.55	1666.67	22.99

deviation. Moreover, the proposed method showed fast processing speed in real time. Since the proposed M-JBF is designed from JBF, its processing speed was similar to BF and JBF.

In addition, the results of enhanced depth map for the target object are shown in Fig. 11. To compare the depth errors clearly, all depth maps were converted to color images with a gray-to-RGB conversion method. As can be seen in Fig. 11(b), the depth values of the black screen are spatially inconsistent, whereas the proposed method, as shown in Fig. 11(h), suppressed depth errors and generated spatially consistent depth values. Although JBF and JBF + KF methods generated spatially consistent depth values, the depth values around object boundaries were smoothed severely.

For an objective evaluation, we applied the temporal enhancement methods to the multiview video sequences, i.e., Book_arrival,[27] Breakdancers, and Ballet,[28] which contain the corresponding depth data. To make a noisy depth data, we added noises onto the region of interest of each frame with a Gaussian noise generator with zero mean and the standard deviation 20. When we apply the proposed method, we measured average standard deviations for the whole region and the processing time in milliseconds for 100 frames, as presented in Tables 2 and 3. Importantly, the standard deviations of the proposed method showed the smaller values compared with the other methods. In the sense of processing speed, the proposed method is rather complex than the MF, the BF, and the JBF. However, those of ours are a competitive method since it suppresses the noise depth values efficiently.

In order to evaluate the subjective quality, we compared the reconstructed depth maps of Breakdancers sequence. Figure 12(a) and 12(b) is the original color image and the corresponding depth data with noise (noisy depth, ND),

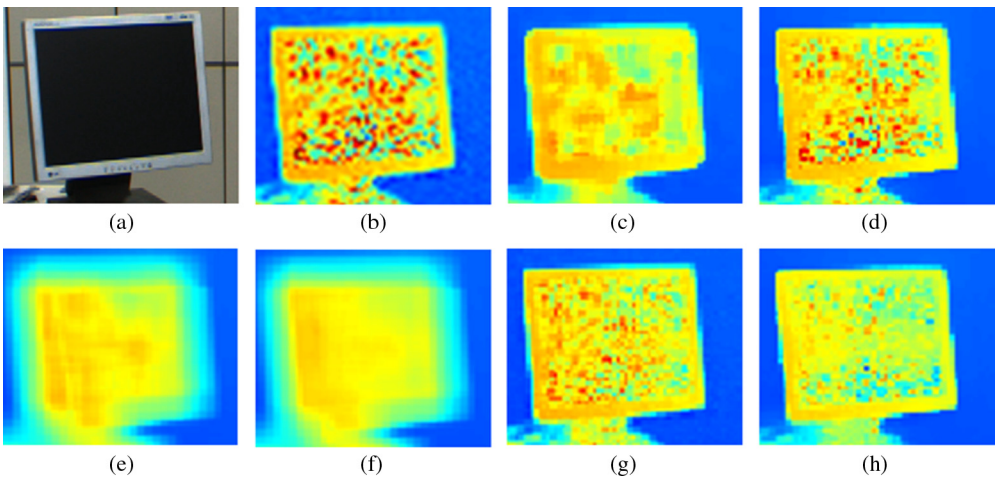


Fig. 11 Comparison of refined depth maps: (a) color image, (b) original, (c) MF, (d) BF, (e) JBF, (f) JBF + KF, (g) TDF + SSIM, (h) proposed.

Table 2 Average standard deviation values for 100 frames.

Test data	ND	MF	BF	JBF	JBF + KF	TDF + SSIM	Proposed
Book_arrival	3.45	2.25	1.53	1.39	8.22	3.08	1.16
Breakdancers	2.95	2.84	2.69	2.65	9.39	1.97	1.73
Ballet	3.03	2.87	2.81	2.79	8.09	3.33	2.78

Table 3 Average processing speed of the temporal compensation methods.

Test data	MF	BF	JBF	JBF + KF	TDF + SSIM	Proposed
Book_arrival	0.04	15.78	17.29	332.19	1152.19	18.04
Breakdancers	0.04	15.96	17.22	333.02	1158.86	18.01
Ballet	0.05	15.91	17.21	333.11	1165.98	18.07

respectively. The results of MF and TDF + SSIM still contain noises on the depth map, whereas those of BF, JBF, JBF + KF, and the proposed method removed the noises efficiently, because those filters are based on BF. When we look at the depth maps, it is hard to distinguish the improvement of the proposed method compared with other BF-based methods. However, the proposed method suppresses the temporal flickering depth values efficiently compared to the other methods regarding the results of Table 2.

4.2 *Results on Depth Map Hole Filling*

With the proposed hole filling method, holes in a warped depth map are filled with the M-BF as represented in Eq. (5). Figure 13 presents the example of the hole filling. Using 3-D warping technique, the depth map at center is shifted to the viewpoints of the RGB cameras. In the depth map warped to left view in Fig. 13(b), the holes are generated on the left side of the foreground object. Since the mixed pixels around object boundaries generate pepper-like noises, we used the MF before conducting the hole-filling process, as shown in Fig. 13(c). As a result, the hole regions became clear. Figure 13(d) shows the resultant hole filled depth maps. As can be seen, the holes are completely filled up by keeping the shape of the foreground object.

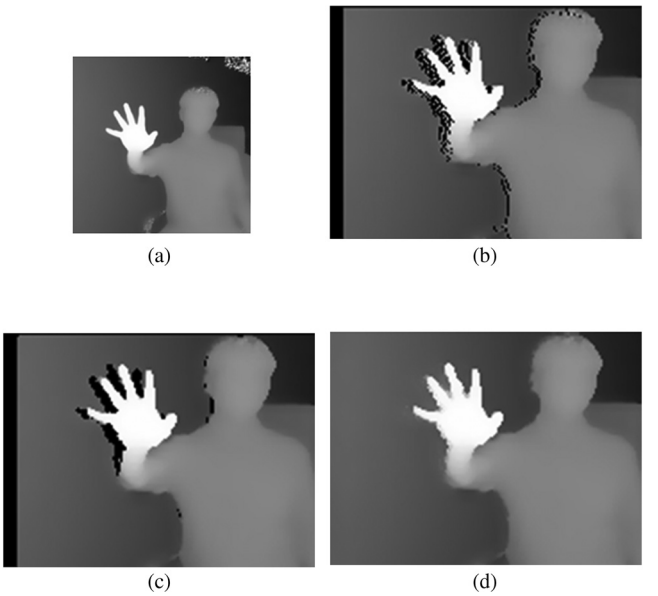


Fig. 13 Result of hole filling on the captured depth map: (a) captured depth map, (b) warped depth map to left view with holes, (c) median filtered depth map, (d) hole filled depth map.

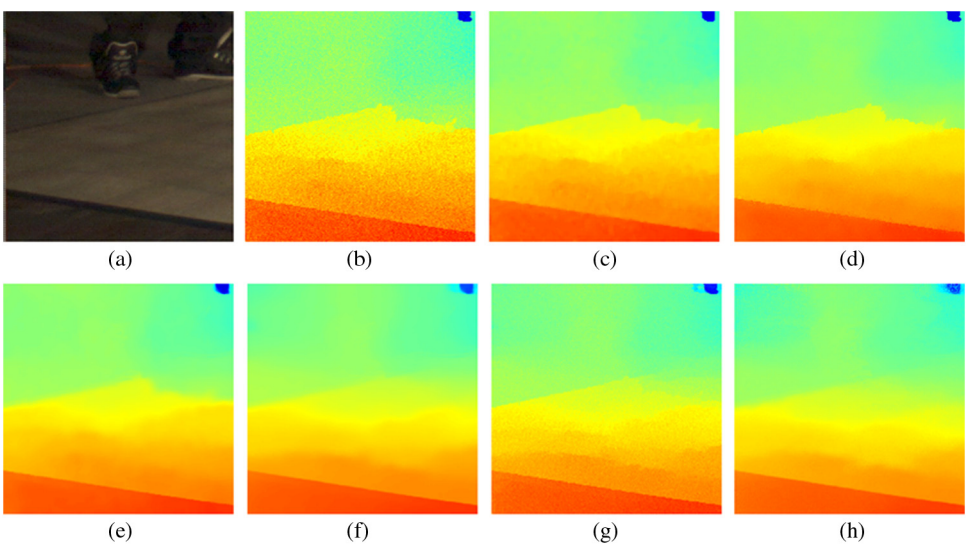
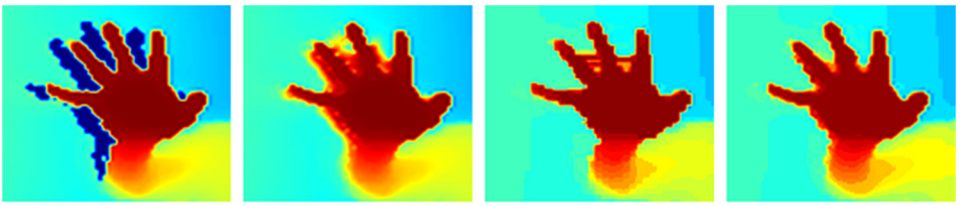


Fig. 12 Refined noisy depth maps of Breakdancers sequence: (a) color image, (b) noisy depth (ND), (c) MF, (d) BF, (e) JBF, (f) JBF+KF, (g) TDF +SSIM, (h) proposed.



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

Fig. 14 Comparison of depth hole filling: (a) warped depth map with holes, (b) inpainting, (c) HOR, (d) proposed.

In order to evaluate the proposed hole-filling method, we tested two hole-filling methods. The first is the inpainting method,[21] which is designed for reconstructing a damaged image. Recently, this method is employed for filling holes in the virtual view generation method.[4] The second method is a horizontal hole filling (HOR)[17] that fills the holes with a lower depth value between the leftmost and rightmost available depth values from the holes. Figure 14 shows the comparison of the resultant depth maps. The inpainting method generated false depth values; the shape of the object has been distorted. The second method, HOR, generated depth errors in-between the fingers. However, the proposed method showed the best results in that the shape of the foreground object has been kept and the holes are filled with the background's depth values successfully.

For the objective evaluation of the hole-filling method, we tested four multiview video sequences provided by MPEG 3-D video group[5]: Newspaper, Book_arrival, Balloons, and Undo_dancer. Using the 3-D warping technique, we obtained viewpoint shifted depth maps with holes and applied three hole filling methods. Then, we calculated the peak-signal-to-noise ratio (PSNR) values for the hole filled depth maps. Given the original depth map D_{org} and the hole filled depth map D_{HF}, MSE_{HF} is defined as:

$$\mathrm{MSE}_{\mathrm{HF}} = \frac{1}{W \cdot H} \sum_{j=0}^{H-1} \sum_{i=0}^{W-1} |D_{\mathrm{org}}(i,j) - D_{\mathrm{HF}}(i,j)|^2. \tag{11}$$

The PSNR value of the hole filled depth map is defined as:

$$\mathrm{PSNR}_{\mathrm{HF}} = 10 \cdot \log_{10}\left(\frac{255^2}{\mathrm{MSE}_{\mathrm{HF}}}\right), \tag{12}$$

where 255 represents the maximum value of the depth map.

Table 4 represents the comparisons of the calculated PSNR values of the hole filled depth maps based on Eq. (12). Overall PSNR values of the proposed methods were higher than those of the other methods. Compared with the inpainting method, the proposed method showed better quality as much as 3.83 dB for Undo_dancer sequence. Table 5 shows the processing speed of each method for 100 frames. Overall processing time of the proposed method was faster than the inpainting method and slower than HOR method. It is because HOR searches a reference depth value only in the horizontal direction, whereas the proposed method uses window-based filtering. In particular, the processing speed of Newspaper sequence was about 14 fps due to wide hole regions.

For the subjective evaluation, we compared the hole filled depth maps. Figure 15 is a comparison of the hole filled depth maps. Figure 15(a) and 15(b) is the color image and its corresponding depth map of the target view (view 3), respectively. Figure 15(c) is the depth map with holes warped from the reference view (view 1). Figure 15(d) is the result of hole filled depth map using the inpainting method, which has expended depth values toward the background. Figure 15(e) is the resultant depth map of HOR method, which has depth errors marked with a red circle. Figure 15(f) is the result of the proposed method, which showed neither expanded depth values nor depth errors around the foreground objects.

4.3 *Results on Mixed Pixel Removal*

After filling holes in the warped depth map, the proposed mixed pixel removal is followed. In order to show the performance of the proposed method, we captured a scene with a highly reflecting object, as shown in Fig. 16. The black object in the color image, Fig. 16(a), is a highly reflecting material, thus its depth map, Fig. 16(b), has distorted depth values. The proposed method detected mixed pixels, as shown in Fig. 16(c). Figure 16(d) shows the final

Table 4 Comparisons of PSNR values of the hole filled depth maps.

Test data	Hole depth map (dB)	Inpainting (dB)	HOR (dB)	Proposed (dB)
Newspaper	20.37	22.83	22.91	23.13
Book_arrival	21.58	24.22	24.24	24.37
Balloons	20.65	26.22	26.21	26.56
Undo_dancer	31.81	33.55	37.24	37.37

Table 5 Processing speed of depth map hole filling.

Test data	Inpainting	HOR	Proposed
Newspaper	590.75	6.06	69.99
Book_arrival	152.88	3.09	15.84
Balloons	132.19	2.07	17.62
Undo_dancer	113.19	4.84	27.52

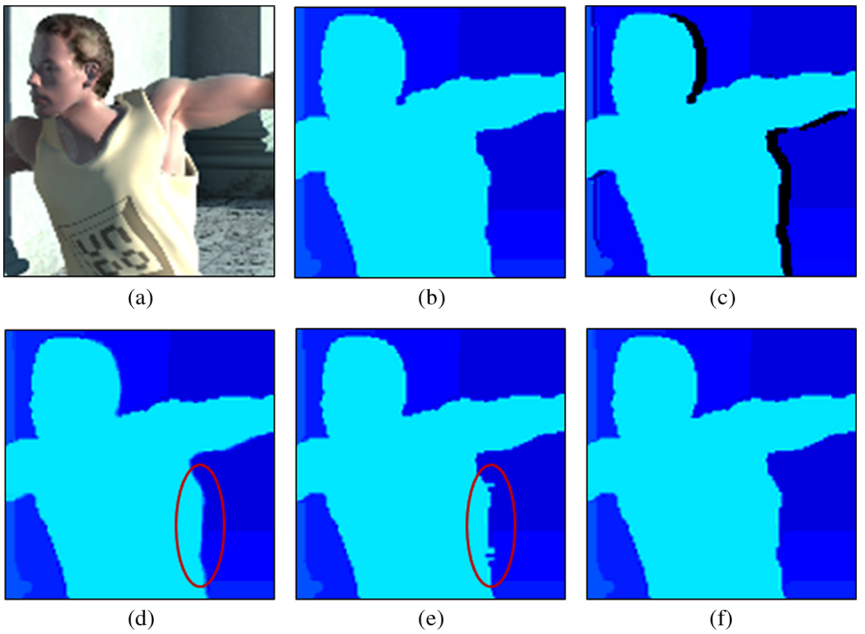


Fig. 15 Results of mixed pixel removal: (a) original color image, (b) original depth map, (c) warped depth map with hole, (d) hole filled depth map via the inpaint method, (e) hole filled depth map via HOR method, (f) hole filled depth map via the proposed method.

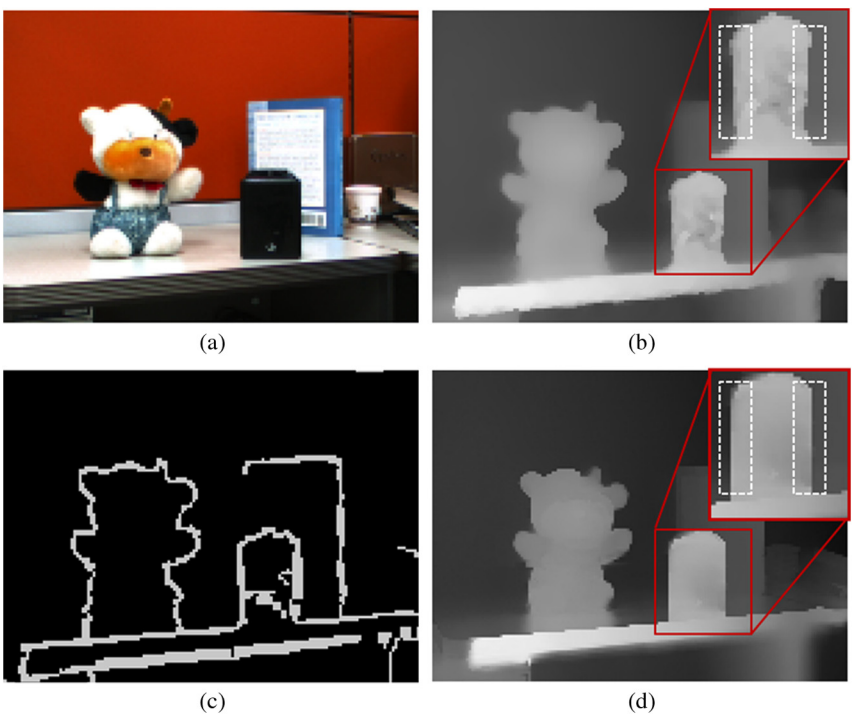


Fig. 16 Results of mixed pixel removal: (a) color image, (b) captured depth map via TOF camera, (c) detected mixed pixels, (d) refined depth map with mixed pixels removal.

depth map in which depth values of the object boundary are efficiently registered.

Additionally, the proposed method was applied to 12 ground truth depth maps provided by the Middlebury stereo: art, barn, bull, cone, dolls, laundry, map, moebius, reindeer, rocks, sawtooth, and venus.[29] To make distorted depth maps from ground truth data, we artificially added Gaussian noises with a standard deviation value 20 along object boundaries. The proposed method is compared with JBF(Ref. 24) and JMF (Ref. 30). For objective evaluation, the quality of output depth maps is measured by bad pixel rates for nonocclusion regions referring to the ground truth data. Let D_{gnd} and D_{rec} be the ground truth depth data and reconstructed depth data, respectively, and $\alpha_{\text{gnd}} = \{0, 1\}$ be the alpha

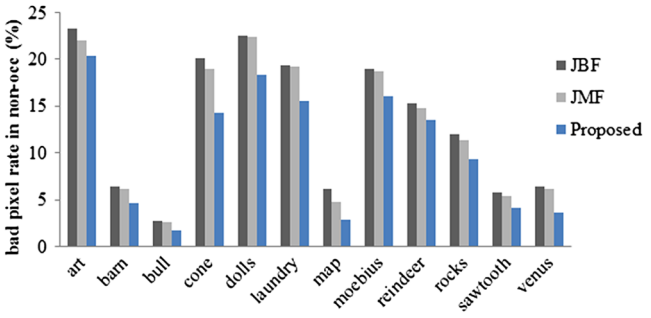


Fig. 17 Comparisons of bad pixels in nonocclusion regions.

map representing the occlusion region; 0 indicates an occluded pixel and 1 indicates a nonocclusion pixel. The bad pixel rate is defined as:

$$\text{badpixelrate} = \frac{\sum_{j=0}^{H-1}\sum_{i=0}^{W-1}\alpha_{\text{gnd}}(i,j)\cdot f_{\text{bad}}(i,j)}{\sum_{j=0}^{H-1}\sum_{i=0}^{W-1}\alpha_{\text{gnd}}(i,j)}, \tag{13}$$

where $f_{\text{bad}}(i,j)$ indicates the bad pixel between two depth maps. If two pixels are far from 1 depth value, we call it a bad pixel defined as:

$$f_{\text{bad}}(i,j) = \begin{cases} 1 & \text{if } |D_{\text{gnd}}(i,j) - D_{\text{rec}}(i,j)| > 1 \\ 0 & \text{otherwise} \end{cases}. \tag{14}$$

Figure 17 shows the comparison of bad pixel rates for the test depth maps. The average bad pixel rates of JBF, JMF, and the proposed method were 13.46%, 12.89%, and 10.59%, respectively. The proposed method produced lower bad pixel rates than JBF and JMF as much as 2.87%, and 2.30%. Figure 18(a)–18(c) demonstrates color image, ground truth depth map and artificially generated depth map of cone. Figure 18(d)–18(f) shows the output depth maps via JBF, JMF, and the proposed A-JMF method. Since the proposed method filters mixed pixels based on the degree of reliability, depth data on cone's edges become sharp while removing mixed pixels. Further improvement is expected when the parameters of the proposed filter are optimized for each sequence.

Table 6 Processing speed of mixed pixel removal.

	JBF	JMF	proposed
Processing speed (ms)	44.65	34.27	116.28

Table 6 presents the processing speed of the mixed pixel removal. For this, we took 100 frames of the warped and hole filled depth data captured by the camera fusion system as shown in Fig. 16. As a result, the proposed method showed the slowest speed. It is because the proposed method employs additional steps such as edge detection and mixed pixel detection based on the filtering. Reducing this high complexity is another challenge of our further work. GPU programming is used for the sampling solution.

4.4 *Generated Multiview Videos using Depth Data*

The error reduced depth data are utilized for the multiview generation process. After the mixed pixel removal, the depth maps are upsampled to the identical resolution of the color image via the multistep depth upsampling method.[30] Figure 19 demonstrates the generated multiview images using two color images and their depth data. The top-left image is the original left image and the bottom-right image is the original right image. As can be seen in Fig. 19, the foreground object, a book, moves from right to left as the viewpoint is shifted from 0 to 35. The black regions are the disocclusions due to viewpoint shifting.

Table 7 presents the overall processing speed of each part of the proposed system. As we described above, only the mixed pixel removal showed lowest speed among the proposed methods. Besides, depth upsampling and multiview generation occupied most of computing power. As well as the mixed pixel removal, efficient depth upsampling method and multiview generation are important issues in 3-D video system but those are not the scope of this work. We have

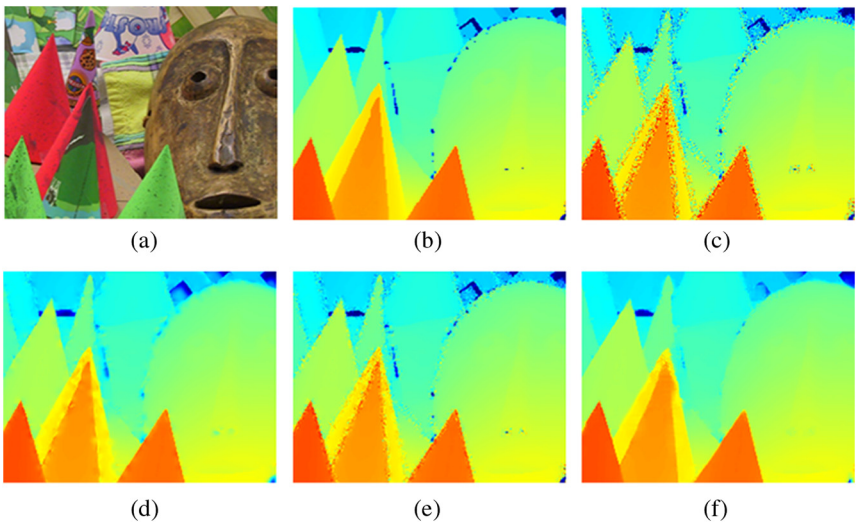


Fig. 18 Results on mixed pixel removal of "Cone" sequence: (a) color image, (b) ground truth depth map, (c) distorted depth map using Gaussian noise generator, (d) enhanced depth map using JBF, (e) enhanced depth map using JMF, (f) enhanced depth map using the proposed A-JMF.

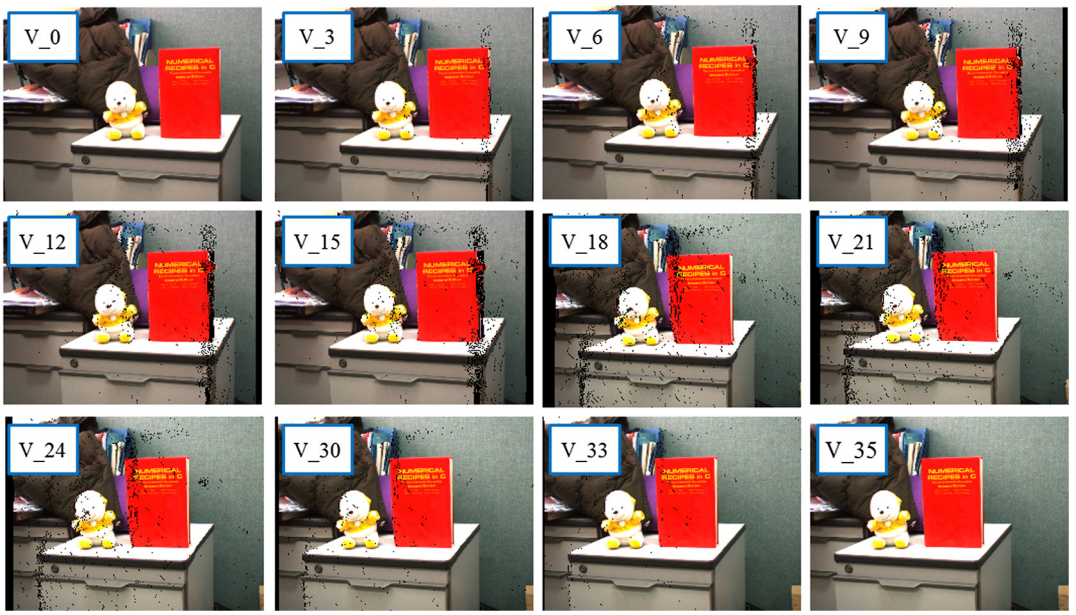


Fig. 19 Twelve-view images among generated 36-view images.

Table 7 Overall processing speed of each process.

Parts	Process	Image resolution	Processing speed	
			ms	fps
Color image	Capturing from RGB camera	1280 × 960	31.29	31.96
Depth map	Capturing from depth camera	200 × 200	27.87	35.88
	Depth temporal enhancement	200 × 200	22.99	43.50
	3-D depth warping	320 × 240	16.43	60.85
	Depth hole filling	320 × 240	12.91	77.46
	Mixed pixel removal	320 × 240	116.28	8.60
	Depth upsampling	1280 × 960	970.87	1.03
Multiview	36-View generation	1280 × 960	2380.00	0.42

plans to develop a fast multiview generation and hole filling with GPU (graphics processing unit) for further works.

5 Conclusions

In this paper, we have proposed three different methods to reduce depth errors in the captured depth map using the camera fusion system. We performed temporal refinement in the depth map captured by the depth camera, hole filling after depth map warping, and mixed pixel removal after hole filling. The temporal refinement method stabilized depth values of static objects, the hole filling method determined the depth value by referring to the background depth values, and the mixed pixel removal aligned depth discontinuities with the object boundaries of the color image. The experimental results showed that the processed depth map had reduced flickering depth values and resulted in less depth errors in the warped depth map. The hole filled and boundary aligned depth map helps to generate more pleasing multiview images.

Acknowledgments

This work is supported in part by the project on "A Development of Interactive Wide Viewing Zone SMV Optics of 3D Display" of the MKE, and in part by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MEST) (Grant No. 2012-0009228).

References

1. A. Kubota et al., "Multiview imaging and 3DTV—special issue overview and introduction," *IEEE Signal Process. Mag.* **24**(6), 10–21 (2007).
2. C. Fehn, "Depth-image-based rendering (DIBR), compression and transmission for a new approach on 3D-TV," *Proc. SPIE* **5291**, 93–104 (2004).
3. ISO/IEC JTC1/SC29/WG11, "Vision on 3D video," in *MPEG Output Document*, N10357 (2009).
4. ISO/IEC JTC1/SC29/WG11, "Report on experimental framework for 3D video coding," in *MPEG Output Document*, N11631 (2010).
5. ISO/IEC JTC1/SC29/WG11, "Call for proposals on 3D video coding technology," in *MPEG Output Document*, N12036 (2011).
6. ISO/IEC JTC1/SC29/WG11, "Overview of 3DV coding tools proposed in the CfP," in *MPEG Output Document*, N12348 (2011).
7. R. Larsen, E. Barth, and A. Kolb, "Special issue on time-of-flight camera based computer vision," *Comput. Vis. Image Underst.* **114**, 1317 (2010).
8. D. Scharstein and R. Szeliski, "A taxonomy and evaluation of dense two-frame stereo correspondence algorithms," *Int. J. Comput. Vis.* **47**, 7–42 (2002).
9. D. Scharstein and R. Szeliski, "High-accuracy stereo depth maps using structured light," in *IEEE Comput. Soc. Conf. Comput. Vision Pattern Recogn.*, pp. I/195–I/202 (2003).
10. G. J. Iddan and G. Yahav, "3D imaging in the studio (and elsewhere...)," *Proc. SPIE* **4298**, 48–55 (2001).
11. M. Lindner, A. Kolb, and K. Hartmann, "Data-fusion of PMD-based distance-information and high-resolution RGB-images," in *Int. Symp. on Signals, Circuits & Systems (ISSCS)*, pp. 121–124 (2007).
12. M. Lambooij et al., "Visual discomfort, and visual fatigue of stereoscopic displays: a review," *J. Imaging Sci. Technol.* **53**, 0302011–03020114 (2009).
13. Y. M. Kim et al., "Design and calibration of a multi-view TOF sensor fusion system," in *IEEE Comput. Soc. Conf. Comput. Vision Pattern Recogn.*, pp. 1–7 (2008).
14. L. M. J. Meesters, W. A. Ijsselsteijn, and P. J. H. Seuntiens, "A survey of perceptual evaluations and requirements of three-dimensional TV," *IEEE Trans. Circuits Syst. Video Technol.* **14**, 381–391 (2004).
15. J. Zhu et al., "Fusion of time-of-flight depth and stereo for high accuracy depth maps," in *Comput. Vision Pattern Recogn. (CVPR)*, pp. 1–8 (2008).
16. E. K. Lee and Y. S. Ho, "Generation of multi-view video using a fusion camera system for 3D displays," *IEEE Trans. Consum. Electron.* **56**, 2797–2805 (2010).
17. C. Lee et al., "3D scene capturing using stereoscopic cameras and a time-of-flight camera," *IEEE Trans. Consum. Electron.* **57**, 1370–1376 (2011).
18. R. L. Larkins et al., "Surface projection for mixed pixel restoration," in *Int. Conf. Image Vision Comput.*, pp. 431–436 (2009).
19. J. Kopf et al., "Joint bilateral upsampling," in *Proc. of the SIGGRAPH conf. ACM Trans. on Graphics* (2007).
20. T. Moeller et al., "Robust 3D measurement with PMD sensors," in Technical report, (PMDTec 2005).
21. A. Telea, "An image inpainting technique based on the fast marching method," *J. Graph. Tools* **9**, 25–36 (2004).
22. C. Lee and Y. S. Ho, "Boundary filtering on synthesized views of 3D video," in *Int. Conf. on Future Generation Communication and Networking*, pp. 15–18 (2008).
23. C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images," in *IEEE Int. Conf. Comput. Vision*, pp. 839–846 (1998).
24. O. P. Gangwal and R. P. Berretty, "Depth map post-processing for 3D-TV," in *IEEE Int. Conf. Consum. Electron.*, pp. 1–2 (2009).
25. D. Fu, Y. Zhao, and L. Yu, "Temporal consistency enhancement on depth sequences," in *Picture Coding Symp.*, pp. 342–345 (2010).
26. M. Camplani and L. Salgado, "Adaptive spatio-temporal filter for low-cost camera depth maps," in *IEEE int. Conf. Emerg. Sig. Processing*, pp. 33–36 (2012).
27. ISO/IEC JTC1/SC29/WG11, "HHI test material for 3D video," in *MPEG Input Document*, m15413 (2008).
28. Microsoft Research 3D Video Download, http://research.microsoft.com/en-us/um/people/sbkang/3dvideodownload/.
29. Middlebury stereo vision, http://vision.middlebury.edu/stereo/data/.
30. A. K. Riemens et al., "Multi-step joint bilateral depth upsampling," *Proc. SPIE* **4298**, 48–55 (2009).

Cheon Lee received his BS degree in electronic engineering and avionics from Korea Aerospace University (KAU), Korea, in 2005 and MS and PhD degrees in information and communication engineering at the Gwangju Institute of Science and Technology (GIST), South Korea, in 2007 and 2013, respectively. His research interests include digital signal processing, video coding, data compression, 3-D video coding, 3-D television and realistic broadcasting, camera fusion system with depth camera.

Sung-Yeol Kim received his BS degree in information and telecommunication engineering from Kangwon National University, South Korea, in 2001, and MS and PhD degrees in information and communication engineering at the Gwangju Institute of Science and Technology (GIST), South Korea, in 2003 and 2008, respectively. From 2009 to 2011, he was with the Imaging, Robotics, and Intelligent System Lab at The University of Tennessee at Knoxville (UTK), USA, as a research associate. His research interests include digital image processing, depth image-based modeling and rendering, computer graphic data processing, 3DTV and realistic broadcasting.

Byeongho Choi received his BS and MS degrees in electronic engineering from the University of Hanyang, Republic of Korea, in 1991 and 1993. From 1993 to 1997, he worked for LG Electronics Co. Ltd as a junior researcher. In 1997, he joined Korea Electronics Technology Institute (KETI), where he was involved in the development of multiview video, stereo vision and other video systems. He is currently a managerial researcher of SoC Research Center. He is also currently pursuing a PhD degree in the Department of Image Engineering at Chung-Ang University. His research interests include digital image processing, and its application, especially-3DTV and stereo vision systems.

Yong-Moo Kwon received his PhD at Hanyang University in 1992. He is now a principal researcher of Imaging Media Research Center at Korea Institute of Science and Technology (KIST). He has done researche in the area of image-processing technology, multimedia databases, virtual heritage technology, and 3-D media technology. Currently, he is now investigating the tangible socia media technology. The key research issues include real-time gaze tracking for the application to human computer interaction, image-based 3-D modeling, tangible and social media platform, and networked collaboration.

Yo-Sung Ho received both BS and MS degrees in electronic engineering from Seoul National University (SNU), Korea, in 1981 and 1983, respectively, and PhD degree in electrical and computer engineering from the University of California, Santa Barbara, in 1990. He joined Electronics and Telecommunications Research Institute (ETRI), Korea, in 1983. From 1990 to 1993, he was with Philips Laboratories, Briarcliff Manor, New York, where he was involved in development of the advanced digital high-definition television (AD-HDTV) system. In 1993, he rejoined the technical staff of ETRI and was involved in development of the Korea direct broadcast satellite (DBS) digital television and high-definition television systems. Since 1995, he has been with the Gwangju Institute of Science and Technology (GIST), where he is currently a professor in the Information and Communications Department. His research interests include digital image and video coding, advanced coding techniques, 3-D television, and realistic broadcasting.