Optical Engineering 49(2), 027002 (February 2010)

Intermediate view synthesis algorithm using mesh clustering for rectangular multiview camera system

Byeongho Choi
Korea Electronics Technology Institute
68 Yatap-dong
Bundang-gu, Seongnam 463-816
Korea
and
Chung-Ang University
221 Heuk Seok-Dong
DongJak-Gu, Seoul 156-756
Korea

Taewan Kim
Korean Agency for Technology and Standards
96 Gyoyukwongil
Gwacheon-Si, Gyonggi-Do 427-723
Korea

Kwan-Jung Oh
Yo-Sung Ho
Gwangju Institute of Science and Technology
261 Cheomdan-gwagiro (Oryong-dong)
Buk-gu, Gwangju 500-712
Korea
kjoh81@gist.ac.kr

Jong-Soo Choi, MEMBER SPIE
Chung-Ang University
221 Heuk Seok-Dong
DongJak-Gu, Seoul 156-756
Korea

Abstract. A multiview video-based three-dimensional (3-D) video system offers a realistic impression and a free view navigation to the user. The efficient compression and intermediate view synthesis are key technologies since 3-D video systems deal multiple views. We propose an intermediate view synthesis using a rectangular multiview camera system that is suitable to realize 3-D video systems. The rectangular multiview camera system not only can offer free view navigation both horizontally and vertically but also can employ three reference views such as left, right, and bottom for intermediate view synthesis. The proposed view synthesis method first represents the each reference view to meshes and then finds the best disparity for each mesh element by using the stereo matching between reference views. Before stereo matching, we separate the virtual image to be synthesized into several regions to enhance the accuracy of disparities. The mesh is classified into foreground and background groups by disparity values and then affine transformed. By experiments, we confirm that the proposed method synthesizes a high-quality image and is suitable for 3-D video systems. [DOI: 10.1117/1.3309459]

Subject terms: intermediate view synthesis; mesh clustering; rectangular multiview video system.

Paper 090100RRR received Feb. 10, 2009; revised manuscript received Dec. 2, 2009; accepted for publication Dec. 11, 2009; published online Feb. 23, 2010.

1 Introduction

With the technical advancement of computer graphics, computer vision, and related fields, the realistic visual can come true in the near future. Recently, various multimedia services have become available and the demands for realistic multimedia systems are growing rapidly and the three-dimensional (3-D) video technologies have been studied to satisfy these demands.[1–5] In Europe, the basic technologies for 3DTV (three-dimensional television) were studied by the ATTEST (advanced three-dimensional television system technologies) project[6] in 2002 and a consortium comprised of over 20 research institutes have carried out a 3DTV project since 2004. The National Aeronautics and Space Administration[7] (NASA) and the Massachusetts Institute of Technology[8] (MIT) studied 3-D video and holography technology in the United States and the Universal Median Research Center (UMRC) of the National Institute of Information and Communications Technology (NIST) studied a 3DTV and ultrarealistic broadcasting in Japan.[9] The Moving Picture Experts Group (MPEG), an international organization for standardization of multimedia, initiated a work for 3-D video and free-viewpoint television (FTV) in 2007 and studied a broader technical scope, including issues such as depth estimation, coding, and rendering.[10]

The virtual view synthesis is a key technology to realize 3-D video, and various view synthesis methods have been studied. Generally, the view synthesis algorithms are categorized into reconstruction-based methods and interpolation-based methods.[11] The reconstruction-based method uses explicit or implicit 3-D information. Since the quality of the synthesized image depends on the accuracy of the disparity, various disparity estimation methods were studied. Sethuraman[12] proposed a quadtree-based disparity estimation. However, the quadtree approach was not efficient for diagonal edges. Altunbasak and Tekalp[13] proposed mesh-based disparity estimation methods. They estimated the disparity based on a triangular mesh and synthesized the virtual image using the estimated disparity. For dealing

0091-3286/2010/$25.00 © 2010 SPIE

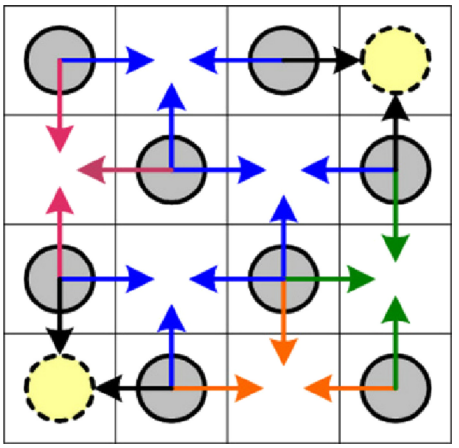


Fig. 1 Rectangular multiview camera arrangement.

with the occlusion problem, they inserted or deleted the mesh elements for the disparity discontinuous region.

In this paper, we propose an intermediate view synthesis method suitable for a rectangular multiview camera system that can offer broader 3-D video to users and is more useful to view synthesis. The proposed view synthesis method employs mesh-based stereo matching. First, we divided the virtual intermediate view into top and bottom regions. The top region existed in a left and a right reference view and the bottom region mainly existed in the bottom view. Second, we extracted the feature points by edge detection methods and then composed the triangular mesh. Third, we determined the disparity for each mesh element and then clustered the mesh elements according to their disparity values. Fourth, we synthesized the intermediate view with reference views and its disparity maps using an affine transform. Finally, we postprocessed the synthesized image to reduce the discontinuous regions.

The rest of this paper is organized as follows. Section 2 explains the details of the proposed view synthesis algorithm. We then demonstrate and evaluate the performance of the proposed scheme in Sec. 3, and conclude the paper in Sec. 4.

2 Proposed Intermediate View Synthesis Method

This section describes the proposed intermediate view synthesis using a rectangular multiview camera system. Figure 1 shows the 4×4 rectangular multiview camera arrangement. For the rectangular multiview camera system, we can employ three reference views for view synthesis by a lat-

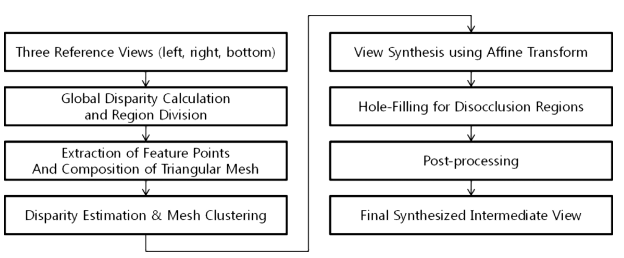


Fig. 2 Diagram of the proposed view synthesis scheme.

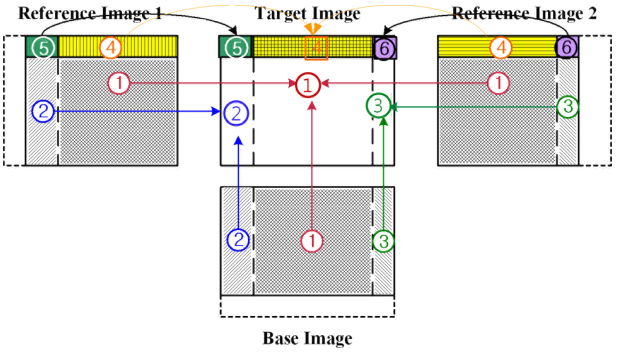


Fig. 3 Region division for the virtual view to be synthesized.

ticed camera arrangement. Thus, we can synthesize the 6 views among the 16 views from the neighboring 3 views.

Figure 2 shows a diagram of the proposed view synthesis scheme and each subalgorithm is detailed in the following subsections. The proposed view synthesis method employs three reference views: left, right, and bottom views.

2.1 *Region Division Using Global Disparity*

We divide the virtual view into six regions, as shown in Fig. 3, to synthesize a better intermediate view by restricting the wrong reference view for each region.

First, we calculate the horizontal distance (HD) and the half-HD using the left and right reference views. The vertical distance (VD) is calculated from the horizontally half-HD shifted left or right reference view and the bottom image. We use the sum of absolute difference (SAD) as a measure to calculate the global disparity. The half-HD is a width of regions 2, 3, 5, 6, and the VD is a height of regions 4, 5, and 6. The height of regions 1, 2, and 3 is equal to the image height—VD. Figure 4 shows examples of HD and VD calculations.

2.2 *Extraction of the Feature Points and Composition of the Triangular Mesh*

Block-based stereo matching is a representative disparity estimation algorithm, however, it does not provide an accurate disparity when the block contains both foreground and background regions and shows the blocking artifacts for rendering. The proposed view synthesis method employs mesh-based disparity estimation. To compose the mesh, we apply the edge detection algorithm and then extract the fea-

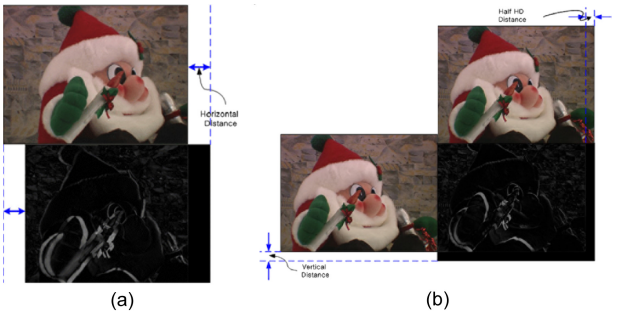
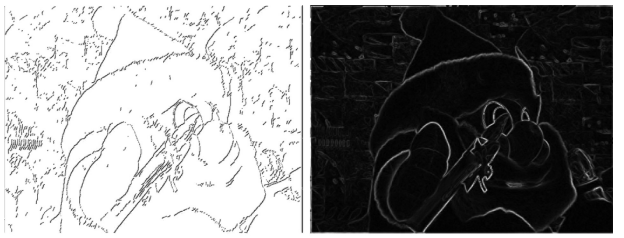


Fig. 4 (a) VD and (b) HD calculations.



(a) (b)

Fig. 5 Edge detection. (a) Canny method and (b) Sobel method.

ture points from edges. Figure 5 illustrates the results of edge detection by the Canny and Sobel methods.[14,15]

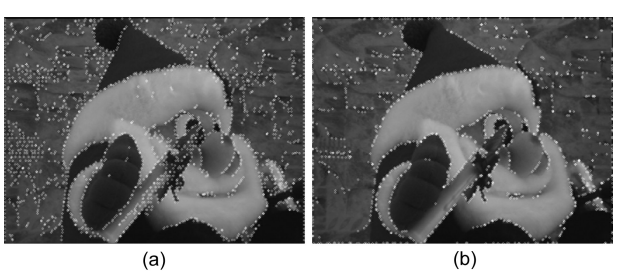
For the edge-detected images, we extract the feature points by variable size windows to properly assign one feature point per each window; that is, whereas a monotonous region is represented by one feature point coming from a large-size window, a complex region has many feature points coming from small-size windows. For the Canny method, the rapidly changed pixel is set as a feature point, and the minimum valued pixel is selected for the Sobel method. Figure 6 shows the extraction of feature points for the Canny and Sobel methods.

After the extraction of feature points, we compose the triangular mesh using a Delaunay triangle. As shown in Fig. 7, the Delaunay triangulation for a set P of points in the plane is a triangulation DT(P) such that no point in P is inside the circumcircle of any triangle in DT(P).

Delaunay triangulations[16] maximize the minimum angle of all the angles of the triangles in the triangulation. The circumcircle of a triangle formed by three points from the original point set is empty if it does not contain vertices other than the three that define it. Other points are permitted only on the very perimeter, not inside. In this paper, we adopt the divide and conquer method to compose the Delaunay triangulation. Figure 8 shows the results of Delaunay triangulation for feature-point-extracted images.

2.3 *Disparity Estimation and Mesh Clustering*

To synthesize the virtual view, the disparity between the virtual view and its neighboring view was previously calculated. For the multiview video sequence, the corresponding point of a certain point should exist on the epipolar line of the given certain point.[17,18] Since the rectangular multiview camera system parallelly sets the cameras, we can simplify the epipoloar geometry.



(a) (b)

Fig. 6 Extraction of feature points. (a) Canny method and (b) Sobel method.

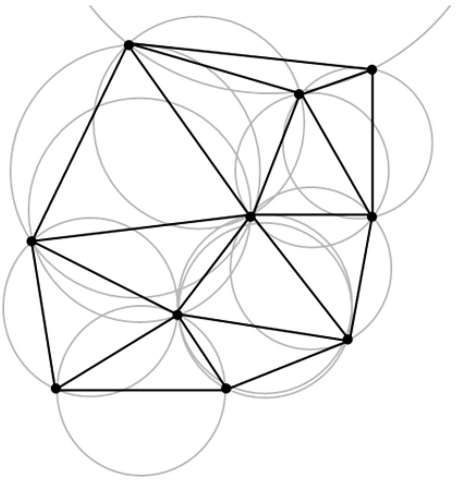


Fig. 7 Delaunay triangulation in the plane with circumcircles.

We assume that the epipolar lines for all points in an image are parallel and all disparities have same direction. From the precalculated HD and VD, we can define the initial disparity under the preceding assumption. We find the disparity for each feature point by SAD-based block matching algorithm as

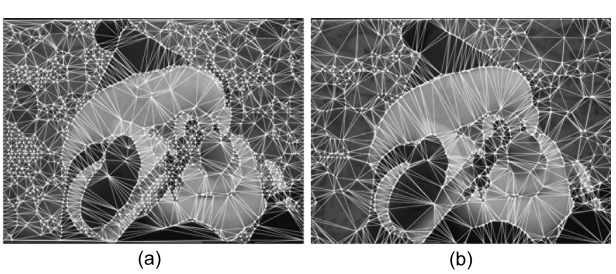
$$\mathrm{SAD}_B = \sum_{(x,y)\in B} |I_B(x,y) - I_L(x+d_x, y+d_y)|, \tag{1}$$

where B is a window containing the feature point at the center of window. The (d_x, d_y) is a disparity.

Regions 1, 2, and 3 of the intermediate view are basically synthesized by regions 1, 2, and 3 of the bottom reference view, as shown in Fig. 3. For that, we must know the vertical disparity between the virtual view and the bottom view. The D_L and D_R are defined as a disparity set between a left view and a bottom view and between a right view and a bottom view, respectively. Regions 2 and 3 of the bottom view have only D_L and D_R, respectively; whereas region 1 of the bottom view has both D_L and D_R. Thus, we can refine the disparity of region 1 of the bottom view as

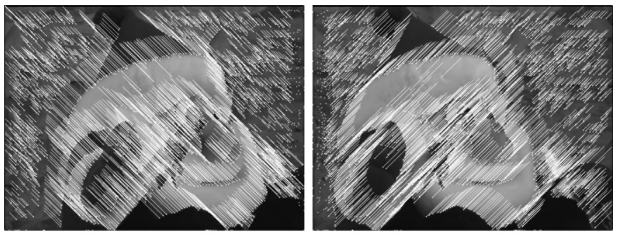
$$d_y = \lfloor D_L(d_y) + D_R(d_y) \rfloor / 2. \tag{2}$$

The disparity refinement provides more accurate disparity and it also affects our ability to handle disocclusion regions later. Figure 9 shows the D_L and D_R for region 1 of the bottom view.



(a) (b)

Fig. 8 Delaunay triangulation. (a) Canny method and (b) Sobel method.



(a) (b)

Fig. 9 Disparities (a) D_L and (b) D_R for region 1 of the bottom view.

In the same manner, the disparity sets for regions 4, 5, and 6 are defined and calculated. In this case, the disparity is defined and calculated between the left and right views. This disparity is represented by horizontal disparity. Figure 10 shows the disparity for region 4.

The handling of continuous/discontinuous regions and disocclusion regions are important since they severely affect the rendering quality. The continuity/discontinuity means the spatial characteristics of successive pixels. Whereas a single object on a certain depth plane should have similar disparities and be continuously synthesized, the border region for such an object boundary has two kinds of disparity sets and is discontinuously synthesized. Disocclusion regions are defined as areas that cannot be seen in the reference image but exist in the synthesized one. The proposed mesh-based view synthesis has a limitation when rendering the discontinuous regions and disocclusion regions as shown in Fig. 11.

To solve the preceding problem, we propose a mesh clustering method. The mesh clustering divides a mesh into a foreground mesh and a background mesh by its disparity value. Figure 12 shows the distribution of disparities for regions 1, 2, and 3 of the bottom view, as shown in Fig. 3. In this case, the disparity value 26 is the best threshold.

Currently, since we have a disparity set for feature points, the points of one mesh element can belong to the different mesh cluster. Thus, we redefine the disparity, as shown in Fig. 13.

Assume that region C_1 belongs to background and region C_2 belongs to the foreground. The values beside the vertices represent the disparity values. Triangles t_1 and t_2 share the one line $(\overline{P_1P_2})$ and it is a border between two triangles. Thus, two triangles should be separated, as shown in Fig. 13(b). The disparity values for vertices P_1' and P_2' of triangle t_1 are redefined. Equation (3) depicts the disparity redefinition.

$$\begin{cases} t(p,q,r) \in C_1 & \text{if } \min[D(p),D(q),D(r)] \in C_1 \\ t(p,q,r) \in C_2 & \text{if } \min[D(p),D(q),D(r)] \in C_2, \end{cases}$$

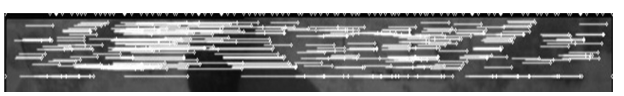


Fig. 10 Disparity set for region 4 of the bottom view.

Fig. 11 Mishandling of discontinuous regions and disocclusion regions.

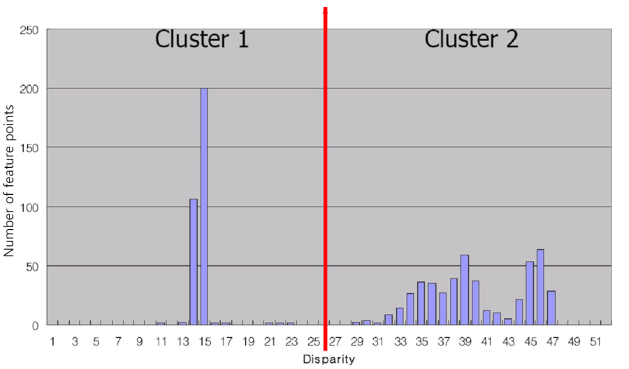
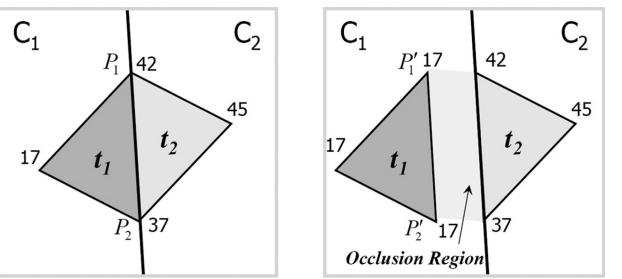


Fig. 12 Distribution of disparity and mesh clustering.



(a) (b)

Fig. 13 Disparity redefinition, (a) before and (b) after.

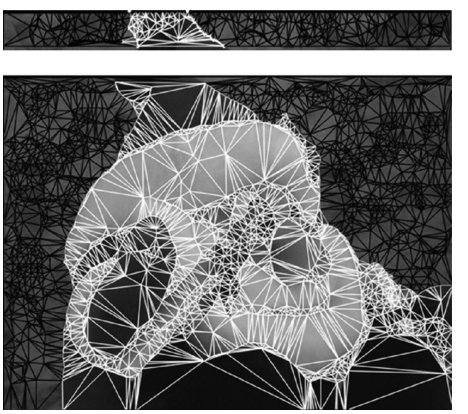


Fig. 14 Mesh clustering.

Fig. 15 Synthesized image after mesh clustering.

Fig. 17 Synthesized image using only a background mesh.

$$\begin{cases} D(q) = D(p) \text{ and } D(r) = D(p) & \text{if } p \in C_1 \text{ and } q,r \in C_2 \\ D(r) = [D(p) + D(q)]/2 & \text{if } p,q \in C_1 \text{ and } r \in C_2. \end{cases} \tag{3}$$

Figure 14 illustrates the mesh clustered image. The black mesh is background mesh and white mesh is foreground mesh. Figure 15 illustrates the image synthesized using mesh clustering. The disocclusion regions between the foreground and background are represented by holes, unlike the image shown in Fig. 11. The holes are filled from the left or right view and are explained in Sec. 2.5.

2.4 *View Synthesis Using Affine Transform*

The proposed view synthesis method utilizes the affine transform. In general, an affine transform is composed of linear transformations such as rotation, scaling, or shear and a translation or shift.[18] Several linear transformations can be combined into a single one. Equation (4) depicts an affine transform between (x,y) and (u,v).

$$\begin{bmatrix} u(x,y) \\ v(x,y) \end{bmatrix} = \begin{pmatrix} a_1 \\ a_4 \end{pmatrix} + \begin{pmatrix} a_2 & a_3 \\ a_5 & a_6 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix}, \tag{4}$$

which is redepicted as

$$\mathbf{d}(x,y) = \begin{bmatrix} u-x \\ v-y \end{bmatrix} = \begin{bmatrix} 1 & x & y & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & x & y \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_6 \end{bmatrix}. \tag{5}$$

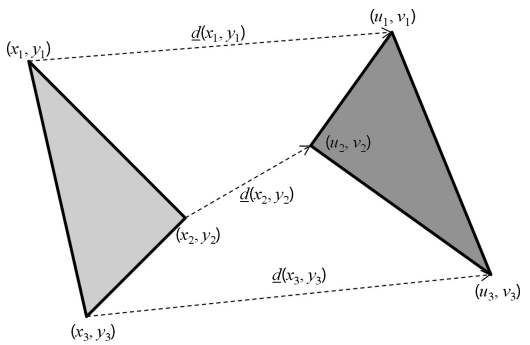


Fig. 16 Affine transform for triangular mesh.

Figure 16 shows an example of an affine transform for one triangular mesh element. From the relations of three vertex pairs, the six components of the affine transform can be obtained and the inner points of a triangular mesh element are affined transformed.

The relation between three vertices (x_1,y_1), (x_2,y_2), and (x_3,y_3) and their affine-transformed points (u_1,v_1), (u_2,v_2), and (u_3,v_3) are

$$\begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ v_1 \\ v_2 \\ v_3 \end{pmatrix} = \begin{pmatrix} 1 & x_1 & y_1 & 0 & 0 & 0 \\ 1 & x_2 & y_2 & 0 & 0 & 0 \\ 1 & x_3 & y_3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & x_1 & y_1 \\ 0 & 0 & 0 & 1 & x_2 & y_2 \\ 0 & 0 & 0 & 1 & x_3 & y_3 \end{pmatrix} \begin{pmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \\ a_6 \end{pmatrix}. \tag{6}$$

If we assume

$$[\phi] = \begin{bmatrix} 1 & x_1 & y_1 \\ 1 & x_2 & y_2 \\ 1 & x_3 & y_3 \end{bmatrix}, \tag{7}$$

then

Fig. 18 Extraction of feature points on a boundary.

(a) (b)

Fig. 19 Matching points for extracted feature points in Fig. 18, (a) left and (b) right reference views.

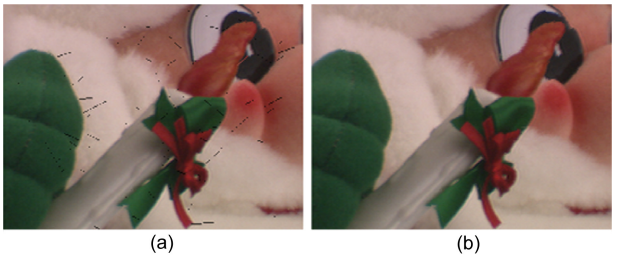
(a) (b)

Fig. 20 Triangular mesh for Fig. 19, (a) left and (b) right reference views.

Fig. 21 Hole-filling for disocclusion regions.

(a) (b)

Fig. 22 (a) Boundary trails and (b) removal of boundary trails.



(a) (b)

Fig. 23 Removal of discontinuous noises, (a) before and (b) after noise removal.

$$\mathbf{d}(x,y)=\begin{bmatrix}1 & x & y & 0 & 0 & 0\\ 0 & 0 & 0 & 1 & x & y\end{bmatrix}\begin{bmatrix}[\phi]^{-1} & 0\\ 0 & [\phi]^{-1}\end{bmatrix}\begin{bmatrix}u_1\\ u_2\\ \vdots\\ v_2\\ v_3\end{bmatrix}, \tag{8}$$

$$[\phi]^{-1}=\frac{1}{\det(\phi)}\begin{bmatrix}x_2y_3-x_3y_2 & x_3y_1-x_1y_3 & x_1y_2-x_2y_1\\ y_2-y_3 & y_3-y_1 & y_1-y_2\\ x_3-x_2 & x_1-x_3 & x_1-x_1\end{bmatrix}=\begin{bmatrix}\alpha_1 & \alpha_2 & \alpha_3\\ \beta_1 & \beta_2 & \beta_3\\ \gamma_1 & \gamma_2 & \gamma_3\end{bmatrix}, \tag{9}$$

where

$$\det(\phi)=x_2y_3+x_1y_1+x_1y_2-x_2y_1-x_3y_2-x_1y_3. \tag{10}$$

Thus, the disparity $\mathbf{d}(x,y)$ for inner point (x,y) is depicted as

$$\mathbf{d}(x,y)=\sum_{k=1}^{3}\phi_k(x,y)d_k,$$

$$\text{where } \phi_k=\alpha_k+\beta_k x+\gamma_k y, \quad \mathbf{d}_k=\begin{bmatrix}u_k\\ v_k\end{bmatrix}. \tag{11}$$

2.5 *Hole-Filling for Disocclusion Regions*

In Sec. 2.3, we proposed the mesh clustering to handle disocclustion regions and we achieved a result in Fig. 16. As you see, the disocclusion regions are represented as holes and they originally belong to background regions. We assume that the left holes of the foreground exist in the left reference view and the right holes of the foreground exist in the right reference view. Figure 17 illustrates the synthesized image only using background mesh. Before the disparity redefinition, the vertices of the boundary mesh element belong to a different cluster, and we can detect the feature points of the disocclusion regions by using the preceding fact. The results are shown in Fig. 18.

For each feature point, we find the matching point in the left or right reference view. The feature points on edge boundary and its matching points are represented as black points and white points, respectively, in Fig. 19. Then we compose the triangular mesh, as shown in Fig. 20, and we fill the holes using mesh elements just obtained. Figure 21 shows the synthesized image with the background mesh and hole-filling for disocclusion regions.

2.6 *Postprocessing*

The proposed view synthesis efficiently handles the disocclusion regions by mesh clustering, however, it causes discontinuous boundary trails, as shown in Fig. 22(a). To remove the boundary trails, we adopt the in-painting technique. The result is shown in Fig. 22(b).

In addition, noises caused by discontinuity between mesh elements and truncation errors of the affine-

(a) (b)

Fig. 24 Test sequences (a) "Santa" and (b) "Kid."

transformed coordinates are removed by median filtering. We replace only the noisy regions with median filtered regions. Figure 23 shows the result of the removal of discontinuous noises.

3 Experimental Results and Analysis

To evaluate the performance of the proposed method, we compared the synthesized results with a conventional mesh-based view synthesis method based on Ref. 19. We used the "Santa" and "Kid" sequences, as shown in Fig. 24. The 2nd, 7th, and 10th view positions in Fig. 2 were set as virtual viewpoints and the neighboring views were used as reference views for view synthesis. As shown in Figs. 25 and 26, the synthesized images are subjectively as natural as captured by a camera.

The objective results are given in Table 1. The objective quality is measured by peak SNR (PSNR) values between the synthesized image and original image, as in Eq. (12). The mean-squared error (MSE) between the original image (I_{org}) and the rendered image (I_{syn}) as in Eq. (13).

$$\mathrm{PSNR} = 10 \times \log_{10}(255^2/\mathrm{MSE}), \tag{12}$$

$$\mathrm{MSE} = \frac{1}{w \times h} \sum_{i=0}^{w-1} \sum_{j=0}^{h-1} \|I_{org}(i,j) - I_{syn}(i,j)\|^2, \tag{13}$$

where w and h represent the image width and image height, respectively.

The proposed method synthesized a high-quality virtual image, especially for a simple contents scene consisting of foreground and background.

Fig. 25 Synthesized Image for "Santa" Sequence (7th view).

Fig. 26 Synthesized Image for "Kid" Sequence (7th view).

4 Conclusions

We proposed an intermediate view synthesis method using a rectangular multiview video camera system to realize 3-D video systems. The proposed view synthesis method employed three reference views and a mesh-based scheme. The mesh was composed of feature points obtained by edge detection, which were clustered to handle disocclusion regions. We used the affine transform for synthesis and adopted the in-painting and median filter to remove the remaining erroneous and discontinuous regions. We are convinced that the proposed intermediate view synthesis and rectangular multiview video are useful to realize 3-D video systems. Our experiments confirmed that the proposed method synthesized a subjectively natural images and its average objective quality was over 34 dB.

Table 1 Experimental results for the proposed view synthesis.

Experimental Results for the "Santa" Sequence

View Number	Previous	Method 1	Method 2
2nd view	29.23 dB	31.60 dB	32.02 dB
7th view	32.33 dB	32.95 dB	33.32 dB
10th view	32.11 dB	32.21 dB	32.40 dB
Average	31.22 dB	32.25 dB	32.58 dB

Experimental Results for the "Kid" Sequence

View Number	Previous	Method 1	Method 2
2nd view	34.85 dB	35.18 dB	35.45 dB
7th view	35.30 dB	35.92 dB	36.07 dB
10th view	34.66 dB	34.60 dB	34.80 dB
Average	34.94 dB	35.23 dB	35.44 dB

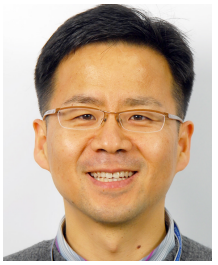
Previous: previous mesh-based method;[19] Method 1: previous +proposed mesh clustering; Method 2: method 1+proposed post-processing.

Acknowledgment

This work was supported in part by the Ministry of Knowledge Economy (MKE) through the next-generation project.

References

1. M. Tanimoto, "Overview of free viewpoint television," *Signal Process. Image Commun.* **21**, 454–461 (July 2006).
2. A. Smolic and P. Kauff, "Interactive 3D video representation and coding technologies," *Proc. IEEE* **93**, 99–110 (2005).
3. M. Tanimoto, "Free viewpoint television—FTV," in *Proc. Lecture Notes in Computer Science (LNCS)* , vol. 3331, pp. 497–504 (2004).
4. F. Isgro, E. Trucco, P. Kauff, and O. Schreer, "3D image processing in the future of immersive media," *IEEE Trans. Circuits Syst. Video Technol.* **14**(3), 288–303 (2004).
5. A. Smolic and D. McCutchen, "3DAV exploration of video-based rendering technology in MPEG," *IEEE Trans. Circuits Syst. Video Technol.* **14**(3), 348–356 (2004).
6. A. Redert, M. Op de Beeck, C. Fehn, W. IJsselsteijn, M. Pollefeys, L. van Gool, E. Ofek, I. Sexton, and P. Surman, "ATTEST: advanced three-dimensional television system technologies," *3D Data Processing Visualization and Transmission, 2002, First Int. Symp.*, pp. 313–319 (June 2002).
7. Mars Pathfinder Mission, http://mars.jpl.nasa.gov/MPF/index0.html.
8. MIT Museum: Collections—Holography, http://web.mit.edu/museum/collections/holography.html.
9. Ultra-Realistic Communications Forum, http://www.scat.or.jp/urcf/forum/index.html.
10. S. C. Chan, H. Y. Shum, and K. T. Ng, "Image-based rendering and synthesis," *IEEE Signal Process. Mag.* **23**(6), 22–33 (Nov. 2007).
11. M. Lhuillier and L. Quan, "Image interpolation by joint view interpolation," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 139–145, Fort Collins CO (1999).
12. S. Sethuraman, "Stereoscopic image sequence compression using multiresolution and quadtree decomposition based disparity—and motion-adaptive segmentation," PhD Diss. Carnegie Mellon University (1996).
13. Y. Altunbasak and A. M. Tekalp, "Occlusion-adaptive, content based mesh design and forward tracking," *IEEE Trans. Image Process.* **6**, 1270–1280 (Sep. 1997).
14. R. C. Gonzales and E. W. Richard, *Digital Image Processing*, 2nd ed., Prentice-Hall, Englewood Cliffs, NJ (2002).
15. J. Canny, "A computational approach to edge detection," *IEEE Trans. Pattern Anal. Mach. Intell.* **8**, 679–714 (1986).
16. P. Cignoni, C. Montani, and R. Scopigno, "DeWall: a fast divide and conquer Delaunay triangulation algorithm in Ed," *CAD* **5**, 333–341 (Apr. 1998).
17. D. V. Papadimitriou and T. J. Dennis, "Epipolar line estimation and rectification for stereo image pairs," *IEEE Trans. Image Process.* **5**(4), 672–676 (1996).
18. R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, Cambridge University Press, UK (2000).
19. J. H. Park and H. W. Park, "Fast view interpolation of stereo images using image gradient and disparity triangulation," *Signal Process. Image Commun.* **18**(5), 401–416 (2003).



Byeongho Choi received his BS and MS degrees in electronic engineering from the University of Hanyang, Republic of Korea, in 1991 and 1993, respectively. From 1993 to 1997, he had 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 chief manager of the Multimedia IP Research Center. He is also 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.

Taewan Kim received his BS degree in electronic engineering from SungKyunKwan University (SKKU), Seoul, Korea, in 2000 and his MS degree in information and communications engineering from Gwanju Institute of Science and Technology (GIST), Korea, in 2002. From January 2002 to May 2006, he was a researcher with the Korean Electronic Technology Institute (KETI), SungNam, where he was involved in several digital signal processing and digital multimedia projects. Since June 2006, he has been a researcher with the Korean Agency for Technology and Standards (KATS), where he has been participating in planning for Korean standards policy and establishing standards for information systems.

Kwan-Jung Oh received his BS degree in electronic computer engineering from Chonnam University, Gwangju, Korea, in 2002 and his MS degree in information and communications engineering from Gwangju Institute of Science and Technology (GIST), Korea, in 2005, where he is currently working toward his PhD degree in the Information and Communications Department. In 2008 he was an intern at Mitsubishi Electric Research Laboratories (MERL), Cambridge, Massachusetts His research interests include digital image and video coding, multiview video coding (MVC), 3-D video, free-viewpoint television (FTV), depth video coding, image-based rendering, and realistic broadcasting.

Yo-Sung Ho received both his BS and MS degrees in electronics engineering from Seoul National University, Korea, in 1981 and 1983, respectively, and his PhD degree in electrical and computer engineering from the University of California, Santa Barbara, in 1990. He joined the 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 the 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, image analysis and image restoration, advanced coding techniques, digital video and audio broadcasting, 3-D television, and realistic broadcasting.

Jong-Soo Choi received his BS degree from Inha University, Inchon, Korea, his MS degree from Seoul National University, Korea, and his PhD degree from Keio University, Yokohama, Japan, all in electrical engineering, in 1975, 1977, and 1981, respectively. He joined the faculty at Chung-Ang University in 1981, where he is now a dean of the Graduate School of Advanced Imaging Science, Multimedia, and Film. His current research interests are computer vision, image coding, and electro-optical systems.