2D/3D Freeview Video Generation for 3DTV System

Dongbo Min, [1] Donghyun Kim, [1] SangUn Yun, [2]
Kwanghoon Sohn *,[1]

Yonsei University, Shinchon-dong, Seodaemun-gu, Seoul, South Korea. [1]
Samsung Electronics, Suwon, South Korea. [2]

Abstract

In this paper, we present a new method of synthesizing novel views from the virtual cameras in multiview camera configurations for 3DTV system. We introduce a semi N-view & N-depth framework in order to estimate disparity maps efficiently and correctly. This framework reduces redundancy on disparity estimation by using information from neighboring views. N views can be classified as reference and target images. The disparity maps on the reference images are only estimated by using the cost aggregation method with the weighted least square. The cost functions on the target images are computed by the proposed warping technique so that significant reduction of computation loads is possible. The occlusion problem, which significantly affects the quality of virtual view rendering, is handled by using cost functions computed with multiview images. The proposed method provides a 2D/3D freeview video for 3DTV system. User can select 2D/3D modes of freeview video and control 3D depth perception by adjusting several parameters in 3D freeview video. Experimental results show that the proposed method yields the accurate disparity maps and the synthesized novel view is satisfactory enough to provide seamless freeview videos for 3DTV system.

Key words:
Occlusion handling, semi N-view & N-depth framework, stereo matching, virtual view rendering.

* Corresponding author
Email address: khsohn@yonsei.ac.kr (Kwanghoon Sohn).
[1] School of Electrical and Electronic Engineering, Yonsei University, South Korea
[2] Samsung Electronics, South Korea

1 Introduction

By recent advance in the multimedia processing fields, 3-dimensional TV (3DTV) is expected to become one of the most dominant markets in the next generation broadcasting system. In conventional TV, the viewpoint of a user is dependent to the acquisition camera, in other words, it can only provide a user subjective video. The basic concept of 3DTV is to provide user interactivity and 3D depth feeling. User interactivity means that 3DTV can provide a user the freedom of selecting viewpoint. 3DTV can also provide a user 3D impression as if he is really over there, by displaying 3D images on the 3D display monitors of glasses/non-glasses types. Development of 3DTV requires the ability of capturing and analyzing the multiview images and compressing and transmitting huge amount of data in communication network [1]. Matusik implemented 3DTV prototype system with real-time acquisition, transmission and 3D display of dynamic scenes [2].

Novel view rendering is an important technique in the 3DTV applications, and many methods have been proposed to solve this problem in the area of image-based rendering (IBR). It can provide reality and interactivity by enabling specific users to select different viewpoints. Since various viewpoints are provided with a limited number of cameras, it is useful to reduce an amount of data and a cost for constructing 3DTV system. It is also necessary in the aspects of compensating for discordances between 3D capturing and display formats. IBR can be classified into three categories according to the estimation of geometry: rendering without geometry, rendering with explicit geometry, and rendering with implicit geometry.

Light field and lumigraph approaches that use the rendering without geometry can perform photorealistic rendering with simple planar geometry representation [3][4][5]. It is possible to synthesize novel views based on densely sampled reference images without estimating accurate geometry information. However, a significant number of 2D images must be used to reconstruct a function that defines the flow of light through the 3D space. A second group of researches reconstruct a complete 3D model from 2D images and render the model from the desired viewpoint. This method consists of estimating the 3D depth information and integrating this depth information to generate a complete 3D model of a given scene. The difficulties of generating complete 3D models have caused these approaches to be used in limited applications only [6][7].

Rendering with implicit geometry synthesizes novel views from a virtual camera with a small number of cameras. These reference images are warped by geometry information and novel views can be computed by the weighted-interpolation of the warped images [8][9]. This method consists of image rectification, disparity estimation, image warping, and view interpolation. A number

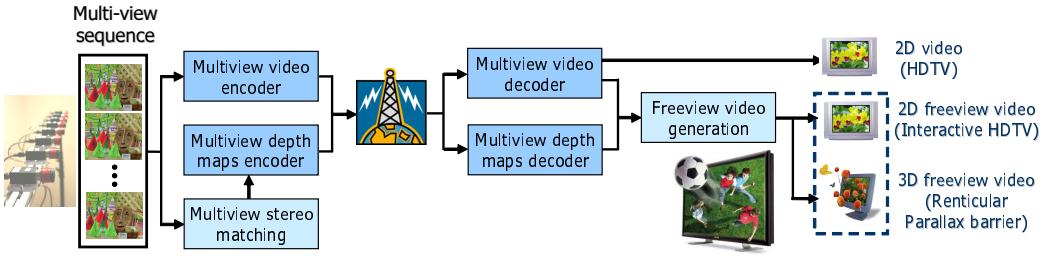


Fig. 1. An example of 3DTV system.

of view interpolation approaches have been proposed to improve performance.

Park *et al.* described the method to estimate a disparity map from multiple images and then warp this map onto neighboring images [18]. Zhang *et al.* proposed a method of reconstructing intermediate views from stereoscopic images [10]. Criminisi *et al.* introduced the novel view synthesis method for one-to-one teleconferencing [11]. These researchers used a stereo algorithm based on improved dynamic programming for efficient novel view generation, and proposed a compact geometric method for novel view synthesis by direct projection of the minimum cost surface. A mesh-based representation method for the disparity map of the stereo images was used for the view interpolation and stereo image compression [12]. Lhuillier and Quan introduced joint view triangulation to efficiently handle the visibility and occlusion problems created by the parallaxes between the images [13].

Redert *et al.* and Kauff *et al.* introduced an advanced approach for 3DTV systems based on the concept of video-plus-depth data representation [14][15]. In this paper, the video-plus-depth data representation method is called the N-view & N-depth framework, where N is the number of cameras in a multiview camera configuration. It focuses on providing a modular and flexible system architecture that can support a wide range of multi-view structures. Zitnick *et al.* proposed a way of performing high-quality novel view interpolation by using multiple synchronized video streams [16] [17]. The depth maps were extracted using the calibration information of cameras directly without rectifying the multiview images in the N-view & N-depth framework. A color segmentation algorithm was used to improve performance and provide robustness to noise and an intensity bias. Visible artifacts on the object boundaries in virtual view rendering were handled by computing the matting information within the pixels from all the depth discontinuities.

In this paper, we propose a new way of synthesizing novel views from virtual cameras based on sparsely sampled reference images. We reduce redundancy when estimating disparity maps in the semi N-view & N-depth framework. The conventional method estimates the disparity maps in the same manner for the N images in the N-view & N-depth framework. The occlusion problem, which significantly affects the quality of synthesized images in virtual view

rendering, is handled by using cost functions computed with multiview images.

We provide the user with 2D/3D freeview videos. 3D freeview videos are provided as stereoscopic images. These images are generated by synthesizing two novel views - one for the left view and one for the right view. Users can select the freeview video mode according to the display device, and control 3D perception by adjusting several parameters in 3D freeview video. Most conventional methods provide users with 2D freeview video [16] [17] or 3D video at one fixed viewpoint by synthesizing intermediate views when stereo images are given [10]. We propose a more flexible system for 3DTV by making it possible for the user to select 2D/3D modes.

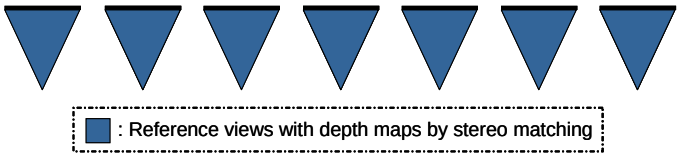
Fig. 1 shows an example of 3DTV system. The multiview images and the associated depth maps estimated by the stereo matching method are transmitted through communication network. In receiver side, the user can select the modes of videos according to his preference, which are 2D video, 2D freeview video and 3D freeview video. We propose a new approach for efficient multiview depth estimation and virtual view generation, which are key technologies for 3DTV system.

The remainder of this paper is organized as follows. In section II, we describe the motivation and overview of the proposed method, and then explain the cost aggregation method for stereo matching in section III. Virtual view rendering in the semi N-view & N-depth framework is described in section IV. Finally, we present the experimental results and conclusions in sections V and VI, respectively.

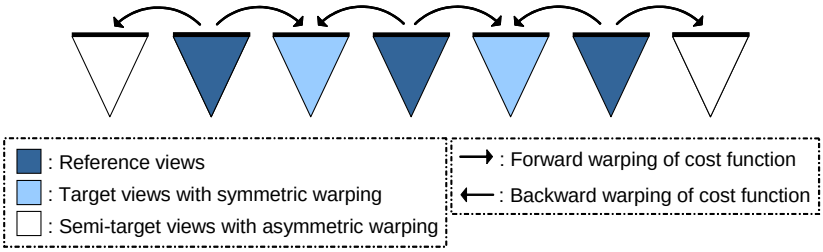
2 Motivation and Overview

When N images are given, it is necessary to acquire N depth maps for rendering novel views from virtual cameras in multiview camera configurations. Given these novel viewpoints, the nearest images are warped with the associated disparity maps, and novel views are synthesized by interpolating these warped images. Zitnick *et al.* proposed a way of synthesizing intermediate views by estimating N depth maps from multiview images [16] [17]. Disparity estimation was performed off-line due to the huge computational loads and intermediate view rendering was applied in real-time by using GPU hardware. One of the most serious implications of the huge computational loads is that the disparity maps for all the images can all be estimated in the same manner. It is known that the disparity maps for neighboring images are generally similar to each other. Fig. 2 shows the 'Breakdancer' image pairs and the disparity maps provided by Zitnick *et al.* [16]. The disparity maps are very similar to each other, except in the occluded parts. By considering this redundancy, we

Fig. 2. 'Breakdancer' color images and depth maps: (from top to bottom) view 3 and 4.



(a) N-view & N-depth framework

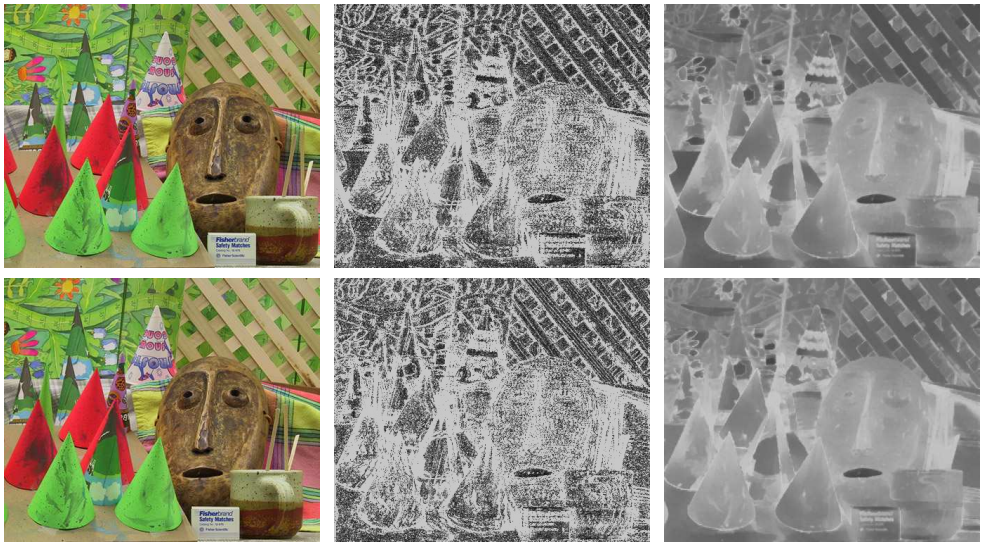


(b) Semi N-view & N-depth framework

Fig. 3. Representation of multiview images and depth data: N-view & N-depth framework and semi N-view & N-depth framework

are able to reduce the complexity in the N-view & N-depth framework.

Based on this observation, we propose semi N-view & N-depth framework to reduce the redundancy of estimating the disparity maps in multiview images. Fig. 3 shows the concepts of the N-view & N-depth and semi N-view & N-depth frameworks. In the N-view & N-depth framework, a disparity map for each image is estimated independently in the same manner, although



(a) (b) (c)

Fig. 4. Costs of the multiview images for (from top to bottom) views 3 and 5 of the 'Cone' image pairs: (a) original images, (b)(c) per-pixel cost $e(p, d)$ and estimated cost $E(p, d)$ when disparity d is 3. The estimated costs are computed with the proposed cost aggregation method.

corresponding pixels, visible pixels on neighboring images, contain the same disparity (depth) values. In contrast, we use the estimated information of neighboring images to estimate the disparity map of one image.

The multiview images (N views) are classified into reference and target images. The target images are divided into target and semi-target images. The efficient cost aggregation method with the weighted least square [19] is only used to estimate the disparity maps of the reference images. The disparity maps of the target images are acquired by warping the cost functions of the reference images. Note that the cost functions of the reference images are transferred into those of the target images, not the disparity values. As shown in Fig. 3, symmetric warping on the target images means that both forward and backward warping are done from neighboring reference images, while asymmetric warping on the semi-target images means that either forward or backward warping are done. Note that the leftmost and rightmost views, in other words, 0 and $N - 1$ views, are always semi-target images. This process is based on the assumption that corresponding pixels on neighboring images have similar cost functions. Fig. 4 shows the estimated cost functions of the 'Cone' image pairs. The cost functions of views 3 and 5 are used for comparison, when the disparities are 3. We find that the cost functions of the two images are very similar to each other.

Fig. 5 shows the overall framework of the proposed system. We acquire N

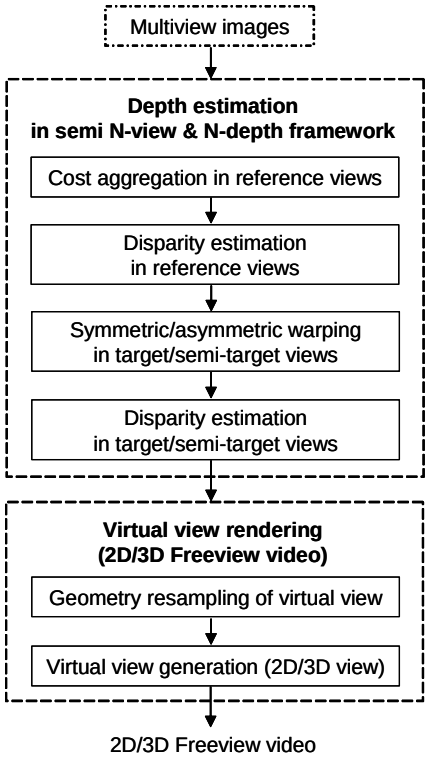


Fig. 5. Overall framework of the proposed method.

depth maps from the semi N-view & N-depth framework, and synthesize the novel views from the virtual camera for the given viewpoint. Each part of Fig. 5 will be explained in the following sections.

3 Stereo Matching with Multiview Images

We use a multiview camera configuration for estimating the disparity maps and rendering virtual views. An extensive review of stereo matching algorithms can be found in [20]. In this paper, our aim is to develop a 2D/3D freeview video generation system. Thus, a parallel camera structure is used, since multiview camera configuration with a toed-in structure may cause a number of holes in the synthesis of 3D freeview videos. We assume that the baseline distances between the captured cameras are the same as B.

3.1 Per-pixel Cost Computation

When estimating the disparity field, two or more images are used. The difference image is computed for each image based on the constant brightness assumption. Let $i-1^{th}$, i^{th} and $i+1^{th}$ images left, center and right images, re-

spectively. Since the multiple images are rectified into horizontal direction, we obtain the difference image of center image by shifting the left (or right) image to the right (or left) direction, and the subtracting the center and shifted left (or right) images. The difference image $e_{i,j}(p,d)$ for i^{th} image is computed with the i^{th} and j^{th} images, as follows:

$$\begin{aligned} e_{i,i+1}(p,d) &= \min\{|I_i(x,y) - I_{i+1}(x+d,y)|, T\} \\ e_{i,i-1}(p,d) &= \min\{|I_i(x,y) - I_{i-1}(x-d,y)|, T\} \end{aligned} \tag{1}$$

where p and d represent the 2D locations of the pixels and the disparity, respectively. I is the intensity when using RGB color, and T is the threshold that defines the upper bound of the matching cost function. We compute the per-pixel cost $e_i(p,d)$ with the $e_{i,i+1}$ and $e_{i,i-1}$ values. When computing the per-pixel cost, we consider whether the pixels in the center image are visible or occluded. We assume that all the pixels in the center image have at least one corresponding point for two neighboring (left and right) images. The occluded pixels are compensated for by using the cost functions of the multiview images. This assumption is useful for handling occlusion, although it is invalid in a few pixels. When the corresponding points are visible in three images, the per-pixel cost is computed with both the $e_{i,i+1}$ and $e_{i,i-1}$ values. However, if it is visible in only one of the two reference images, the per-pixel cost have to be computed with only one visible point among them. Based on the principle that the matching cost of visible pixels is generally smaller than that of occluded pixels, we compute the per-pixel cost for the center (i^{th}) image as follows:

$$e_i(p,d) = \min(e_{i,i+1}(p,d), e_{i,i-1}(p,d)) \tag{2}$$

While most approaches detect the occluded regions by using a uniqueness constraint and assign pre-defined values to the occluded pixels, we address the occlusion problem by using the cost functions of the multiview images.

3.2 Cost Aggregation with Weighted Least Square [19]

In order to estimate the optimal cost $E_i(p,d)$ for given the per-pixel cost $e_i(p,d)$ of the i^{th} image, we use a prior knowledge that costs should vary smoothly, except at object boundaries. Moreover, it is necessary to gather sufficient texture in the neighborhoods for computing optimal cost. From this assumption, we are able to estimate the cost by using nonlinear iterative filtering in the weighted least square framework [19] as follows:

$$
\begin{aligned}
E^{k+1}(p) &= \bar{e}(p) + \bar{E}^k(p) \\
&= \frac{e(p) + \lambda \sum_{m \in N_c(p)} w(p,m) E^{k+1}(m) + \lambda \sum_{m \in N_n(p)} w(p,m) E^k(m)}{1 + \lambda \sum_{m \in N(p)} w(p,m)}
\end{aligned} \tag{3}
$$

$$
N(p) = \{(x + x_n, y + y_n) | -M \leq x_n, y_n \leq M, \;\; x_n + y_n \neq 0\}
$$

where $N_c(p)$ and $N_n(p)$ are the causal and non-causal parts of $N(p)$, and $N(p) = N_c(p) \cup N_n(p)$. w represents the weighting function between corresponding neighboring pixels. w is a weighting factor that controls an ratio of the per-pixel cost and estimated cost. We simplify $E_i(p, d)$ to $E(p)$, since the same process is performed for each disparity value. Eq. (3) consists of two parts: normalized per-pixel matching cost and weighted neighboring pixel cost. By running the iteration scheme, the cost function E is regularized with the weighted neighboring pixel cost. In the proposed method, we use the asymmetric Gaussian weighting function with the CIE-Lab color space in Eq. (4). r_c and r_s are the weighting constants for the color and geometric distances, respectively. If C_i is the color distance that is computed with the i^{th} image, the weighting function is defined as follows.

$$
\begin{aligned}
w(p,m) &= \exp\left(-\left(\frac{C_i(p,m)}{2r_c^2} + \frac{S(p,m)}{2r_s^2}\right)\right) \\
C_i(p,m) &= (L_i(p) - L_i(m))^2 + (a_i(p) - a_i(m))^2 + (b_i(p) - b_i(m))^2 \\
S(p,m) &= (p - m)^2
\end{aligned} \tag{4}
$$

We use a multiscale approach to accelerate the convergence of Eq. (3). We can initialize the value close to the optimal cost in each level by using the final value in the coarser level. We first compute the 3D cost volume and then perform the proposed multiscale scheme for each 2D cost function. The proposed multiscale method runs the iterative scheme at the coarsest level by initializing the cost function to $e(p, d)$. After K iterations, the resulting cost function is used to initialize the cost function at the finer level, and this process is repeated until the finest level is reached. The proposed multiscale scheme is shown in Fig. 6, which includes adaptive interpolation.

When the cost function on the $(l+1)^{th}$ level is defined as $E_{l+1}(p)$, we are able to refine the resolution of the cost function $E_l(p)$ on the finer level by using adaptive interpolation [19]:

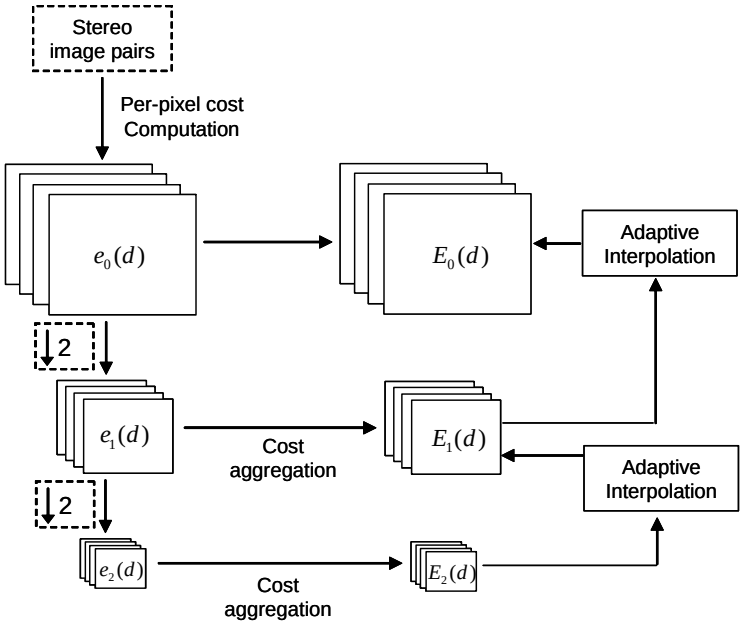


Fig. 6. Overall framework of the proposed cost aggregation.

$$E_l(p) = \frac{e_l(p) + \lambda_a \sum\limits_{p_m \in N(p_c)} w_{p,p_m} E_{l+1}(p_m)}{1 + \lambda_a \sum\limits_{p_m \in N(p_c)} w_{p,p_m}} \tag{5}$$

where $p_c = (x_c, y_c)$ represents a pixel on the coarser level, and $N(p_c)$ on the $(l+1)^{th}$ level is a set of 4-neighboring pixels. We set the weighting factor to $\lambda_a = 15$. Another advantage of adaptive interpolation is to increase the resolution of the cost function so that no blocking artifact exists, so that it is not necessary to perform the cost aggregation scheme on the finest level, and this makes the proposed method faster.

4 Virtual View Rendering

We estimate the disparity maps by using cost aggregation, and handle the occlusion problem by using cost functions computed with multiview images. It is necessary to acquire N depth maps for virtual view rendering in multiview camera configuration. In this section, we propose a new approach which eliminates the redundancy of estimating the disparity maps in the semi N-view & N-depth framework. The virtual view can be synthesized by warping each image with its disparity map. Since the proposed system provides 2D/3D freeview video, users can select 2D/3D modes and control 3D depth perception by adjusting several parameters in 3D freeview video.

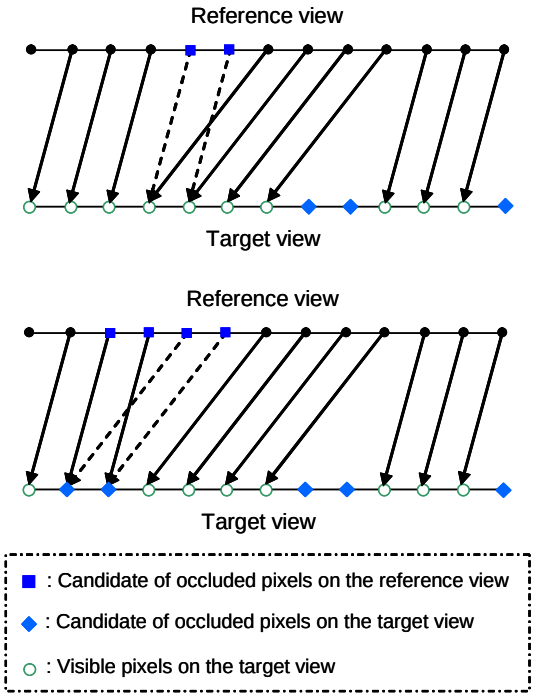


Fig. 7. Several cases of forward warping: (from top to bottom) when occluded pixels in the reference view are blocked by visible pixels, and when occluded pixels in the reference view block visible pixels.

4.1 Semi N-view & N-depth Framework by Warping of Aggregated Cost

In this section, we propose a new way of eliminating redundancy and reducing computational loads in the cost aggregation scheme. The cost functions in the reference images are estimated by using the proposed cost aggregation method with the weighted least square. The cost functions in the target and semi-target images are estimated through the warping of those in the reference images. The cost functions of the reference images are transferred into those of the target (semi-target) images with the corresponding disparity maps of the reference images. Since both forward and backward warping are performed in the target images, we are able to compensate for the occluded pixels, which are caused by other warping so that only few holes exist. However, since either forward or backward warping is performed in the semi-target images, the occluded parts as well as the holes remain. For example, since backward warping is only performed in the leftmost view 0, the occluded parts appear in the left side of the given object and in the left border region of the given image. For assigning a reasonable cost function to the occluded pixels and holes, we use the method of handling the occluded pixels and holes with reliable neighboring pixels in the cost aggregation scheme.

In general, the cost functions of visible pixels on two images should only be transferred through forward/backward warping. To determine whether a pixel

on the reference image is visible or not, we use geometric and photometric constraints. First, we explain the process of forward warping. We are able to estimate the visibility of the pixels by evaluating the disparity values of the neighboring pixels. The occluding pixel has the largest disparity among multiple matching pixels, so that the disparity of the occluding pixels is generally larger than that of the occluded pixels. Before we define the visibility function of the pixels based on this principle, we describe the function $S_t(j)$ for target image as a set of pixels in the reference image:

$$S_t(j) = \{i | i - d_r(i) = j, \;\; all\ i\ with\ \; 0 \leq i \leq W - 1\}$$

where i and j represent the x coordinates of the reference and target images, respectively. W represents the width of the image and d represents the disparity of the pixel. We define a visibility function O_t which takes the value 1 (or 0) when the pixel is visible (or occluded and hole). Approaches which exploit the uniqueness constraint determine the visibility function of the reference images with the disparity fields estimated from other images when there are multiple matching points at the pixels of the other images. However, the proposed method only uses the disparity fields on the reference image. When there are multiple matching points at pixels in the target image, that is, $\#(S_t(j)) \geq 1$, the pixel with the largest disparity among $S_t(j)$ is considered as visible and the remaining pixels as occluded. This is valid only if the occluding pixels have reliable disparities. Fig. 7 shows several cases of forward warping. If the disparities in the occluded pixels are smaller than those of the visible pixels, we are able to accurately detect the occluded region. Otherwise, the occluded pixels block the other visible pixels. We use the photometric constraint to evaluate the reliability of the occluding pixels. We determine a set of occlusion candidates instead of a set of occlusions on the target image by using this constraint. The occlusion candidates consist of both occlusion and holes. The costs at the occluded pixels are generally larger than those of the visible pixels. If the cost at the pixel, which is determined as occluding pixels by geometric constraints, is not smaller than that of the remaining occluded pixels, we can not guarantee the reliability of the occluding pixels. Therefore, all the pixels in $S_t(j)$ are used as occlusion candidates as shown in Fig. 7 (b), and $\#(S_t(j))$ is reset to 0. Then, the visibility function $O_t(j)$ on the target image is set to 0 when $\#(S_t(j)) = 0$, and otherwise, $O_t(j) = 1$. By using the visibility function O_t on the target image, we warp cost functions of reference image as follows:

$$E_t(i - d_r, d_r) = E_r(i, d_r), \qquad if\ O_t(i - d_r) = 1. \tag{6}$$

In Eq. (6), the y coordinate is omitted, since the same process is performed for each scanline. The process of backward warping is similar to that of forward warping. In backward warping, we define the function $S_t(j)$ as follows:

$$S_t(j) = \{i | i + d_r(i) = j, \;\; all\ i\ with\ \; 0 \leq i \leq W-1\}$$

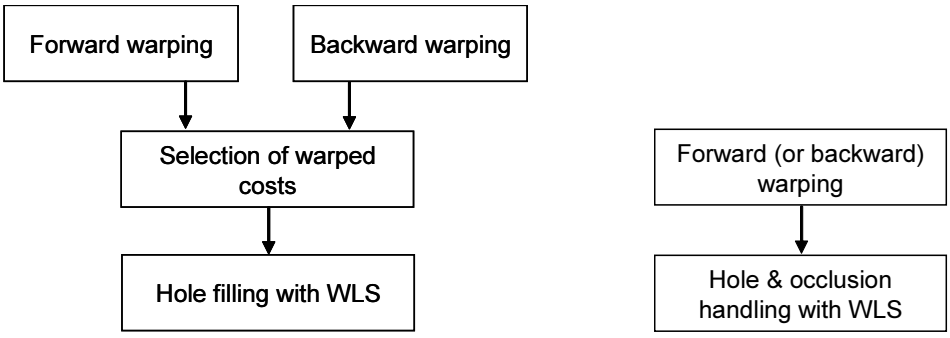
By using the visibility function O_t on the target image, we perform backward warping.

$$E_t(i + d_r, d_r) = E_r(i, d_r), \qquad if\ O_t(i + d_r) = 1 \tag{7}$$

Note that the cost functions of the reference images are transferred into those of the target images through the warping process, not the disparity values of the reference images. Both forward and backward warping compensate for the occluded pixels that are caused by other warping, and there are only a few holes in the target image. In the pixels of the target image where the cost functions are transferred by both forward and backward warping, we select the warping process in which the cost function for the warped disparity value is smaller. The occluded parts and holes in the target (semi-target) images are handled in the cost aggregation process. By using the visibility function O_t on the target (semi-target) image, we estimate the costs of the pixels in the candidate of occlusion in Eq. (8) as follows:

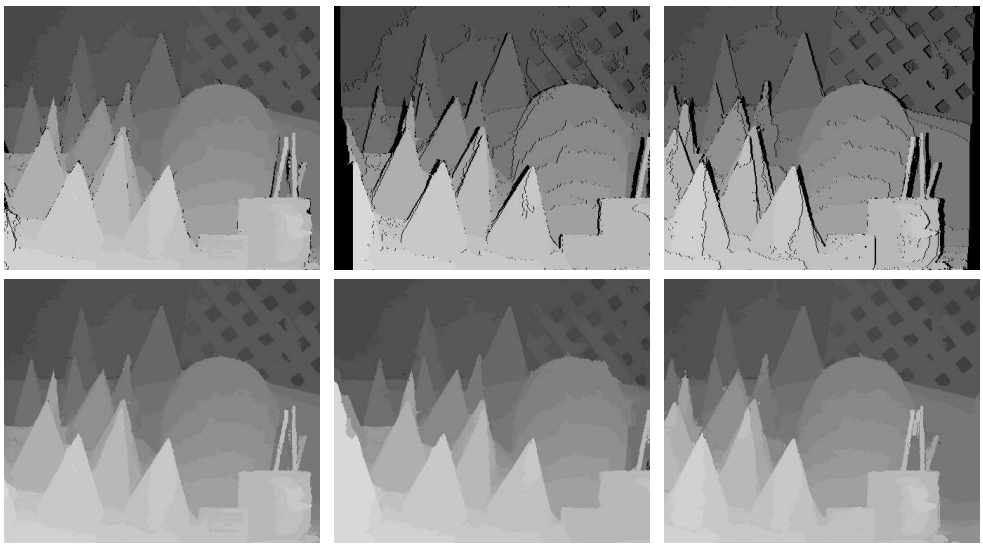
$$\begin{aligned}
&if\ O_t(p) = 0, \\
&\quad for\ k = 1 : K \\
&\qquad E_t^{k+1}(p) = \frac{\begin{array}{r} O_t(p)e(p) + \lambda \sum\limits_{m \in N_c(p)} O_t(m)w(p,m)E_t^{k+1}(m) \\ +\lambda \sum\limits_{m \in N_n(p)} O_t(m)w(p,m)E_t^{k}(m) \end{array}}{O_t(p) + \lambda \sum\limits_{m \in N(p)} O_t(m)w(p,m)} \\
&\quad end \\
&\quad O_t(p) = 1 \\
&end
\end{aligned} \tag{8}$$

O_t is 0, when a pixel is in the candidate of occlusion on the semi-target image, and 1 if otherwise. The costs of pixels in the candidate of occlusion are computed with those of visible pixels only. In Eq. (8), the number of iterations is $K = 1$, and we can estimate the costs of pixels in the candidate of occlusion after only one iteration. The updated pixels through Eq. (8) become visible, and they are sequentially used when the costs of other pixels are computed. Fig. 8 shows the process of symmetric/asymmetric warping. Generally, asymmetric warping is slower than symmetric warping, since occlusion handling is



(a) (b)

Fig. 8. (a) Symmetric and (b) asymmetric warping in the target and semi-target views.



(a) (b) (c)

Fig. 9. Results of symmetric and asymmetric warping for the 'Cone' image pairs: (a) symmetric warping in target view 4 (forward/backward warping and hole filling), (b) asymmetric warping in semi-target view 2 (backward warping and occlusion/hole handling), (c) asymmetric warping in semi-target view 6 (forward warping and occlusion/hole handling).

done in the asymmetric warping process. The results of symmetric and asymmetric warping are shown in Fig. 9. Five images (view $2 \sim 6$) from the 'Cone' image pairs are used [24]. Fig. 9 (a), (b) and (c) shows the results of symmetric warping in target view 4, and asymmetric warping in semi-target views 2 and 6, respectively. In the asymmetric warping process, reasonable disparity values are assigned to occluded pixels and holes through the proposed handling process. This is different from the extrapolation technique widely used for occlusion handling. While the extrapolation technique is just filling by us-

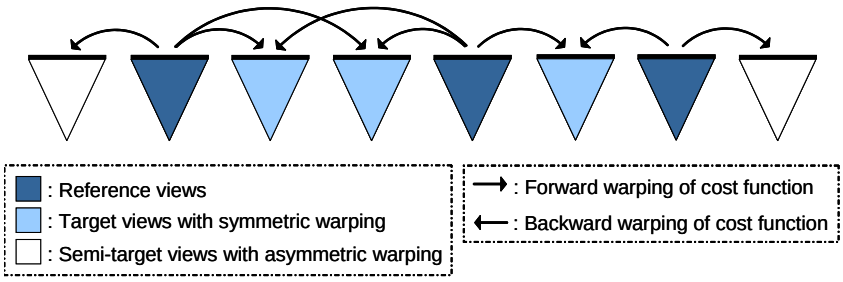


Fig. 10. Semi N-view & N-depth framework (when N is even): target views 2 and 3 are made by the warping reference views 1 and 4.

ing the disparities of the visible pixels, the proposed method propagates the information of the visible pixels into that of the occluded pixels and holes. In this paper, we use WTA (Winner-Takes-All) method as the optimization method for disparity estimation. Other optimization techniques such as graph cut and belief propagation [21] [22] can be used to perform disparity estimation in the warped cost function on the target and semi-target images instead of the WTA method.

Fig. 10 shows the semi N-view & N-depth framework when N is even. The images 1, 4 and 6 are used as for reference images. The cost function of the target images 2, 3 and 5 are made by symmetric warping of the reference images. The target images 2 and 3 are made by symmetric warping with the reference images 1 and 4. It will have been possible to compensate for the occluded pixels which are caused by other warping, although the reference images 1 and 4 are not the neighboring views of 3 and 2, respectively. There are always two semi-target views, except when N is 4. The number of reference and target images is defined as follows:

$$\#\text{ of reference views} = \left[\frac{\mathrm{N}-1}{2}\right]$$

$$\#\text{ of target views} = \left[\frac{N}{2}\right] - 1$$

$$\#\text{ of semi} - \text{target views} = 2$$

The ratio of the complexity value R with the N-view & N-depth and the semi N-view & N-depth framework is defined as follows:

$$R = \frac{C_1\left[\frac{N-1}{2}\right] + C_2\left(\left[\frac{N}{2}\right] - 1\right) + 2C_3}{C_1 N}$$

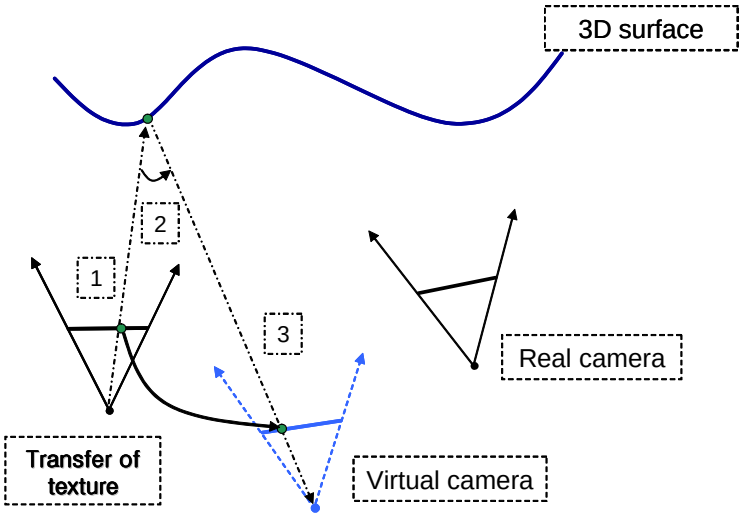


Fig. 11. Novel view rendering process (1: back-projection, 2: transformation of camera coordinates, 3: projection).

$$\cong \frac{C_1\left[\frac{N-1}{2}\right] + 2C_3}{C_1 N} \tag{9}$$

where C_1, C_2 and C_3 represent the complexity value for the reference, target and semi-target images, respectively. Since there are only a few holes in the target images as shown in Fig. 9 (a), the complexity of the target image C_2 is negligible compared to that of the reference image C_1. Moreover, the complexity value of the semi-target view is nearly $10 \sim 20$ percent of that of the reference view in our experiments. Therefore, we are able to reduce the complexity value by about 50 percent or more in the semi N-view & N-depth framework.

4.2 Novel View Generation

Given the N images and the associated disparity maps, the virtual views are synthesized by warping each image with their disparity maps. All the images are warped and a novel view is generated by performing a weighted-interpolation procedure. The method of synthesizing novel views from a virtual camera proceeds as follows:

1. We perform back-projection for all the pixels in the reference image into a 3D space by using the disparity map.
2. We transform the coordinate of the reference camera into the coordinate of the virtual camera.
3. We perform the projection of the 3D points into an image plane of the novel view.

Using the above process, the texture in the reference image is mapped into a novel view from the virtual camera. This process is performed for all the reference images. Fig. 11 shows the process of projecting the reference images. In this paper, all the images are disposed in the parallel structure. All the images are rectified and the viewing directions are the same, in other words, there is only translation between cameras. The translation of the virtual camera is only considered in the novel view rendering process. When using a novel viewpoint, the nearest two images (camera i and $i+1$) are selected and projected into the virtual view. A point $m_i(x, y)$ with the disparity value d_i on the i^{th} image is converted into the 3D point M_i as follows:

$$\left(\frac{(x_i - x_0)B}{d_i}, \frac{(y_i - y_0)B}{d_i}, \frac{fB}{d_i}\right) \tag{10}$$

where (x_0, y_0) represents the center of the image plane. When the transformation between the real and virtual cameras consists of the translation (T_x, T_y, T_z), we compute the 3D point M_i^v in the virtual camera coordinates:

$$\left(\frac{(x_i - x_0)B}{d_i} + T_x, \frac{(y_i - y_0)B}{d_i} + T_y, \frac{fB}{d_i} + T_z\right) \tag{11}$$

By projecting the 3D point M_i^v into an image plane of the virtual camera, we acquire the relationship between the corresponding pixels in the reference and virtual images. A point in the novel view $m_i^v(x_i^v, y_i^v)$ is computed as follows:

$$\begin{aligned} x_i^v - x_0 &= f\frac{(x_i - x_0)B/d_i + T_x}{fB/d_i + T_z} = \frac{x_i - x_0 + d_i\alpha_x}{1 + d_i\alpha_z/f} \\ y_i^v - y_0 &= f\frac{(y_i - y_0)B/d_i + T_y}{fB/d_i + T_z} = \frac{y_i - y_0 + d_i\alpha_y}{1 + d_i\alpha_z/f} \end{aligned} \tag{12}$$

To simplify this notation, we use a normalized coordinate $(\alpha_x, \alpha_y, \alpha_z) = (T_x, T_y, T_z)/B$, and set the baseline distance to 1. The novel view from the virtual camera is synthesized by projecting the reference images into the image plane of the virtual camera and then performing the weighted-interpolation process. If I_i and I_i^v are the reference and projected images, respectively, then $I_i^v(x_i^v, y_i^v) = I_i(x_i, y_i)$.

When the novel view is synthesized with the forward mapping of the texture information, there are some problems. Since the relationship in Eq. (12) is generally not one-to-one mapping, multiple projections and holes in the novel view can usually exist. Two or more pixels of the reference image can be projected into the same point of the novel view, and the holes can be generated

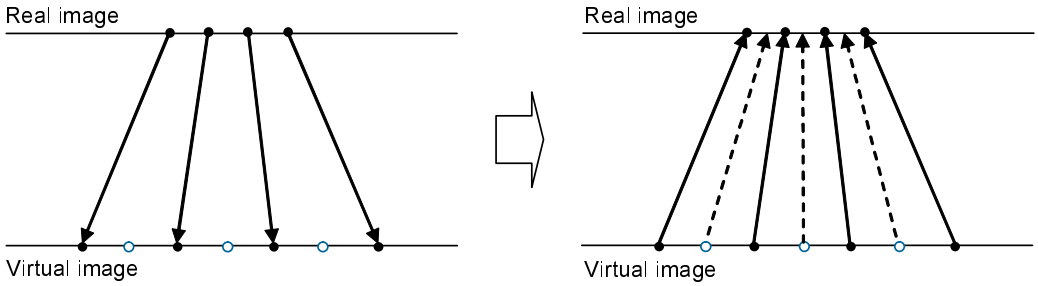


Fig. 12. Reverse mapping of disparity value and bilinear interpolation of intensity value.

by occlusion. Multiple projections into the novel view can be caused by two reasons: depth discontinuity and image resampling. The pixels on the depth discontinuities can be projected into the same point in the novel view, although they have different disparity values (depths). In this case, the pixel with the smallest depth value among the projected pixels should be retained since the pixel should cover the remaining pixels of objects farther from the camera. Since the distance between the objects and the camera is inversely proportional to the disparity value, we retain the pixel with the largest disparity value when it comes to rectified camera configuration. Another problem occurs with regard to image resampling. In general, when objects zoom out (or in) in novel view rendering, multiple projections (or holes) may be found, although they are equal disparity values. Moreover, the point (x_i^v, y_i^v) in the novel view may not be an integer.

In order to solve these problems, we adopt the reverse mapping and bilinear interpolation in the novel view rendering process. Given the novel viewpoint, we perform geometric resampling in the novel view, by transferring the depth and occlusion information to the novel view for each reference image, as shown in Fig. 12. Simple median filtering is performed in the depth and occlusion map to eliminate small holes. The reverse mapping process prevents the quality of the novel view from being degenerated by image resampling. Since it is known that disparity (or depth) varies smoothly, geometric resampling does not affect the quality of novel view rendering. This is different from image resampling.

Fig. 13 shows the movement of the virtual camera. The distance between the real cameras is normalized as 1.0 to simplify the notation. $(\alpha_x^g, \alpha_y^g, \alpha_z^g)$ is the global location of virtual camera. Virtual camera can move along x and z-axes, which include left, right, forward and backward movements. The y-axis movement is limited since this may cause some holes in the novel view. In general, it is possible to generate 2D or 3D freeview video by synthesizing one or two novel views, respectively. The final reconstructed novel view is computed by interpolation with the projected images as follows:

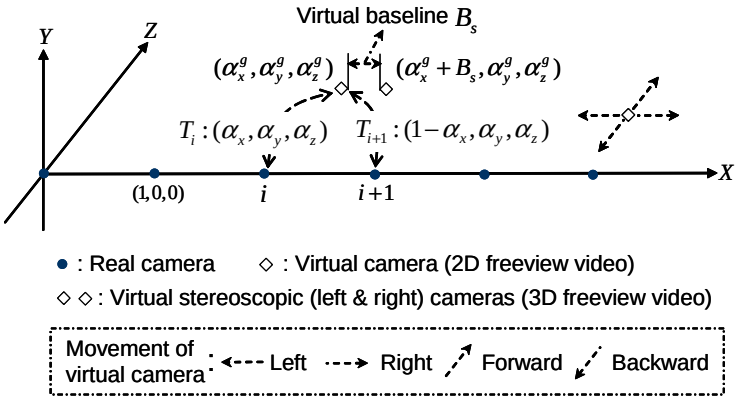


Fig. 13. Movement of virtual camera.

$$I_v(p) = V^i(p)(1-\alpha_x)I_v^i(p) + V^{i+1}(p)\alpha_x I_v^{i+1}(p) \tag{13}$$

where $\alpha(\alpha_x, \alpha_y, \alpha_z)$ represents the relative locations of the virtual camera. $V(p)$ is a visibility function that shows whether a pixel in the novel view is visible in the reference views, with values of 1 (or 0) when visible (or not). The visibility function $V(p)$ is defined when geometric resampling is performed.

4.3 Virtual 3D View Generation

We synthesize the stereoscopic novel view from the virtual camera. In general, this synthesis can be generated by synthesizing two novel views - one for the left view and one for the right view. The distance between the two novel views can be defined as B_s. In order to establish the zero parallax setting (ZPS), the CCD sensors of the stereoscopic cameras in the parallel structure are translated by a small shift h relative to the position of the lenses [23]. This makes us choose the convergence distance Z_c in the 3D scene. In general, this shift sensor concept is usually used as an alternative to the "toed-in" approach, because it does not cause keystone distortions and depth-plane curvature in stereoscopic images [23]. The sensor shift can be simply formulated as the displacement of a camera's principal point. When the horizontal shift of the principal point is defined as h, the point in the novel view is computed as follows:

$$x_i^v - (x_0 \pm h) = \frac{x_i - x_0 + d_i\alpha_x}{1 + d_i\alpha_z/f}$$

$$y_i^v - (y_0 \pm h) = \frac{y_i - y_0 + d_i\alpha_y}{1 + d_i\alpha_z/f} \tag{14}$$

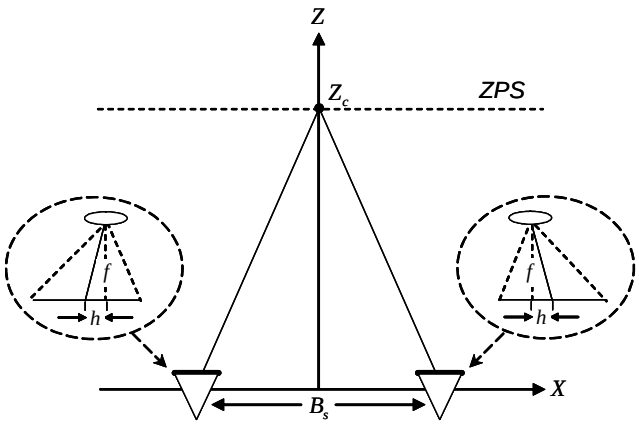


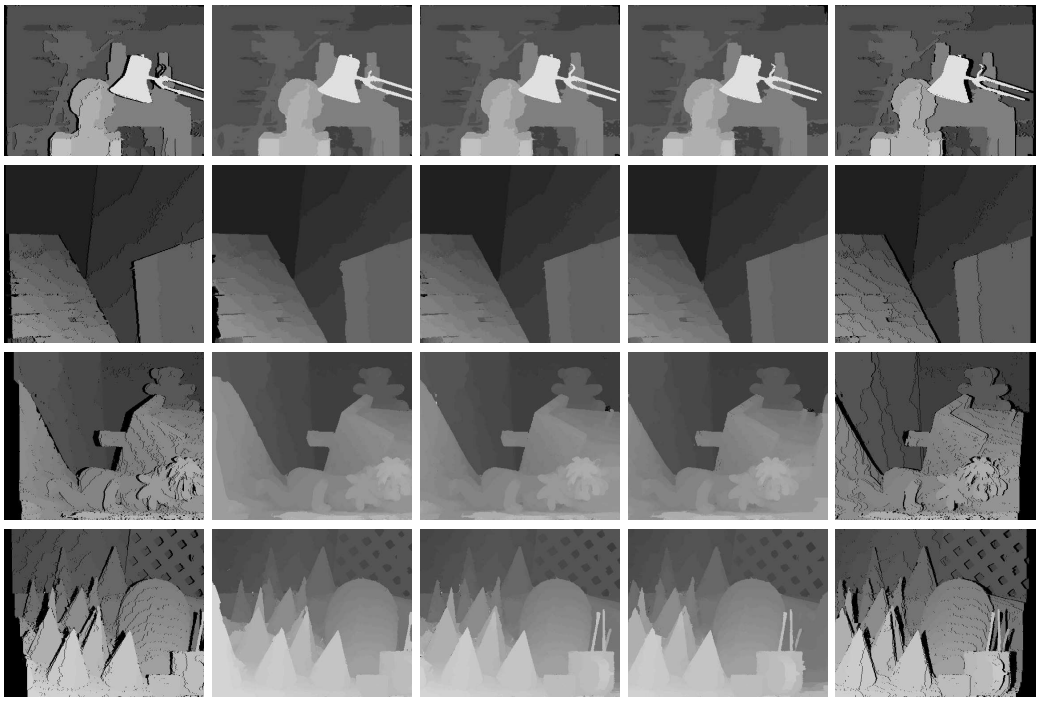
Fig. 14. Stereoscopic imaging using the shift sensor method. This figure is from [23].

where $\pm h$ means the shifted right and left images of the novel stereoscopic views, respectively. Fig. 14 shows stereoscopic imaging using the shift sensor method [23]. Scene parts that lie further away than the convergence distance Z_c are visualized behind the screen in a 3D display, and areas closer than Z_c are reproduced in front of the display in the viewer space [23]. Please refer to [23] for more detailed explanations for this method.

5 Experimental Results

To validate the performance of the semi N-view & N-depth framework, we performed the experiments with the Middlebury test sequences [24]. We used the following test data sets: 'Tsukuba' (384×288 pixels, search range: 16), 'Venus' (434×383 pixels, search range: 20), 'Teddy' (450×375 pixels, search range: 60), and 'Cone' (450×375 pixels, search range: 60). The 'Tsukuba' image set contains five color images (views 0-4), and the 'Venus', 'Teddy' and 'Cone' image sets contain nine color images (views 0-8). We used the images of the even views for 'Venus', 'Teddy' and 'Cone' image pairs. Note that the Middlebury stereo test bed performs objective evaluation with views 2 and 3 for the 'Tsukuba' image set, views 2 and 6 for the 'Venus', 'Teddy', and 'Cone' image sets, and provides ground truth maps of view 2 for the 'Tsukuba' image set and views 2 and 6 for the 'Venus', 'Teddy', and 'Cone' image sets.

The proposed method was tested using the same parameters for all the test images. The two parameters in the weighting function were $r_c = 8.0$, $r_s = 8.0$, and the weighting factor was $\lambda = 1.0$. We used the multiscale approach on four levels, and the number of iterations was $(3, 2, 2, \times)$, on a coarse to fine scale. The iteration number of the finest level was not defined since we used the adaptive interpolation technique in the up-sampling step, as mentioned in section III. The sizes of the sets of neighboring pixels were 5×5, 7×7, 9×9,

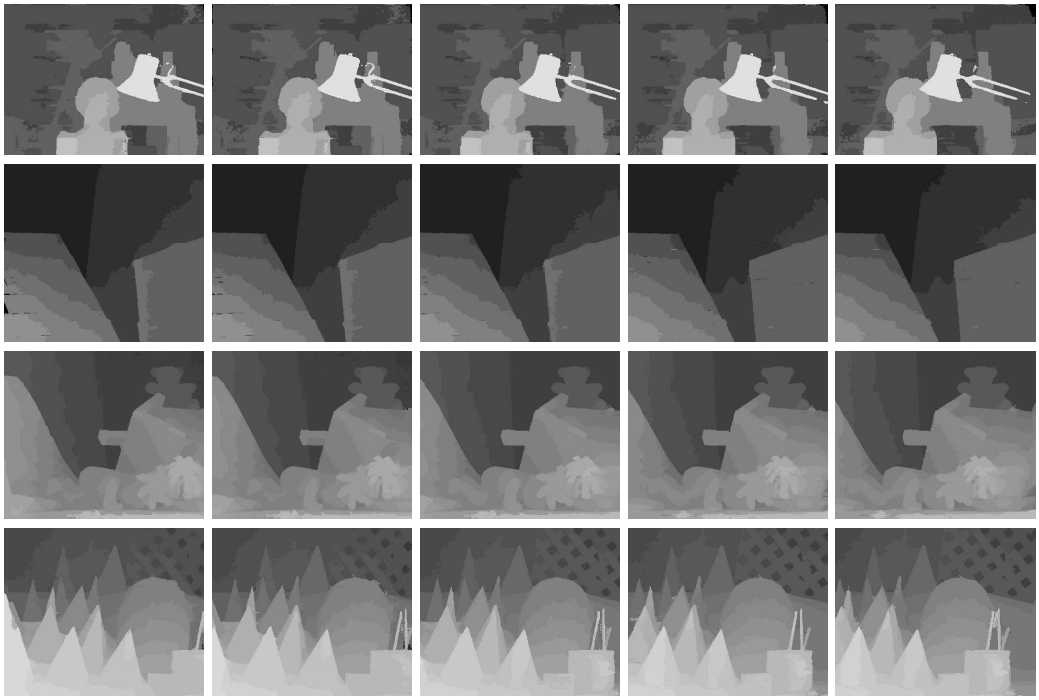


(a) (b) (c) (d) (e)

Fig. 15. Results for (from top to bottom) 'Tsukuba', 'Venus', 'Teddy' and 'Cone' image pairs in the semi N-view & N-depth framework ($N = 3$): (a)(e) Disparity maps on target images 0 and 2 before occlusion handling, (b)(d) Disparity maps on target images 0 and 2 after occlusion handling, (c) Disparity maps on reference image.

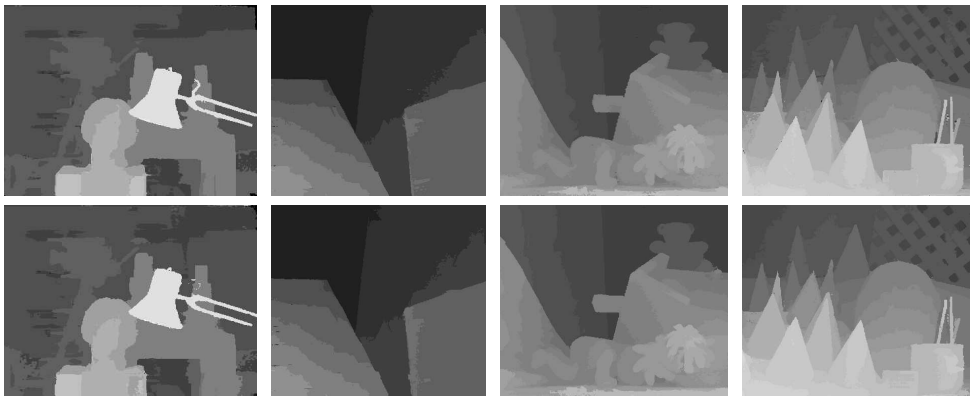
Table 1
Processing time for N-view & N-depth and semi N-view & N-depth frameworks ($N = 5$).

View #	*Tsukuba (s)*		*Venus (s)*		*Teddy (s)*		*Cone (s)*	
	N.N.	Semi	N.N.	Semi	N.N.	Semi	N.N.	Semi
View 0	2.43	0.59	3.03	0.48	6.58	1.97	6.32	2.09
View 1	2.39	2.42	2.92	3.05	6.42	6.65	6.27	6.20
View 2	2.36	0.11	2.89	0.13	6.38	0.28	6.24	0.34
View 3	2.37	2.39	2.93	3.02	6.33	6.55	6.22	6.22
View 4	2.41	0.53	3.08	0.47	6.52	1.91	6.37	2.05
Total	11.96	6.05	14.84	7.14	32.23	17.36	31.41	16.91



(a) view 0 (b) view 1 (c) view 2 (d) view 3 (e) view 4

Fig. 16. Disparity maps for the multiview image pairs in the semi N-view & N-depth framework ($N = 5$).



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

Fig. 17. Disparity maps estimated in the N-view & N-depth (top) and semi N-view & N-depth (bottom) frameworks.

and 9×9.

To evaluate the proposed cost aggregation method, the disparity maps estimated for the standard stereo image pairs are shown in Fig. 15. The disparity map of view 2 for the 'Tsukuba' image set was estimated with views 1, 2 and

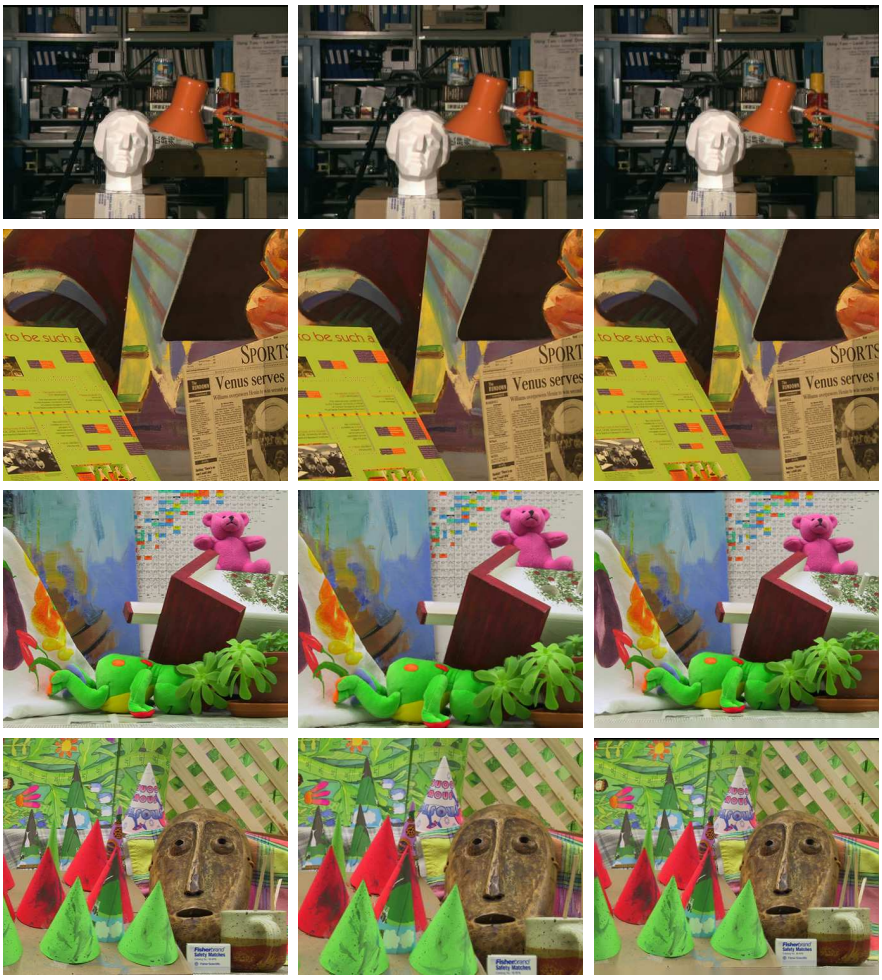
Table 2
PSNR results of reconstructed views in N-view & N-depth and semi N-view & N-depth frameworks.

$N = 5$	Tsukuba (dB)	Venus (dB)	Teddy (dB)	Cone (dB)
N-N.	*N.A.*	36.99	33.66	31.57
Semi N-N.	*N.A.*	36.98	33.57	31.67

3, and the disparity map of view 4 for the 'Venus', 'Teddy' and 'Cone' image sets with views 0, 4 and 8 in order to perform disparity estimation in the same search range. In other words, the disparity maps were estimated in the semi N-view & N-depth framework, when N was 3. Fig. 15 (c) shows the disparity map estimated with cost aggregation method on the reference image. Fig. 15 (a) and (e) show the disparity maps of the target images before occlusion handling. They were acquired by warping the cost function of reference image. Fig. 15 (b) and (d) show the disparity maps after occlusion handling. We could find that the disparity maps of the target images were accurate and had good localization on the object boundary, although these were acquired by warping technique. The proposed method yielded accurate results for the discontinuity, occluded, and textureless regions. We found that correct disparity fields were estimated in the occluded pixels by using a simple technique that compared the cost functions of multiview images. We think that the error in some parts, for example, the 'Venus' image pairs, may have been caused by using the asymmetric weighting function in the proposed cost aggregation process as shown in Eq. (4), which is different from the result obtained in [19].

The estimated disparity maps for the multiview image pairs are shown in Fig. 16. The disparity maps were estimated in the semi N-view & N-depth framework, when N was 5. Fig. 16 (b) and (d) show the disparity maps in the reference images, which were acquired by the proposed cost aggregation method. The disparity map Fig. 16 (c) in the target image was computed by symmetric warping. The disparity maps Fig. 16 (a) and (e) in the semi-target image were computed by backward (or forward) warping only, therefore reasonable cost functions were assigned into the occluded pixels with the proposed occlusion handling process. We find that the disparity maps for the target and semi-target images were accurate and had good localization on the object boundaries, although these were acquired by warping techniques.

Fig. 17 shows the estimated disparity maps in the N-view & N-depth and the semi N-view & N-depth frameworks. The disparity maps in the semi N-view & N-depth framework were estimated by using symmetric warping, in other words, the disparity maps in the target images were used for comparison. We found that the disparity maps estimated in the semi N-view & N-depth framework were as good as those estimated in the N-view & N-depth framework with trivial computational loads only. Table 1 shows the processing times when

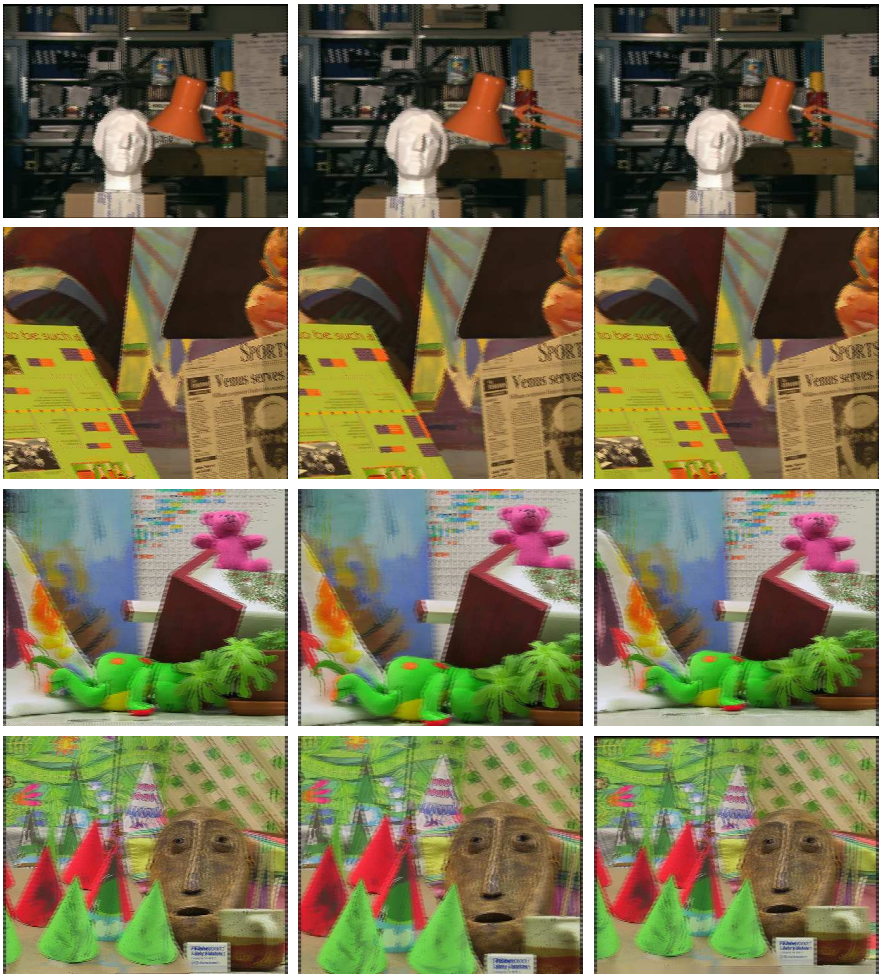


(a) (b) (c)

Fig. 18. Results for 2D freeview generation, when N is 5: (from left to right) (a) $(1.5, 0.0, 0.0)$, (b) $(1.5, 0.0, 1.0)$, (c) $(1.5, 0.0, -0.5)$.

(a) (b) (c)

Fig. 19. Difference images for original and reconstructed images in the semi N-view & N-depth framework (Table 2): (a) Venus, (b) Teddy, (c) Cone.

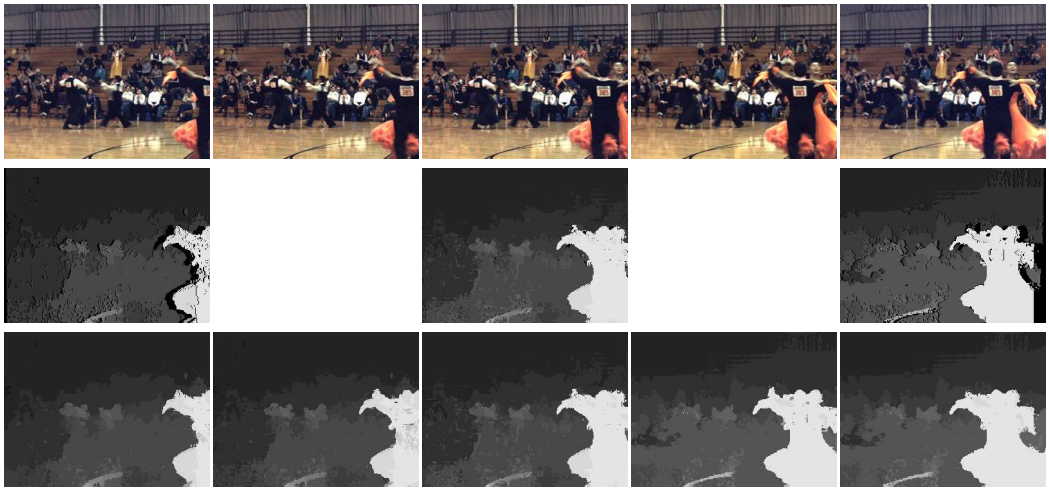


(a) (b) (c)

Fig. 20. Results for 3D freeview generation, when N is 5: (from top to bottom) 'Tsukuba', 'Venus', 'Teddy' and 'Cone' image pairs.

comparing levels of complexity with those of other methods. The processing time of the semi N-view & N-depth framework was nearly half of that of the N-view & N-depth framework.

Fig. 18 shows the synthesized novel views that were obtained by the virtual camera. We found that seamless images were synthesized in the object boundaries and the occluded regions. The quality of the synthesized images was satisfactory enough to provide users with natural freeview videos for 3DTV. For objective evaluation, we compared with PSNR results of reconstructed images in N-view & N-depth and semi N-view & N-depth frameworks, as shown in Table 2. In the 'Venus', 'Teddy' and 'Cone' image pairs, we synthesized view 3 with view 2 and 4 and computed PSNR. We found that the PSNR of reconstructed images in two frameworks was nearly same. The difference images for original and synthesized images (view 3) are shown in Fig. 19. The



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

Fig. 21. Disparity maps for 'Ballroom' image pairs: We used 5 images (views 2-6) in the experiment.

synthesized stereoscopic images are shown in Fig. 20. The virtual left view is the same as that in Fig. 18. The virtual baseline distance B_s is 0.2, and the sensor shift h is $SR/3$, where SR defines the search range of the stereo matching process. Users could see the freeview 3D video through a 3D stereoscopic monitor. In these experiments, we used a $G170S$, a stereoscopic MIRACUBE LCD monitor [25]. This monitor supports various 3D display formats such as 'interlaced stereo', 'frame sequential', 'sub-field', and 'side-field'. Moreover, it supports both 2D and 3D display modes, and the maximum resolution is 1280×512 (1280×1024) in the 3D (2D) mode. The synthesized 2D and 3D freeview videos are available at [27].

Fig. 21 shows the estimated disparity maps for 'Ballroom' (640×480 pixels) image pairs, multiview video coding (MVC) test sequences which consists of 8 rectified views. In the experiments, we used 5 images (views 2-6), and search range was set to 40. In order to minimize the error that might be caused by the difference of baseline distances between cameras, we used the images of 320×240 pixels by performing subsampling. Fig. 21 (b) and (d) show the disparity maps in the reference images. Fig. 21 (c) show the disparity maps in the target image, and (a) and (e) the disparity maps in the semi-target images. The figures in the second column show the intermediate results before hole filling and occlusion handling. We found that reasonable disparity values in the occluded parts were obtained on the target and semi-target images. Fig. 22 shows the synthesized 2D and 3D novel views. The virtual baseline distance and the sensor shift are the same as those of Fig. 18.

The experiment was additionally performed using other images such as 'Mans'

(a) (b) (c)

Fig. 22. Results for 2D and 3D virtual views: (from left to right) (a) $(1.5, 0.0, 0.0)$, (b) $(1.5, 0.0, 1.0)$, (c) $(1.5, 0.0, -0.5)$.

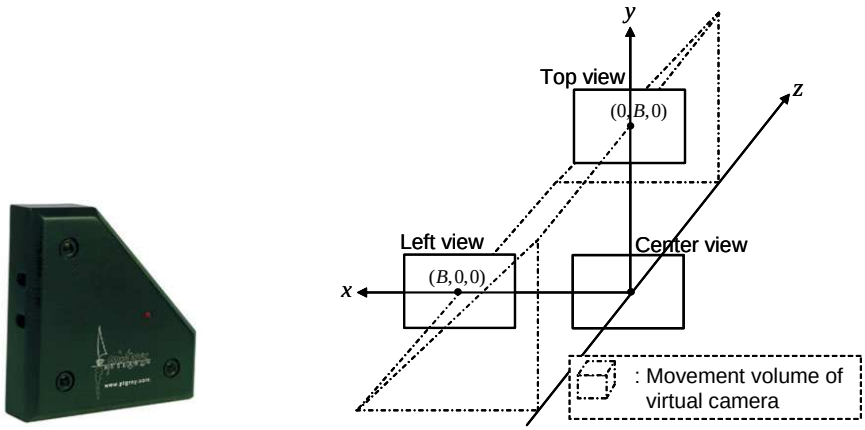


Fig. 23. Trinocular camera configuration.

(640 × 480 pixels). The 'Mans' image pairs were captured by the Digiclops camera of Point Grey Research Inc. [26]. The search range was 35. Since the Digiclops is the trinocular camera which consists of the left, center and top views, the virtual views were synthesized in 3D volume, that is x, y and z axes. Fig. 23 shows the trinocular camera configuration. The novel view from the virtual camera can be synthesized in the volume which is encircled by the dotted lines. The estimated disparity maps and synthesized 2D and 3D views are shown in Fig. 24 and 25, respectively. The virtual baseline distance was set to 0.3, and the sensor shift was 0. The quality of the synthesized images was satisfactory, especially, in the occluded parts or object boundaries.

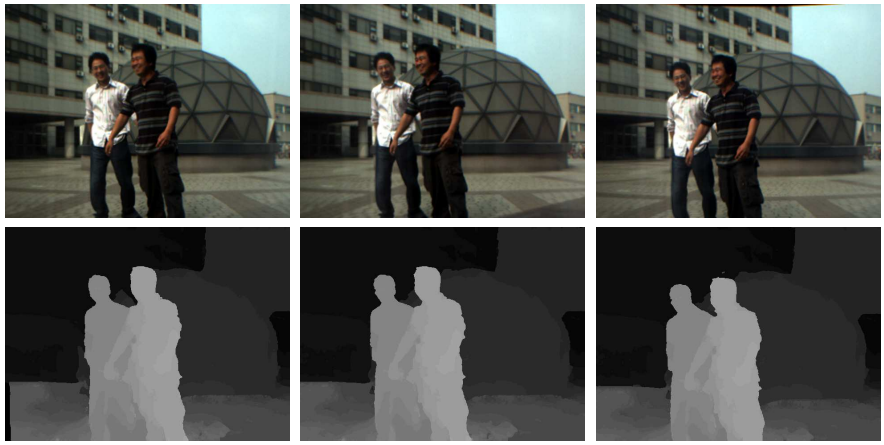


Fig. 24. Disparity maps for 'Mans' (640×480 pixels) image pairs captured by the Digiclops camera of Point Grey Research Inc. [26]: (from left to right) Left, center, and top views.

Fig. 25. Results for 2D and 3D virtual views: (from left to right) $(0.7, 0.5, 0.0)$ $(0.7, 0.5, 2.0)$ $(0.7, 0.5, -1.0)$.

6 Conclusions

In this paper, we have presented a novel approach for generating 2D/3D free-view video in multiview camera configurations. By using the estimated cost functions of neighboring images, the redundancy of estimating disparity maps in multiview images was reduced in the semi N-view & N-depth framework. The disparity maps in the reference images, which were estimated by the proposed method, were accurate and robust to occlusion. Since the cost functions on the target images were computed by the proposed warping technique, the computation loads were reduced significantly. The occlusion problem was efficiently handled by using the cost functions of multiview images. Novel views

could be selected among the 2D or 3D stereoscopic images according to user selection. In further work, we will investigate virtual view rendering systems for various camera configurations, and investigate the backward warping method, which is more robust to disparity map errors.

Acknowledgments

This work was financially supported by the Ministry of Education, Science and Technology (MEST), the Ministry of Knowledge Economy (MKE) and the Ministry of Labor (MOLAB) through the fostering project of the Lab of Excellency, and was partially supported by the Korea Science and Engineering Foundation (KOSEF) through the Biometrics Engineering Research Center(BERC) at Yonsei University.

References

[1] https://www.3dtv-research.org.

[2] W. Matusik and H. Pfister, "3D TV: A scalable system for real-time acquisition, transmission and autostereoscopic display of dynamic scenes," SIGGRAPH, , pp. 814-824, 2004.

[3] M. Levoy and P. Hanrahan, "Light field rendering," *SIGGRAPH*, pp. 31-42, 1996.

[4] S.J. Gortler, R. Grzeszczuk, R. Szeliski and M.F. Cohen, "The lumigraph," *SIGGRAPH*, pp. 43-54, 1996.

[5] C. Buehler, M. Bosse, L. McMillan, S. Gortler, M. Cohen, "Unstructured lumigraph rendering," *SIGGRAPH*, pp. 425-432, 2001.

[6] P. E. Debevec, C. J. Taylor, and J. Malik, "Modeling and rendering architecture from photographs: a hybrid geometry- and image-based approach," *SIGGRAPH*, pp. 11-20, 1996.

[7] P. Debevec, Y. Yu and G. Borshukov, "Efficient view-dependent image-based rendering with projective texture-mapping," *Eurographics Rendering Workshop*, 1998.

[8] E. Chen and L. Williams, "View interpolation for image synthesis," *SIGGRAPH*, pp. 279-288, 1993.

[9] S. Seitz and C. Dyer, "View morphing," *SIGGRAPH*, pp. 21-30, 1996.

[10] L. Zhang, D. Wang, and A. Vincent, "Adaptive Reconstruction of Intermediate Views From Stereoscopic Images," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 16, no. 1, pp. 102-113, Jan. 2006.

[11] A. Criminisi, A. Blake and C. Rother, "Efficient Dense Stereo with Occlusions for New View-Synthesis by Four-State Dynamic Programming," *International Journal of Computer Vision*, vol. 71, no. 1, pp. 89-110, 2007.

[12] J. Park and H. Park, "A Mesh-Based Disparity Representation Method for View Interpolation and Stereo Image Compression," *IEEE Trans. on Image Processing*, vol. 15, no. 7, pp. 1751-1762, July 2006.

[13] M. Lhuillier and L. Quan, "Image-Based Rendering by Joint View Triangulation," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 13, no. 11, pp. 1051-1063, Nov. 2003.

[14] A. Redert, M. O. de Beeck, C. Fehn, W. IJsselsteijn, M. Pollefeys, L. Gool, E. Ofek, I. Sexton, P. Surman, "ATTEST: Advanced three-dimensional television system technologies," Proc. IEEE 3DPVT, pp. 313-319, 2002.

[15] P. Kauff, N. Atzpadin, C. Fehn, M. Muller, O. Schreer, A. Smolic, and R. Tanger, "Depth map creation and image-based rendering for advanced 3DTV services providing interoperability and scalability," *Signal Processing: Image Communication*, vol. 22, pp. 217-234, 2007.

[16] L. Zitnick, S. Kang, M. Uyttendaele, S. Winder, and R. Szeliski, "High-quality video view interpolation using a layered representation," *SIGGRAPH*, pp. 598-606, 2004.

[17] L. Zitnick and S. Kang, "Stereo for Image-Based Rendering using Image Over-Segmentation," *International Journal of Computer Vision* , vol. 75, no. 1, pp. 49-65, 2007.

[18] J. Park and S. Inoue, "Arbitrary view generation from multiple cameras," *Proc. IEEE ICIP*, pp. 149-152, 1997.

[19] D. Min and K. Sohn, "Cost Aggregation and Occlusion Handling with WLS in stereo matching," *IEEE Trans. Image Processing*, vol. 17, no. 8, pp. 1431–1442, Aug. 2008.

[20] D. Scharstein and R. Szeliski, "A taxonomy and evaluation of dense two-frame stereo correspondence algorithms," *International Journal of Computer Vision*, vol. 47, no. 1-3, pp. 7-42, Apr. 2002.

[21] V. Kolmogorov and R. Zabih, "Computing visual correspondence with occlusions using graph cuts," *Proc. IEEE International Conf. Computer Vision*, pp. 508-515, 2001.

[22] J. Sun, N-N. Zheng, and H-Y. Shum, "Stereo Matching Using Belief Propagation," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 25, no. 7, pp. 787-800, 2003.

[23] C. Fehn, R. Barre, and S. Pastoor, "Interactive 3-DTV - Concepts and Key Technologies," *Proceedings of the IEEE*, vol. 94, no. 3, pp. 524-538, Mar. 2006.

[24] http://vision.middlebury.edu/stereo.

[25] http://www.miracube.net.

[26] http://www.ptgrey.com.

[27] http://diml.yonsei.ac.kr/∼forevertin.