VIRTUAL VIEW RENDERING SYSTEM FOR 3DTV

Dongbo Min, Donghyun Kim and Kwanghoon Sohn

Dept. of Electrical and Electronic Eng., Yonsei University, Seoul, Korea
khsohn@yonsei.ac.kr

ABSTRACT

In this paper, we propose a new approach for efficient multiview stereo matching and virtual view generation, which are key technologies for 3DTV. We propose semi N-view & N-depth framework to estimate disparity maps efficiently and correctly. This framework reduces the redundancy on disparity estimation by using the information of neighboring views. The proposed method provides a user 2D/3D freeview video, and the user can select 2D/3D modes of freeview video. Experimental results show that the proposed method yields the accurate disparity maps and the synthesized novel view is satisfactory enough to provide user seamless freeview videos.

Index Terms— Stereo matching, semi N-view & N-depth framework, virtual view rendering.

1. INTRODUCTION

By recent advance in the multimedia processing fields, 3DTV is expected to become one of the most dominant products of markets in the next generation broadcasting system. 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. 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].

Novel view rendering is one of the most important techniques in the 3DTV. 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. There is the rendering approach with implicit geometry among image-based rendering. A number of view interpolation approaches have been proposed to improve the performance [2]. Zhang et al proposed the method of reconstructing intermediate views from stereoscopic images [3]. Kauff et al introduced an advanced approach for 3DTV system based on the concept of video-plus-depth data representation [4]. In this paper, video-plus-depth data representation method is called N-view & N-depth framework, where N is the number of cameras. Zitnick et al proposed the method for performing high-quality novel view interpolation using multiple synchronized video streams [5].

In this paper, we propose a new method for synthesizing novel views from the virtual camera. We reduce the redundancy of estimating the disparity maps in semi N-view & N-depth framework, while the conventional method estimates the disparity maps in the same manner for N images in N-view & N-depth framework. We can provide user 2D/3D freeview videos in the proposed method. Most conventional methods provide a user 2D freeview video [5] or 3D video at one fixed viewpoint when stereo images are given [3]. We propose more flexible system for 3DTV by making it possible that the user selects 2D/3D modes of freeview videos.

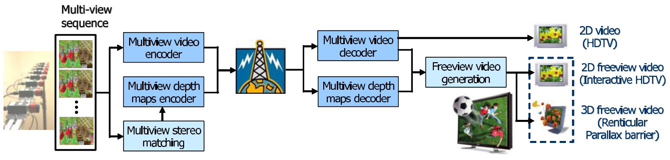
Multi-view sequence
Multiview video encoder
Multiview depth maps encoder
Multiview stereo matching
Multiview video decoder
Multiview depth maps decoder
Freeview video generation
2D video (HDTV)
2D freeview video (Interactive HDTV)
3D freeview video (Renticular Parallax barrier)

Fig. 1. Overview of 3DTV system.

2. OVERVIEW OF 3DTV SYSTEM

Fig. 1 shows the overall framework 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. In this paper, we propose a new approach for efficient multiview depth estimation and virtual view generation, which are key technologies for 3DTV system.

3. MULTIVIEW STEREO MATCHING

We use the multiview camera configuration for estimating disparity maps and rendering virtual view. An extensive review of stereo matching algorithms can be found in [6]. In this paper, we use multiview camera system where 3 cameras are disposed in 1D parallel structure ($N = 3$). Our aim is to develop 2D/3D freeview video generation system. Thus, the parallel camera structure is used, since multiview camera configuration with toed-in structure may cause a number of holes in the synthesis of 3D freeview video. We assume that the baseline distances between captured cameras are same to B.

3.1. Per-pixel cost computation

When estimating the disparity field, two or more images are used. Let $i-1^{th}$, i^{th} and $i+1^{th}$ images left, center and right images, respectively. 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.

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

p and d represent the 2D locations of pixels and disparity, respectively. I is the intensity with RGB color, and T is the threshold that defines the upper bound of matching cost. We compute the per-pixel cost $e_i(p,d)$ with the $e_{i,i+1}$, $e_{i,i-1}$. When computing the per-pixel cost, we should consider whether 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. The assumption is useful

This research was supported by the MIC, Korea, under the ITRC support program supervised by the IITA (IITA-2005-(C1090-0502-0027))

978-1-4244-1755-1/08/$25.00 ©2008 IEEE

for handling occlusion, although it is invalid in a few pixels. Based on the principle which matching cost of visible pixel is generally smaller than that of occluded pixel, we compute the per-pixel cost $e_i(p,d)$ on the center (i^{th}) image as follows:

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

3.2. Cost aggregation with weighted least square

In order to estimate the optimal cost $E_i(p,d)$ on the i^{th} image, we use a prior knowledge that costs should vary smoothly, except at object boundaries. From this observation, we are able to estimate the cost function by minimizing the following energy model with weighted least square:

$$\begin{array}{l} \varepsilon(E) = \int_\Omega (E(p) - e(p))^2\, dp \\ \quad +\lambda \int_\Omega \sum\limits_{n \in N_1} \left\{ \begin{array}{c} w_{p,p+n}(E(p) - E(p+n))^2 \\ +w_{p,p+n^\perp}(E(p) - E(p+n^\perp))^2 \end{array} \right\} dp \ , \\ N_1 = \{(x_n, y_n) | 0 < x_n \le M, \ \ 0 \le y_n \le M\} \end{array} \tag{3}$$

where w represents the weighting function between corresponding neighbor pixels. We simplify $E_i(p,d)$ to $E(p)$, since the same process is performed for each disparity. n and $n^\perp$ represent the 2D vectors, which are perpendicular to each other. M represents the size of a set of neighbor pixels. Taking the first derivative of Eq. (3) with respect to E, we obtain the following equation:

$$E(p) - e(p) + \lambda \sum_{n \in N_1} \left\{ \begin{array}{l} w_{p,p+n}(E(p) - E(p+n)) \\ -w_{p-n,p}(E(p-n) - E(p)) \\ +w_{p,p+n^\perp}(E(p) - E(p+n^\perp)) \\ -w_{p-n^\perp,p}(E(p-n^\perp) - E(p)) \end{array} \right\} = 0 \tag{4}$$

To simplify the above equation, we redefine the set of neighbor pixels. When p is (x,y), the set can be expressed as:

$$N(p) = \{(x + x_n, y + y_n)| - M \le x_n, y_n \le M, \ \ x_n + y_n \ne 0\}$$

By using the above notation, Eq. (4) is expressed as:

$$E(p) - e(p) + \lambda \sum_{m \in N(p)} w_{p,m}(E(p) - E(m)) = 0 \tag{5}$$

The solution of the $(k+1)^{th}$ iteration is obtained by the following equation:

$$E^{k+1}(p) = \bar{e}(p) + \bar{E}^k(p) = \frac{e(p) + \lambda \sum\limits_{m \in N(p)} w_{p,m} E^k(m)}{1 + \lambda \sum\limits_{m \in N(p)} w_{p,m}} \tag{6}$$

Eq. (6) 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. The iteration scheme is similar to the adaptive weight approach [7] when the number of iterations is 1. In the proposed method, we use the asymmetric Gaussian weighting function with the CIE-Lab color space in Eq. (7). r_c and r_s are weighting constants for the color and geometric distances, respectively. When C_i is the color distance that is computed with i^{th} image, the weighting function can be defined as follows.

$$\begin{array}{l} w(p,m) = \exp\left(-\left(\frac{C_i(p,m)}{2r_c^2} + \frac{S(p,m)}{2r_s^2} \right) \right) \\ C_i(p,m) = (L_p^i - L_m^i)^2 + (a_p^i - a_m^i)^2 + (b_p^i - b_m^i)^2 \\ S(p,m) = (p-m)^2 \end{array} \tag{7}$$

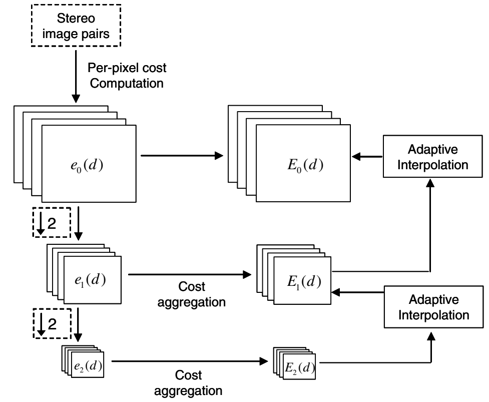

Fig. 2. Overall framework of the cost aggregation.

3.3. Acceleration Scheme

3.3.1. Gauss-Seidel Acceleration

One reason for slowing down the convergence in Eq. (6) is that the updated components in each pixel are used only after one iteration is complete. We compensate for this problem by using the updated components in each pixel intermediately after they are computed. We divide a set of neighbor pixels $N(p)$ into two parts: the causal part $N_c(p)$ and the non-causal part $N_n(p)$. Eq. (6) is expressed as follows, based on this relationship:

$$E^{k+1}(p) = \frac{e(p) + \lambda \sum\limits_{m \in N_c(p)} w_{p,m} E^{k+1}(m) + \lambda \sum\limits_{m \in N_n(p)} w_{p,m} E^k(m)}{1 + \lambda \sum\limits_{m \in N(p)} w_{p,m}} \tag{8}$$

3.3.2. Multiscale Approach

As previously mentioned, it is necessary to gather pixel information at a large distance to ensure reliable matching. This implies that a number of iterations are required to estimate the correct cost function. We use a multiscale approach to solve this problem. Our method is different from the conventional approaches in the sense that it is applied in the cost domain. We can initialize the value close to the optimal cost in each level by using the final value in the coarser level. Using Eq. (8), the proposed method performs cost aggregation independently in each section with the same disparity of the 3D cost volume. We first compute the 3D cost volume and then perform the proposed multiscale scheme in 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 in the finer level, and this process is repeated until the finest level is reached. The proposed multiscale scheme is shown in Fig. 2, which includes adaptive interpolation.

When the cost function on the $(l+1)^{th}$ level is defined as $E_{l+1}(p)$, we can refine the resolution of the cost function $E_l(p)$ on the finer level by using bilinear interpolation. However, if bilinear interpolation is used, the error can be propagated into the neighborhood regions, especially on the boundary region. To avoid this problem, we propose an adaptive interpolation method based on the weighted least square:

$$E_l(p) = \frac{e_l(p) + \lambda_a \sum\limits_{p_m \in N(p_i)} w_{p,p_m} E_{l+1}(p_m)}{1 + \lambda_a \sum\limits_{p_m \in N(p_i)} w_{p,p_m}}, \tag{9}$$

where $p_i = (x_i, y_i)$ represents a pixel on the coarser level, and $N(p_i)$ on the $(l+1)^{th}$ level is a set of 4-neighboring pixels. In Eq. (9), w represents the weighting function, equivalent to that in Eq. (6). We set the weighting factor to $\lambda_a = 15$. The adaptive

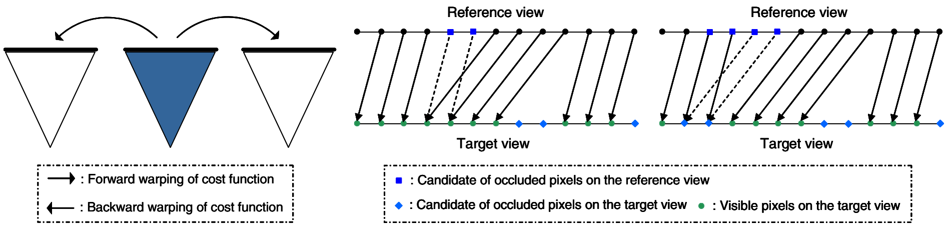
Reference view
Reference view
Target view
Target view
: Forward warping of cost function
: Backward warping of cost function
: Candidate of occluded pixels on the reference view
: Candidate of occluded pixels on the target view
: Visible pixels on the target view

Fig. 3. semi N-view & N-depth framework with warping techniques: (a) semi N-view & N-depth framework, (b) Several cases of forward warping: when occluded pixels on the reference are blocked by visible pixels, and when occluded pixels on the reference block visible pixels.

interpolation by the intensity values on two successive levels leads to the up-sampling scheme, which preserves the discontinuities on the boundary region. Thus, it is not necessary to perform the cost aggregation scheme on the finest level, and this makes the proposed method faster.

3.4. Warping of aggregated cost

Most approaches have acquired the disparity maps for N images independently in the same manner. They have huge redundancy of estimating the disparity maps. In this section, we propose a new approach for eliminating the redundancy of estimating disparity maps in the cost aggregation scheme. Fig. 3 (a) shows semi N-view & N-depth framework, when N is 3. Color and white views are reference and target views, respectively. The cost functions in the reference images are estimated by using the proposed cost aggregation method with weighted least square. The cost functions in the target images are estimated through warping of those in the reference images. It is based on the assumption that the corresponding pixels on neighboring images have generally the similar cost functions. The cost functions of reference images are transferred into those of target images with the corresponding disparity maps of the reference images. Since asymmetric warping is performed only, the occluded parts in cost function remain. For assigning reasonable cost function into the occluded pixels, we use the method of handling occluded pixels with reliable neighboring pixels in the cost aggregation scheme.

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. At first, we explain the process of forward warping. We can estimate the visibility of pixels by evaluating the disparity values of the neighboring pixels. 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. 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. 3 (b) 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 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. 3 (b), and $\#(S_t(j))$ is reset to 0. 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:

$$E_t(i - d_r, d_r) = E_r(i, d_r), \qquad if\ O_t(i - d_r) = 1 \qquad (10)$$

In Eq. (10), the y coordinate is omitted, since the same process is performed for each scanline. The process of backward warping is also similar to that of forward warping. Note that the cost functions of reference images are transferred into those of target images through the warping, not disparity values. The occluded parts in target images are handled in the cost aggregation. By using the visibility function O_t on the target image, we can redefine the iterative scheme in Eq. (8) as follows:

$$E_t^{k+1}(p) = \frac{\begin{array}{c} O_t(p)e(p) + \lambda \sum\limits_{m \in N_c} O_t(m) w_{p,m} E_t^{k+1}(m) \\ + \lambda \sum\limits_{m \in N_n} O_t(m) w_{p,m} E_t^{k}(m) \end{array}}{O_t(p) + \lambda \sum\limits_{m \in N(p)} O_t(m) w_{p,m}} \qquad (11)$$

It is different from the extrapolation technique widely used for occlusion handling. While the extrapolation technique is just filling by using the disparities of the visible pixels, the proposed method propagates the information of the visible pixels into that of the occluded pixels. The proposed occlusion handling is similar to the concept of edge-preserving nonlinear diffusion. In this paper, we use WTA (Winner-Takes-All) method as optimization method for disparity estimation. Other optimization techniques such as graph cut and belief propagation can be used to perform disparity estimation.

4. VIRTUAL VIEW RENDERING

4.1. Virtual view generation

Given N images and the associated disparity maps, the virtual view can be synthesized by warping each image with its disparity map. All the images are warped and the novel view is generated by performing the weighted-interpolation. The rotation of the virtual camera is not considered in the novel view rendering, since the rotation of the virtual camera may cause a number of holes in the novel view and it is not appropriate in 3DTV or video-conferencing. Given a novel viewpoint, the nearest two images (camera i and $i+1$) are selected and projected into virtual view. The relation between $m_i(x, y)$ in the i^{th} image and $m_i^v(x_i^v, y_i^v)$ in the novel view can be computed as follows [10]:

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

where (x_0, y_0) is the center of the image plane and (T_x, T_y, T_z) represents translation between the real and virtual cameras. To simplify the 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. Let I_i and I_i^v

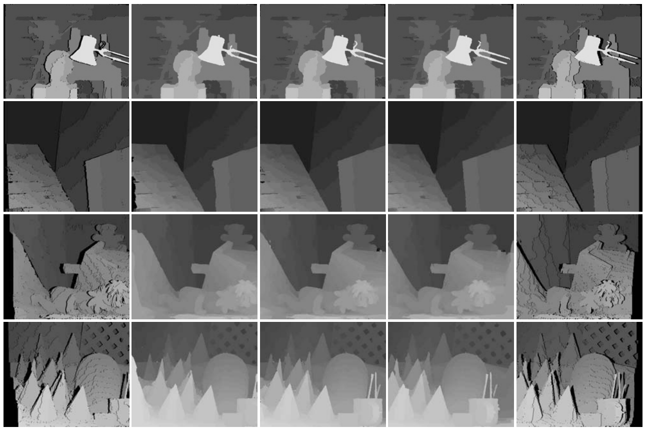

Fig. 4. Results for (from top to bottom) 'Tsukuba', 'Venus', 'Teddy' and 'Cone' image pairs: (a)(e) Disparity maps on target images 0 and 2 before occlusion handling, (b)(d) Disparity maps on target images 0 and 2 after occlusion handling, (c) Disparity maps on reference image.

the reference and projected images, respectively, then $I_i^v(x_v, y_v) = I_i(x, y)$. The virtual camera can move with x and z-axes, which consists of left, right, forward and backward movements. The movement with y-axis is limited since this may cause some holes in the novel view. We can generate 2D or 3D freeview video by synthesizing one or two novel views, respectively. When $V(p)$ is a visibility function whether a pixel in the novel view is visible in the reference views, the final reconstructed novel view is computed by interpolation with the projected images as follows.

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

4.2. Virtual 3D view generation

The synthesis of stereoscopic novel view can be generated by synthesizing two novel views - one for left view and one for right view. The distance between two novel views is defined as B_s. To establish the zero parallax setting (ZPS), the CCD sensor of the stereoscopic cameras in the parallel structure are translated by a small shift h relative to the position of the lenses [8]. It makes us choose the convergence distance Z_c in the 3D scene. The sensor shift can be simply formulated as a displacement of a camera's principal point. When a horizontal shift of the principal point is defined as h, the point in the novel view can be computed in Eq. (14). $\pm h$ means the shifted right and left images of novel stereoscopic views, respectively. Please refer to [8] for more detailed explanation.

$$\begin{aligned} x_i^v - (x_0 \pm h) &= \frac{x_i - x_0 + d_i\alpha_x}{1 + d_i\alpha_z/f} \\ y_i^v - (y_0 \pm h) &= \frac{y_i - y_0 + d_i\alpha_y}{1 + d_i\alpha_z/f} \end{aligned} \qquad (14)$$

5. EXPERIMENTAL RESULTS

To validate the performance of semi N-view & N-depth framework, we performed the experiments with the Middlebury test sequences [9]. We use the following test data sets: 'Tsukuba', 'Venus', 'Teddy', and 'Cone'. We perform the experiments with multiview images which N is 3. The proposed method is tested using the same parameters for all the test images. The two parameters in the weighting function are $r_c = 8.0$, $r_s = 8.0$, and the weighting factor is $\lambda = 1.0$. We use the multiscale approach at four levels, and the number of iterations is $(3, 2, 2, \times)$, on a coarse to fine scale. The iteration number of the finest level is not defined since we use the adaptive interpolation technique in the up-sampling step. The sizes of the sets of neighbor pixels are 5×5, 7×7, 9×9, and 9×9.

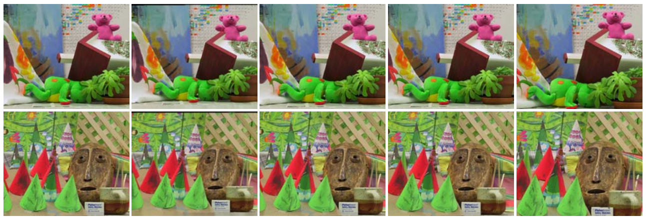

Fig. 5. Synthesized virtual view for 'Teddy' and 'Cone' image pairs: (from left to right) (a) $\alpha_x = 0.75$, $\alpha_z = 0.0$ (b) $\alpha_x = 0.75$, $\alpha_z = -0.5$ (c) $\alpha_x = 0.9$, $\alpha_z = 0.0$ (d) $\alpha_x = 1.35$, $\alpha_z = 0.0$ (e) $\alpha_x = 1.35$, $\alpha_z = 1.0$.

The estimated disparity maps for multiview image pairs are shown in Fig. 4. Fig. 4 (c) shows the disparity map estimated with cost aggregation method on the reference image. Fig. 4 (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. 4 (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. Fig. 5 shows the synthesized novel views from the virtual camera. The quality of the synthesized images was satisfactory enough to provide user the natural freeview videos for 3DTV. We show 2D novel views for 'Teddy' and 'Cone' image pairs only due to limitation of space. The 2D and 3D freeview videos for other images are available at [11].

6. CONCLUSION

In this paper, we have presented a novel approach for generating 2D/3D freeview video in multiview camera configuration. By using estimated cost functions of neighboring images, redundancy of estimating disparity maps in the multiview images is reduced in semi N-view & N-depth framework. The occlusion problem was efficiently handled by using the cost functions of multiview images. The novel view can be selected among 2D or 3D stereoscopic images according to the selection of the user. In further work, we will investigate virtual view rendering system for various camera configurations, and develop 3DTV system including coding structure of multiview images and depth maps.

7. REFERENCES

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

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

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

[4] 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," *SPIC*, vol. 22, pp. 217-234, 2007.

[5] 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.

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

[7] K. Yoon and I. Kweon, "Adaptive support-weight approach for correspondence search," *IEEE Trans. PAMI*, vol. 28, no. 4, pp. 650-656, Apr. 2006.

[8] 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.

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

[10] D. Min, D. Kim, S. Yun, and K. Sohn "Freeview rendering with trinocular camera," *IEEE Proc. ISCAS*, May 2008.

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