28th Picture Coding Symposium, PCS2010, December 8-10, 2010, Nagoya, Japan

# FREE-VIEWPOINT IMAGE GENERATION USING DIFFERENT FOCAL LENGTH CAMERA ARRAY

*Kengo Ando*[*a], *Norishige Fukushima*[b], *Tomohiro Yendo* [a], *Mehrdad Panahpour Tehrani* [a], *Toshiaki Fujii* [c], *Masayuki Tanimoto* [a]

[a] Graduate School of Engineering, Nagoya University, Furo-cho, Chikusa-ku, Nagoya-shi, Aichi, 464-8603 Japan
[b] Graduate School of Engineering, Nagoya Institute of Technology, Gokiso-cho, Showa-ku, Nagoya-shi, Aichi, 466-8555 Japan
[c] Graduate School of Engineering, Tokyo Institute of Technology, 2-12-1 Ookayama, Meguro-ku, Tokyo, 152-8550 Japan

## ABSTRACT

The availability of multi-view images of a scene makes new and exciting applications possible, including Free-Viewpoint TV (FTV). FTV allows us to change viewpoint freely in a 3D world, where the virtual viewpoint images are synthesized by Image-Based Rendering (IBR). In this paper, we introduce a FTV depth estimation method for forward virtual viewpoints. Moreover, we introduce a view generation method by using a zoom camera in our camera setup to improve virtual viewpoint-ts' image quality. Simulation results confirm reduced error during depth estimation using our proposed method in comp-arison with conventional stereo matching scheme. We have demonstrated the improvement in image resolution of virtually moved forward camera using a zoom camera setup.

***Index Terms*** **—** Free-viewpoint image generation, Image Based Rendering

## 1. INTRODUCTION

The FTV[3] system is based on Image-Based-Rendering (IBR).In the early age of IBR research, Chen and Williams [1] proposed smooth interpolation between images by modeling the motion of pixels. This method allows limited image synthesis between cameras. A light ray in 3D space can be defined using the Plenoptic Function with seven parameters: $f(x, y, z, \theta, \varphi, \lambda : t)$. The IBR techniques, namely Ray-Space [2], Light Field [4], and the Lumigraph [5], were proposed around the same time. These methods define rays in 3D space with four parameters assuming that the ray travels in a straight line and does not attenuate. For example, the Ray-Space method represents rays as a function $f(x, y, \theta, \varphi)$.

Practically, the number of rays captured by cameras are limited. So, unavailable rays are obtained by the linear interpolation. However, far from the camera the density of rays is sparse. Therefore, only sparse rays are available to synthesize the image of a virtual camera far from a real camera. Thus, depth estimation errors occur easily at the virtual viewpoint, resulting a poor synthesized image quality. Moreover, the resolution lowering problem of the generated image from the moved forward viewpoint is caused. In this paper, we describe two methods to generate a free-viewpoint image. One is the method to improve the accuracy of depth estimation for a forward virtual camera. Our second method improves the resolution of the synthesized image for the moved forward viewpoint by making use of a zoom camera in our camera setup.

## 2. FREE VIEWPOINT IMAGE GENERATION

This chapter describes the fundamental principle of free viewpoint image generation, and the depth estimation method [7]. Furthermore, the problem of the image resolution from the moved forward position compositing is also described.

### 2.1. Fundamental principle of free viewpoint image generation

To synthesize the free viewpoint image, we need an appropriate interpolation of the ray according to the geometry structure of the scene. First of all, Lambert reflection is assumed i.e. rays are reflected equally in all directions so the same color and intensity is observed from all viewpoints. Then, ray information at an arbitrary position is obtained by specifying the position of the source of light from the relation between object and camera, and linear interpolating the intensity of the ray that has been acquired. We explain this method using X-Z plane diagram of Figure 1. First of all, we input the position of the virtual camera $(x, z)$. Next, for each pixel $(u, v)$ the following steps are done.

- Calculate the intersection of the ray from the virtual camera with the plane that composes the camera array (reference plane). Select the two cameras by the side of the intersection point.

978-1-4244-7133-1/10/$26.00 ©2010 IEEE

- Change the depth along the ray, and calculate the correlation value for the two selected cameras. The depth with the highest correlation value is the assumed depth value.
- Using this depth, the intensity value of the pixel $(u,v)$ is decided using linear interpolation according to the ratio of the distance of the ray in the reference plane.

The above-mentioned steps are done for all pixels, to generate the free viewpoint image. The intensity value of the output virtual camera $VC(u,v)$ is given by Eq.(1), where $d$ indicates the obtained best disparity value.

$$VC(u,v) = \bar{a}LC(u+ad,v) + aRC(u-\bar{a}d,v) \tag{1}$$

Where $\bar{a} = 1-a$, and $LC(i,j)$ and $RC(i,j)$ indicate the intensity value of the left camera and the right camera. The relational equation is shown in Eq. (2).

$$aLC(u+ad_{suitable},v) \cong aRC(u-\bar{a}d_{suitable},v) \tag{2}$$

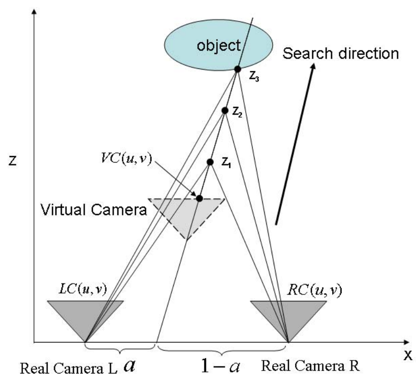



**Fig.1.** Depth estimation and ray interpolation

Thus, the intensity value in the two cameras almost become equal for a certain depth $d_{suitable}$. However, in case of noise or low texture, it is difficult to obtain correct correspondence by comparing intensity values of one pixel. This can be improved by block matching using a block of (2*m* + 1) ×(2*m* + 1) centered around the pixel of interest. The correlation value *S* is defined as in Eq. (3)

$$S(z) = \frac{\sum_{i=0}^{N} |I_1(i,z) - I_2(i,z)|}{N} \tag{3}$$

Where $N = (2m + 1) \times (2m + 1)$ shows the number of pixels in the block, and $I_k(i,z)$ is the intensity value of the *i*-th pixel in the block projected in image *k* at depth *Z* . The closer to 0 this correlation values *S* is, the higher the correlation of the block is. Depth *Z* is changed, and the one with the highest correlation value is decided as the correspondence point.

### 2.2. Problem of Depth Estimation

In the depth estimation by block matching, when estimating the depth in an area of low texture, the peak of the correlation value becomes ambiguous. Therefore, the depth is frequently wrongly estimated. To improve the accuracy of depth estimation, it is necessary to enlarge the size of the block. However, the calculation cost increases when the size of the block is enlarged. Therefore, in this paper we propose a technique to reduce the calculation cost by using pre-calculated depth maps. These depth maps can be calculated off-line beforehand, for example using Graph Cuts or Belief Propagation, and improve the accuracy of depth maps at the virtual viewpoints.

### 2.3.Image Resolution Problem of moved forward viewpoint

We think in the case of Figure 2. There is planar object at distance *Z* from a real camera, and a virtual camera is moved forward by distance *d*. The focal length of the real camera and the virtual camera is assumed to be $f_1$ and $f_2$ respectively. It is assumed that one pixel at the real camera is projected to plane length of $P$ . In this paper, we refer to $P$ as the resolution expansion rate which can be expressed as in Eq.(4).

$$P = \frac{f_2 Z}{f_1 (Z-d)} \tag{4}$$

Therefore, the resolution of a corresponding square in a 2D image, decreases by $1/P^2$ .

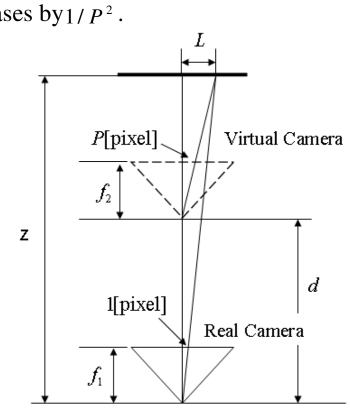



**Fig.2.** Resolution problem

When $f_1 = f_2$, the expansion rate $P_0$ is given by Eq. (5)

$$P_0 = \frac{Z}{Z-d} \tag{5}$$

If the real camera is substituted for a zoom camera, that is $f_1 > f_2$, and the expansion rate at this time is $P'$ , is given by Eq.(6) .

$$P' = \frac{f_2}{f_1} P_0 \qquad \therefore P' < P_0 \tag{6}$$

Therefore, using a zoom camera for the capturing camera, the resolution of the forward virtual image improves.

## 3. PROPOSED METHOD

This chapter explains the proposed method of depth estimation and the method of view generation of a virtual image at a forward viewpoint using a zoom camera.

### 3.1. Depth estimation

As an input, our method uses the depth maps at the camera position obtained beforehand by off-line depth estimation. The Graph-cut algorithm [7] is used for generating the depth maps. Using these depth maps, the depth map of the virtual camera is estimated. We will explain our method using the diagram Figure 3. First of all, we input the position of the virtual camera $(x,z)$, and then for each pixel $(u,v)$ following steps are done:

- Set the depth of a pixel in the virtual camera $V_Z(u,v)$, looping all possible depth values in the search range.
- Using this depth, 3D point $(x,y,z)$ is projected to left depth map $L_Z(u,v)$ and right depth map $R_Z(u,v)$.
- Depth with the highest correlation is assumed to be depth of $V_Z(u,v)$. Here, the correlation is calculated based on depth $Z$, and projection to left depth $Z_L$ and a right depth $Z_R$ rather than camera views as in section 2.1.

This process is done for all pixels to generate the depth map of the virtual camera. As correlation value *V*, we use a variance value (Eq.7). Depth $Z$ for which *V* becomes minimum is assumed to be depth at the virtual camera. Where $\bar{Z}$ is the average of $Z_L$, $Z_R$ and $Z$.

$$V = \frac{1}{3}\left((Z_L - \bar{Z})^2 + (Z_R - \bar{Z})^2 + (Z - \bar{Z})^2\right) \tag{7}$$

The advantage of our method includes reduction of computing time. Depth estimation using block matching should calculate the correlation by using all pixels in the block. Therefore, it takes more time if the size of the block is large. Our method has to calculate the variance for one pixel only. Additionally, the accuracy of depth estimation can be improved over the conventional method as described in section 2.1.

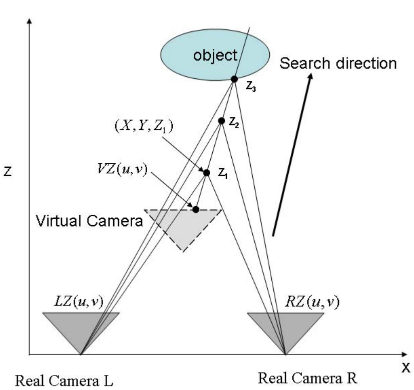



Fig.3. Proposed method of depth estimation

### 3.2 View synthesis using zoom-camera

Next, we will explain the simplest system for view synthesis using two cameras with a zoom camera. Three cameras are arranged in a straight line. The zoom camera is placed in the middle between the left and right camera as shown in Figure 4. In this system, view synthesis for the forward virtual viewpoint is done as follows:

1. The depth of the virtual camera is calculated based on the depth map of the right and left camera.
2. Project all pixels of the zoom camera to the virtual view based on the depth of the virtual camera.
3. The right and left camera views are projected to the virtual view and blended, and then used to fill pixels which could not be projected from the zoom camera.

The image for the forward viewpoint can be synthesized with high resolution, by using this method and giving priority to the image of the zoom camera.

## 4. SIMULATION

### 4.1 Simulation Condition

In this paper, we carry out following three simulations:

1. Depth estimation and view synthesis for the forward virtual viewpoint by conventional block matching.
2. Depth estimation and view synthesis for the forward virtual viewpoint by the proposed method.
3. View synthesis using a zoom camera.

Figure 4 shows the system as used in the experiments, and corresponding parameters are shown in Table 1.

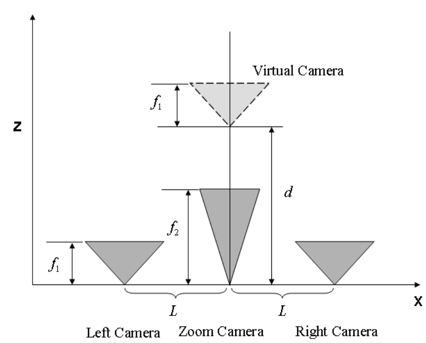



**Fig.4. Camera Setting for Simulation**

**Table 1: Simulation Parameters**

| | |
|---|---|
| Focal length (LRcamera) $f_1$ | 100[pixel] |
| Focal length (Zoom Camera) $f_2$ | 140[pixel] |
| Camera distance $L$ | 50[mm] |
| Forward distance $d$ | 30[mm] |
| Image Resolution | 320×240[pixel] |

In experiment 1, the size of the block was set to 3×3pixels. The zoom camera is only used in experiment 3, and the depth is estimated by the proposed method.
Experiments 1 and 2 assume no zoom camera exits, and synthesis of the forward virtual viewpoint is based on only the right and left camera.

### 4.2 Simulation Result and discussion

Figure 5(a) shows depth estimation result of the forward viewpoint in experiment 1, and Figure 6(a) shows the corresponding synthesis result at the forward virtual viewpoint. The depth estimation result of our proposed method is shown in Figure 5(b), and the corresponding synthesized view is shown in Figure 6(b). From Figure 5(a) and (b) it is clear that the depth estimation result of our method is much more accurate than by block matching, for example, in the hat of the stuffed animal. Therefore, we are able to synthesize a much better quality image at the forward viewpoint compared with conventional method as shown in Figure 6(a).
Figure 6(d) shows the result of the view synthesis using the zoom camera of experiment 3.Compared with Figure 6(c) where the zoom camera is not used, improvement of the resolution was seen in part of the scene both with forward comparatively and captured with zoom camera (like the face of the stuffed animal). However, the resolution of the input image originally: from two points, using the rough one and where magnification percentage of the zoom camera was small, the resolution has improved only to the extent that understands gazing.

## 5. CONCLUSION

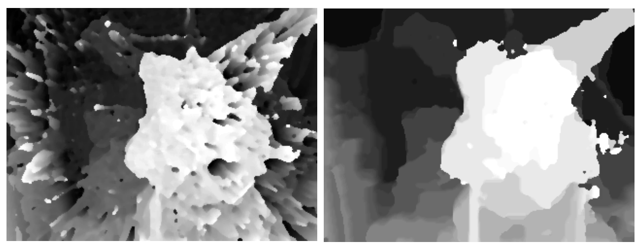
In this paper, we proposed a method of depth estimation for a moved forward virtual viewpoint. Furthermore, we proposed a method for improving the resolution of the synthesized image at the forward viewpoint by using a zoom camera.
As a result of simulation, in the depth estimation, our proposed method confirmed the depth error reduction compared with conventional block matching. Moreover, we confirmed that the image resolution of the virtually moved forward camera can be improved using a zoom camera.
In our future work, we evaluate the proposed method using images of CG simulation. And, we simulate it by various camera placements with a zoom camera.

## 6. ACKNOWLEDGMENT

This work was partially supported by Strategic Information and Communications R&D Promotion Programme (SCOPE) 093106002 of the Ministry of Internal Affairs and Communications.
This work was partly supported by the Grand-In-Aid for Young Scientists (B) of Japan Society for the Promotion of Science under Grant 22700174 and The Hori Information Science Promotion Foundation.

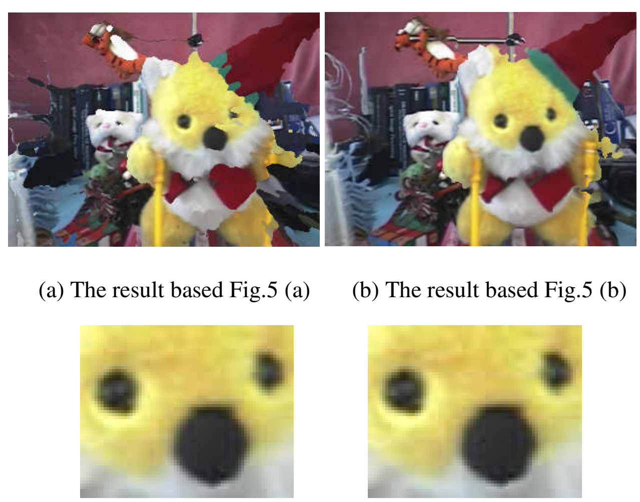
## 7. REFERENCE

[1] S. E. Chen, L. Williams: "View Interpolation for Image Synthesis", Proc. ACM SIGGRAPH' 93, pp. 279–288, 1993.
[2] T. Fujii, T. Kimoto, and M. Tanimoto: "Ray space coding for 3D visual communication", Proc. Picture Coding Symp.' 96, vol. 2, pp. 447–451, 1996.
[3] T. Fujii and M. Tanimoto, "Free-viewpoint TV system based on ray-space representation", in Proc. SPIE ITCom, vol. 4864-22, 2002, pp. 175–189.
[4] M. Levoy, P. Hanrahan: "Light Field Rendering", Proc. ACM SIGGRAPH' 96, pp. 31–42, 1996.
[5] S. J. Gortler, R. Grzeszczuk, R. Szeliski, and M.F. Cohen: "The Lumigraph", Proc. ACM SIGGRAPH' 96, pp. 43–54, 1996.
[6] Fukushima, N., Yendo, T., Fujii, T., Tanimoto, M.: "Real-time arbitrary view interpolation and rendering system using Ray-Space.", Proc. SPIE Three-Dimensional TV. Video. and Display IV.6016,250-261 (2005)
[7] Y. Boykov and V. Kolomogorov, "An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision", Transactions on Pattern Analysis and Machine Intelligence, vol. 26, no. 9, pp. 1124-1137, 2004.



(a) Block matching (b) Proposed method

**Fig.5.** Depth estimation result



(a) The result based Fig.5 (a) (b) The result based Fig.5 (b)

(c) Not use a zoom-camera (d) use a zoom-camera

**Fig.6.** View synthesis result