IMAGE-BASED RENDERING FOR MIXED REALITY

Hiroshi KAWASAKI, Hiroyuki ARITAKI, Katsushi IKEUCHI, Masao SAKAUCHI

Institute of Industrial Science, University of Tokyo
4–6–1 Komaba, Meguro-ku, Tokyo, 153-8505, Japan
{h-kawa, hiroyuki,sakauchi}@sak.iis.u-tokyo.ac.jp, ki@iis.u-tokyo.ac.jp

ABSTRACT

In this paper, we propose an image-based approach to synthesize a novel view image for mixed reality (MR) systems. Theoretically, the image- based method is good for synthesizing realistic images, but it is difficult to achieve interactive handling of the object. As a solution, we propose a new method based on the “surface light field rendering” technique . With this method, we can synthesize the objects with arbitrary deformation and illumination changes. To demonstrate the efficiency of this method, we describe successful experiments that we performed using objects with non-rigid effects (e.g. velvet and Tatami carpet) which are difficult to render correctly by the use of general model-based rendering techniques.

1. INTRODUCTION

Image-based rendering (IBR) technique has now become one of the major topics of the computer graphics (CG) and vision research areas; this is due to IBR’s its great potential for photo-realistic image synthesis with complicated shapes and non-rigid effects (e.g., animal fur and velvet) whose rendering has been historically difficult. In recent years, principles and various kinds of implementation and theoretical analyses of IBR have been proposed and published one after another.

However, for practical use of IBR, for example, for MR systems, little research has been done and few actual applications have been developed. Although there may be many reasons for this, the following two reasons are significantly important:

- Huge data size
- Lack of Interactivity

The data size of IBR is very large compared with that of other major model-based methods. The data size is a very important and crucial issue for actual implementation of IBR because, theoretically, real rays can be described in 7 dimensions, an amount that too big for efficient handling and storage. There have been many attempts to reduce the data size without degrading the quality of synthesized images; also, many theoretical analyses have been conducted.

Even if we can conquer the data size problem, there still remains the important issue of “interactivity” for practical application. So far, little research has been devoted to realizing interactivity for IBR. In this paper, we describe an initial attempt to achieve an interactive extention for IBR.

1.1. Related works

One of the key concepts developed in the IBR is the plenoptic function[1]. However, in reality, it is difficult to apply this method for rendering novel view images because of huge data size, some reasonable methods have been presented. “Lumigraph”[2] and “Light Field Rendering”[3] are the 4D plenoptic function with clever parameterization. Recently H.Shum et al.[4] proved the relationship among three elements for IBR: the depth, the minimum number of sampling images, and the rendering resolution.

On the other hand, there are several recent researches that extend the IBR for more efficient handling and illumination changes. The surface light field, a term coined by Miller et al. [5], is a function that assigns an RGB value to every ray emanating from every point on a surface.

Nishino et al.[6, 7] and Wood et al.[8] extend a surface light field rendering in an efficient manner. The basic idea of their research is almost the same, but their purpose and data compression algorithms are different. Wood et al. also proposed an editing system and managed to render a slightly deformed object under a limited condition.

1.2. Purpose and Goal

As mentioned in the Introduction, we attempt to realize the interactivity for the IBR technique. The word “interactivity” has many meanings and ambiguities depending on research and individual applications; therefore, it is necessary for us to define the word precisely, especially with regard to MR systems. Once defined, we can then clarify the purpose and the goal of our research more concretely.

Interactivity Basically, we assume that the interactivity has three key elements. The first element is the arbitrary motion of the object, including deformation of the object. The second is the arbitrary illumination change. The third is time consistency.

Purpose and Goal Based on the previous definition, the purpose of our research to achieve the interactivity for IBR can be translated as **“rendering arbitrarily positioned objects with arbitrary deformation and illumination in real time”**. Especially, in this paper, we are interested in rendering objects which are dynamically moved and/or deformed correctly.

Significance There are only a few papers which describe attempts to synthesize deformed objects with IBR. Also, we realize BTF(bidirectional texture function)[9] for actual 3D object which have never before been done. (Note that, to realize this, efficient data acquisition, accurate alignment and calibration is essential.)

2. THEORY AND ALGORITHM OVERVIEW

Rendering the moved and/or deformed object with non-rigid surface effects correctly is hard to achieve using previously available IBR techniques. In this paper, we make a new data structure and propose an original synthesis method to accomplish our purpose. In the following sections, we will introduce the algoritm and data structure.

2.1. Image capturing process

To make the data for our purpose, we configured the original data aquisition system “light dome” shown in Fig.1. In this system, the rotation table turns automatically to δ_o direction and CCD images are also taken automatically. We can also change the ϕ_o and θ_o degree of the turn table precisely with manual operation. In terms of the illumination position, this dome is covered by the flood lamp and each lamp has its own θ_l and ϕ_l to define the position. Also this light dome is equipped with a range sensor; therefore, we can easily measure the precise 3D shape. After acquiring the image sequence and 3D shape, we project the 3D shape to the image and can consequently acquire the texture image for mesh. Fig4 (a) is the sample acquired images. This texture image consists of triangle patch textures; the global mesh consists of these triangle patches.

2.2. Data structure

To realize efficient rendering, we have to configure a suitable data structure for actual implementation. Considering the BTF (bidirectional texture function)[9], we need the viewing direction (θ_o, ϕ_o) and light direction (θ_l, ϕ_l) for each argument to determine the ray. Thus, the dimension of the data is 4D(Actually +2D for texture coordinate). Fig.2 shows the structure of the 4D texture database.

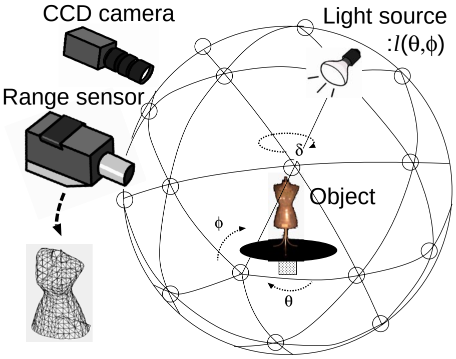


Fig. 1. Light dome

Originally, each polygon of the 3D object has its own suface normal direction; therefore, each texture patch of Fig4 (a) has its own 4D parameters $(\theta_l, \phi_l,\ \theta_w, \phi_w)$. With our 4D database as shown in Fig.2, each texture patch is arranged orderly with 4D parameters. So, we sort each texture patch to make the actual 4D texture database for our purpose as shown in Fig3. Fig4 (b) shows the sorted 4D texture database of Fig4 (a).

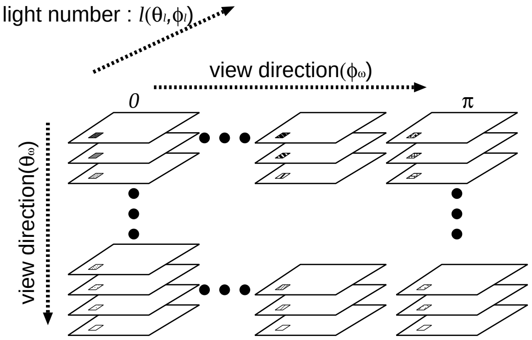


Fig. 2. 4D data structure concerned BTF

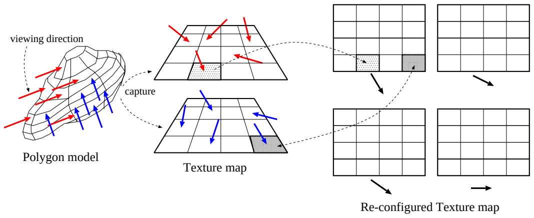
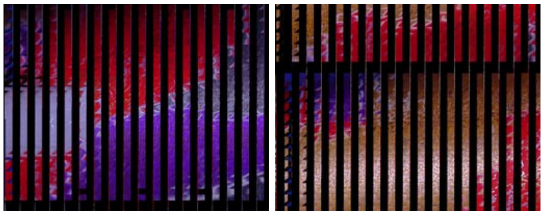
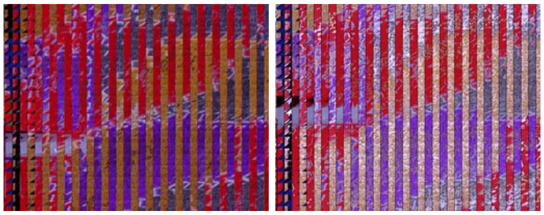


Fig. 3. Data reconfiguration



object pose θ= 0° θ= 120° (ϕ = 90°)

(a)



view direction θ= 30° θ= 90° (ϕ = 90°)

(b)

Fig. 4. actual data of 4D texture:each image represent the whole texture of the 3D object,(a)before arrangement (b)after arrangement:note that the luminance of all patches are almost same in (b)

2.3. Image synthesis

To synthesize an object whose position and pose change arbitrarily, we have to consider the relative relationship between the light source direction, the surface orientation and view direction. We introduce the function (1) for rendering.

$$F_i(\vec{\omega_l}, \vec{o}, \vec{n}) \quad (1)$$

Here, $\vec{\omega_l}$ denotes the lth light source direction, $\vec{o}$ denotes the viewing direction and $\vec{n}$ is the surface normal vector of the deformed mesh(actually, we also need to define the top direction for each triangle). This function represents the RGB value and is defined at each triangle i.

2.4. Polygon deformation

To achieve further interactivity for the objects, not only moving the objects, but also deformation of the objects is effective. Since the proposed image synthesis algorithm and the data structure are configured for each triangle, we can apply the same method to the deformed object in the same manner.

3. EXPERIMENTS AND RESULTS

We performed several experiments to show the effectiveness of our method. Fig. 5 shows the overview of the experiments. First, we used the "light dome" to capture the images and subsequently made the 4D texture data. Then, we deformed the shape of the objects and rendered the deformed objects using our proposed method.

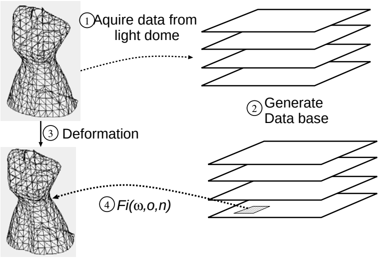


Fig. 5. Experiment overview

3.1. Velvet cloth

Cotton velvet is usually difficult to synthesize because of complicated BRDF; however, by using IBR technique, such non-rigid effects can be successfully rendered. In this experiment, we used cotton velvet and synthesized the cloth after deformation. In Fig. 6, left column images are the synthesized image with our method; right column images are of the actual velvet cloth taken by CCD camera which was carefully set up so as to be same as the rendered image's illumination, camera position and deformation of the cloth. When we compared these results, we observed that the synthesized deformed velvet cloth was almost the same as actual velvet cloth even though we did not use any of the material's properties for rendering.

3.2. Tatami block

In this experiment, we used a 3D block made of tightly bound straw. This material also has non-rigid effects on surface and is difficult to render with common techniques. This 3D block also had small textures on it and matching accuracy between polygon and texture image would affect the results. Fig.7(a) and (b) are the original images and (c) and (d) are the deformed images. Note that the surface texture of the deformed 3D block gradually changes its color dependent on its degree of bend and is naturally rendered.

4. DISCUSSION

There remain several problems for our research to solve, some of which are fundamental to IBR. First, we can not correctly render materials with strong reflectance properties, e.g., mirrors. Another significant problem derived from our algorithm was the self shadow on the object which disappeared after deformation. Another issue of our research was data size. Even if we ignore the arbitrary illumination

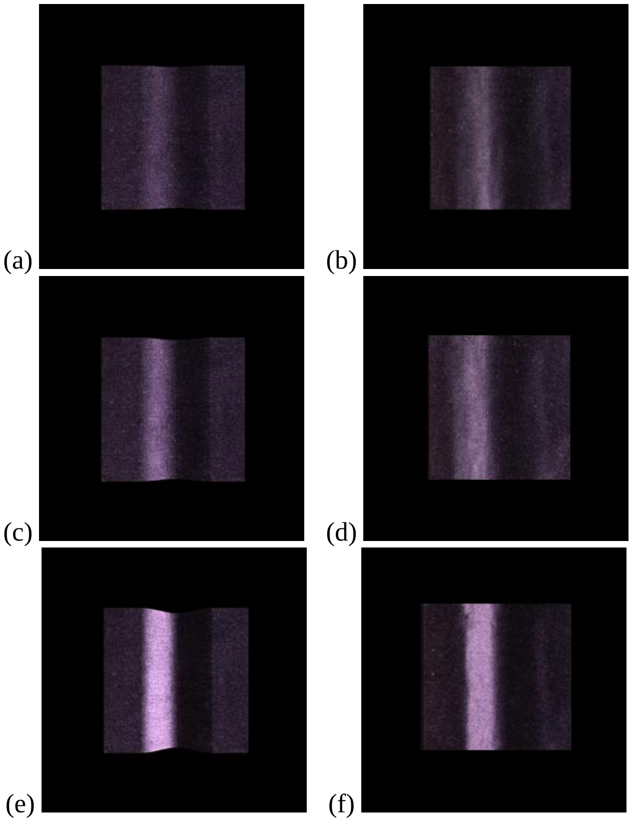


Fig. 6. Results of velvet cloth: (a,c, e) synthesized images using our proposed method, (b,d,f)Actual captured image

changes(we only considered intensity in this paper), the size of data is 4D, a huge amount for common usage.

5. CONCLUSION

We proposed an interactive extension for IBR to apply IBR techniques to MR systems. In this paper, we assumed the important interaction to the object for MR systems was to freely move and deform the object under arbitrary illumination. Therefore, to achieve this goal, we proposed and implemented an efficient IBR algorithm and 4D data structure.

To demonstrate the effectiveness of our proposed method, we conducted several experiments using several objects. With our proposed algorithm, we successfully rendered the deformed objects with non-rigid surface effects. In the future, we need to compress the data size and realize the synthesis of the deformed object into the real-world environment for MR systems.

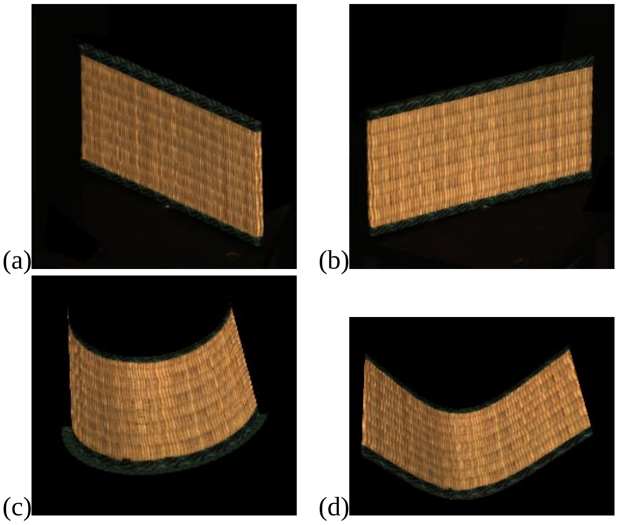


Fig. 7. Results of Tatami: (a,b)Actually captured image,(c,d) synthesized images using our proposed method

6. REFERENCES

[1] E H Adelson and J Bergen, *The Plenoptic function and the elements of early vision*, MIT Press Cambridge, MA, 1991.

[2] S J Gortler, R Grzeszczuk, R Szeliski, and M F Cohen, “The lumigraph,” *ACM SIGGRAPH*, pp. 43–54, 1996.

[3] Marc Levoy and P Hanrahan, “Light field rendering,” *ACM SIGGRAPH*, pp. 31–42, 1996.

[4] Jinxiang Chai, Xin Tong, Shing-Chow Chan, and Heung-Yeung Shum, “Plenoptic sampling,” *ACM SIGGRAPH*, pp. 307–318, July 2000.

[5] Gavin Miller, Steven Rubin, and Dulce Ponceleon, “Lazy decompression of surface light fields for precomputed global illumination,” in *Rendering Techniques(Eurographics Proceedings)*, June 1998, pp. 281–292.

[6] Ko Nishino, Yoichi Sato, and Katsushi Ikeuchi, “Eigen-texture method: Appearance compression based on 3D model,” in *Computer Vision and Pattern Recognition*, June 1999, vol. 1, pp. 618–624.

[7] Ko Nishino, Yoichi Sato, and Katsushi Ikeuchi, “Appearance compression and synthesis based on 3D model for mixed reality,” in *Proc. of SeventhInternational Conference on Computer Vision*, Sept. 1999, vol. 1, pp. 38–45.

[8] Daniel Wood, Daniel Azuma, Wyvern Aldinger, Brian Curless, Tom Duchamp, David Salesin, and Werner Steutzle, “Surface light fields for 3d photography,” *ACM SIGGRAPH*, July 2000.

[9] Kristin J Dana, Bram van Ginneken, Shree K Nayar, and Jan J Koenderik, “Reflectance and texture of real-world surfaces,” in *Computer Vision and Pattern Recognition97*, 1997.