自由視点テレビためのリアルタイム処理システム

プリム ナ・バンチャン† 藤井 俊彰† 谷本 正幸†

†名古屋大学 工学研究科 〒464-8603 名古屋市 千種区 不老町

E-mail: †purin@tanimoto.nuee.nagoya.ac.jp †{fujii, tanimoto}@nuee.nagoya-u.ac.jp

あらまし 我々はリアルタイムの画像光線空間データ獲得とその処理を行うシステム、「自由視点テレビ」を提案する。このシステムを用いて、ユーザはリアルタイムに実世界で視点を移動するかの如く自由に視点を変更することができる。この手法は実画像を元に視点修正を行う光線空間法を基本としており、我々の試作システムでは一直線上に配置した 16 台の CCD カメラを使用してこれを実現している。補間は光線空間データ補間フィルタリングを基本に行っている。我々の試作システムは民生機のみの使用であるが、処理速度と画質の双方において良好な結果が得られている。

キーワード 光線空間法、補間、マルチカメラシステム、リアルタイム画像生成

Realtime System of Free Viewpoint Television(FTV)

Purim Na Bangchang† Toshiaki Fujii† and Masayuki Tanimoto†

† Graduate School of Engineering, Nagoya University Furo-cho, Chigusa-ku, Nagoya, 464-8603 Japan

E-mail: †purin@tanimoto.nuee.nagoya.ac.jp †{fujii, tanimoto}@nuee.nagoya-u.ac.jp

Abstract We proposed a new realtime video ray-space data acquisition and rendering system named the "Free Viewpoint Television". With this system, the user can freely change the viewpoint of any dynamic real-world scene in realtime. The basic idea is based on the ray-space method which the arbitrary photo-realistic view can be generated from a collection of real images. For our prototype system, 16 CCD cameras are used to form one dimensional camera array. The interpolation is based on the adaptive filtering ray-space data interpolation. Although only consumer-class hardware is used, the result is good in term of both image quality and rendering speed.

Keyword Ray Space Method, Ray Space Interpolation, Multi Camera System, Realtime Image generation.

1. Introduction

Over the last few years, the ray-space method has become one of the most important tools in 3D visual communication. The main reason is based on its ability to generate an arbitrary photo-realistic view image from a collection of view input images. Similar ideas that have been proposed in computer graphics field as parts of the "image-based rendering technique" are called "Light Field"[4] or "Lumigraph"[5].

Although the basic idea of ray-space can be applied to both static and dynamic real-world scene, most of the previous works has been concentrated mainly on a static scene or object. The main limitation for processing a dynamic scene ray-space data lies on the difficulty to build a fast and accurate dynamic ray-space acquisition system. The dynamic ray-space system must rely on an array of cameras to continually capture rays from many different view positions. Based on the ray-space theory, the large number of cameras needs to be placed densely in order to achieve enough ray data for further processing. But, the large number of cameras also means a long processing time and expensive cost in building the system. So the actual number of cameras that has been used in the many dynamic ray-space researches is quite small and varies from 6-16 cameras[6-10].

Because the limited number of view input images, the interpolation that can obtain the missing ray information has played an important role in every dynamic ray-space systems. Note that the interpolation that suitable in this case must have abilities to due with 2 main problems; wide camera interval and limited input images.

There are 2 main interpolation concepts that are widely used among different dynamic ray-space system.

The first concept is based on the idea that the approximate geometric information can be used to improve the quality of the reconstructed image. Before any interpolation, the approximate geometric models are inferred from the input images. The systems that based on this concept have been proposed in [6],[9-10]

The second concept is rely on the idea that only the object plane depths information is enough to obtain a good interpolated image as the systems have been proposed in [7-8]

In this paper, we propose a new realtime video ray-space data acquisition and rendering system named the "Free Viewpoint Television" or "FTV". The system fully operates in realtime from capturing the dynamic ray-space data until the process to generate the new view image. With this system, the user can freely change the viewpoint of any captured dynamic real-world scene in realtime.

2. Ray-Space Method

2.1 Ray-Space Representation of 3D space

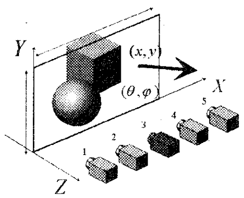


Fig.1 Definition of the Ray-Space

Ray-space is originally proposed as a common data formats for 3D image communication [1]. A similar idea has been proposed in computer graphics field for generating photo-realistic images into computer generated virtual world. It is one of the "image-based rendering technique" and is called Light Field [4][5]. It has been widely used to create photo-realistic virtual world. Both of them are based on the idea that a view image from an arbitrary viewpoint can be generated from a collection of real view images.

Fig. 1 shows an example of the definition of ray-space. Let (x,y,z) be three space coordinates, and (θ, ϕ) be the parameters of direction. A ray going through the space is uniquely designated by the intersection (x,y) with plane z = 0 and its direction (θ, ϕ). These ray parameters construct a 4D space. In fact, this is a 4D subspace of 5D ray-space (x,y,z,θ,ϕ). In this ray-parameter space, we define a function f whose value corresponds to an intensity of the specified ray. Thus, all the intensity data of rays can be expressed by

$f(x,y,\theta,\varphi)$ where $-\pi\leq\theta<\pi$ and $-\pi/2\leq\varphi<\pi/2$.

2.2 Ray-Space Acquisition System

An acquisition system of ray-space $f(x,y,\theta,\varphi)$ requires a photo sensor that is set at a positon (x,y) and senses the intensity of a ray coming from the direction (θ,φ). In practical, the rays are captured by a 2D camera.

Static Ray-Space Acquisition System

To record the light field of a static scene, it is customary practice to use one single still-image camera and consecutively record multiple views by moving the camera around the scene. The camera position can be control preciously by attach camera with a camera gantry or a turntable. The effect of camera intrinsic parameters cancels as only one camera is used to capture the whole ray-data.

Dynamic Ray-Space Acquisition System

In the contract with static case, the dynamic ray-space acquisition relies on a set of cameras to capture a set of input image. The cameras are placed still at various view location to capture the whole ray-data in realtime.

3. Proposed System

3.1 Hardware Configuration

In our system, the collection of real images is acquired through 16 CCD cameras placed parallel in one dimensional array with 2 cm interval. Thus total ray-space capture range of our system is 30 cm. This configuration allows high degree of freedom to move the viewpoint in x and z direction but prohibits any movement in y direction.

Each camera is controlled by one client processor thus we need totally 16 client processors. One server processor is used to receive the user input and to rendering the new view image. All computers in the system are connected together and formed the network using an Ethernet connection. Note that no special hardware is used and the computers are based on only consumer-class hardware.

The basic hardware configuration can be depicted as in Fig1.

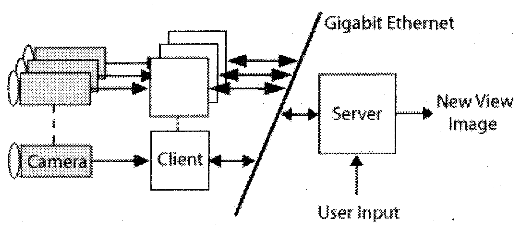


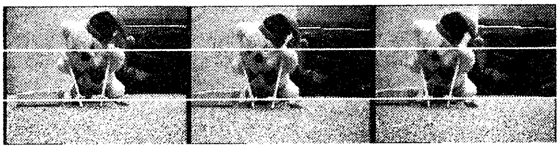
Fig.2 Basic Hardware Configuration

3.2 Rectification

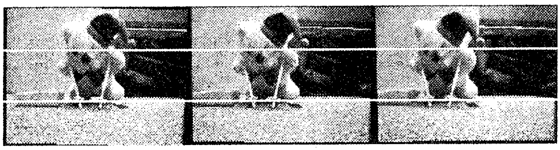
In our system, a set of input images is captured by using the array of cameras. By using more than one camera, it inevitably leads to a slight mismatch in each image. Typical reason is the difference in camera's extrinsic and intrinsic parameters which are very difficult or impossible to adjust correctly.

The rectification is the process to minimize these mismatch effects and prepares the images to be ready for the interpolation. The rectification matrix can be found by calculate the projection matrix that transforms the actual test pattern image to the corrected test pattern image. We need to calculate the rectification matrixes only once after each new camera setup.

We perform the rectification in the client processor immediately after the images are captured in order to relief some processing task from the server.



(a) Input Images



(b) Rectified Images

Fig.3 Rectification Result

3.3 Interpolation

The interpolation is one of the most important parts in our system as it allows the missing ray data to be obtained. The technique we use is based on the adaptive filtering ray-space data interpolation. This technique combines the characteristics of block matching and Filtering interpolation.

The Fig.4 shows a block diagram of this technique. First, interpolation filters are prepared so as to be able to reflect the local characteristics of the ray-space data. Next, each ray-space data is up-sampled in the direction that we want to obtain the input image. The number of interpolated view determines the number of up-sampled line. Next, the pixels around the one to be interpolated are analyzed. The optimum filter is selected based on the results of this analysis. Convolution processing is then performed on all pixels to be interpolated to obtain an output image.

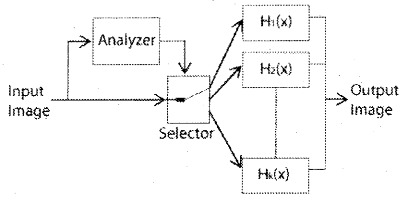


Fig.4 Interpolation Diagram

Although the optimum filter is selected based on the block searching, the interpolated pixel value is calculated using the only values of the center of these two blocks as shown in Fig.5

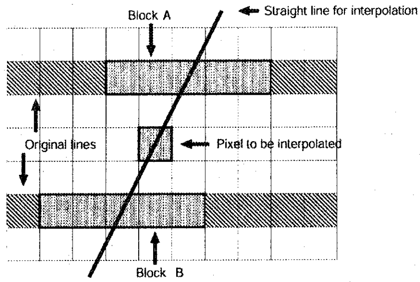
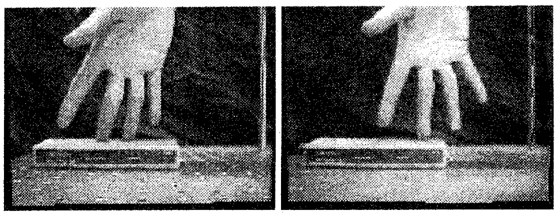


Fig.5 Block matching and Interpolated pixel value calculation

The advantage of this method is the approximate geometric models are not required before interpolation. Thus, the system can be applied to any type of objects. We show the result of the interpolation in Fig.6. In this case the object is human hand which is one of objects that is difficult to model.



(a) Input Image (b) Interpolated Image

Fig.6 Interpolation Result.

3.2 Image Synthesis

The last past of our system is concerned about how to generate the new view image. Given a view position, a new view image can be generated by performing the ray tracking. Because the cameras are placed only in the horizontal direction, we only need to track the ray along this direction and the group of rays that belong to the same x position can be tracked at the same time. We don't need to track the ray in vertical direction but when the requested viewpoint moves away from the camera plane, a magnification along y-axis is need to be performed.

4. Results

We implemented the complete system described above. The specifications of the computers in this experiment are as follows:

- Server(OS:Linux)
  - CPU : Pentium3 850MHZ
  - Memory : 256MB
- Client(OS:Linux)x16
  - CPU : Pentium3 850MHZ
  - Memory : 128MB
  - Video Capture : IO Data GV-VCP2/PCI
- Network
  - Gigabit Network

All parameter is carefully chosen based on the need of realtime operation. The distance between object plane and camera plane is about 30 cm. and the distance between each camera is about 2 cm. The input image is captured at 20 frames/sec, 160x120 pixel resolution. The rectification and interpolation are carried on with the same image resolution.

Between each pair of cameras, 15 view images are interpolated. Totally 241 different viewpoints can be selected which allows smoothly move of viewpoint along the camera plane. The high number of interpolated views also insures that no aliasing effect will occur. The results of system are shown in Fig.7. The black strip on the side of the images shows the missing ray information.

(a) At camera plane (b) At 10cm. closer to object plane

Fig.7 Rendering Result at different depth

5. Conclusions and Future Direction

In this paper we proposed the experimental system of Free Viewpoint Television. Using this system, the arbitrary view image of various dynamic scenes and objects can be generated in realtime. The constructed image quality is good and the generation rate is sufficient for the general operation.

Although the result is satisfied, many points still need to be improved, for example, generated image quality and rendering speed.

Reference

[1] T.Fujii, T.Kimoto, and M. Tanimoto, "Ray space coding for 3D visual communication", Proc. Picture Coding Symp.'96, vol.2, pp. 447-451, Mar 1996.

[2] T.Fujii, T.Kimoto, and M. Tanimoto, "A New Flexible Acquisition System of Ray-Space Data for Arbitrary Objects", IEEE Tran. on Circuit and Systems, vol.10, pp. 218-224, Mar 2000.

[3] A. Nakanishi, T.Fujii, T.Kimoto, and M. Tanimoto, "Ray-Space Data Interpolation for View Image Generation", IDW'01, pp. 1385-1388, 2001.

[4] M. Levoy and P. Hanrahan, "Light field rendering", ACM SIGGRAPH'96, pp. 31-42,1996

[5] S.J. Gortler, R. Grzesczuk, R.Szeliski, and M.F. Cohen, "The Lumigraph", ACM SIGGRAPH'96, pp. 31-42,1996

[6] M.Sekitoh, T.Fujii, T.Kimoto, and M. Tanimoto, "Bird's Eye View System for ITS", IEEE, Intelligent Vehicle Symposium, pp. 119-123, 2001.

[7] T. Naemura,and H. Harashima,"Real-Time Video -Based Rendering for Augmented Spatial Communication", Visual Commun. Image Process, pp 620-631, 1998.

[8] T. Naemura, J. Tago, H. Harashima,"Real-Time Video Based Modeling and Rendering of 3D Scenes", IEEE Computer Graphics and App., pp 66-73, 2002

[9] H. Schirmacher, L. Ming and H.P. Seide, "On-the-fly processing of generalized lumigraphs" Computer Graphics Forum, vol.20, pp. c165-c173, 2001

[10] B. Goldlucke, M. Magnor, and B. Wilburn, "Hardware accelerated Dynamic Light Field Rendering", VMV 2002, Nov 2002.

[11] T. Naemura and H. Harashima, "The Ray-Based Approach to Integrated 3D visual Communication", in Three Dimensional Television, Video and Display Technologies,pp 183-209, Springer-Verlag , Berlin.