A Novel Image-Based Rendering System With A Longitudinally Aligned Camera Array

Jiang Li, Kun Zhou*, Yong Wang* and Heung-Yeung Shum

Microsoft Research, China

{jiangli, hshum}@microsoft.com

Abstract

This paper introduces a novel image-based rendering system to capture, represent and render real world and synthetic scenes. In our system, a longitudinally aligned camera array is mounted on a rotating arm supported by a tripod. The cameras are always aimed along the radial direction. The scene is captured by the camera array that rotates along a circle. Each pixel of the captured images is indexed by 4 parameters, i.e. the rotation angle of the camera array, the longitudinal number of the camera, the image column number and the image row number. Given the position and the viewing direction of an observer, the system can generate novel views by interpolating the captured pixels in real time without any geometric representation. If the observer is constrained to move on a plane, the size of the scene data can be further reduced to that of an approximately 3.5D plenoptic function. Compared with light field and Lumigraph, our method provides an easier inside-looking-out capture configuration and a uniform spatial sampling pattern. Our system goes a step further than concentric mosaics by allowing users to move continuously within a 3D cylindrical space, thus users can experience significant lateral as well as longitudinal parallaxes and lighting changes of a scene. Moreover, our method provides an image-based solution to the wandering of a large environment through concatenation of various wandering circles. Our technique has potential applications in entertainment, e-commerce and communication.

1. Introduction

In recent years, image-based rendering techniques have been developed to generate novel views of an environment from a set of pre-acquired images. These techniques have contributed significantly to the wandering around in virtual environment. With the use of image-based rendering techniques, the cost of rendering a scene is independent of the scene complexity and truly compelling photo-realism can be achieved since the images can be directly taken from the real world. While some approaches have been developed based on view interpolation[4], view morphing[10], and geometric recovery[12, 13], a branch of approaches which requires less interaction while constructing is based on plenoptic functions.

The original 7D plenoptic function[2] was defined as the intensity of light rays passing through every position, at every possible angle, for every wavelength and at every time instant. These rays are used to reconstruct virtual scenes in which people can wander and look around in it. However, at that time, many problems including how to capture a scene and what a uniform sampling pattern of a scene is have not been considered. By ignoring time and wavelength, a 5D plenoptic function[9] is proposed. It is performed via interpolating a set of panoramic images at different 3D locations, but difficult feature correspondence problems remain to be solved. If the scene can be constrained to a bounding box, a 5D plenoptic function can be reduced to a 4D plenoptic function called the light field[8] or a Lumigraph[7]. The Lumigraph has been used mostly to represent small objects that are viewed from outside. To capture the light field or a Lumigraph, precise camera poses have to be known or recovered. Since 4D data sets are extremely large and the sampling of a box is irregular, walkthroughs of a real scene using the light field or a Lumigraph have not yet been fully demonstrated.

QuickTime VR[5] using a collection of panoramas is a practical system that lets users stand at one position and look around in an environment. The small file size of a panorama, which represents a 2D plenoptic function, makes

*This work was completed while Kun Zhou and Yong Wang were interns at Microsoft Research, China.

the system applicable. The weakness of the system is that users have to jump between different capture positions if they want to navigate in the environment; therefore the goal of continuously wandering is still not achieved. As a further step, Concentric Mosaics[11] that represent a 3D plenoptic function capture a scene by spinning an off-centered camera on a rotary table and render novel views by combining appropriate captured rays. This method allows users to move continuously in a circular region and observe lateral parallax and lighting changes in the scene. Compared with the light field or a Lumigraph, concentric mosaics are easy to capture and have much smaller file size because only a 3D plenoptic function is constructed.

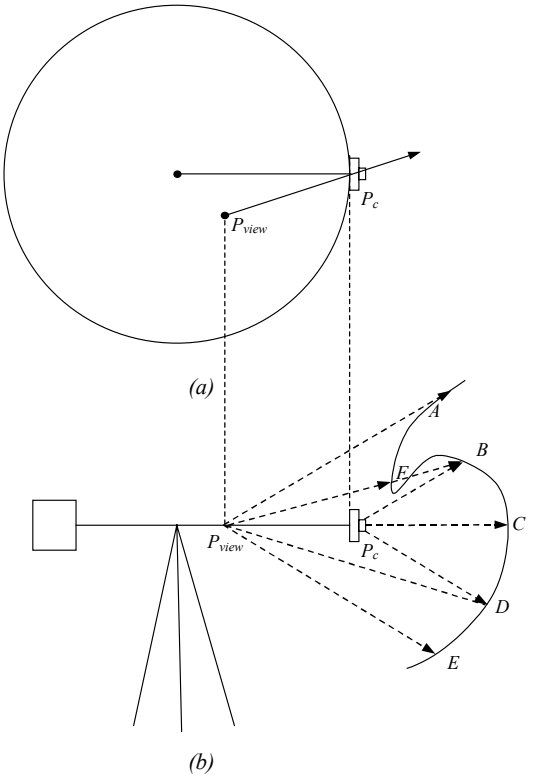


Figure 1: *The problems in concentric mosaics method. (a) Top view. (b) Side view.*

Unfortunately, concentric mosaics method inevitably contains vertical distortions and lacks vertical parallax. As illustrated in Figure 1, the view at any point P_{view} within the capture circle should be reproduced from the images captured at some point P_c on the circle. For those viewing directions that are parallel to the capture plane, the captured rays such as P_cC are identical to the required viewing rays such as $P_{view}C$. However, for other viewing directions that are not parallel to the capture plane, the captured rays such as P_cB are much different to the required viewing rays such as $P_{view}A$. Even when depth correction is introduced, not all the viewing rays can be reproduced from the captured rays. As shown in Figure 1(b), although the viewing rays, e.g. $P_{view}D$ can be reproduced from the captured rays, e.g. P_cD, the viewing rays, e.g. $P_{view}F$ could never be reproduced from the captured rays, e.g. P_cB since the cameras have never captured the part of the scene around point F.

The weakness of concentric mosaics lies in the less sampling of the vertical information of the scene. How can we capture more information along vertical direction and still limit the size of data file below that of a 4D plenoptic function? The answer is straightforward – utilizing a camera array instead of only one camera in the capture process. Using this setup, we can always retrieve those viewing rays that are off the capture plane from certain cameras in the vertical array and no depth corrections are needed. Without depth correction, the system becomes a purely image-based rendering system that can automatically generate novel views regardless of any geometric recovery. In addition, large environments can be easily constructed via the concatenation of various capture circles. At the same time, we observed that not all the captured rays need to be stored if users were constrained to move on a plane as in the case of concentric mosaics. This makes the file size of this system equivalent to an approximately 3.5D plenoptic function.

The remainder of this paper is organized as follows. In Section 2, we introduce the setup of our capture system and discuss our sampling considerations. The process that indexes and compresses the image data will be discussed in Section 3. Section 4 is devoted to the rendering of a novel view. The concatenation of various capture circles will be discussed in Section 5. A demo that shows the wandering in a room is illustrated in Section 6. Finally we conclude our work and discuss future directions in section 7.

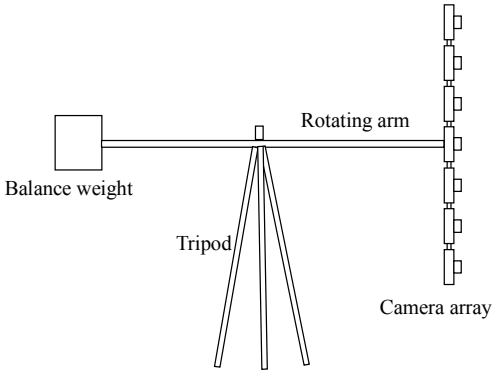


Figure 2: *The setup of the capture system.*

2. The capture system

Figure 2 illustrates the setup of our capture system. A longitudinally aligned camera array is mounted on a horizontal arm supported by a tripod. The cameras are always aimed along the radial direction. The scene is captured while the array rotates along a circle, which is referred to as the "capture circle". The plane swept out by the rotating arm is referred as the "capture plane".

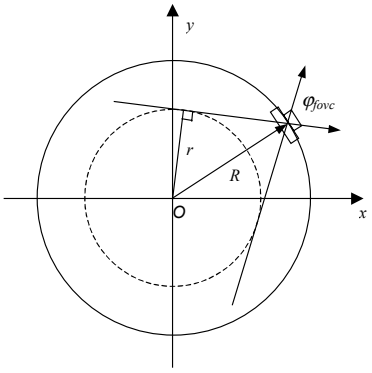


Figure 3: *The relationship between the radius of the wandering circle and the lateral field of view of the capture camera.*

Let us first explain some parameters that are related to the capture and rendering processes.

As illustrated in Figure 3, the radius r of a circle in which users can freely move and view depends on both the lateral field of view φ_{fovc} of the capture camera and the radius R of the capture circle. It is expressed as:

$$r = R\sin\left(\frac{\varphi_{fovc}}{2}\right) \tag{1}$$

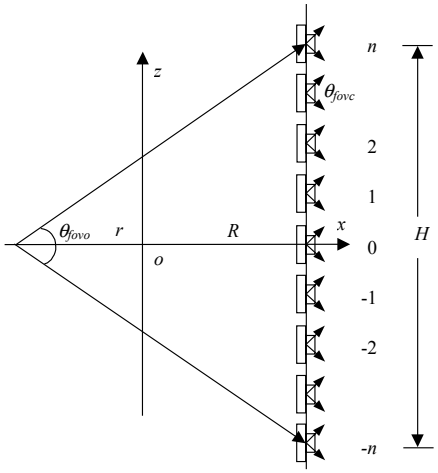


Figure 4: *The relationship between the height of the camera array and the longitudinal field of view of the observer.*

It is obvious that any ray that originates from any viewpoint within the circle and passes through the capture camera must be within the field of view of the capture camera. Therefore, any novel view of the user can always be reproduced from the captured images. We refer this circle as the "wandering circle".

In addition, as shown in Figure 4, the height H of the camera array should be so designed that the longitudinal field of view θ_{fovo} of the observer is still covered by the camera array even if he/she is located at the far end of the wandering circle. We have

$$H \geq 2(r+R)\tan\left(\frac{\theta_{fovo}}{2}\right) \tag{2}$$

Of course, the longitudinal field of view of the observer should not be wider than that of the capture camera, i.e.

$$\theta_{fovo} \leq \theta_{fovc} \tag{3}$$

Now let us consider how densely we should deploy the cameras on the array and how many images each camera should take in one circle.

Assuming that the width and height of the novel image of the observer are w_o and h_o, respectively, the ideal longitudinal interval d_c between adjacent cameras should be

$$d_c = \frac{2(r+R)\tan\left(\frac{\theta_{fovo}}{2}\right)}{h_o} \tag{4}$$

It means that each row of the novel image corresponds to each camera on the array. This guarantees that the longitudinal parallax of the scene will be reproduced. On the other hand, the angular rotation increment $\Delta\varphi_i$ of the camera array should be approximately

$$\Delta\varphi_i \approx \frac{\varphi_{fovo}}{w_o} \tag{5}$$

It means that each column of the novel image corresponds to each angular position of the camera array on the capture circle. This guarantees the reproduction of the lateral parallax of the scene. Finally, the resolution of the capture camera should be chosen as that of the observer's view.

3. Processing of captured image data

Since the size of the captured image data of a scene is usually very large, it is necessary to compress the data file before the entire image data are loaded into the main memory of a computer. Compression makes sense because there are significant correlation and redundancy between adjacent images. The technique we choose is vector quantization[6], which is a compression method with quick selective decoding[14].

It is worthy to note that in the case where the observer is constrained to move on a 2D circular plane as in the case of concentric mosaics, we can further reduce the captured image data by discarding the part of image area that would never be seen. As illustrated in Figure 5, we draw a line

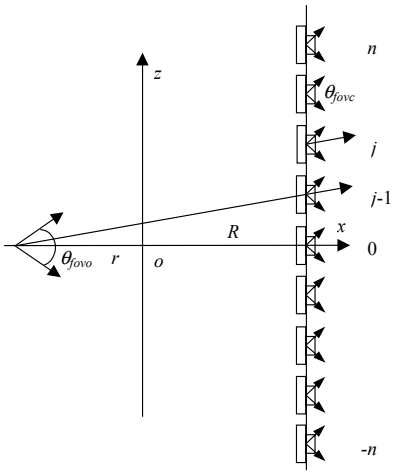


Figure 5: *The reduction of the captured image data.*

connecting the far end of the wandering circle and the (j-1)th camera. The novel viewing rays that originate within the wandering circle and pass through the interval between camera j-1 and camera j should be reproduced by interpolating between captured rays of camera (j-1) and camera j. The elevation angles of these rays are always larger than that of the line of camera j, which is parallel to the above connecting line. Therefore, for camera j, the captured rays with their elevation angles smaller than that of the above parallel line would never be used in the rendering. So we only need to store the parts of image rows with their corresponding elevation angles lying between angles

$$\arctan\left(\frac{(j-1)d_c}{(r+R)}\right)$$

and $\theta_{fovc}/2$ for cameras with $j > 0$ or

$$\arctan\left(\frac{(j+1)d_c}{(r+R)}\right)$$

and $-\theta_{fovc}/2$ for cameras with $j < 0$. It is obvious that the farther away is the camera from the array center, the fewer are the image rows needed to be stored. Thus the amount of captured data as a 4D plenoptic function is effectively reduced to the amount of an approximate 3.5D plenoptic function. Comparing with light field and Lumigraph, this system significantly reduces the data size without sacrificing any 3D parallaxes. Contrasting with concentric mosaics, this system eliminates vertical distortions and displays significant longitudinal parallax and lighting changes.

4. Rendering novel views

The principle of rendering novel views is to interpolate each novel viewing ray by finding the nearest captured rays. The captured rays are indexed by 4 parameters, i.e. the rotation angle of the camera array, the longitudinal number of the camera, the image column number and the image row number. We will discuss the determination of these 4 parameters in the following.

4.1 Determination of the rotation angle of the camera array

As illustrated in Figure 6, an observer is supposed to stand at point P. One of the viewing rays from the observer is denoted as V. P_0 and V_0 are the projection vectors of P and V on the capture plane respectively.

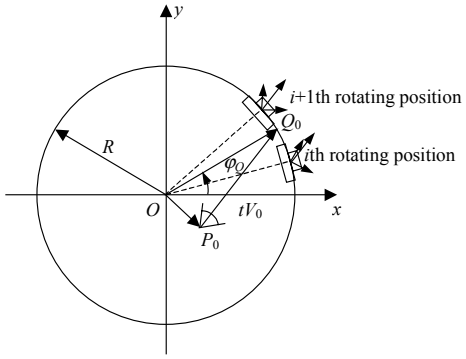


Figure 6: *The determination of the rotation angle of the camera array.*

The intersection point Q_0 of the viewing ray V_0 and the capture circle is obtained by

$$Q_0 = P_0 + tV_0 \tag{6}$$

where t is the positive root of the equation

$$\| P_0 + tV_0 \|^2 = R^2 \tag{7}$$

The direction from the circle center to the intersection point Q_0 may either coincide with one of the rotating positions of the camera array or lie between two adjacent rotating positions. In the latter case, the interpolation weights are inversely proportional to the angular differences between the direction and the two rotating positions.

4.2 Determination of the image column number

According to geometric relation, the azimuth angle $\Delta\varphi$ between V_0 and the direction of camera at the above rotating position is equal to the azimuth angle φ_{V0} of V_0 minus the azimuth angle φ_Q of the rotating position Q_0. The angle $\Delta\varphi$ may correspond to either one or two adjacent columns of the images captured by the camera array at the rotating position. In the latter case, the interpolation weights are inversely proportional to the angular differences between $\Delta\varphi$ and those of the two adjacent columns of the images.

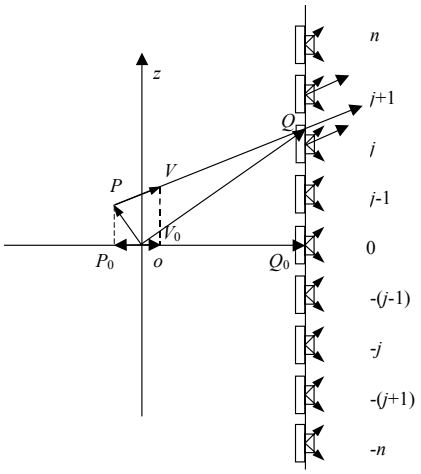


Figure 7: *The determination of the longitudinal camera number.*

4.3 Determination of the longitudinal camera number

Figure 7 shows a diagram of a section plane determined by the point P and the viewing direction V and its projection V_0. The intersection point Q of the viewing ray V and the capture cylinder can be obtained by

$$Q = P + tV \tag{8}$$

where t is the positive root of Eq.(7). The height of the intersection point Q may be equal to the height(s) of either one or two longitudinally adjacent cameras. In the latter case, the interpolation weights are inversely proportional to the differences between the height of Q and the heights of the two longitudinally adjacent cameras.

4.4 Determination of the image row number

Figure 8 shows a diagram of viewport of a camera. Assume that the above viewing ray V intersects with image plane of the viewport at point A. Let B be the projection of A on the lateral axis of the viewport, therefore $\angle AOB$ is equal to the elevation angle of vector V. Suppose that a line that is parallel to the lateral axis and passes through A intersects with the longitudinal axis of the viewport at D. It is angle $\angle COD$ instead of angle $\angle AOB$ that directly determines the corresponding rows of the captured images. The relationship between $\angle COD$ and $\angle AOB$ is

$$\tan(\angle COD) = \tan(\angle AOB)\sec(\angle BOC) \tag{9}$$

where $\angle BOC$ is exactly the angular difference $\Delta\varphi$ mentioned in section 4.2. Angle $\angle COD$ may either be equal to one of the corresponding angles of the image rows or lie between the corresponding angles of two adjacent image rows. In the latter case, the weights of the linear interpolation are determined by the angular differences of $\angle COD$ and the corresponding angles of two adjacent image rows.

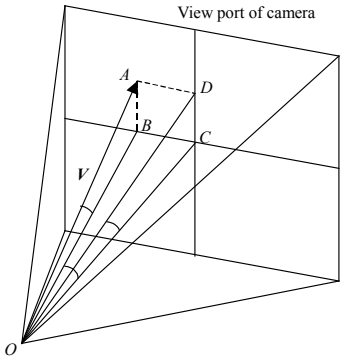


Figure 8: *The determination of the image row number.*

5. Concatenation of the capture circles

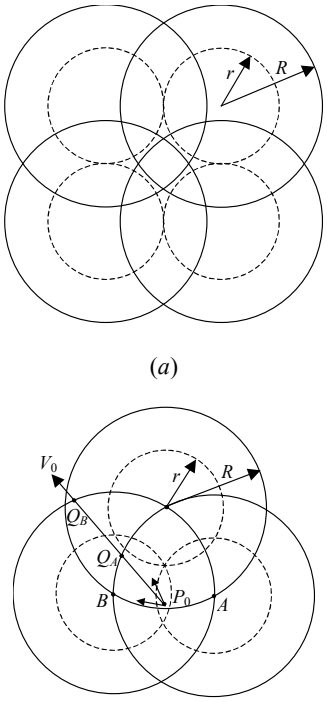


Figure 9: *The concatenation of capture circles.*

One of the advantages of this system is that it provides a theoretical and practical image based solution to the wandering of large environments. Previous QuickTime VR method requires observers to jump between hot spots, and light field and Lumigraph are all local methods and never provide concatenation solutions. As illustrated in Figure 9, in this system, it is very easy to concatenate various capture circles and extend the wandering space of observers. In

Figure 9, capture circles with radius R are represented by solid lines and wandering circles with radius r are drawn in dash lines. Figure 9(a) indicates one of the most sparse situations of the concatenation of the capture circles, which allows an observer to continuously move from one wandering circle to another wandering circle through their tangent points, whereas Figure 9(b) shows one of the most dense situations, which allows observers to move freely between wandering circles without any restriction. Other configurations may be designed according to the distribution of scene objects. The rendering of novel views is also very easy when observers are walking through various wandering circles since this method does not rely on any geometric representation of the scene. If the observer stands at a common area of some wandering circles, essentially anyone of the related capture circles can be used in the rendering of novel view (See the definition of wandering circle in Section 2). For consistence, we usually choose the capture circle nearest to the viewpoint. For example, assuming that an observer stands within a common area of wandering circles A and B (See Figure 9 (b)). The horizontal projection of one of the rays originated from the field of view of the observer is denoted as V_0. It intersects with circle A and circle B at Q_A and Q_B respectively. We use capture circle A to reproduce pixel corresponding to the viewing ray in the novel view since Q_A is nearer than Q_B to P_0.

6. Experiment results

We simulate the capture process in a synthetic scene, which is modified from the scene "Brians Beach Bungalow" downloaded from 3DCAFE[1]. We choose the radius of the capture circle as 1.57 meters. Both of the vertical and lateral fields of view of the camera are 45°. Therefore the radius of wandering circle is 0.6 meters according to Eq.(1). We arrange 61 cameras in an array of 2.7 meters high. Each camera capture 360 pictures with a resolution of 256×256 pixels as the array rotates one round. It costs about 30 hours to render a total of 21960 images in a Pentium III 500 PC. The amount of the resultant raw data is about 4GB. After vector quantization (12:1) and Lempel-Ziv coding (4:1), the size of the data file is reduced to 80MB. Our system achieves a frame rate of 15 frames per second in the wandering. As illustrated in Figure 10, users can move left (a_1) and right (a_2), up (b_1) and down (b_2), forward (c_1) and backward (c_2), and look upward (d_1) and downward (d_2). One can see that lateral and longitudinal parallaxes obviously exist between these pictures. In addition, the vertical lines of a wall are obviously inclined when the observer looks upward (d_1) and download (d_2). Therefore, our system correctly reproduces the perspective effects. Interested readers can visit http://research.microsoft.com/~jiangli/eg2000demo.htm to view a video demo of the wandering.

7. Conclusion

In this paper we have described a novel image-based rendering system. According to the configuration of the system, images of scenes are captured using a longitudinally aligned camera array rotating along a circle with its orientation kept outward. Each pixel of the captured images, which represents a viewing ray of a scene, is indexed by 4 parameters, the rotation angle of the camera array, the longitudinal number of the camera, the image column number and the image row number. When the viewing position and direction of an observer are given, the system generates novel views by interpolating the captured pixels in real time without any geometric models.

This system represents a novel sampling pattern of a 4D plenoptic function. Compared to the light field, Lumigraph and other sampling method of 4D plenoptic function[3], our method provides an easier capture configuration, a uniform spatial sampling and an outward looking experience. Because our method does not require any geometric recovery, it is suitable for wandering around in a large environment.

In the case that observers only need to move inside a circle on a plane, concentric mosaics provided a 3D plenoptic function solution. However, vertical distortions exist and vertical parallax is absent due to insufficient sampling along the vertical direction. Indeed, for the situation of wandering on a 2D plane, a 4D plenoptic function is still needed with 2D for position and 2D for ray direction. Rather than employing a standard 4D plenoptic function that captures rays at every point on the plane along every direction, our method only captures rays by rotating a vertically aligned camera array. By discarding parts of captured image areas that will never be seen by the observer, our method reduces the file size to that of an approximately 3.5D plenoptic function.

We are working on a number of problems towards the improvement and the practical use of the system. First, since it is difficult to practically mount dozens of cameras on a vertical bar, we are designing a similar device using only one camera but elevating the rotation arm a certain height in each round of rotation. If it costs 2 minutes to rotate one round, the total capture time would be several hours. Second, since there are significant correlation and redundancy between adjacent images, more efficient compression methods such as prediction-based IBR compression that employs MPEG4 codec techniques are under study. Third, as the file size of the scene data is still much larger than the currently available bandwidth on the Internet, a random access method that only retrieves parts of the data necessary for rendering current views is under development. Finally, special purpose CCD array equipments that can replace the ordinary camera array in the capture setup could also be designed.

Acknowledgements

We would like to thank Min-Sheng Wu for providing compression code and Hong-Hui Sun for implementing the first prototype of the system. Yiying Tong, an intern from Zhejiang University also made his contributions to the completion of the system. Finally, special thanks go to Ka Yan Chan for patiently proofreading the paper.

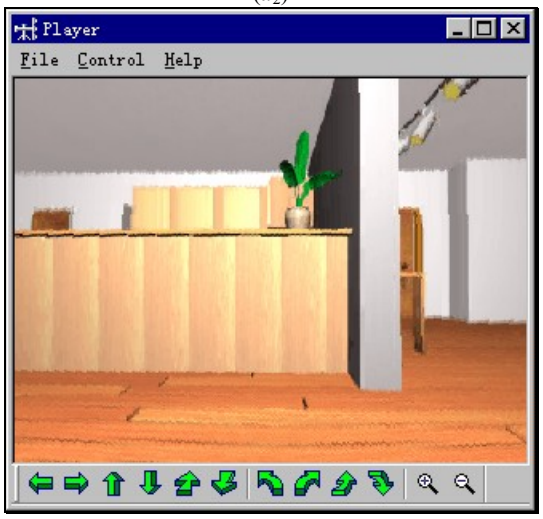
References

1. http://www.3dcafe.com/ 6
2. E. H. Adelson and J. Bergen, The Plenoptic Function and the Elements of Early Version, Computational Models of Visual Proceeding, pp.3-20, MIT press, Cambridge, MA, 1991. 1
3. Emilio Camahort, Apostolos Lerios and Donald Fussell, Uniformly Sampled Light Fields, Rendering Techniques'98, pp.117-130, 1998. 6
4. S. E. Chen and L. Williams, View Interpolation for Image Synthesis, ACM Computer Graphics, Proc. of SIGGRAPH 1993, pp.279–288, 1993. 1
5. S. E. Chen, QuickTime VR – An Image-based Approach to Virtual Environment Navigation, ACM Computer Graphics, Proc. of SIGGRAPH 1995, pp.29-38, 1995. 1
6. A. Gersho, R. M. Gray, Vector Quantization and signal compression, Kluwer Academic Publishers, 1992. 3
7. S. J. Gortler, R. Grzeszczuk, R. Szeliski, and M. F. Cohen, The Lumigraph, ACM Computer Graphics, Proc. of SIGGRAPH 1996, pp.43-54, 1996. 1
8. M. Levoy and P. Hanrahan, Light Field Rendering, ACM Computer Graphics, Proc. SIGGRAPH 1996, pp.31-42, 1996. 1
9. L. McMillan and G. Bishop. Plenoptic Modeling: An Image-based Rendering System, ACM Computer Graphics, Proc. of SIGGRAPH 1995, pp.39-46, 1995. 1
10. S. M. Seitz and C. M. Dyer, View Morphing, ACM Computer Graphics, Proc. of SIGGRAPH 1996, pp.21-30, 1996. 1
11. H. Y. Shum and L. W. He, Rendering with Concentric Mosaics, ACM Computer Graphics, Proc. of SIGGRAPH 1999, pp.299-306, 1999. 1
12. Y. Yu and J. Malik, Recovering Photometric Properties of Architectural Scenes from Photographs, ACM Computer Graphics, Proc. of SIGGRAPH 1996, pp.207–218, 1998. 1
13. Y. Yu, P. Debevec, J. Malik and T. Hawkins, Inverse Global Illumination: Recovering Reflectance Models of Real Scenes from Photographs, ACM Computer Graphics, Proc. of SIGGRAPH 1999, pp.215-224, 1999. 1
14. J. Ziv and A. Lempel, A Universal Algorithm for Sequential Data Compression, IEEE Transactions on Information Theory, Vol.23, pp.337-343, 1997. 3

(a_1)

(a_2)

(b_1)



(b_2)

Figure 10: *The wandering in a living room. A user can move left (a_1) and right (a_2), up (b_2) and down (b_2), forward (c_1) and backward (c_2), look up (d_1) and down (d_2), and take other conventional actions such as turning left and right, and zooming in and zooming out.*