1998 International Computer Symposium
Workshop on Computer Graphics and Virtual Reality
December 17-19, 1998, N.C.K.U., Tainan, Taiwan, R.O.C.

A PERSPECTIVE PERCEPTION ENHANCEMENT ALGORITHM FOR IMAGE-BASED VIRTUAL ENVIRONMENT

Gia-Hong Tsai and *Zen-Chung Shih*
Department of Computer and Information Science
National Chiao Tung University, Hsinchu, Taiwan, R.O.C
Pei-Sen Liu
Multimedia Laboratory
Institute of Information Industry, Taipei, Taiwan, R.O.C

ABSTRACT

Recently image-based rendering becomes an important technique in virtual reality. It can generate photo-realistic images interactively. Among image-based virtual environments, however, there is one major deficiency – the lack of perspective perception. Because of the loss of depth information, users may not have immersive feeling when walkthrough in the scene. In this paper, we extract the foreground object images from the original panoramic image. When walking through in the environment, users can feel that the foreground objects are in front of the background, and provide strong perspective perception.

1. Introduction

Traditionally, a virtual environment is modeled as a collection of three-dimensional geometric entities, such as points, lines, and polygons. To provide interactive walkthrough, these geometric entities need to be rendered in real-time. However, it is not an easy work. More detail objects need more complicate geometric models. So it takes more computation time in rendering process. Thus, it is a tradeoff between rendering quality and performance. In image-based VR system, the whole environment is constructed by a set of real-world images. So there are no three-dimensional geometrical entities to be modeled. The rendering time depends on the resolution of panoramic image rather than scene complexity. It usually takes constant rendering time.

Among all image-based virtual reality system, however, there is one major deficiency – the lack of perspective perception. All objects in the virtual environment belong to a panoramic image. When users walk through in the environment, the occlusions among objects and the background will never change. It seems looking at a picture, rather than staying in a three-dimensional environment.

Horry, Anjyo, and Arai[3] presented a method for making animations in a two-dimensional picture. It extracts foreground objects from the background image, creates simplified model for the background, and then rendering via two-to-three dimensional transformations. Then users can walkthrough (rotation, translation, zoom, etc.) in a single image. The occlusion among foreground objects and the background will change during walking through the scene. Thus, users can obtain a little perspective perception. But this approach deals with only a single image. In this paper, we improve it and apply to panoramic images to enhance the perspective perception in image-based virtual reality systems.

The rest of this paper is organized as follows. In section 2, we review the related works. In section 3, we introduce the traditional panoramic approach. The details of how to construct a perspective image-based virtual environment and the rendering process are discussed in section 4. In section 5, we show our experimental results. Conclusions and future works are given in section 6.

2. Related Works

Recently there have been a lot of researches in image-based system. The first commercial product of image-based virtual reality is the *QuickTime VR*[2]. It suggests that the traditional modeling and rendering process can be skipped. Instead, a series of captured environmental maps allow a user to look around a scene from fixed points in the three-dimensional space. The interactive environment comprises two types of players, they are panoramic movie player and object movie player. The panoramic movie player provides the functions of panning, zooming and navigating in a scene for this VR system. It also includes an interactive application, "hot spot" picking capability. The object movie player uses frame indexing to allow the user to rotate an object or look the object from different viewing directions. Walking in space is accomplished by "hopping" to different panoramic points.

Plenoptic modeling[7] is a modeling system which is similar to image-based approach. The whole scene consists of a series of reference images. The "plenoptic function" is a parameterized function for describing everything that is visible from a given point in space. *Plenoptic modeling* presents a real-time system that uses panoramic photographs with depth computed, from stereo correspondence. They implement three key steps of sampling, reconstructing, and resampling the plenoptic function for providing a consistent framework. However, this method is able to obtain acceptable results for nearby

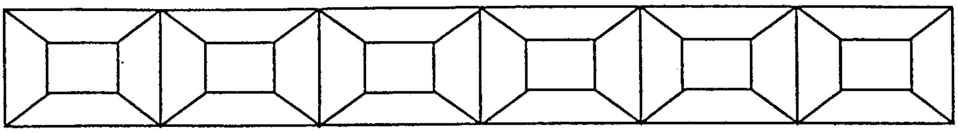


Figure 1: Multiple perspective views inside a panorama.

views using user input to aid the stereo depth recovery.

Disparity Morphing[5] proposed a method for automatic generation of high-quality photo-realistic 360-degree stereo panoramas using a sequence of stereo images acquired by two cameras. A stereo panoramic imaging system (SPISY) has been implemented by using the proposed method. In order to generate stereo views, two cameras are used for simultaneous image acquisition in the SPISY. There is only one vertical rotation axis, at least one camera cannot have its lens center location on the rotation axis. Assume the rotation axis passes through the lens center of the left camera. This makes the seaming of the right panoramic images difficult to be done automatically. In this work, they analyze the image disparity caused by the dislocation of the lens center from the vertical rotation axis, and then propose a disparity morphing algorithm to solve this problem successfully.

Horry, Anjyo, and Arai[3] presented a method for easily making animations from one two-dimensional picture or photograph of a scene. In this work, "*Tour into the Picture* (TIP)", the animation is created from the viewpoint of a camera which can be three-dimensionally "walked or flown-through" the two-dimensional picture or photograph. This method is not intended to construct a precise three-dimensional scene model. A "spidery mesh" is employed in this method to obtain a simple scene model from the two-dimensional image of the scene using a graphical user interface. There is a vanishing point in a perspective projection. The modeling process starts by specifying the vanishing point in the two-dimensional image. Then it distinguishes foreground objects from background. This model is then a polyhedron-like form with the vanishing point being on its base. The "billboard-like" representation and its variation are used for foreground objects. The rendering process is implemented via the two-to-three dimensional transformations. While walking through the scene, users will feel that the foreground objects are standing on a polygon of the background model, and the occlusion among foreground objects and the background will change.

3. Panoramic Image

In this section, we introduce the process of panoramic image generation. There are several methods for creating panoramic images. One is using the panoramic cameras. The advantage of using the panoramic camera is that it can capture the 360-degree panoramic image at one shot, without stitching a lot of overlapped images. But the panoramic cameras are still very expensive.

Another way of panoramic image generation is to use a general-purpose camera set on a tripod, taking a shot at certain rotation angle. This kind of panoramic imaging system stitches the overlapped shots on a cylinder[2][5] or on a sphere[9]. To generate the seamless panoramic image, it should deal with the following problems. Because of the varying lighting environment, the duration of camera exposure is not fixed. Thus the intensity between neighboring images could be different. The smoothing process[1][5] is needed here. Next, while taking pictures, the camera could be tilted at each shot. We should perform a tilt correction[5] to correct this error. Another problem is the distortion caused by perspective projection. In order to correct this distortion, we should do a projection warping[2][5]. Finally, the warped images are stitched together to generate the panorama. An image registration algorithm[4][6][10] should be performed. The task of image seaming can be completed by using an image blending algorithm.

4. Perspective Panorama

Perspective perception is very weak in the traditional image-based virtual environment system. In this section, we discuss the concept of perspective panoramic image. We also propose the procedure to construct a perspective image-based virtual environment.

Perspective projection provides good depth cue for two-dimensional images. Users can easily distinguish the depth sequence of objects in the perspective-projected image. Traditionally, we process the perspective projection only for a single image. It is not an easy work for panoramic images.

4.1 Perspective Panoramic Image

For the construction of the perspective panoramic image, there are multiple vanishing points for a perspective panoramic image. As shown in Figure 1, the background panoramic image looks like.

A panoramic image is composed of multiple subimages. Each shows a perspective view of the scene. Figure 2 depicts the top-view of the three-dimensional model.

The circle is the original panoramic image, we warp it into six flat subimages. For each subimage, we create the corresponding three-dimensional scene as shown in Figure 2. It seems that the virtual camera can walk into the scene freely. The view plane can also be projected onto the

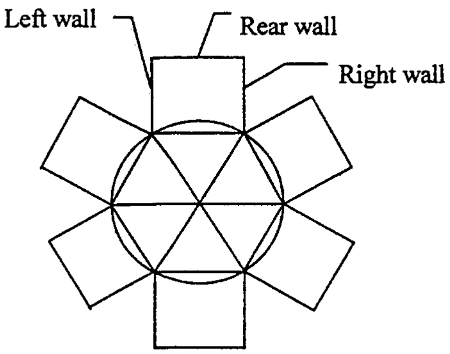


Figure2:The 3D model of multi-perspective image.

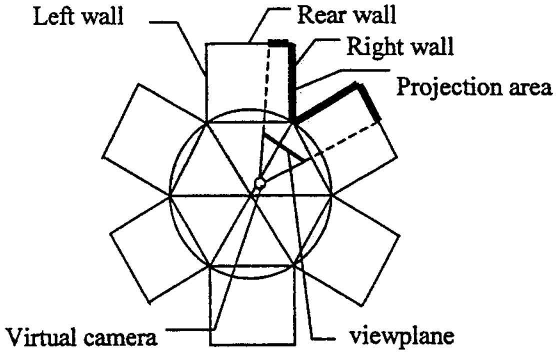


Figure3:The distortion occurs at the boundary between two

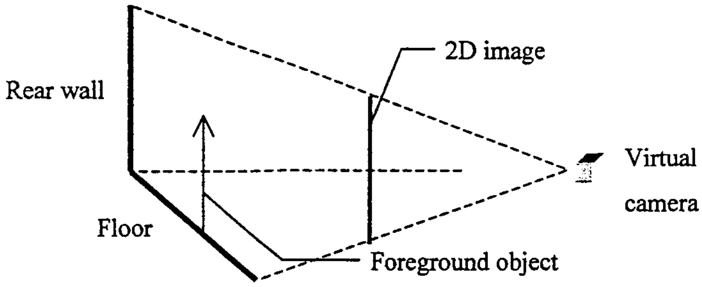


Figure 4: The simplified model.

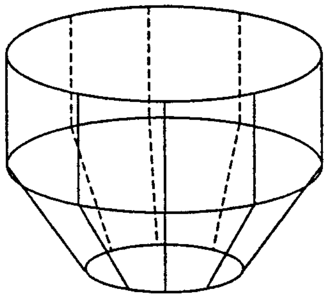


Figure 5:The modified model suitable for panoramic images.

Figure 6: Part of the original panoramic image.

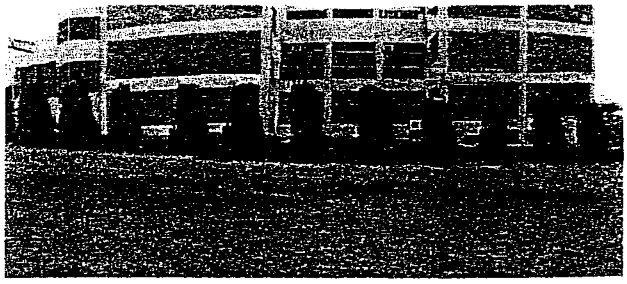
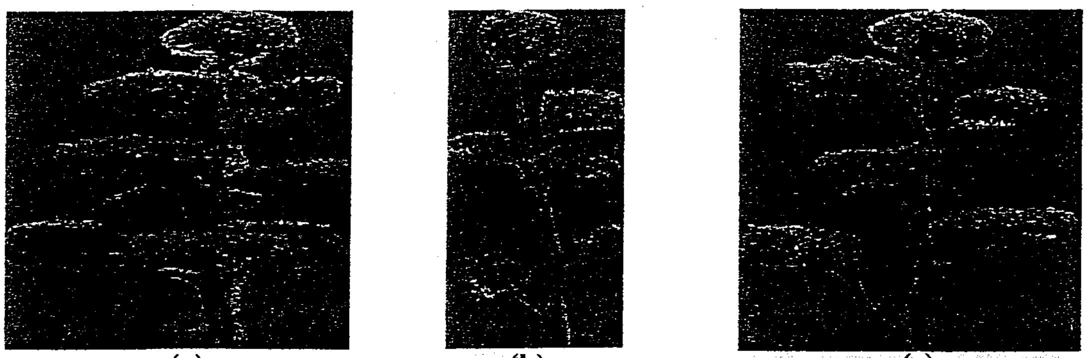


Figure 7: After extracting foreground objects.



(a) (b) (c)

Figure 8: Foreground object images.

three-dimensional model. Then we calculate the color of each pixel on the plane via a two-to-three dimensional transformation. However, there are still some problems.

While the virtual camera moves (translates, rotates, ...etc.), the final image of the new viewplane should be warped. This problem is not so serious in the animation within a single image. But, in panoramic image, the result of distortion is not acceptable, especially when virtual camera moving from one perspective view to another.

Besides, the two-to-three dimensional transformation can not be performed in real-time. Moreover how to divide the panoramic image into subimages depends on the scene. Thus, we should do some modification

Consider Figure 3. The distortion usually occurs at the left and right walls. So we eliminate the left and right walls of the scene model. In fact, we don't need the ceiling either. Furthermore we also eliminate the limitation that the floor must be on the plane $y = 0$. The foreground objects still stand on the floor, as shown in Figure 4.

In order to eliminate the distortion, enough subimages are needed. The whole three-dimensional model looks like Figure 5.

We use the texture mapping technique to replace the two-to-three dimensional transformation. The advantage is that, with texture mapping, the rendering process can be done easily.

4.2 Foreground Objects Extraction

Now we discuss the extraction of foreground objects from the original panoramic image. The foreground objects in the tree-dimensional scene are closer to the camera and standing on the floor. We have decided which objects belong to foreground while taking the pictures.

Extracting foreground objects is an easy work. But for the reconstruction of background panoramic image, we should occlude the traces of the foreground objects. The only way is to replace pixels in the area of foreground objects by the color information of neighborhood. Figure 6 shows part of the original panoramic image. Figure 7 shows the background panoramic image after extracting the foreground objects. Figure 8 depicts several foreground images. The foreground objects are modeled as quadrangles in the three-dimensional scene. The blue area is transparent at the rendering stage. Figures 9(a) and (b) show the whole panorama and its background image respectively.

4.3 Preprocessing of Background Images

From now on, we already have foreground object images and background panoramic images. But we can not use the background panoramic images directly. The three-dimensional model is created by our imagination. We need a preprocess for the background panoramic images

Since the original image should be projected on the three-dimensional model, we need to warp the image before rendering.

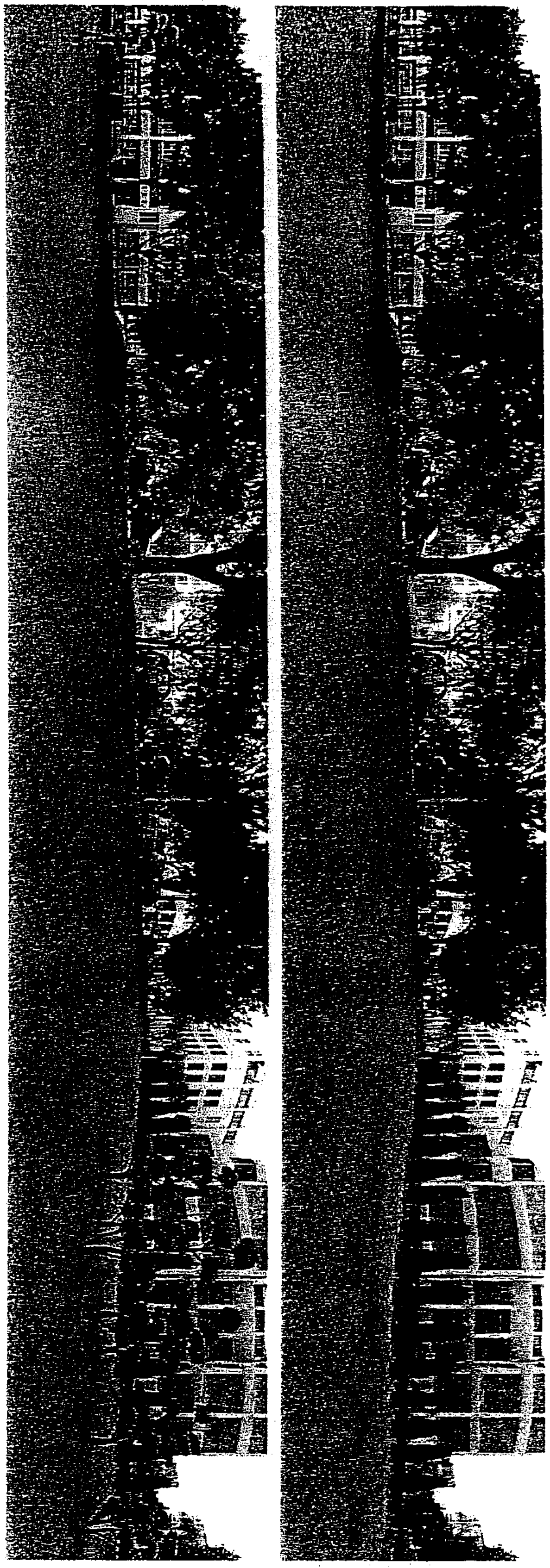

(a) (b)

Figure 9:The whole panorama and its background image of the scene.

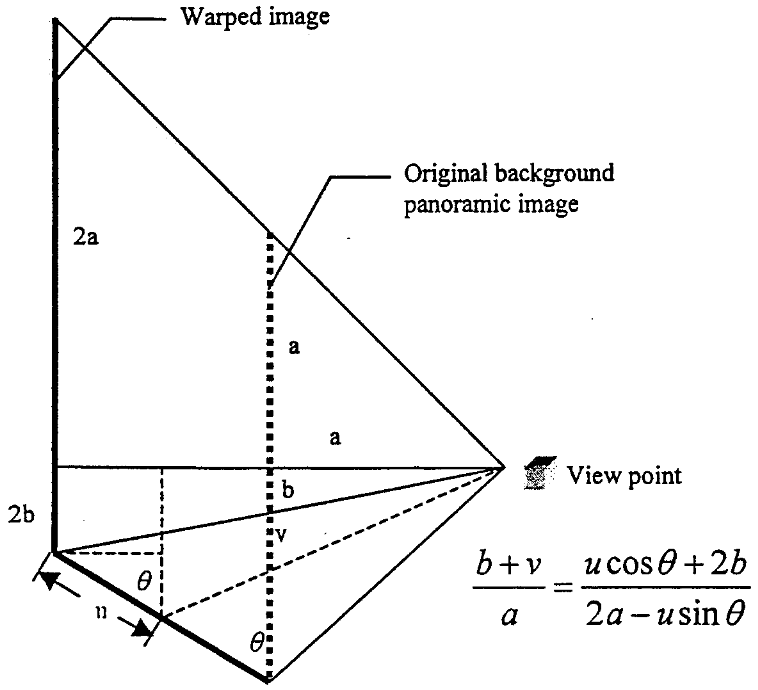


Figure 10: Projection warping of the background image.

Consider Figure 10. The original background panoramic image is projected onto the scene model according to the view point. We warp the image. Assume that the pixel of any y value is on the resulted warped image. We can use the following equation to find y' value of the corresponding pixel on the original background panoramic image.

$$\text{if } y<2a+2b$$
$$y'=y/2$$
$$\text{if } y\geq 2a+2b$$
$$y'=\frac{(y-(2a+2b))\times\cos\theta+2b}{2a-(y-(2a+2b))\times\sin\theta}\times a-b+a$$

The warp function is selected case by case. After obtaining the warped background panoramic images, we use it as textures of the three-dimensional scene. Figure 11 shows the textures of the scene model. The texture include the warped background panoramic image and the foreground images.

The rendering process of two-to-three dimensional transformation in TIP can not be performed in real-time. In our work, we use the technique of texture mapping in rendering process. The advantage is that the rendering process can be performed efficiently, and users can walkthrough in the scene interactively.

Figure 12 shows the whole scene. We use background panoramic images as the texture of background models and foreground object images as the texture of foreground model. The color format of textures are R, G, B, and A. The A value of the blue area of foreground images is set to zero. Thus, while rendering, the blue area of foreground image will be transparent.

5. Experimental Results

Our implementations are using Visual C++ environment with OpenGL library as a MFC application. And the hardware platform is Pentium 200 MHz with a three-dimensional accelerator. In TIP, the resolution of the images in animation is of 640x480. The wall clock time for rendering a frame is around 0.1 sec. In our experiments, users can walkthrough in the scenes in real-time, but the rendering performance depends on the graphics accelerator. The rendering image size in our system is 400x300 pixels.

The example in our work is in a grassplot in our campus. A few plants in the grassplot are the foreground objects in this case. Figure 9(a) and (b) show the original panoramic image and its corresponding background image. Figures 13 and 14 are panning and zooming processes respectively.

6. Conclusions

In this paper, we propose a method which enhance the perspective perception in image-based virtual environment system. The main idea is extracting the foreground objects from the original panoramic images. Traditional panoramic imaging system stitches the

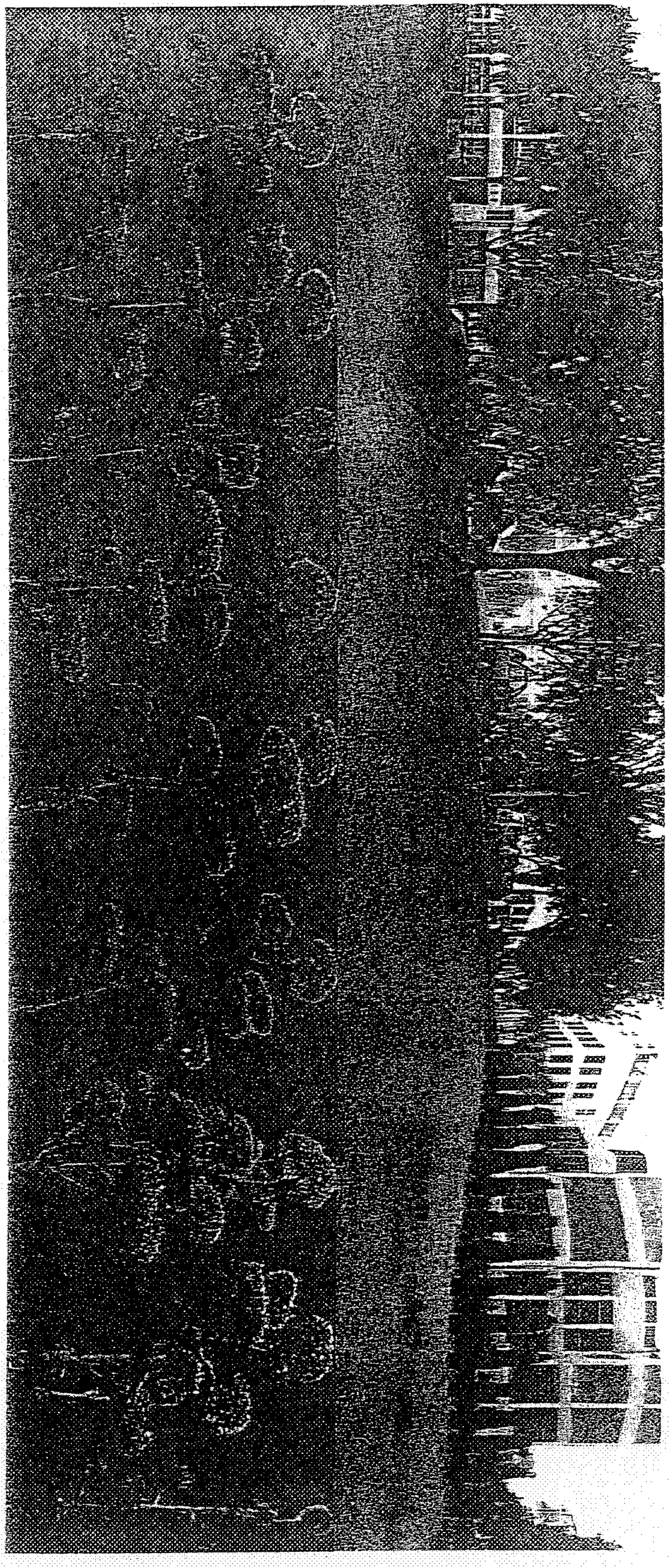


Figure11:Textures of the scene

overlapped shots on a cylinder. In our work, we make some modification on the scene model. There are rear wall and floor in our model and we use them be the projection of the background panoramic image. The foreground objects are modeled as quadrangles and standing on the floor. In our system, users can zoom and pan (horizontally

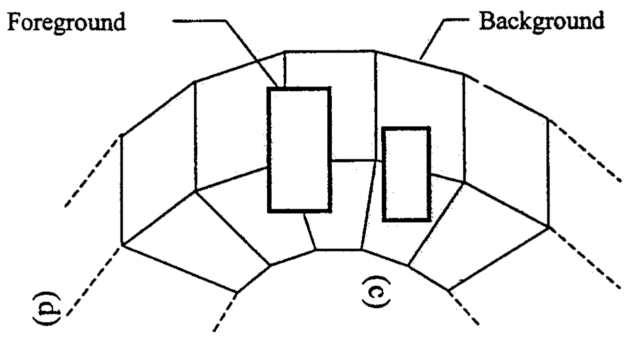
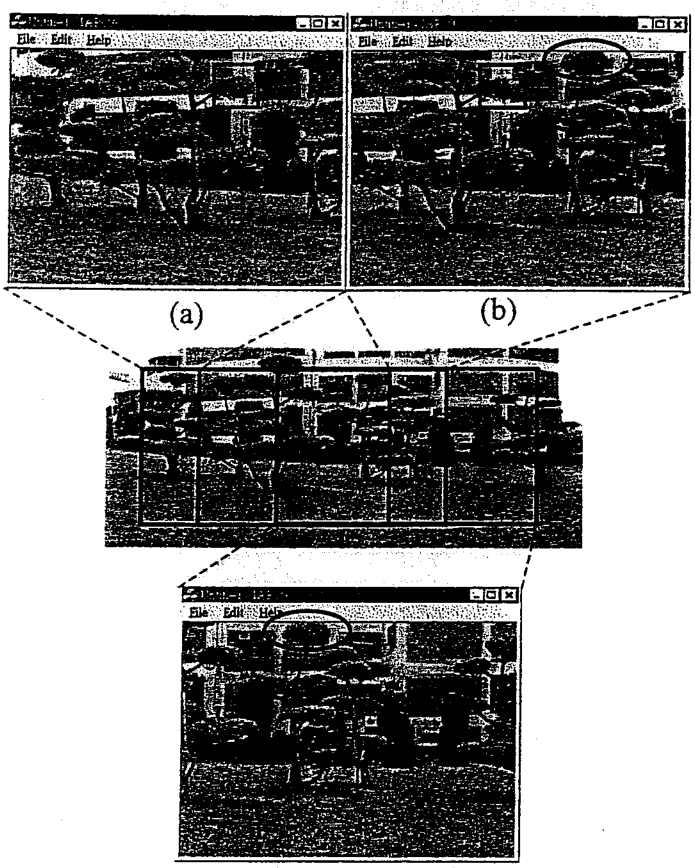


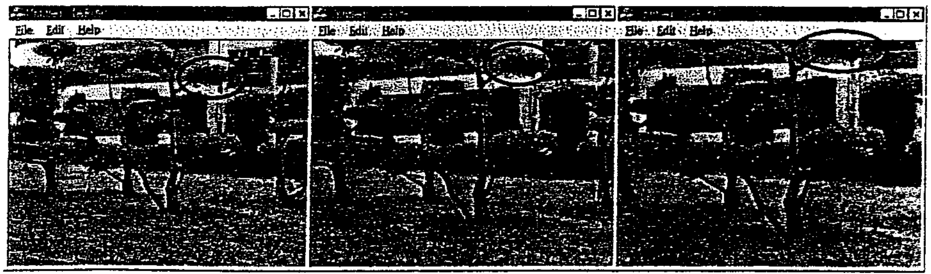
Figure 12: The model of the texture



(a) (b)

(c)

Figure 13: Panning of the scene.



(a) (b) (c)

Figure 14: Zooming of the scene.

and vertically) the virtual camera. While walking through the virtual environment, users will feel that the foreground objects are in front of the background, and the occlusion between them will change from different view point. Thus our system can provide strong perspective perception.

Currently, our system deals with single-node environment. We are trying to deal with multi-node environment in the near future. To accomplish this, we must provide continuous views during switching among nodes. The *view morphing*[8] approach is a good solution to this problem.

References

[1] K. M. Chen and Z. C. Shih, "*An Approach for Smoothly Switching in Image-Based Virtual Environment Navigation*", Department of Computer Information Science, National Chiao Tung University.

[2] S. E. Chen, "*QuickTime VR — An Image-Based Approach to Virtual Environment Navigation*", Computer Graphics (Proc. SIGGRAPH '95), p.29-p.38.

[3] Y. Horry, K. I. Anjyo, and K. Arai, "*Tour Into the Picture: Using a Spidery Mesh Interface to Make Animation from a Single Image*", Computer Graphics (Proc. SIGGRAPH '97).

[4] H. C. Huang and Y. P. Hung, "*Adaptive early jump-out technique for fast motion estimation on video coding*", Proceeding of ICPR, p-864-p868, Vienna, August 1996.

[5] H. C. Huang and Y. P. Hung, "*Disparity Morphing and Automatic Generation of Stereo Panoramas for Photo-Realistic Virtual Reality System*", Institute of Information Science, Academia Sinica, Taipei, Taiwan, Roc.

[6] C. F. Lin and C. L. Lee, "A Robust Method for Image Registration", Institute of Electrical Engineering and Computer Engineering and Science, Yuan-Ze Institute of Technology.

[7] L. McMillan and G. Bishop, "*Plenoptic Modeling: An Image-Based Rendering System*", Computer Graphics (Proc. SIGGRAPH '95), p.39-p.46.

[8] S. M. Seitz and C. R. Dyer, "*View Morphing*", Computer Graphics (Proc. SIGGRAPH '96), p.21-p.30.

[9] W. K. Tsao, J. J. Su, B. Y. Chen and M. Ouhyoung, "*Photo VR: A System of Rendering High Quality Images for Virtual Environments Using Sphere-like Polyhedral Environment Maps*", Proc. of RAMS, p.397 - p.404, Sinica, Taipei, R.O.C., 1996.

[10] Q. Zheng, "*A Computational Vision Approach to Image Registration*", IEEE Transactions on Image Processing, vol. 2, No. 3, p.311-p.326.