IMAGE-BASED VIRTUAL NAVIGATION SYSTEM FOR ART CAVES

Dongming Lu and Yunhe Pan
Zhejiang University, P R China

ABSTRACT

Dunhuang Art Cave (DAC), one of the most famous cultural heritage sites in the world, is confronted with serious natural efflorescence. How to virtually rebuild the caves and support their investigation and exploitation with computer graphics (CG) and Virtual Reality (VR) technologies is an urgent and important project. In this paper we discuss in detail the essential technical issues in our image-based virtual DAC navigation system. For DAC modeling of the cave architecture we employ 3D surface modeling, texture and 2D image-based modeling, murals, and painted statues. For DAC rendering we use the *level of detail* (LOD) method for murals and image-based rendering for painted statues. For DAC shadow generation we investigate a simple 3D model-based solution. Finally, this paper will show the general system architecture as well as demos of virtual navigation.

1. Introduction

Dunhuang Art Cave (DAC) is one of the most famous cultural heritage sites in the world. Its main components are its architecture, murals, and painted statues, all of which possess great artistic and cultural value. Therefore DAC was ranked as a national key culture relic by the State Council of the P. R. China in 1961, and as a world history and cultural heritage relic by UESCO in 1987.

Located in the desert, DAC is confronted with serious natural efflorescence and dangers. The problem of investigating and exploiting the cultural relics is a very complicated and urgent task. Our project [1][2] is just to do research and develop a virtual navigation system as an effective tool for DAC research. In this paper we shall discuss in detail the key technical issues, namely image-based modeling, rendering, and shadow generation. In the last section we shall outline the general system architecture and show demos of DAC virtual navigation.

2. Image Based DAC Modeling

Based on different construction architectures, we can divide the caves that make up DAC into five main types: Nirvana Cave, Central Pole Cave, Big Buddha Cave, Back Screen Cave, and the rest. Figure 2.1 illustrates.

Figure 2.1 Classification of Caves: (1) Nirvana Cave;
(2) Central Pole Cave; (3) Back Screen Cave;
(4) Big Buddha Cave
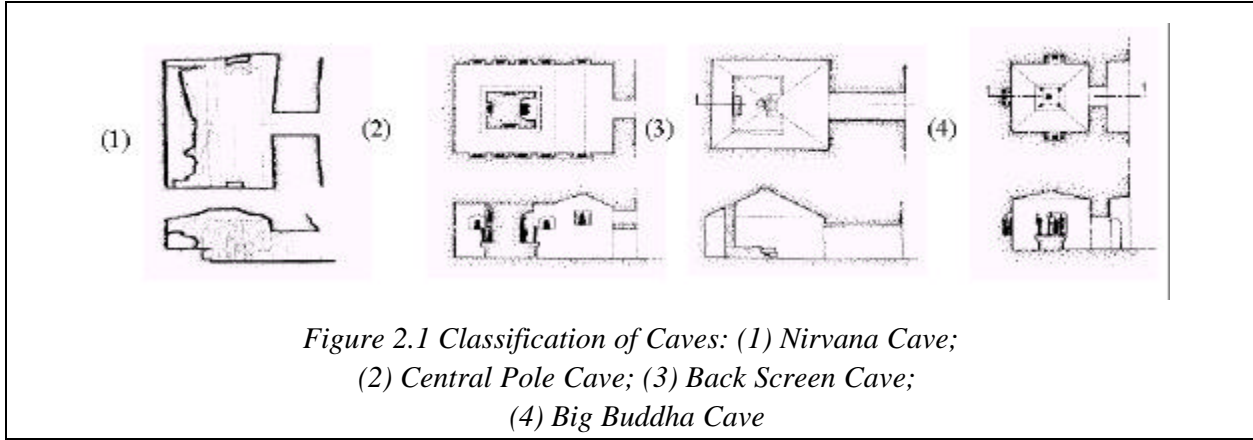
Shown in Figure 2.2 is a typical cave construction architecture composed of rooms, corridor, pole/alcove, roof, and painted statues. Most surfaces are regular and covered by murals. Exceptions are certain complex Buddha niches and painted statues. Therefore the three most important components of DAC are architecture, murals, and painted statues. We use 3D surface modeling for the architecture, texture images for the murals, and image-based modeling for the painted statues. In the following subsections we discuss the main pre-processing techniques for mural texture images and painted statue images.

Figure 2.2 The platform and side view of a sample cove
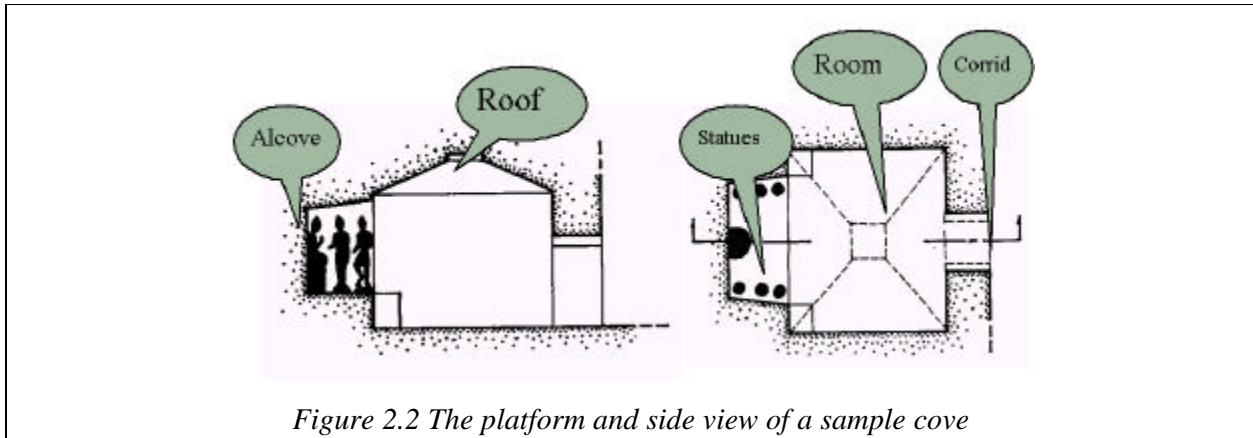
2.1 MOSAIC OF MURAL IMAGES

To obtain high resolution, one mural image on each wall has to be generated from several overlapped photo images. Therefore we need to solve the problem of creating a mosaic. For a planar mosaic, the images from one plane are merged via a 3x3-transform matrix:

$$\begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} = \begin{bmatrix} m_{00} & m_{01} & m_{02} \\ m_{10} & m_{11} & m_{12} \\ m_{20} & m_{21} & m_{22} \end{bmatrix} \bullet \begin{bmatrix} x \\ y \\ z \end{bmatrix} = M_{2d} \cdot u' \qquad \cdots \qquad (0)$$

The Graig G. Method [3] is adopted to get the matrix automatically. The basic principle is first to

compute the square sum of the pixel intensity differences between two images, and then choose the smallest one. The formulas are as follows:

$$E = \sum_i [I'(x_i', y_i') - I(x_i, y_i)]^2 = \sum_i e_i^2 \qquad \ldots \quad (1)$$

$$x_i' = \frac{m_0 x_i + m_1 y_i + m_2}{m_6 x_i + m_7 y_i + 1} \qquad \ldots \quad (2\text{-}1)$$

$$y_i' = \frac{m_3 x_i + m_4 y_i + m_5}{m_6 x_i + m_7 y_i + 1} \qquad \ldots \quad (2\text{-}2)$$

$$E = \sum_i [I'(x_i', y_i') - I(x_i, y_i)]^2 = \sum_i e_i^2$$

Due to the wide variation in illumination, image distance, and overlapping area, some images have gradient distortion and magnitude differences. Through experiments we found that if the images have more overlapping area, we can get a better effect. Furthermore, the changes of image distortion, rotation, and chromatism have a great influence on the characteristics of the mosaic.

We found some adverse effects from experiments. First, the method has some limitations because it merges the images through a matrix that is correlated with the difference of intensity. Some matrices will be abnormal, making the square sum of the pixel intensity differences smaller (maybe the smallest), showing that it is not a suitable mosaic matrix, especially for images with less color contrast. Fewer overlapping areas will also increase the possibility of errors. Second, because of their age, Dunhuang murals often have fading problems, reducing the color contrast of the murals and making the creation of mosaics more difficult.
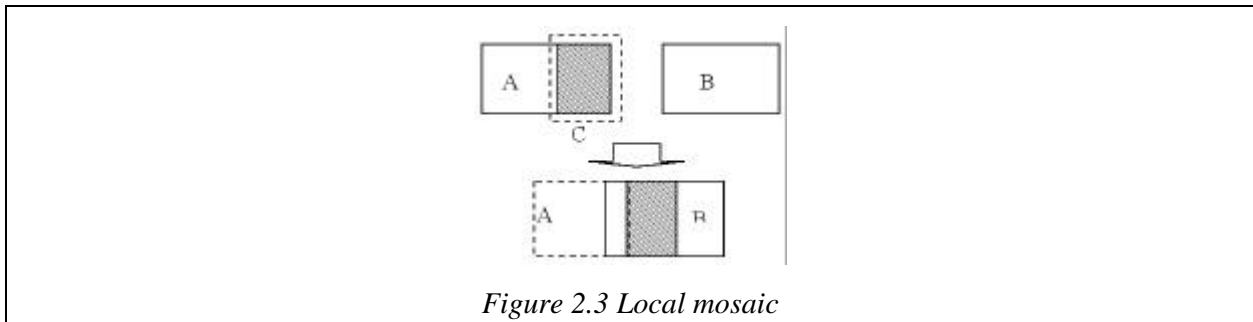

Figure 2.3 Local mosaic

If we make the overlapping area between images larger, the error will be smaller. Thus we use a *local mosaic* strategy. Here is the basic principle. Suppose the shadow of image A is the overlapping area between image A and image B. First we merge parts of A and C.   Suppose the position of C in A is (x, y). Then we compute the transform matrix M using C and B.   As C is part of A, it follows that C and A have a homologous transform used to create the mosaic between A and B. The experimental results are shown in figure 2.4.

Figure 2.4 Examples of mural mosaics
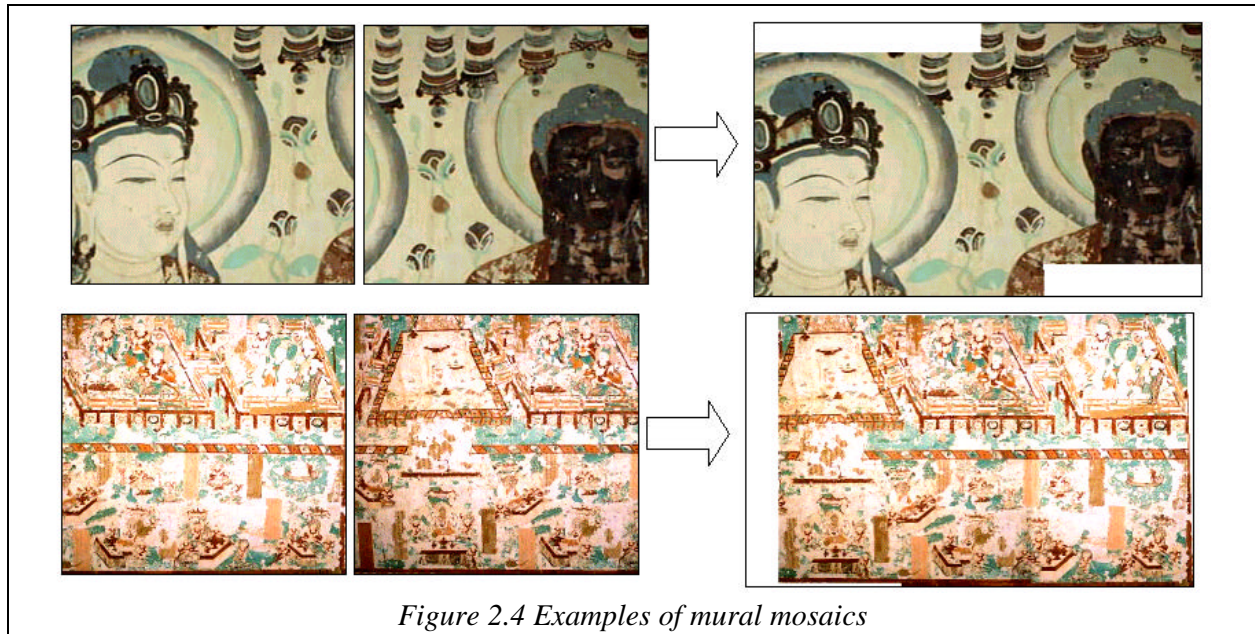
2.2 IMAGE MORPHING
FOR PAINTED STATUES

Due to their complex appearances, it is very difficult to make solid modeling of painted statues. *Image-Based Rendering* (IBR) methods have emerged using 2D information to represent 3D complex scenes[4]. Now there are many IBR methods. Based on different preconditions of images and process modes, we divide the IBR techniques into four main types: methods based on panoramic view[5], methods based on morphing[6], methods based on depth image information[7], and methods based on light field information[8]. Each method has unique features. Most of them are inefficient. One exception is the method based on panoramic views.

We use the IBR method to display the statues of DAC. Due to restrictions in location (i.e. the distances between statues are very short), it is impossible to take the photos of statues from all different viewpoints. Therefore, we integrate the field morphing method[9] to solve the problem.

Source image 1   Synthesized Image   Source Image 2

Figure 2.5 Image Morphing Sample
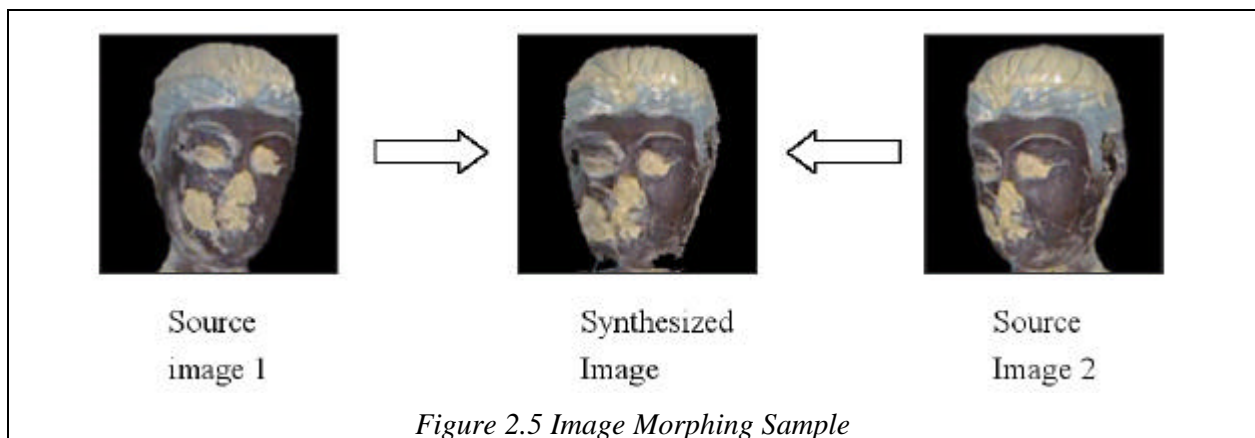
The field morphing method is based on feature lines in the images. First we consider the condition of a single-feature line. In Figure 2.6 we can define vector lines *AB* and *A'B'*, respectively, in image *I* and *I'*, which specify the corresponding relationships of pixels in the two images. Then we take

$$u = \frac{(\vec{X} - \vec{A}) \cdot (\vec{B} - \vec{A})}{\left\| \vec{B} - \vec{A} \right\|^2} \qquad (6)$$

$$v = \frac{(\vec{X} - \vec{A}) \bullet Perpendicular(\vec{B} - \vec{A})}{\left\| \vec{B} - \vec{A} \right\|} \qquad (7)$$

$$X' = A' + u \cdot (\vec{B'} - \vec{A'}) + \frac{v \cdot Perpendicular(\vec{B'} - \vec{A'})}{\left\| \vec{B'} - \vec{A'} \right\|} \quad (8)$$

where *Perpendicular* represents the vector line that is perpendicular to *A'B'* and has the same length as *A'B'*; *v* is the distance from one point *X* (or *X'*) in the image *I* (or *I'*) to the vector line *AB* (or *A'B'*); and *u* is used to indicate how the vector line *AB* (or *A'B'*) is divided by the perpendicular line. 0 u   1, indicates that the point of intersection lies between *A* and *B* (or *A'* and *B'*); while u < 0 or u > 1, indicates that it lies outside of *AB(A'B')*.
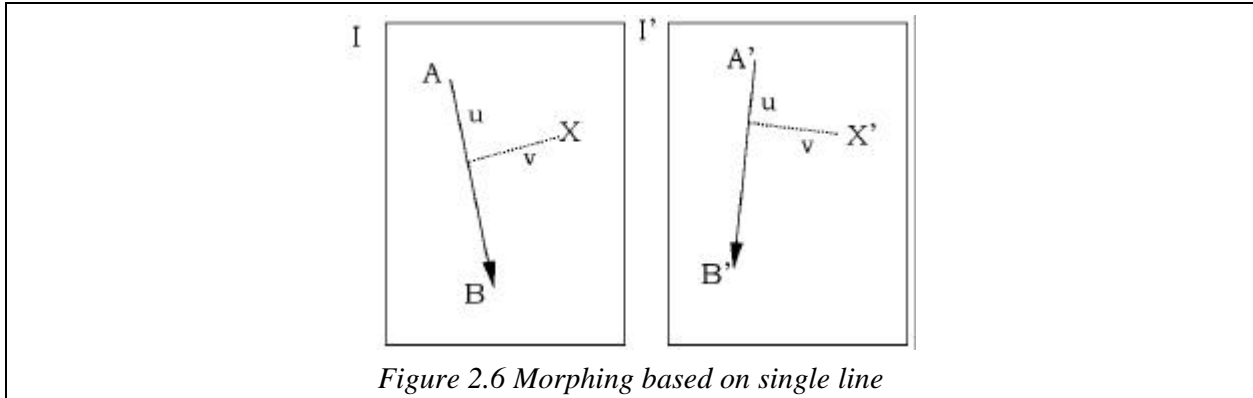

Figure 2.6 Morphing based on single line

Next we consider multiple feature lines such as shown in Figure 2.7. For an arbitrary point *P* in the source image *I* we first compute the corresponding points P_1', P_2', ... , P_n' in the destination image *I'* for all feature lines, respectively, and then find the vector difference D_i between *P* and P_i' in each case. By adding the average sum of these vector differences with weights and the initial *P*,   we can get the final point P'. The weight *w* is varies inversely with the distance between *P* and the feature line. Then we take

$$w = (\frac{length^{p}}{a + dist})^{b} \qquad (9)$$

$$P' = P + \frac{\sum\limits_{i=1}^{N} w_{i} \cdot (\overrightarrow{P_{i}'} - \overrightarrow{P})}{\sum\limits_{i=1}^{N} w_{i}} \qquad (10)$$

where *length* is the length of the feature line, *dist* is the distance from the point to the feature line, and *a, b, p* are constants used to adjust the influence of the feature line.

Figure 2.7 Morphing based on multiple lines
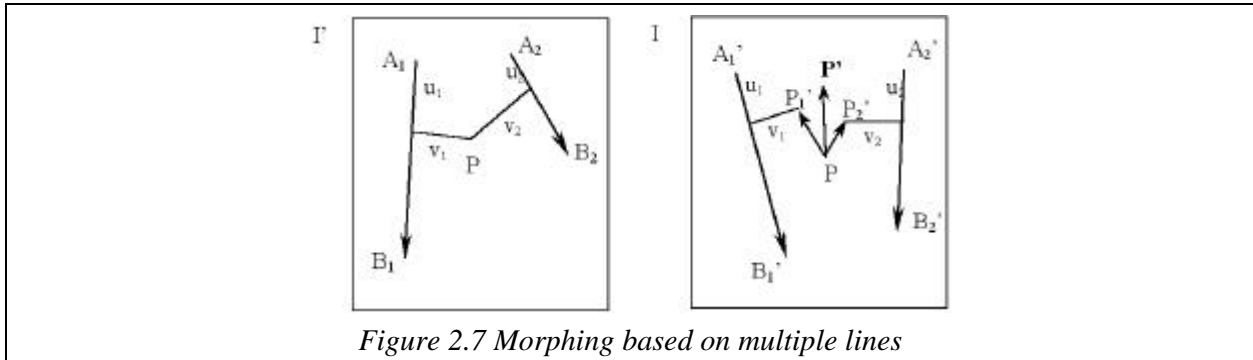
Using the above techniques, we have succeeded in modeling of 3D painted statues of DAC. Figure 2.8 shows the morphing result of the head part of the statue in cave No. 85. However, the remaining problem is that many minor facial details have insufficient detail, such as in the Buddha niche structure. Because of the limitations of image information, we cannot guarantee the precise expression when applying 2D images to model the 3D statues. We believe that 3D modeling can achieve better results, but a great deal of artificial manipulation is needed. Taking advantage of image information can solve the modeling problem, but some of the 3D features will be lost.

Feature line

Figure 2.8 Morphing based on feature lines
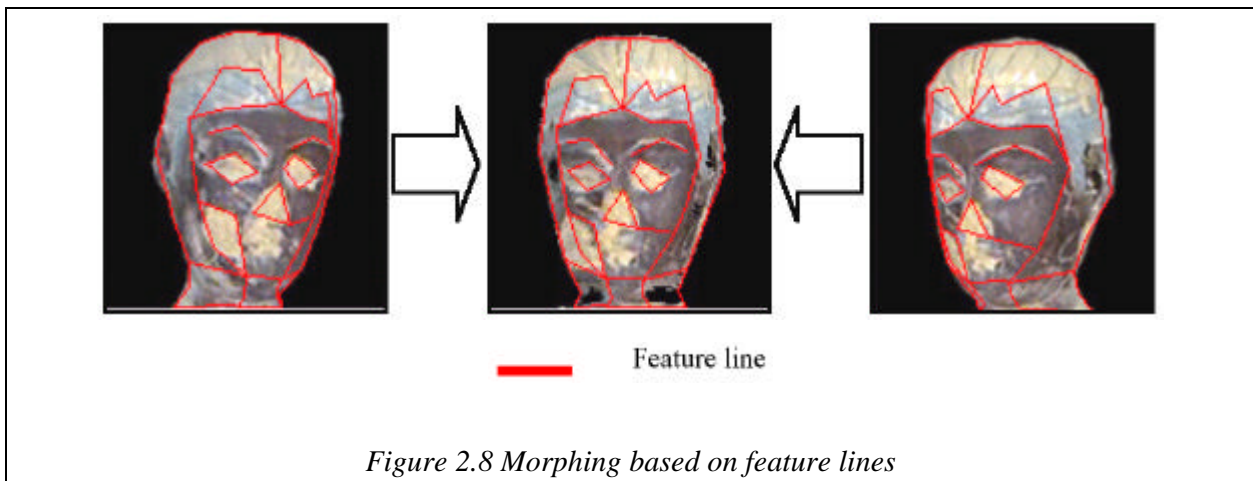
3. Image based DAC Rendering

Scene rendering is one of the most important problems in virtual navigation systems because it will directly affect navigating performance, such as the refresh speed and the perception of three dimensions. Our system uses three main technologies. The first is mural rendering based on LOD, the second is painted statue rendering based on 2D images, and the third is shadow generation based on a simple 3D model.

3.1 MURAL SCENE RENDERING

In order to get a better rendering quality in navigation, we must use higher resolution textures. Certainly they reduce rendering efficiency, so we adopt the multi-resolution method to represent textures. We design multiple layer models of murals: one is the LOD model of murals; another is a story model of murals. During rendering, the system will choose the proper cave model architecture according to eye location. The cave story model, which includes text description about murals, can be used for the efficient mural retrieval.
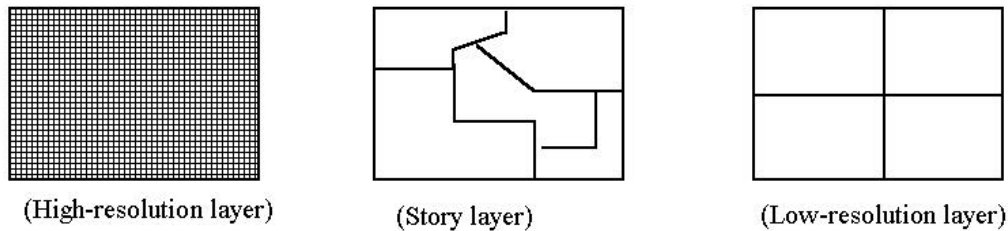
(High-resolution layer)   (Story layer)   (Low-resolution layer)

Figure 3.1 Multiple layers model

3.2 PAINTED STATUE RENDERING

In the previous section we discussed the modeling of painted statues based on 2D images. In rendering we just use 2D images to render 3D painted statues. First we need to build a transparent template for supporting image textures, such as a rectangle. During navigation this template faces the user all the time and has image texture changing according to user-eye location. The mechanism of choosing photos is described in Figure 3.2.

Figure 3.2 Painted statue rendering based on images
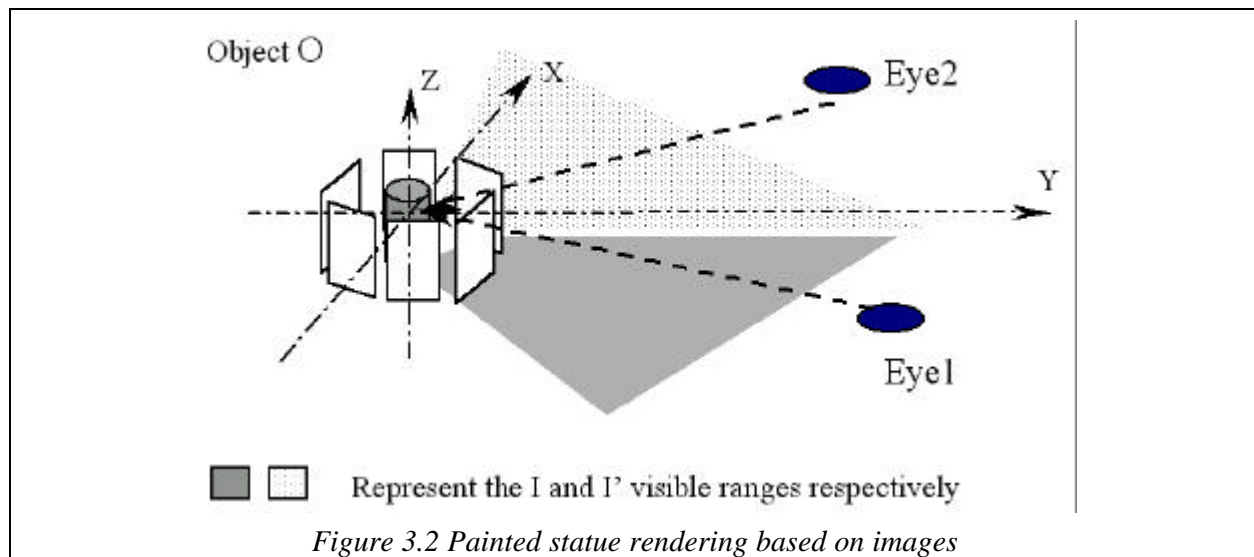
Suppose there is a set of 360-degree viewpoint photos (including those got by morphing) surrounding one painted statue object O. We assume there are N pieces of photos totally, labeled *1, 2, 3,..., N*. The interval angles between two contiguous photos are all the same so $\theta = 360 / N$. The coordinates of the object O are (O_x, O_y, O_z), and eye location is (V_x, V_y, V_z).   Then Tangent ($i\theta$) = ($V_x - O_x$) / ($V_y - O_y$).   If $\theta < 0$; $\theta = 360 + \theta$. It's easy to extend this solution to 180-degrees in the vertical direction as well using both horizontal and vertical photos. If enough photos are obtained, we can support a continuous movement in the virtual environment.

3.3 SHADOW GENERATION
BASED ON A SIMPLE MODEL

The IBR technique can simplify the modeling procedure and have higher rendering speed without significantly reducing the reality of 3D painted statues. However, the existing shadow generation techniques depend on 3D geometric architecture and cannot work based solely on a 2D model[10]. In order to generate reasonable painted statue shadows without losing the advantages of IBR we augment the image-based model with a simple 3D model. In the augmented model the images are used to render the scene of the painted statue while the simple 3D model is used to project its shadow onto the scene surrounding the statue under some specific set of light sources. The idea is illustrated in Figure 3.3.
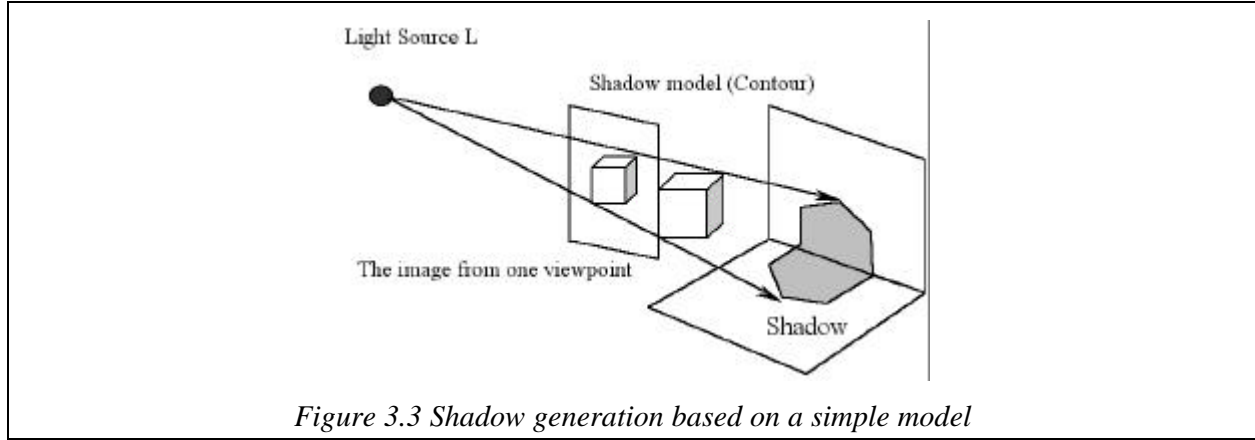
Figure 3.3 Shadow generation based on a simple model

Under the condition of a single light source, the shadow-generation algorithm is as follows:

(1) According to the location of light source, we first find the surfaces of the simple 3D model facing the light source, tagged as F_i (i=1, n). The implementation method is as follows:

Suppose the surface equation is $Ax + By + Cz + D = 0$, and the location of the light source is $L(x_l, y_l, z_l)$.

If the following inequality is valid, then the light source point is in the outside space of the surface.

$$Ax_l + By_l + Cz_l + D > 0$$

When we know the coordinates of edge points, such as $p_1(x_1, y_1, z_1)$, $p_2(x_2, y_2, z_2)$, ... , $p_n(x_n, y_n, z_n)$, then we can compute the parameters of the plane equation by

$$A = \sum_{i=1}^{n} (y_i - y_j)(z_i + z_j)$$

$$B = \sum_{i=1}^{n} (z_i - z_j)(x_i + x_j)$$

$$C = \sum_{i=1}^{n} (x_i - x_j)(y_i + y_j)$$

where $j = (i+1)$ mod n, and

$$D = -(Ax_k + By_k + Cz_k), \qquad 1 \le k \le n$$

(2) For each surface F_I, we generate its project shadow A_i on the surrounding scene. The computing method is as follows:

Suppose the project plane is XoY and the light source coordinate is $L(0,0,z_l)$. For each point x, y, z in F_I we can compute its projecting point as follows

$$\begin{bmatrix} x_p & y_p & z_p \end{bmatrix} = \begin{bmatrix} x & y & z & 1 \end{bmatrix} \begin{bmatrix} \dfrac{-z_l}{z-z_l} & 0 & 0 & 0 \\ 0 & \dfrac{-z_l}{z-z_l} & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

(3) By merging all projecting shadow A_I, we generate the entire project shadow A.

(4) According to the intensity of the light source, we regenerate the shadow texture by synthesizing the project shadow A and corresponding scene texture.

Obviously, when multiple light sources exist we can generate the overall shadow by repeating the above process several times. The experimental results are shown in Figure 3.4.

(A) Fixed viewpoint with changing light

(B) Changing viewpoint with fixed light
Figure 3.4 Shadows based on a simple 3D model

(C) Demo of a dynamic shadow
Figure 3.4 (continued)

4. Conclusion

Based on the above technique, we have implemented a prototype DAC virtual navigation system, the general architecture of which is illustrated in Figure 4.1. From the experimental results in figure 4.2, we found that image-based techniques are quite effective, although there are still some limitations.
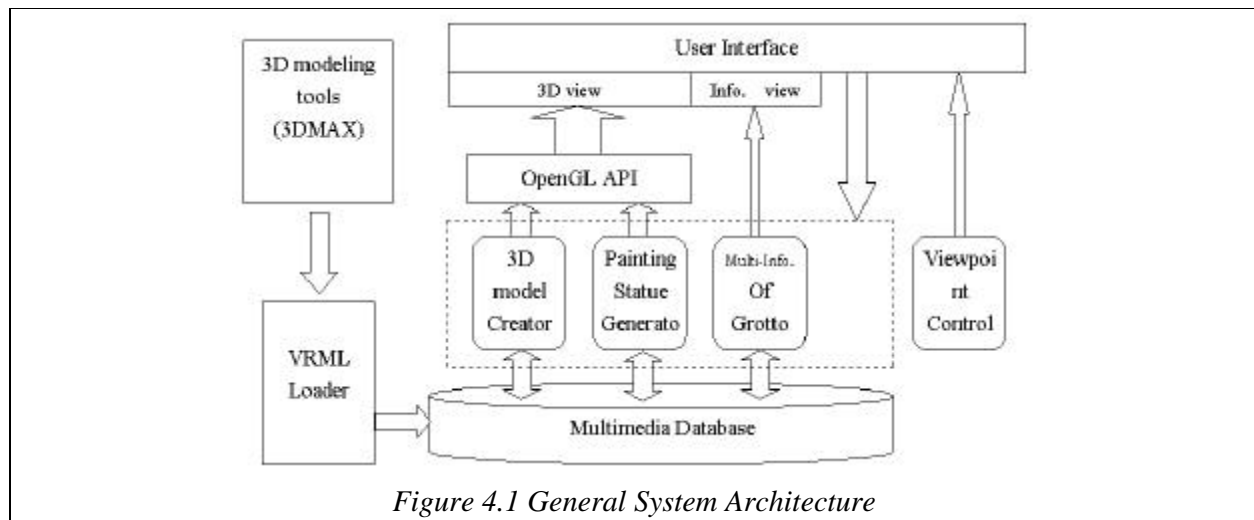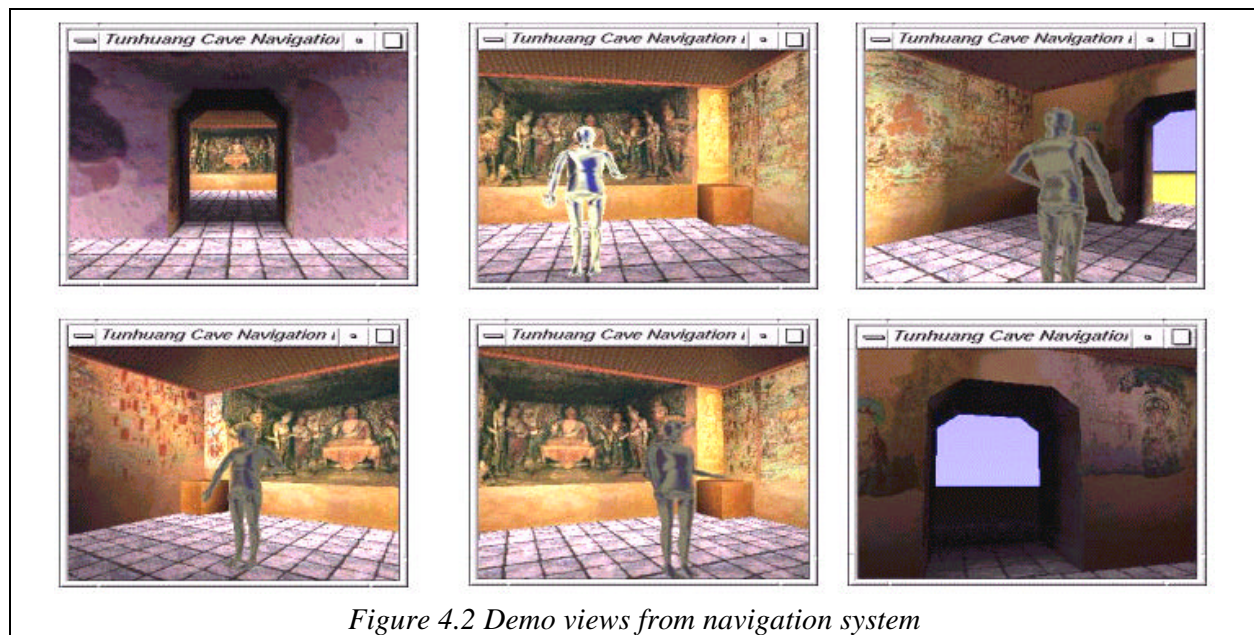

Figure 4.1 General System Architecture

Figure 4.2 Demo views from navigation system

We will investigate the two main tasks in the future. The first is to improve rendering efficiency. In DAC virtual navigating, the main bottleneck is the requirement for rendering with high-resolution textures and plenty of statues. Integrating the 3D models and 2D images to represent the statues will improve the rendering speed.

The second is to enhance 3D display**.** Shadow is always the key problem, especially for the object rendered based on IBR technology. We have studied only the hard shadow until now, but in the real world, most of the shadows are soft ones. This will be our future focus.

5. Acknowledgments

REFERENCES

[1] Z.Gong, D.M. Lu and Y.H. Pan, *Dunhuang Art Cave Presentation and Preservation based on VE,* Proc. of MMVS'98, Nov. 1998, GiFu, Japan, pp612-618.

[2] Y.H. Pan, *Dunhuang Art Cave Virtual Rebuilding and Navigation*, Proc. of CAADRIS'99, May, 1999, Shanghai, China, pp1-20.

[3] Craig G. and Arnon N., *Image Based Rendering: Mosaicing*, Technology Report.

[4] Brian C. and Marc L, *A Volumetric Method for Building Complex Models from Range Images*, SIGGRAPH'96, 1996, pp303-312.

[5] Chen S.E., *QuickTime VR – An Image-based Approach to Virtual Environment Navigation*, Computer Graphics Proceedings, 1995, pp29-38.

[6] Seitz S.M. and Dyer C.R., *View Morphing*, SIGGRAPH'96*,* 1996.

[7] Stephane L. And Oliver F., *3-D Scene Representation as a Collection of Images and Fundamental Matrices*, INRIA Report, 1994.

[8] Marc L. and Pat H., *Light Field Rendering*, SIGGRAPH'96, 1996, pp31-42.

[9] Thaddeus B. and Shawn N., *Feature-Based Image Metamorphosis*, SIGGRAPHI'92, 1992, pp35-42.

[10] Manuel M and Gary B., *Dynamic Shading in Image-based Rendering*, 1997.

BIOGRAPHIES

Dongming Lu is an associate professor of the Department of Computer Science and Engineering, Zhejiang University. He received MS and Ph.D. degrees from the Department of Computer Science and Engineering, Zhejiang University. His current work concentrates on Multimedia, VR and CSCW. He has published more than 20 papers.

Contact information:

Dr. Dongming Lu
Artificial Institute
Zhejiang University, Hangzhou, China
Phone: +86-571-7951884
Fax: +86-571-7951670
Email: mailto:ldm@appmail.zju.edu.cn

Yunhe Pan is a professor in the Department of Computer Science and Engineering, Zhejiang University. He is the president of Zhejiang University and the director of the Artificial Intelligence Institute of Zhejiang University. He received an MS degree from the Department of Computer Science and Engineering of Zhejiang University. Since the 1980s his work has concentrated on intelligent CAD, computer art, multimedia, and VR. He has published more than 200 papers.

Contact information:

Prof. Yunhe Pan
President Office
Zhejiang University, Hangzhou, China
Phone: +86-571-7951109
Fax: +86-571-7951358
Email: mailto:Panyh@sun.zju.edu.cn