Generation of Binocular Object Movies from Monocular Object Movies

Ying-Ruei Chen[3], Wan-Yen Lo[3], Yu-Pao Tsai[1,4], and Yi-Ping Hung[1,2,3]

[1]Institute of Information Science, Academia Sinica, Taipei, Taiwan
[2]Institute of Networking and Multimedia, National Taiwan Univ., Taipei, Taiwan
[3]Dept. of Computer Science and Information Engineering, National Taiwan Univ., Taipei, Taiwan
[4]Dept. of Computer and Information Science, National Chiao Tung Univ., Hsinchu, Taiwan
Email: hung@csie.ntu.edu.tw

ABSTRACT

Object movie (OM) is a popular technique for producing interactive 3D artifacts because of its simplicity in production and it photo-realistic ability to present the artifacts. At the same time, many stereoscopic vision techniques are developed for a variety of applications. However, the traditional approach for generating binocular object movies require duplicate effort compared with monocular ones both in the process of acquisition and image processing. Therefore, we propose a framework to generate stereo OMs from monocular ones with the help of an automatically constructed 3D model from the monocular OM. Here, a new representation of the 3D model, named billboard clusters, is proposed for efficient generating binocular views. In order to obtain better results, a novel approach to extract view-independent texture is developed in this work. Besides, billboard clusters can be used to compress the storage capacity of OMs, and to perform relighting so that the binocular OMs can be well augmented into virtual environments with different lighting conditions. This paper describes the methods in detail and reports on its wide applications.

1. INTRODUCTION

To date, image-based techniques for modeling and rendering high quality and photo-realistic 3D objects have become a popular research topic. Having the advantage of being photo-realistic, object movie is especially suitable for delicate artifacts and thus has been widely applied to many areas, e.g., E-Commerce, digital Archive, Digital Museum, etc. An object movie is a set of images taken from different perspectives around a 3D object, as shown in Figure. 1; when the images are played sequentially, the object seems to be rotated around itself. This technique was first proposed in Apple QuickTime VR.[1] When captured, each image is associated with distinctive pan and tilt angles of the viewing direction, and thus some particular image can be chosen and shown on screen according to mouse motion of the user. In this way, users can interactively rotate the virtual artifacts arbitrarily and enjoy what he/she cannot see or feel in general.

Providing simulated environment that is hard to experience in real world, virtual reality systems are more and more attracting and are coming into people's lives. For constructing a photo-realistic virtual environment, we proposed a pure imageVbased approach, named augmented panorama,[2]which does not have to reconstruct the geometric models of the 3D objects, to augment a panorama with object movies in a visually 3D-consistent way. Fig 2 shows an example of augmented

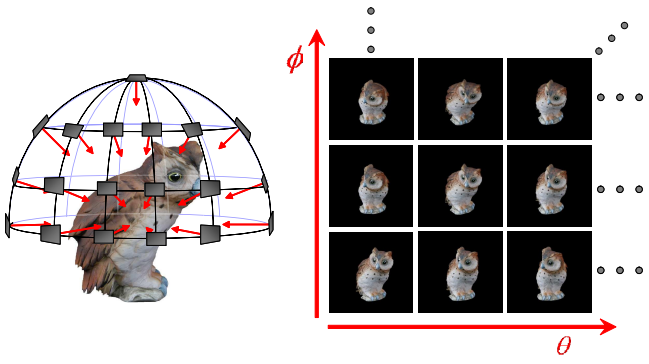


Figure 1. The 3D configuration and corresponding images of object movies.

Stereoscopic Displays and Virtual Reality Systems XIV, edited by Andrew J. Woods, Neil A. Dodgson, John O. Merritt, Mark T. Bolas, Ian E. McDowall, Proc. of SPIE-IS&T Electronic Imaging, SPIE Vol. 6490, 64901G, © 2007 SPIE-IS&T · 0277-786X/07/$18

(a) An object movie is rotated in the 3D scene.

(b) The result after zooming.

Figure 2. Results of augmentation of panorama with object movies.

Figure 3. A stereoscopic kiosk, a practical application of object movies. Visitors can browse the digital museum in the upper monitor, and watch the stereo OMs of artifacts in the lower monitor.

panorama. Most museums are taking digital exhibition as an important feature now. However, as the device and technology for virtual reality are made progressing, having stereoscopic visual experience of the virtual artifacts are made possible. The binocular OM is thus proposed as an improvement of traditional monocular OM to keep stereoscopic effects. By placing one more camera by the side of the original one in the acquisition process, binocular OM simulates how human eyes work. Hence, two sets of OMs, one for left eye's view and the other for right eye's view, are composed for producing a binocular OM to represent an object. In this way, users can observe a stereo object with a binocular OM via a stereoscopic display system. Based on augmented panorama, we developed a stereoscopic kiosk[3] for virtual museum, which consists of two display devices: one is a touch screen and the other is a stereoscopic display, as shown in Fig 3. In the kiosk system, artifacts are presented as object movies, and can be integrated with both image-based panoramas and geometry-based scenes for constructing virtual museum. Through the touch screen, the users can arbitrarily navigate in the virtual museum, select artifacts, and interactively view the detail in-formation of the selected artifacts. Once an artifact on the touch screen is selected, the stereoscopic object movie of the selected artifact will be synchronously shown in the stereoscopic display. The kiosk has been used in several virtual museums including National Palace Museum and National Museum of History. The kiosk system provides the user a better experience for browsing the 3D object through the stereoscopic display, however, the exhibition environment and the stereo OMs are displayed in separated devices. Therefore, we proposed a method, augmented stereo panorama,[4] to integrate them together so that the user can navigate the virtual exhibition and browse the 3D objects using a stereoscopic display. Fig 2 shows the result of a stereo virtual museum with a stereo OM augmented.

As mentioned above, binocular OM is getting more and more important in many area. Nevertheless, existing methods for acquiring a binocular OM takes duplicate efforts and time, that is, two sets of monocular OMs must be acquired separately to compose a binocular one–acquiring binocular would become time-consuming and the stereo baseline of the produced binocular OM is fixed. Furthermore, there are already many existing monocular OMs, which were generated for

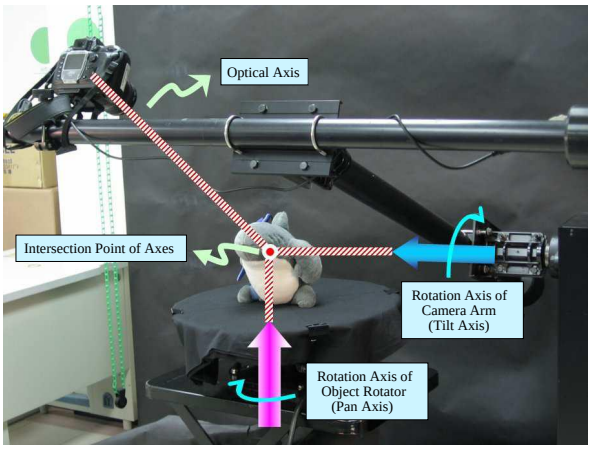


Figure 4. Motorized object rig–AutoQTVR

archived artifacts, with the traditional approaches, the whole process should be repeated again. This may be unacceptable for many OM holders. For this reason, this paper presents a framework to generate binocular OMs from monocular ones, which reduces the efforts of image acquisition and image processing, and can arbitrarily produce the binocular OM with the desired baseline based on application scenarios.

In this framework, we take a set of calibrated monocular OM images as input, and generate novel views for stereo vision based on the sparse set of images and the automatically-reconstructed 3D model. We can generate the stereo OM for the other eye with the information from the 3D model instead of acquiring another OM with a camera. Thus, our framework reduce both acquiring time and storage of stereo OM. Our framework is composed of several steps. We first calibrate the object rig shown in Fig. 4 so as to do image acquisition. After acquiring a set of images around the target object, we then develop a system to remove the background of the images with few user intervention and to obtain the 3D geometry. Finally, the last process in this framework is to generate novel views for binocular vision. To do this efficiently, we first exploit some intersected, partially transparent billboards to simplify the constructed geometry, and extracted the albedos from each image so that the remained view-independent images can be directly texture-mapped onto the geometry. After relight the new representation of the object according to the original illumination condition, the images that should be seen by the other eye can be generated effectively. The remainder of this paper is organized as follows. The procedure of our framework is described in more detail in Section 2. Section 3 explains how to generate the billboard clusters, and Section 4 describes how to extract view-independent textures. the contents of our virtual museum are produced. Section 5 describes how to generate novel views and shows some results. Section 6 gives a conclusion.

2. OVERVIEW OF OUR FRAMEWORK

In this work, we use the motorized object rig, AutoQTVR, as shown in Fig.4, to capture OM. In OM acquisition, the center of the object should be placed at the crossing point of the two rotation axes and the optical axis of the camera, so that the resulted OM can rotate smoothly. However, since the optical axis of the camera is invisible, aligning these three axes is inherently a difficult problem. In our framework, we first adopt the method proposed by Huang et al.[5] to calibrate the object rig and to acquire high quality OMs that rotate smoothly. We use the camera mounted on AutoQTVR to capture some feature points, whose 3D positions are known beforehand. The 2D and 3D of the feature points are used to estimate intrinsic and extrinsic camera parameters. This information is used to reconstruct the kinematic model of the rig. Then, we apply a simple and practical model to formulate the relation among the three axes. A tool shows the virtual axes, which helps us adjust the motorized object rig iteratively.

Second, to obtain better rendering result, OM segmentation is done. Although blue screen and green screen methods are effective for background removal in movie making, a black screen is preferable during the OM acquisition process to prevent the object from reflecting the blue or green light. Black screen, however, often results in ambiguous shadowed regions that significantly raise difficulty in segmentation. Because there are always hundreds of images to be segmented, it will take quite a long time to remove the backgrounds manually. We then develop a system to do OM segmentation

Figure 5. Stitching result of a stereo panorama.

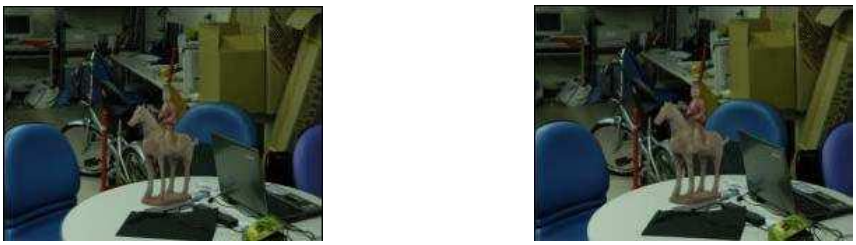


Figure 6. Result of the augmented panorama with a stereo OM (a) shows the rendered left view, and (b) shows the right view.

with only a small amount of user intervention, and have implemented it in our framework.[6] Given an OM with camera parameters calibrated from first step mentioned above our method starts from automatic initial segmentation. After user selected a subset of acceptably segmented images, the 3D reconstruction form these selected images is followed. For every image in the OM, we refine the segmentation by using the prior knowledge of the shape learned from the reconstructed 3D model.

Finally, we propose a new representation for 3D objects to generate binocular views. Although it is possible to create an extreme detailed 3D model manually for rendering novel views, such a detailed model is not only costly but also slow in rendering. Besides, some systems exploit feature correspondence among images for interpolating the novel views[7],[8] but unsatisfying artifacts may appear in the rendering results due to false feature correspondence. Hence, we developed our approach for efficiently generating fine binocular views for OMs, with the new representation of billboard clusters. *Billboard clusters* combines both advantages of image-based and model-based rendering. Décoret et al.[9] has proposed the idea of billboard clouds, which are collections of intersecting textured quadrilaterals that look like a real object from a distance. We adopt the basic concept of their original work, while reform billboard clouds as billboard clusters in order to overcome the problem of losing fine details in the process of extreme simplification. Our representation constitutes of two steps. One is billboard cluster generation, and the other is view-independent texture extraction. In billboard cluster generation, the reconstructed 3D model is first simplified with large billboards using our grouping algorithm, and then each billboard primitive is spread into layer imposters.[10] Second, lighting effects, shading and specular, are removed from original images, and only view-independent albedos are remained for texture mapping.

The remainder of this paper consists as follows. In section 3, we explain in detail the algorithm for generating billboard cluster representation from the constructed 3D geometry, which is in the form of point clouds. In section 4, we introduce our method for extracting textures that consist of only albedos from the acquired OM images. We then describe how to render a view-independent virtual object and how to generate images for binocular vision in Section **??**. Finally, we have applied our work on two existing applications. In Section 5, we discuss these results, and our conclusion is stated in Section 6.

3. BILLBOARD CLUSTER GENERATION

We propose a new representation for 3D object, called billboard clusters, which is efficient for generating novel views. As a hybrid of billboard cloud[9] and layer impostors,[10] this representation has merits of both: it is an extreme simplification,

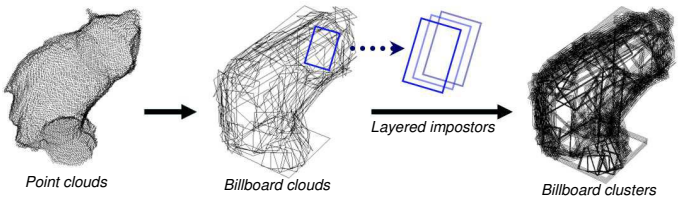


Figure 7. The basic concept of billboard clusters. We first find the primitive billboard clouds according to the input point clouds, then spread each billboard into multiple layers to form the billboard clusters representation.

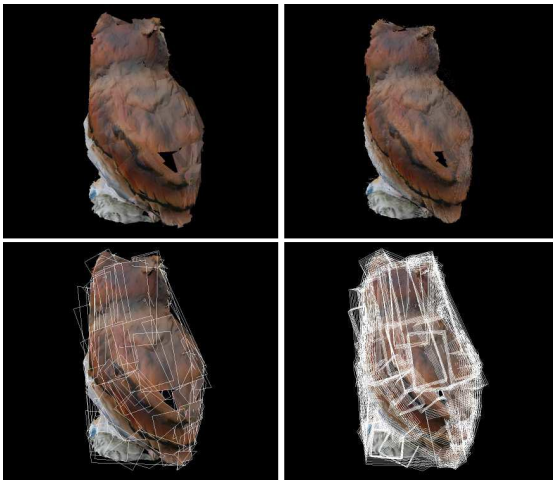


Figure 8. Rendering results of the pottery owl. The images in the lower part display the quadrilaterals used. (a) The result of billboard cloud. (b) The result of billboard clusters, with each billboard being spread into 7 quadrilaterals. Note the round shape of owl is better preserved in this representation, since the accuracy increases with the number of layers.

but can capture better the shape of the original model, even for curved surfaces. In the next subsections, we will discuss how to construct the billboard cloud, and how to use it as a primitive to generate a layered impostor representation. The basic concept of billboard clusters is illustrated in Figure 7.

3.1. Finding Primitive Billboard Cloud

In billboard representation, a 3D model is covered by billboards. Thus, selection of these billboards determines the quality of the result. To select a good set of planes that approximate the input point clouds, we first define the features of a well-fitting plane. First, it must preserve the original shape of the object to prevent loss in accuracy in the rendering result, that is, the orientation and position of the plane should approximate those of the surfaces of the object. Second, it should be as large as possible, since a large plane is more efficient than many overlapped small ones, no matter for storage or rendering speed. Based on these criteria, we then develop an error-based construction strategy. Our algorithm iterates the following steps until the representation costs are too high:

1. Do grouping in the input point clouds to find a set of groups.
2. For each group of points, find an optimal plane that minimizes the representation error.
3. If two planes have similar position and orientation, and their corresponding groups overlap much, merge the groups to recompute a new plane.
4. Construct a billboard from each plane, by projecting all the points in that group onto the plane and finding the minimal bounding rectangle.

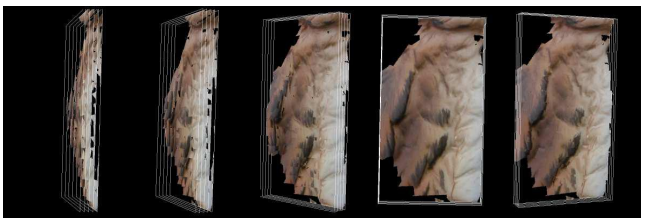


Figure 9. Different views of a layered impostor.

5. Remove all the representable points from the input data.

We first describe how grouping algorithm works. For each point P represented in no planes, construct a group G_P. For each member m in G_P, if any of the 26 neighbors of m has similar normals with P, group it into G_P. This is repeated until there are no new members grouped in G_P.

Second, to find an optimal plane $\mathbf{T}$ for a point group, we have to compute plane parameters in a way that minimize the total error measure. Point i can be well approximated by impostor plane j, if the distance between them is small and if their normals are parallel enough. Hence, we can formulate the error measure for plane $\mathbf{T}$ and 3D point $\mathbf{p}$ as following:

$$E(\mathbf{p}, \mathbf{T}) = Dist^2(\mathbf{p}, \mathbf{T}) + \alpha(1 - (\frac{n^T \cdot N}{\mid n \mid\mid \ N \mid})^2),$$

where n and N are the given normals of $\mathbf{p}$ and $\mathbf{T}$ respectively, and α denotes the relative importance of orientation and distance. The first term stands for distance between point and plane, and the second term stands for if their normals are close. Minimizing the total error measure $\sum_i E(\mathbf{p}_i, \mathbf{T})$ with a constraint that the plane normal is a unit vector, we can then find the optimal plane by computing the partial derivatives of the following equation according to all parameters respectively, and making all the derivatives equal to zero:[11]

$$f(N, d) = \sum_i E(\mathbf{p}_i, \mathbf{T}) - \lambda(N^T \cdot N - 1), \tag{1}$$

where N and d are parameters of plane $\mathbf{T}$, and λ is the Lagrange multiplier.

3.2. Layer Imposter Representation

In the representation of billboard cloud, the shape of the object seems to be compromised due to the nature of billboards, as demonstrated in Figure. 8(a).

Schaufler proposed the idea of Layer imposers[10] in 1998. Each layered impostor consists of a stack of texture-mapped, partially-transparent quadrilaterals. Using stacks of parallel quadrilaterals is to approximate the volume of the target object, and to well capture the shape of it. Figure. 9 shows an example of layered imposters. We then apply this concept to improve billboards. We first use the method stated in the previous section to construct the billboard cloud, then use it as a primitive to further create layer impostors, by means of shifting every billboard forward and backward according to the direction of its normal. Further, for layered impostors, the texture mapped on each layer is depth augmented, and therefore only those pixels having similar depth value are drawn onto each layer. As Figure. 8(b) demonstrates, with our new representation, the round shape of the pottery owl in the left side is better preserved.

To sum up, since billboard cloud provides extreme simplification and layer imposters increase only a little overhead, the final representation of our billboard cluster is very efficient. Some rendering results of billboard clusters are shown in Figure. 10.

Figure 10. Rendering result of the pottery owl with our billboard cluster representation.

4. VIEW-INDEPENDENT TEXTURE EXTRACTION

To achieve photo-realistic view-dependent rendering, the view-independent components must be first separated from the input image sequences and then be used for texture mapping. Otherwise, the varied refection effects will result in color-inconsistent rendering results, for the textures mapped on the billboards are warped from disparate input images. Furthermore, only if the reflection parameters of diffuse and specular components are estimated for recovering the view-independent reflectance images, can the virtual object be relighted realistically under different illumination conditions.

Here follows our proposed step for lighting compensation.

1. Separate the diffuse and specular reflection components, with the help of the input geometry.
2. Recover the reflectance images from the input sequence with diffuse and specular reflection parameters, estimated by using the maximum-a-posteriori (MAP) technique.

4.1. Reflection Model

A mechanism of reflection is described in terms of three reflection components, namely the diffuse lobe, the specular lobe, and the specular spike.[12]We omit the specular spike component due to uncommonly observation in many actual applications. Then the light reflected on the object can then be approximated as a linear combination of two reflection components: diffuse reflection component I_D and specular reflection component I_S[13,14,15]:

$$I = I_D + I_S. \tag{2}$$

The Lambertian and Torrance-Sparrow reflection model[16] are further used for modeling the diffuse and specular reflection in our analysis, i.e.,

$$I_D = K_D cos\theta_i, \tag{3}$$

and

$$I_S = K_S \frac{1}{cos\theta_r} e^{-\alpha^2/2\sigma^2}. \tag{4}$$

After substituting (3) and (4) for (2), we can get the following equation:

$$I = K_D cos\theta_i + K_S \frac{1}{cos\theta_r} e^{-\alpha^2/2\sigma^2}, \tag{5}$$

which can further be simplified as:

$$I = m_D K_D + m_S K_S, \tag{6}$$

where K_D and K_S are diffuse and specular color vectors respectively, and m_D and m_S are their corresponding reflection parameters. K_D and K_S are both color constants, consisting of RGB three channels, while all the factors relating with viewing direction, lighting direction and surface normals are all absorbed into parameters m_D and m_S. Therefore, in order to remove the shading and specular effects from the image sequences, and to obtain the view-independent reflectance images, every pixel should be restored to its diffuse color, K_D. We then use this simplified model in our analysis to find robust estimation of m_D and m_S and to recover K_D with the estimated parameters.

4.2. Separating Reflection Components

We first gather color samples of each voxel from all the views, except those that are invisible for the voxel. We then find the colors with maximum K_D and minimum K_S of every voxel in the collected samples with our priori knowledge of acquisition environment. We then use this colored model as our initial guess to estimate view-independent reflectance maps from the original image sequence. We formulate this problem in a Bayesian framework and solve it using the maximum-a-posteriori (MAP) technique.

We assume the environment lights are all white in our work. Then we express this problem as a maximization over a probability P with constrain that K_D, K_S, and I should be coplanar, that is,

$$(I \times K_S) \cdot K_D = 0. \tag{7}$$

Bayes's rules are then used to express the result as a maximization over a sum of log likelihood:

$$\begin{aligned}
&\underset{m_D,m_S,K_D}{\operatorname{argmax}} \ \textit{belief probability function} f_B \\
&= \underset{m_D,m_S,K_D}{\operatorname{argmax}} \ P(m_D, m_S, K_D|I) - \lambda \cdot ((I \times K_S) \cdot K_D) \\
&= \underset{m_D,m_S,K_D}{\operatorname{argmax}} \ P(I|m_D, m_S, K_D)P(m_D)P(m_S)P(K_D)/P(I) \\
&= \underset{m_D,m_S,K_D}{\operatorname{argmax}} \ L(I|m_D, m_S, K_D) + L(m_D) + L(m_S) + L(K_D) \\
&\qquad\qquad -\lambda \cdot ((I \times K_S) \cdot K_D),
\end{aligned} \tag{8}$$

where $L(\cdot)$ is the *log likelihood* $L(\cdot) = logP(\cdot)$, λ is the Lagrange-Multiplier, and the $P(I)$ term is dropped because it is a constant with respect to the optimization parameters. We will then state in the following how to model the log likelihoods $L(I|m_D, m_S, K_D)$, $L(m_D)$, $L(m_S)$, and $L(K_D)$ respectively.

We can model the first term by measuring the difference between the color observed and that predicted by the estimated m_D, m_S and K_D:

$$L(I|m_D, m_S, K_D) = \frac{-\|I - m_D K_D - m_S K_S\|^2}{2\sigma_I^2}. \tag{9}$$

This log likelihood models error in the measurement of I and corresponds to a Gaussian probability distribution centered at $\overline{I} = m_D K_D + m_S K_S$ with standard deviation σ_I.

We further use the spatial coherence to estimate the term $L(K_D)$. We first calculate the value of $\overline{K_D}$, which is a weighted mean of the known and previously estimated albedos within each pixel p's neighborhood N, and the contribution of each nearby pixel q in N neighborhood is weighted according to three factors. First, we use a spatial Gaussian falloff G_q to stress the contribution of nearby pixels over those that are further away. Second, the albedo of those having similar measured color with p are more trustworthy. Third, we also weight a neighbor pixel's contribution by its belief probability value, as defined in (8). As a result, the combined weight of a neighbor q for pixel p is

$$w_q = \frac{G_q \times f_B(q)}{(I_q - I_p)}, \tag{10}$$

and the weighted mean $\overline{K_D}$ and covariance matrix Σ_{K_D} are given by

$$\overline{K_D} = \frac{1}{W} \sum_{q \in N} w_q K_{D,q} \tag{11}$$

$$\Sigma_{K_D} = \frac{1}{W} \sum_{q \in N} w_q (K_{D,q} - \overline{K_D})(K_{D,q} - \overline{K_D})^T \tag{12}$$

where $W = \sum_{q \in N} w_q$. The likelihood for the albedo $L(K_D)$ can then be modeled as being derived from a Gaussian distribution with the weighted covariance matrix:

$$L(K_D) = -(K_D - \overline{K_D})^T \Sigma_{K_D}^{-1} (K_D - \overline{K_D})/2. \tag{13}$$

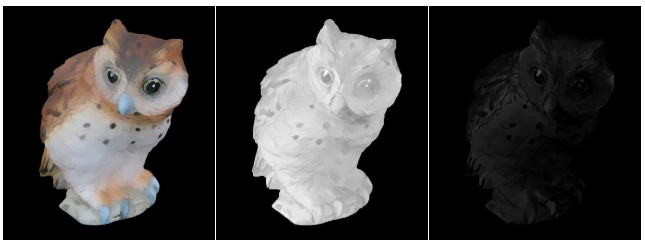


Figure 11. Results of lighting compensation after optimization with Bayesian's framework. (a) The reflectance map. (b) The shading map. (c) The specular map.

Furthermore, $L(m_D)$ and $L(m_S)$ can also be modeled in a similar way as follows:

$$L(m_D) = \frac{-(m_D - \overline{m_D})^2}{2\sigma_{m_D}^2} \tag{14}$$

$$L(m_S) = \frac{-(m_S - \overline{m_S})^2}{2\sigma_{m_S}^2}, \tag{15}$$

where

$$\overline{m_D} = \frac{1}{W}\sum_{q\in N} w_q m_{D,q} \tag{16}$$

$$\sigma_{m_D}^2 = \frac{1}{W}\sum_{q\in N} w_q (m_{D,q} - \overline{m_D})^2, \tag{17}$$

and

$$\overline{m_S} = \frac{1}{W}\sum_{q\in N} w_q m_{S,q} \tag{18}$$

$$\sigma_{m_S}^2 = \frac{1}{W}\sum_{q\in N} w_q (m_{S,q} - \overline{m_S})^2. \tag{19}$$

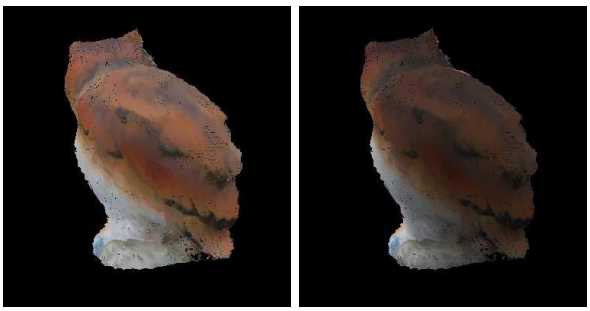
Finally, substituting (9), (13), (14), and (15) into the belief probability function defined in (8), we can get:

$$\begin{aligned} f_B = & \frac{-\|I - m_D K_D - m_S K_S\|^2}{2\sigma_I^2} \\ & - (K_D - \overline{K_D})^T \Sigma_{K_D}^{-1} (K_D - \overline{K_D})/2 \\ & - \frac{(m_D - \overline{m_D})^2}{2\sigma_{m_D}^2} - \frac{(m_S - \overline{m_S})^2}{2\sigma_{m_S}^2} \\ & - \lambda \cdot ((I \times K_S) \cdot K_D). \end{aligned} \tag{20}$$

The albedos and reflection parameters for all pixels can then be computed by optimizing (20).

5. EXPERIMENTAL RESULTS

Having removed the lighting effects from the OM images, we can obtain color-consistent rendering results, which are texture-mapped with only albedos of the object. The result is shown in Figure. 12(a), and as what is demonstrated, the resulted image of the pottery owl has no shading or specular effects. We then further adopt a coarse 3D triangle model to relight the rendering results of billboard clusters according to any user-specified illumination condition. This rough model is invisibly sits behind the billboard clusters and only help for relighting. The relighted result is shown in Figure 12(b).



(a) (b)

Figure 12. Rendering results before and after relighting. (a) The rendering result of billboard clusters. Each billboard is textured with view-independent reflectance images. (b) The relighted result corresponding to the pose in (a).

To generate binocular views, we first set the viewpoint of the billboard clusters to be identical with that of the monocular OM. And then we shift the viewpoint slightly along the horizontal line. This distance is usually about 5cm to 7cm, which is the average interval of general human eyes, while the real distance depends on where the object is placed in virtual environment. Hence, the final binocular OM consists of a set of these images that are generated in this way for each view in the original monocular OM, as demonstrated in Figure. 13.

Many virtual museums have already adopted OM technique for artifact exhibition. Chen et al.[4] have proposed a method for building an augmented stereo panoramas without any explicit 3D model of the scene. However, many existing OMs generated for the archived artifacts are monocular, and re-capture two sets of OM images to produce binocular OMs for all the artifacts are very time-consuming and undesirable. What is worse, even a re-generated binocular OM may not work perfectly in a stereo environment. Since OM is a view-dependent image-based rendering technique, viewpoints in binocular OMs are limited to where they were taken. This property restricts not only the center position of the stereo panorama, but also where the virtual objects should be placed. Our framework helps generate a new 3D representation for an OM, and this representation can gives images from arbitrary viewpoint and with arbitrary light source. This means the virtual objects can now be put anywhere in virtual environment, and can be illuminated freely without generating weird shading and shadow effects.

Furthermore, the storage size of the OMs can be highly reduced, since we can subtract the original images with the newly rendered results of billboard clusters in order to encode the residuals. Hence, only the billboard clusters, which is usually less then 3MB (including geometry and textures) for each OM, and the encoded residuals are required for transmission. Our work is thus very applicable for many practical applications that need Internet transmission for sharing. At the receiver side, the original OM images can be finally recovered with the use of the received billboard clusters and encoded residuals, and binocular views can then be further generated. In this way, eye-pleasing stereo OMs can be successfully obtained and displayed at the receiver side.

6. CONCLUSION

In this paper, we propose a framework for generating stereo OMs directly from monocular OMs, which includes camera calibration, background removal, 3D reconstruction, view-independent texture extraction and novel view generation. With our framework, a stereo OM can be generated, once the stereo baseline length is determined by the application. Compared with traditional approaches, which take two sets of OMs separately, our approach saves more than half of the processing time including acquisition and segmentation.

Besides generating stereo OMs, some additional merits of the proposed framework are summarized as follows. First, high-quality OMs can be acquired with the help of our calibration process. Second, the user intervention for removing background of OM is significantly reduced, which in our work is just a selection of subset images of the initial segmentation results. Finally, a new representation of 3D model is proposed. This technique has a very high compression ratio on OM with photo-realistic quality and real-time rendering speed. Even more, this technique can be used to relight the OM so that it can be well augmented into virtual environments with different lighting conditions.

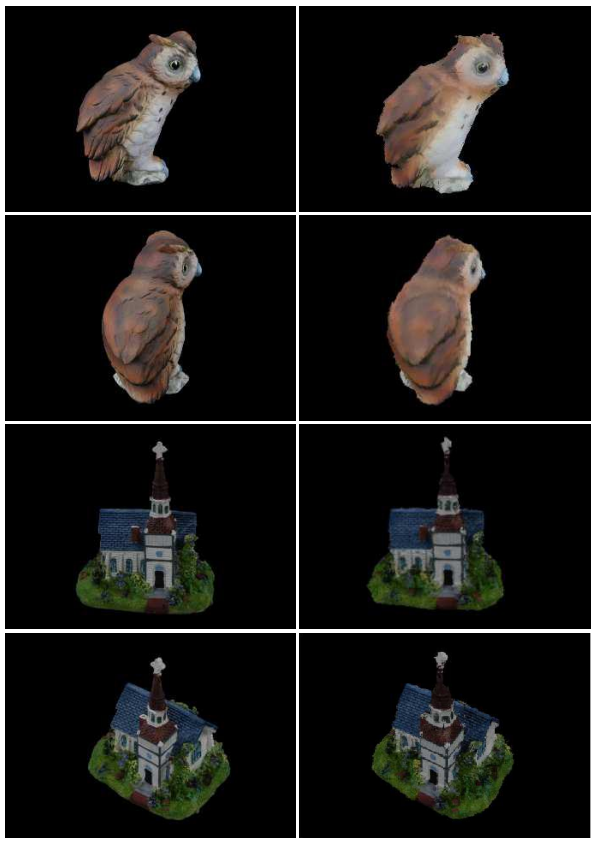


Figure 13. (a) The original OM images. (b) Our rendering results of binocular views.

Acknowledgments

This work was partially supported by National Digital Archives Program under the grants of NSC 94-2422-H-002-019.

REFERENCES

1. S. E. Chen, "Quicktime VR: an image-based approach to virtual environment navigation," in *SIGGRAPH*, 1995, pp. 29–38.
2. Y.-P. Hung, C.-S. Chen, Y.-P. Tsai, and S.-W. Lin, "Augmenting panoramas with object movies by generating novel views with disparity-based view morphing," *Journal of Visualization and Computer Animation*, vol. 13, no. 4, pp. 237–247, 2002.
3. W.-Y. Lo, Y.-P. Tsai, C.-W. Chen, and Y.-P. Hung, "Stereoscopic kiosk for virtual museum," in *Proceedings of International Computer Symposium*, 2004.
4. C.-W. Chen, L.-W. Chan, Y.-P. Tsai, and Y.-P. Hung, "Augmented stereo panoramas," in *ACCV (1)*, 2006, pp. 41–49.
5. P.-H. Huang, Y.-P. Tsai, W.-Y. Lo, S.-W. Shih, C.-S. Chen, and Y.-P. Hung, "A method for calibrating a motorized object rig," in *ACCV (1)*, 2006, pp. 379–388.
6. C.-H. Ko, Y.-P. Tsai, Z.-C. Shih, and Y.-P. Hung, "A new image segmentation method for removing background of object movies by learning shape priors," in *Proc. IEEE Int'l Conf. Pattern Recognition*, 2006.
7. S. M. Seitz and C. R. Dyer, "View morphing," in *SIGGRAPH*, 1996, pp. 21–30.
8. J. Xiao and M. Shah, "From images to video: View morphing of three images," in *VMV*, 2003, pp. 495–502.
9. X. Décoret, F. Durand, F. X. Sillion, and J. Dorsey, "Billboard clouds for extreme model simplification," *ACM Trans. Graph.*, vol. 22, no. 3, pp. 689–696, 2003.
10. G. Schaufler, "Image-based object representation by layered impostors," in *VRST*, 1998, pp. 99–104.

11. I. Garcia, M. Sbert, and L. Szirmay-Kalos, "Leaf cluster impostors for tree rendering with parallax," in *Eurographics*, 2005.
12. S. K. Nayar, K. Ikeuchi, and T. Kanade, "Surface reflection: Physical and geometrical perspectives," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 13, no. 7, pp. 611–634, 1991.
13. S. A. Shafer, "Using color to separate reflection components," pp. 43–51, 1992.
14. G. Klinker, S. Shafer, and T. Kanade, "A physical approach to color image understanding," *IJCV*, vol. 4, no. 1. January 1990, pp. 7–38, 1990.
15. K. Nishino, Z. Zhang, and K. Ikeuchi, "Determining reflectance parameters and illumination distribution from a sparse set of images for view-dependent image synthesis," in *ICCV*, 2001, pp. 599–606.
16. K. E. Torrance and E. M. Sparrow, "Theory for off-specular reflection from roughened surfaces," pp. 32–41, 1922.