VIRTUAL REALITY USING THE CONCENTRIC MOSAIC: CONSTRUCTION, RENDERING AND DATA COMPRESSION

Heung-Yeung Shum

Microsoft Research, China

King-To Ng and Shing-Chow Chan

Department of Electrical and Electronic Engineering,
The University of Hong Kong

ABSTRACT

This paper proposes a new image based rendering technique called concentric mosaic for virtual reality applications. It is constructed by capturing vertical slit images when a camera is moving around a set of concentric circles. Concentric mosaic allows the user to move freely in a circular region and observe significant parallax and lighting changes without recovering the geometric and photometric scene model. The rendering of concentric mosaic is very efficient, which amounts to reordering and interpolating of previously captured slit images in the concentric mosaic. Concentric mosaic typically consists of hundreds of high-resolution images, which consumes significant amount of storage and bandwidth for transmission. A MPEG-like compression algorithm is therefore proposed in this paper taking advantages of the access patterns and redundancies of the mosaic images. Experimental results show that real-time reconstruction of novel views with good image quality can be achieved in a Pentium II 300 MHz PC.

1. INTRODUCTION

Images and videos are effective means to convey information of objects, environment or scenes. To provide the users with better experience such as navigation through a virtual environment and to interact with virtual objects, virtual reality techniques are becoming more and more important. Virtual reality (VR) can broadly be classified as immersive and non-immersive. Immersive VR systems typically employ head-mounted or stereo displays and data gloves to convince users that they are interacting with an artificial world. Non-immersive VR systems, on the other hand, are not fully immersive. They typically attempt to recreate a virtual environment as convincingly as possible using conventional or much simpler input and display devices. Apple's QuickTime VR [2] for example, creates an environmental map using a panorama and displays a novel view at any angle around a given point on a graphics monitor. Panorama belongs to a class of techniques called image based rendering which use real photographs to recreate true reality [3,6]. Image based rendering is an excellent alternative if we just want to re-render images at a collection of viewpoints. It not only provides superior image quality than 3D model building but also requires much less computational power for rendering. The panorama mentioned earlier in QuickTime VR is a special case of a plenoptic function with two dimensions. The *plenoptic function* [1] describes all of the radiant energy that can be perceived by the observer at any point in space and time. In its most general form, it is a 7-dimensional function allowing one to reconstruct any novel view at any point in space and time. More sophisticated simplifications of the plenoptic functions include the 4D light field [5] and the 4D lumigraph [4].

In this paper, we report an image based rendering system for realizing virtual reality or virtual walkthrough applications using a novel 3D plenoptic function called concentric mosaic with viewpoints constrained on a plane. The concentric mosaic is set of slit images created by capturing a vertical slit image when a camera is moving around a set of concentric circles, Fig. 2. It is parameterized by three parameters: the radius of the concentric circles, the rotation angle and vertical elevation. Compared with light field and lumigraph, concentric mosaic has much smaller file size because it is a 3D plenoptic function. In contrast to panorama in which the viewpoint is fixed, concentric mosaic allows the user to move freely in a circular region and observe significant parallax and lighting changes without recovering the geometric and photometric scene model. The rendering of concentric mosaic is very efficient, which amounts to reordering and interpolating of previously captured slit images in the concentric mosaics. Concentric mosaic typically stores hundreds of high-resolution images and requires considerable amount of storage. A MPEG-like algorithm for compressing concentric mosaic is therefore proposed in this paper. Experimental results show that real-time reconstruction of novel views with good image quality can be achieved in a Pentium II PC running at 300 MHz. The paper is organized as follows: the principle of concentric mosaic, its construction and rendering are discussed in Section 2. Section 3 is devoted to the compression of concentric mosaics. Finally, we summarize our works in Section 4, the conclusion.

2. THE CONCENTRIC MOSAIC

2.1. Representation and Construction

Concentric mosaic is a set of manifold mosaics [7] constructed from slit images taken by cameras rotating on concentric circles. Fig. 1 shows a possible system for capturing concentric mosaic. A number of cameras C_k, $k = 0,...,n$, are mounted on a rotating horizontal beam that is supported by a tripod. It is assumed that each camera C_k is a slit camera so that only a vertical line of image is taken at a certain viewpoint such as $v_j^{(k)}$ in Fig. 2. To capture the concentric mosaics, the beam is rotating slowly so that each camera C_k is moving continuously along circle CO_k with a radius R_k. Consider the viewpoint $v_j^{(k)}$ on the circle CO_k, the ray that is tangential to the circle CO_k at $v_j^{(k)}$ is captured. Alternatively, we can capture the ray in the opposite viewing direction as shown in dotted lines in Fig. 2 using for example another camera facing the opposite direction. If we put together such slit images at different rotation angles along the circle, a concentric mosaic is formed. For simplicity, we only consider the concentric mosaics *CM*, instead of *CM'*. It can be seen that the entire concentric mosaics consist of *n* concentric mosaics CM_k captured by the *n* cameras. Each of them is in turn consisted of line images taken at different rotation angles by camera C_k. The concentric mosaics are conveniently indexed by its radius and rotation angle, which are much simpler than those of light field and lumigraph. If the number of concentric mosaics *n* is large enough, any such rays in the free space can be reconstructed either from the mosaic images or from their interpolation. It is therefore feasible to reconstruct or render a novel view at any position in the same horizontal plane as the cameras inside the circular free space.

Instead of using many cameras as shown in Fig. 1, a much simpler capturing method is to use a single off-centered camera that rotates along a circle, Fig. 3. The camera can be placed for example in a rotary table with known rotation. At each rotation angle, instead of a slit line image, a regular image with multiple vertical lines (depending on the horizontal field of view (FOV) of the image) is captured. Fig. 3 shows one possible setup called the normal setup. No matter which point the camera is located at the circle CO_n, the same indexed ray captured on the image plane (e.g. the line r_k. r_k' captures the rays in CM'_k.) is always tangential to the same inner circle (CO_k). Putting together the same vertical lines in the image planes, the concentric mosaic CM_k of some inner circle CO_k is constructed. The normal setup covers all the inner circles, from CO_0 to CO_k in this case. This capturing method is very simple because only one circular motion is necessary. The resulting visible (or movable) region is, however, significantly limited by the camera's horizontal FOV. Indeed, for the normal setup, the maximum radius of the visible inner circle CO_k is given by $R_k = R_n \sin(h_{FOV}/2)$, where h_{FOV} is the horizontal FOV of the camera.

2.2. Rendering of a Novel View

Consider the rendering of a novel view at a point P with a polar co-ordinate (R,θ) measured from the center of the CM as shown in Fig. 5. The ray $\overline{PV}_i$ is not captured at the novel view point P. Since the circular region is a free space, we can use the ray previous captured at point v_i in the concentric mosaic CM_k [4,5]. Similarly, the ray $\overline{PV}_j$ can be retrieved from the point v_j in the concentric mosaic CM_j. Therefore, the novel view at P can be completely constructed from the concentric mosaics. In practice, however, only a small subset of the rays is stored in the concentric mosaics. For those rays that are not recorded in the concentric mosaics, they have to be approximated from adjacent ones that have previously been recorded. By taking the whole line from a different location, however, we are making an implicit assumption that the depth is at infinity. This approximation will cause vertical distortion in rendered images. Due to page length limitation, interested readers are referred to [9] for methods that alleviate this problem and other aspects of concentric mosaic.

2.3. Experimental Results

In our experiments of real environment, we have used a Sony Mini DV digital video camera. A Parker 5'' rotary table is used to slowly rotate the camera along an off-centered circle and to provide accurate rotation parameters. The reasons that we rotate the camera slowly are 1) to get sufficient samples along the angular direction for the concentric mosaic, and 2) to avoid motion blur during the capture. A full circle motion takes about 90 seconds and a total of 1350 image frames are recorded. The resolution of each digitized frame is 320×240. The images are then re-binned to construct the concentric mosaic as described in Section 2.1. It took only about 10 minutes to setup, capture and digitize a complete sequence of video needed for constructing concentric mosaic. On a Pentium II 300 MHz PC, we achieved a frame rate of 20 to 30 frames per second for rendering with concentric mosaic.

Fig. 8 (a) shows a mosaic image of a concentric mosaic called "lobby". The total number of mosaic images is 160 (because out of the 320 lines, half of them correspond to CM', see Fig. 2). Fig. 8 (b) and (c) show two novel views rendered from the "lobby" mosaics. It can be seen that strong parallax can be seen between the plant and the poster.

3. COMPRESSION OF CONCENTRIC MOSAICS

3.1. The Random Access Problem

Because concentric mosaics have large spatial resolution, they have to be compressed to reduce the amount of digital storage and bandwidth for transmission. For instance, the entire concentric mosaic "lobby" consists of 1350 (320×240) video frames which require 300 Mbytes of storage without compression. It is natural to apply standard image compression techniques like transform coding, vector quantization and wavelet transform to compress these images. Most of these compression algorithms employ entropy coding, such as Huffman or arithmetic coding, to achieve a better compression ratio. Therefore, the symbols after compression are of variable size, which complicates the rendering of concentric mosaics. In fact, it will be very time-consuming to retrieve the line images if the bit stream does not support any mechanism for randomly or efficiently accessing the compressed line images. As an illustration, let's consider the mosaic image in Fig. 4. Without loss of generality, we assume that the image is compressed by some block-based techniques, such as transform coding using the discrete cosine transform (DCT). Other coding scheme can also be used after appropriate modifications, as suggested in the following, to achieve fast decoding. In Fig. 4, the image is divided into non-overlapping blocks of the macroblock size (16×16). Here, the blocks are scanned vertically so that pixel data of each vertical line are contained in a group of consecutive blocks. In order to retrieve the pixel data of the line L, the compressed data of blocks kM to $(k+1)M-1$ have to be located and decoded. Locating the required data by searching the headers of the blocks can be very time consuming, especially for real-time rendering. To overcome this problem, a set of pointers to the starting locations of the vertical group of blocks in the compressed data is first determined and stored in an array prior to rendering. In this paper, we choose to embed the pointers in the compressed bit streams. This avoids creating the pointer arrays and parsing the entire compressed bit stream when new mosaic images are loaded into the memory. During rendering, the compressed data for the required group of blocks can be located very quickly. Apart from this practical issue, successive concentric mosaics CM_k's also exhibit significant amount of redundancies. In next section, two coding schemes similar to the MPEG-2 standard [10] are proposed to reduce these spatial redundancies.

3.2. Predicting Mosaic Images

As mentioned earlier, successive mosaic images have significant amount of spatial redundancies which can be exploited by prediction technique similar to motion estimation in video coding. A MPEG-like codec is chosen because it can support random access to the individual pictures. In particular, each mosaic image can be treated as a video frame and compress by a MPEG codec. In this paper, we investigate the coding of the mosaic images using the multiperspective panoramas and the image sequences obtained in the normal setup, Fig. 3, to remove the spatial redundancies. The multiperspective panoramas are obtained by putting the lines say $L_i(\theta)$ at the same horizontal position, i, together in the normal setup [8]. Fig. 6 shows how the MPEG-2 algorithm can be used to compress the multiperspective panoramas.

For an I-frame, the pointer structure mentioned earlier can be used to access the compressed data of a group of blocks. If B-pictures are added, for higher compression ratio,

the pointer structure would only allow us to efficiently decode the motion vectors and the prediction residuals of that group of blocks. We still have to retrieve their predictors in the I-frames. Since predictors in the I-frames can in general be located in different group of blocks, several such blocks in the I-frames have to be decoded. The situation is even worse if P-frames are involved because they are in turns predicted from previous P-frames. One solution to this problem is to decode all the I- and P-frames and saved them in the memory for later use, at the expenses of more memory requirement. Hence, there is a tradeoff between fast rendering speed and the amount of compression achieved. Fortunately, we found that by using the multiperspective panoramas, the number of I- and P-pictures can be significantly reduced. Out of the 320 multiperspective panoramas of the "lobby" concentric mosaic, we only need 9 I-pictures. Therefore, there are approximately 39 B-pictures between two I-pictures. If higher compression is needed, the I-pictures in between can further be encoded as P-pictures. During rendering, they will be decoded and loaded into the memory so that fast random access can be achieved. Another reason for the increased separation between the I-pictures is that we have applied global motion estimation to the multiperspective panoramas before using ordinary motion estimation in MPEG-2. This reduces the search range in the MPEG-2 algorithm and hence the data used to represent the motion vectors. More precisely, the block size used in the global motion estimation is (16×224), which consists of a vertical stripe as shown in Fig. 7. Because of the uncovered scene, the upper portion of a multiperspective panorama will not be seen in previous one, Fig. 7. Therefore, only the lower portion is used to estimate the global displacement (motion) vectors for a group of blocks in the "forward prediction". Similarly, in predicting a multi-perspective panorama MP_i from another one with higher position, say MP_{i+1}, the lower portition is not used in the global motion estimation. The global motion vectors will be used as initial positions for carrying out the motion estimation and the differential motion vectors are coded using the MPEG-2 algorithm. We have also studied the encoding of the image sequence obtained from the normal setup. This can be viewed as a video sequence of resolution (320×240). For efficient rendering, we do not employ P-pictures in our algorithm, due to their inter-dependences. The compression ratio of using the normal setup sequence is higher than that of using multiperspective panoramas. The separation of the I-pictures, on the other hand, is much smaller than the former. Therefore, it is impossible to decode all the I-pictures initially and load them into memory for fast rendering. This means that the rendering speed will be slightly slower than using multiperspective pananomas.

3.3. Experimental Results

The "lobby" concentric mosaic described in Section 2 is compressed using the proposed coding algorithms. As all the mosaic images will be used to render the novel views, it is important to ensure that the quantization error to be acceptable throughout the images. Because of this reason, no rate control algorithm is applied and a uniform quantizer is used for the I- and B-pictures. Tables 1 shows the compression performances of the normal setup sequence with quantization scale factor Q=14 and 16, respectively. The performances of using different combinations of I- and B-pictures are also given. It can be seen that using more B-pictures will improve the coding performance when the separation between successive I-pictures are less than 6 for Q=14 and 16. Table 2 shows the compression performances of using the multiperspective panorama. It can be seen that the coding performance is reasonable even when the separation

between successive I-pictures is increased to 39. Fig. 9 and 10 show a rendered view from the compressed normal setup sequence and a typical decompressed multiperspective panorama. It shows good quality reconstruction with a compression ratio of 65 and 42, respectively.

4. CONCLUSION

A new image based rendering technique called concentric mosaic for virtual reality applications is presented. It is constructed by capturing vertical slit images when a camera is moving around a set of concentric circles. Compared with other more sophisticated 4D plenoptic functions such as the light field and lumigraph, the file size of concentric mosaic is much smaller. In contrast to panorama, concentric mosaic allow the user to move freely in a circular region and observe significant parallax and lighting changes without recovering the geometric and photometric scene model. The rendering of concentric mosaic is very efficient, which amounts to reordering and interpolating of previously captured slit images in the concentric mosaic. Concentric mosaic typical consists of hundreds of high-resolution images, which consume significant amount of storage and bandwidth for transmission. A MPEG-like compression algorithm is presented which takes into account the access patterns and redundancies of the mosaic images. Experimental results show that real-time reconstruction of novel views with good image quality can be achieved in a Pentium II 300 MHz PC.

REFERENCES

[1] E. H. Adelson and J. Bergen, "The plenoptic function and the elements of early vision," in *Computational Models of Visual Processing*, pages 3-20. MIT Press, Cambridge, MA, 1991.
[2] S. E. Chen, "QuickTime VR – an image-based approach to virtual environment navigation," in *Computer Graphics (SIGGRAPH'95)*, pp. 29-38, August 1995.
[3] P. E. Debevec, C. J. Taylor, and J. Malik,"Modeling and rendering architecture from photographs : A hybrid geometry – and image-based approach," in *Computer Graphics (SIGGRAPH'96)*, pp.11-20.
[4] S. J. Gortler, R. Grzeszczuk, R. Szeliski, and M. F. Cohen, "The lumigraph," in *Computer Graphics (SIGGRAPH'96)*, pp.43-54.
[5] M. Levoy and P. Hanrahan, "Light field rendering," in *Computer Graphics Processings (SIGGRAPH'96)*, pp. 31-42, August 1996.
[6] L. McMillan and G. Bishop,"Plenoptic modeling: An image-based rendering system," in *Computer Graphics (SIGGRAPH'95)*, pp.39-46.
[7] S. Peleg and J. Herman, "Panoramic mosaics by manifold projection," in *IEEE CVPR'97*, pp. 338-343, June 1997.
[8] D. N. Wood et al, "Multiperspective panoramas for cel animation," in *Computer Graphics (SIGGRAPH'97)*, pp. 243-250.
[9] H. Y. Shum and L. W. He, "Rendering with Concentric Mosaics" in *Computer Graphics (SIGGRAPH'97)*, pp. 299-306, August 1999.
[10] ITU-T Rec. H.262|ISO/IEC 13818-2,"Generic Coding of Moving Pictures and Associated Audio Information: Video," Nov. 1994.

Q = 14	CR	PSNR	Q = 16	CR	PSNR
I only	27.548	35.902	I only	30.491	35.147
I 1B	41.954	36.310	I 1B	47.036	35.585
I 3B	53.602	36.415	I 3B	60.891	35.697
I 5B	56.554	36.444	I 5B	64.830	35.734
I 6B	56.833	36.437	I 6B	65.280	35.724
I 7B	56.155	36.405	I 7B	64.683	35.688
I 8B	55.190	36.380	I 8B	63.626	35.657

Table 1. Compression performance using normal setup sequence.

Q = 16	CR	Mean PSNR
I only	24.277	33.673
I 5B	43.220	34.523
I 10B	46.495	34.525
I 20B	45.033	34.537
I 39B	42.012	34.495

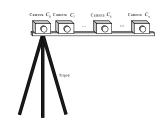Table 2. Compression performance using multiperspective panoramas.

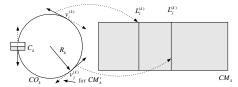Figure 1. A setup for constructing concentric mosaics.
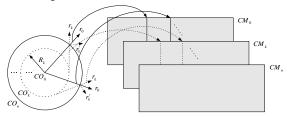

Figure 2. Construction of a concentric mosaic.

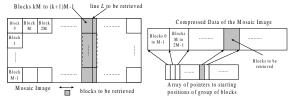Figure 3. Construction of concentric mosaic from one circle : camera along normal.

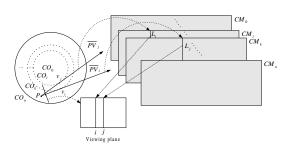Figure 4. Accessing a line L in a mosaic image.

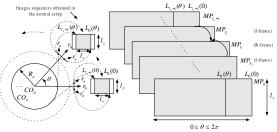Figure 5. Rendering a novel view with concentric mosaic.

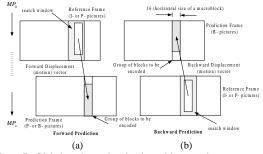Figure 6. Spatial prediction of multiperspective panoramas (MP).

Figure 7. Global motion estimation in multiperspective panoramas : (a) Forward prediction; (b) Backward prediction.

Figure 8 (a). A mosaic image of the "lobby" concentric mosaic.

Figure 8 (b), (c). Two rendered views from the "lobby" concentric mosaic.   Figure 9. Rendered view after decompression.

Figure 10. A decompressed multiperspective panorama.