A Simplified Concentric Mosaics System with Non-uniformly Distributed Pre-captured Images

Xiaoyong Sun and Eric Dubois

School of Information Technology and Engineering (SITE)
University of Ottawa, 161 Louis Pasteur, Ottawa, Canada

ABSTRACT

In this paper, a simplified implementation of the Concentric Mosaics image-based rendering technique is proposed. The greatest difficulty for an ordinary user in obtaining the pre-captured images for use in the Concentric Mosaics technique is the precise control of the rotation of a long beam. In the proposed Simplified Concentric Mosaics technique, the camera positions where the pre-captured images will be taken are not precisely controlled, but are estimated from the pre-captured images. We use a stereo technique for estimation of camera positions in this special scenario instead of using the traditional camera pose estimation methods of computer vision. The estimation errors have been analyzed and a closed-loop constraint is used to achieve better rotation angle estimation. In addition, a ratio fitting technique is proposed to select good matching features in the special tri-view matching detection scenario, which subsequently improves rotation angle estimation. Another contribution of the paper is a pre-processing step to eliminate or reduce the possible vertical offsets and other distortions in the pre-captured images, which are caused by any possible motions of the camera that deviate from the ideal one. In a column-based view synthesis technique like the proposed method and the conventional Concentric Mosaics technique, these vertical offsets and distortions in the pre-captured images will lower the quality of the synthesized images. Thus our pre-processing can be applied on both the proposed method and the ordinary Concentric Mosaics technique. The pre-captured image data structures of both Concentric Mosaics and the proposed method have been illustrated and the comparison has been made. The proposed technique has a similar data structure and thus a similar rendering algorithm as the conventional Concentric Mosaics technique. As a result, it meets our objective that an ordinary user can obtain the Concentric Mosaics type image data and plug it into a common Concentric Mosaics rendering framework. Simulation results show that the proposed method can achieve good rendering results.

Keywords: Image-based Rendering, Concentric Mosaics

1. INTRODUCTION

Image-based rendering (IBR) has attracted a great deal of interest recently. With the increase of computer power and communication bandwidth, one goal of IBR, to virtually navigate in real-image-based environments, has become feasible. Compared with traditional computer-graphics-based virtual environments, real-image-based virtual environments are easy to obtain because the IBR techniques are usually relatively scene independent. Moreover, they look more realistic because real scenes are usually too complex to model accurately with computer-graphic primitives.

Many methods have been proposed for IBR,[1] among which three basic types in the category of rendering without geometric information are panoramas,[2,3] Concentric Mosaics[4] and Light Field Rendering.[5] The ideas involve representation of the plenoptic function[6] based on pre-captured image samples. However, these methods strongly depend on the method used to obtain the pre-captured images; different methods lead to different rendering strategies.

Different approaches have respective advantages and disadvantages and may be used for different applications. The major considerations for the user to choose one particular method may be the cost to obtain the pre-captured images for the given applications and the fidelity of the represented environments with respect to the real ones.

Xiaoyong Sun: xsun@site.uottawa.ca; Eric Dubois: edubois@site.uottawa.ca

Visual Communications and Image Processing 2005, edited by Shipeng Li, Fernando Pereira, Heung-Yeung Shum, Andrew G. Tescher, Proc. of SPIE Vol. 5960 (SPIE, Bellingham, WA, 2005) · 0277-786X/05/$15 · doi: 10.1117/12.632659

Usually there is a tradeoff between these two issues. This can be illustrated from the comparison between Concentric Mosaics and Light Field Rendering. In Light Field Rendering, the camera positions are controlled in a two-dimensional plane and it is a strict implementation of plenoptic functions. Theoretically, an accurate representation should result. However, the precise control of the camera position in two dimensions is expensive for the current technology, especially when dealing with large environments. The Concentric Mosaics technique is a clever way to reduce the technical requirement from two-dimensional camera position control to one-dimensional camera position control. However, as a consequence, depth distortion is unavoidable; the technique reduces the representation fidelity but it may be adequate for many applications. In the future, a complete IBR system might be hybrid and compatible with different structures of the pre-captured image data obtained from different methods, using the concept of plenoptic primitives.[7]

In this paper, a method that is based on the Concentric Mosaics data structure but further reduces the technical requirement is proposed. Our system to obtain the pre-captured images is similar to the Concentric Mosaics setup but simpler. We eliminate the precise camera position control part in the Concentric Mosaics technique, because we believe it is still expensive for an ordinary user to use the Concentric Mosaics technique for IBR applications. Thus, we name our method the Simplified Concentric Mosaics (SCOM) technique. In the proposed method, the non-uniform camera positions have to be determined from the pre-captured images. Fortunately, since the camera trajectory is on a circle, this constraint can considerably simplify the camera position estimation.

In section 2, the mechanical setup of the SCOM technique is briefly described; it is very similar to that in the conventional Concentric Mosaics technique. The methods to estimate the camera positions on a circle are described in Section 3, which is the key issue in this paper. The pre-processing methods to eliminate or reduce the possible vertical offsets and distortions in the pre-captured images, which might be brought by any motions of the camera that deviate from the ideal ones, are addressed in Section 4. Section 5 gives the image data structure, comparing to that in the ordinary Concentric Mosaics technique. The simulation results are provided in Section 6, with our conclusions and future work in Section 7.

2. SIMPLIFIED CONCENTRIC MOSAICS SETUP

The setup of SCOM is similar to the Concentric Mosaics setup, but without the precise rotation control part. A long bar is mounted on a tripod (or a more stable platform) and can be freely rotated manually. A video camera is connected to the other side of the long bar, with the imaging direction of the camera along the radial direction of the bar, pointed to the outside. The basic setup is similar to the Concentric Mosaics setup as described in ref. 4.

The number of pre-captured images to represent an environment using the Concentric Mosaics technique has been studied[8] based on plenoptic sampling theory.[9] In the SCOM technique, the long bar is rotated as slowly as possible to obtain as many pre-captured images as possible, so that we over-sample the environment. The reason for over-sampling will be explained in the following section.

3. ESTIMATION OF THE CAMERA POSITIONS FROM THE PRE-CAPTURED IMAGES

Camera position estimation from an image sequence has been studied in computer vision.[10] Assume that an image sequence has been taken by a calibrated camera. By tracking the positions of a number of feature points in the image sequence, the projection matrices can be obtained. The relative camera positions can be calculated as external parameters after decomposing the projection matrices. Then, bundle adjustment is applied in the camera position estimation refinement to get the converged solutions. However, it requires a large amount of computation to obtain the camera position using this technique. Furthermore, when the differences between the camera positions are too small, the epipolar constraints are not reliable. The bundle adjustment algorithms that are commonly used in computer vision may not converge in this situation.

Fortunately, the camera position estimation can be greatly simplified due to the special trajectory of the camera movement in SCOM. In this paper, we use a stereo approach to estimate the camera positions.

3.1. Depth estimation using the stereo technique

The stereo technique is the fundamental method for depth estimation[11]; the principle of the stereo technique is illustrated in Fig. 1.

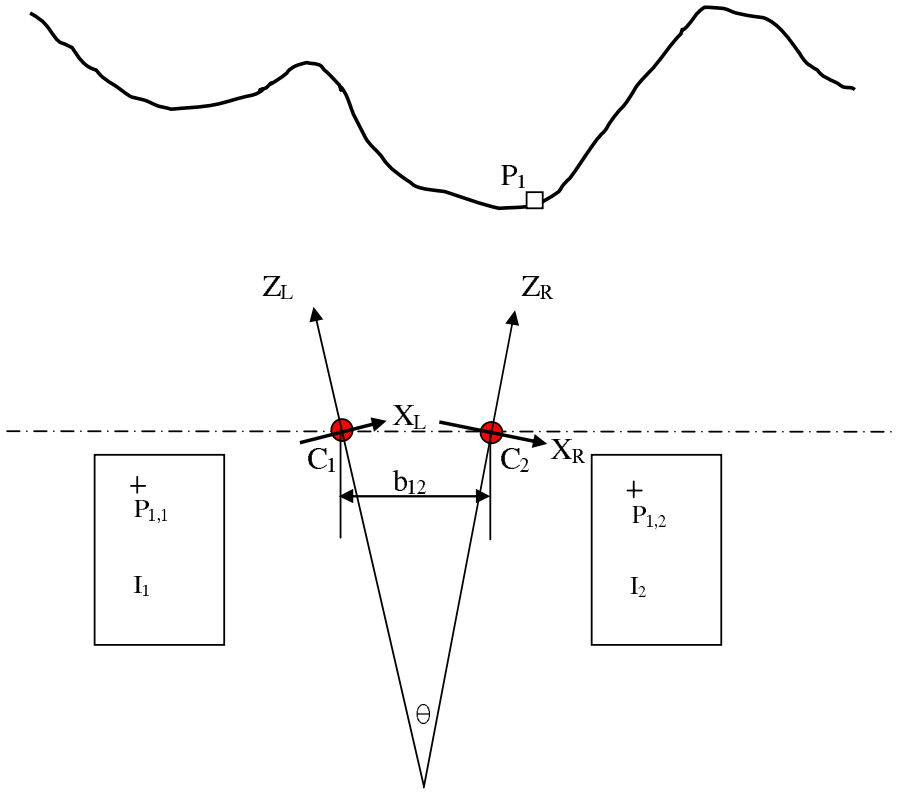

Figure 1. Setup for depth estimation using the stereo technique with two views

C_1 and C_2 are the positions of the camera (in this paper, we refer to the projection center of a camera as its position). Two images I_1 and I_2 are taken at these two positions, respectively. For a scene point P_1, its projection in the image I_1 is at $P_{1,1}$ and at $P_{1,2}$ in image I_2. The line segment C_1C_2 is called the baseline for the stereo pair I_1, I_2 and its length is denoted by $b_{1,2}$. The camera coordinate systems (X, Y, Z) at different positions are shown in the figure, with Y directions pointing out of paper. (X_{1L}, Z_{1L}) and (X_{1R}, Z_{1R}) denote the (X, Z) coordinates of the scene point P_1 in the two camera systems. The angle between the optical axis (in the same plane) of the two cameras is θ. Because the scene point is far away and the angle θ is small, the relationship between the baseline $b_{1,2}$ and the coordinates Z_{1L} and Z_{1R} can be approximated by[11]

$$Z_{1L} \approx Z_{1R} \approx \frac{f \cdot b_{1,2}}{d(P_{1,1}, P_{1,2}) - f \cdot \theta} \tag{1}$$

where, $d(P_{1,1}, P_{1,2})$ is the disparity between points $P_{1,1}$ and $P_{1,2}$ in the horizontal direction. We assume that there is no vertical disparity for the ideal situation. f is the focal length, and all the above values are in units of pixels.

3.2. Relationship between the baselines and the disparities on a circular camera path

Using the above stereo technique, we can show that the distance between two cameras which are closely spaced on a circle is proportional to the horizontal disparity of a scene point in the correspondent images. That is the camera movement situation for the SCOM.

In Fig. 2, C_1, C_2, and C_3 are the camera positions that are on an arc. The distance between C_1 and C_2 is $b_{1,2}$, and the distance between C_2 and C_3 is $b_{2,3}$. The images I_1, I_2, and I_3 are taken at C_1, C_2, and C_3, respectively. Assume that the scene point $P_{i_1}^{(1)}$ is projected into images I_1, I_2 and I_3 at positions $P_{i_1,1}^{(1)}$, $P_{i_1,2}^{(1)}$ and

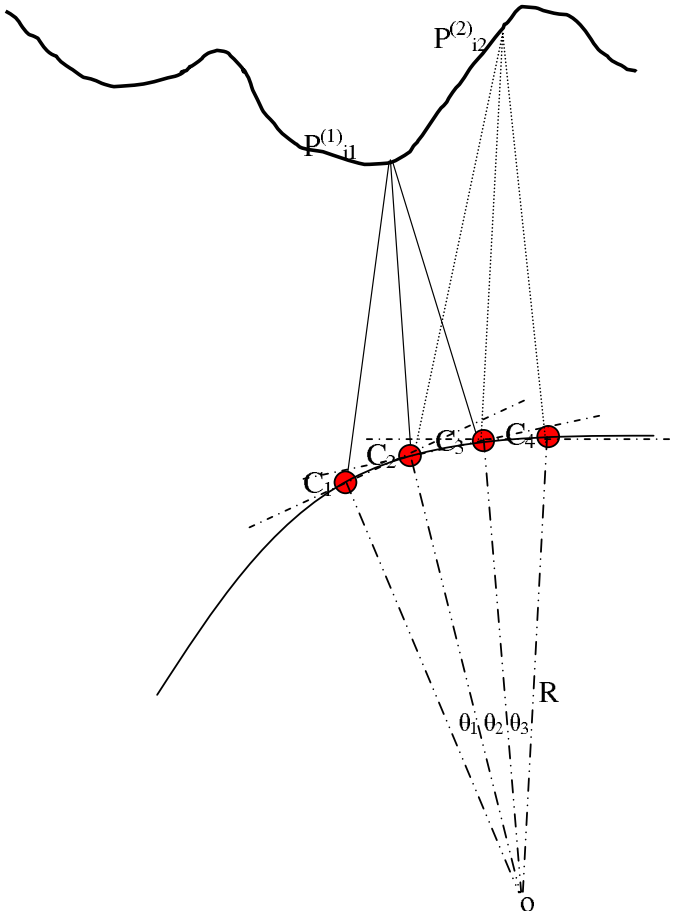

Figure 2. Setup for depth estimation using the stereo technique with multiple views

$P_{i_1,3}^{(1)}$. There should be many scene points that are shared by images I_1, I_2, and I_3, so we use i_1 to represent one of them, where $i_1 \in \{1, 2, .., M_{123}\}$, if the total number of matching features among images I_1, I_2, and I_3 is M_{123}. If the horizontal disparity between $P_{i_1,1}^{(1)}$ and $P_{i_1,2}^{(1)}$ is $d(P_{i_1,1}^{(1)}, P_{i_1,2}^{(1)})$ and the horizontal disparity between $P_{i_1,2}^{(1)}$ and $P_{i_1,3}^{(1)}$ is $d(P_{i_1,2}^{(1)}, P_{i_1,3}^{(1)})$, then

$$Z_{i_1,1L}^{(1)} \approx Z_{i_1,1R}^{(1)} \approx \frac{f \cdot b_{1,2}}{d(P_{i_1,1}^{(1)}, P_{i_1,2}^{(1)}) - f \cdot \theta_1} \tag{2}$$

and similarly,

$$Z_{i_1,2L}^{(1)} \approx Z_{i_1,2R}^{(1)} \approx \frac{f \cdot b_{2,3}}{d(P_{i_1,2}^{(1)}, P_{i_1,3}^{(1)}) - f \cdot \theta_2} \tag{3}$$

where, $Z_{i_1,1L}^{(1)}$ and $Z_{i_1,1R}^{(1)}$ are the Z coordinates of scene point $P_{i_1}^{(1)}$ in the stereo system with camera positions at C_1 and C_2. $Z_{i_1,2L}^{(1)}$ and $Z_{i_1,2R}^{(1)}$ are the Z coordinates of scene point $P_{i_1}^{(1)}$ in the stereo system with camera positions at C_2 and C_3. Images I_1 and I_2 and images I_2 and I_3 form two stereo pairs, respectively. Image I_2 is the right image for stereo pair I_1 and I_2 and the left image for image pair I_2 and I_3, thus $Z_{i_1,1R}^{(1)} = Z_{i_1,2L}^{(1)}$.

Because the camera is moving on a circle, we have $\theta_1 = \frac{b_{12}}{R}$ and $\theta_2 = \frac{b_{23}}{R}$, where R is the radius of the circle. Substituting all these relationships into equations (2) and (3), we obtain

$$\frac{b_{12}}{d(P^{(1)}_{i_1,1}, P^{(1)}_{i_1,2})} \approx \frac{b_{23}}{d(P^{(1)}_{i_1,2}, P^{(1)}_{i_1,3})}. \tag{4}$$

This demonstrates our conclusion that the distance between two cameras that are on a circle is proportional to the horizontal disparity of a scene point in the correspondent images.

3.3. Estimation of the camera positions on a circle

From equation (4), we obtain

$$b_{2,3} \approx \frac{d(P^{(1)}_{i_1,2}, P^{(1)}_{i_1,3})}{d(P^{(1)}_{i_1,1}, P^{(1)}_{i_1,2})} b_{1,2}. \tag{5}$$

Similarly, we assume that images I_2, I_3 and I_4 share scene point $P^{(2)}_{i_2}$ with its imaging points on I_2, I_3 and I_4 at positions $P^{(2)}_{i_2,2}$, $P^{(2)}_{i_2,3}$ and $P^{(2)}_{i_2,4}$. Then we have

$$b_{3,4} \approx \frac{d(P^{(2)}_{i_2,3}, P^{(2)}_{i_2,4})}{d(P^{(2)}_{i_2,2}, P^{(2)}_{i_2,3})} b_{2,3} \tag{6}$$

or

$$b_{3,4} \approx \frac{d(P^{(2)}_{i_2,3}, P^{(2)}_{i_2,4})}{d(P^{(2)}_{i_2,2}, P^{(2)}_{i_2,3})} \frac{d(P^{(1)}_{i_1,2}, P^{(1)}_{i_1,3})}{d(P^{(1)}_{i_1,1}, P^{(1)}_{i_1,2})} b_{1,2}. \tag{7}$$

If we assume that there are N images taken by the camera at the positions $C_1, C_2, ..., C_N$ around the entire circle, then for any distance $b_{k,k+1}$ between C_k and C_{k+1}, we have

$$b_{k,k+1} \approx \frac{d(P^{(k-1)}_{i_k,k}, P^{(k-1)}_{i_k,k+1})}{d(P^{(k-1)}_{i_k,k-1}, P^{(k-1)}_{i_k,k})} ... \frac{d(P^{(2)}_{i_2,3}, P^{(2)}_{i_2,4})}{d(P^{(2)}_{i_2,2}, P^{(2)}_{i_2,3})} \frac{d(P^{(1)}_{i_1,2}, P^{(1)}_{i_1,3})}{d(P^{(1)}_{i_1,1}, P^{(1)}_{i_1,2})} b_{1,2} \tag{8}$$

with $k = 2, ..., N-1$. We also calculate the distance between the last image and the first image related to b_{12}.

In the Concentric Mosaics technique-based rendering algorithm, the relative angles around the rotation center between different camera positions are required, instead of the real camera positions. If we assume the perimeter is $2\pi R$ and then,

$$\sum_{k=1}^{N} b_{k,k+1} = 2\pi R \tag{9}$$

where, $b_{N,N+1} = b_{N,1}$, and R is the length of the rotation beam. This yields a set of linear equations that we can solve for the $b_{k,k+1}$

3.4. Using multiple sets of matching features to improve the estimation

In the above approach, we only use one common matching feature among three adjacent images to estimate the relative camera positions on a circle. If we want to use multiple matching features among three adjacent views, the optimal value can be calculated in the sense of minimizing the mean square errors. For the ratio of the distance between the first two camera positions, or $\eta_1 = \frac{b_{2,3}}{b_{1,2}}$, we have

$$\eta_1 \approx \frac{d(P^{(1)}_{i_1,2}, P^{(1)}_{i_1,3})}{d(P^{(1)}_{i_1,1}, P^{(1)}_{i_1,2})} \tag{10}$$

or $d(P^{(1)}_{i_1,2}, P^{(1)}_{i_1,3}) \approx \eta_1 d(P^{(1)}_{i_1,1}, P^{(1)}_{i_1,2})$. The optimal ratio η_1 using multiple matching points in sense of minimizing the mean square errors can be estimated as

$$\hat{\eta}_1 = \arg\min_{\eta_1} \sum_{i_1=1}^{M_{123}} (d(P^{(1)}_{i_1,2}, P^{(1)}_{i_1,3}) - \eta_1 d(P^{(1)}_{i_1,1}, P^{(1)}_{i_1,2}))^2. \tag{11}$$

Thus, solving this linear least-square problem,

$$\hat{\eta}_1 = \frac{\sum_{i_1=1}^{M_{123}} d(P^{(1)}_{i_1,2}, P^{(1)}_{i_1,3}) \cdot d(P^{(1)}_{i_1,1}, P^{(1)}_{i_1,2})}{\sum_{i_1=1}^{M_{123}} d(P^{(1)}_{i_1,1}, P^{(1)}_{i_1,2})^2}. \tag{12}$$

Similarly, we can estimate the successive ratios $\hat{\eta}_2 = \frac{b_{3,4}}{b_{2,3}}$, $\hat{\eta}_3 = \frac{b_{4,5}}{b_{3,4}}$, ..., up to $\hat{\eta}_N$.

3.5. Closed-loop constraint for regularization

Equations (5) to (8) are approximations. The approximation errors come from two sources. The first one is that we make an assumption $Z^1_{i_1,1L} \approx Z^1_{i_1,1R}$ in section 3.1 as well as neglecting several small terms to obtain. The second one is that there may be some errors in the disparity estimation. The approximation errors will propagate from the baseline ratio in one stereo pair to those in the following ones. This may end up with ruining our camera-position estimation results. Fortunately, the camera is moving on a circle so that we can use a closed-loop constraint as regularization term to relieve the problem.

Considering the approximation errors, equations (5) to (8) can be rewritten as

$$b_{2,3} = \beta_1 \frac{d(P^{(1)}_{i_1,2}, P^{(1)}_{i_1,3})}{d(P^{(1)}_{i_1,1}, P^{(1)}_{i_1,2})} b_{1,2} \tag{13}$$

$$b_{3,4} = \beta_2 \frac{d(P^{(2)}_{i_2,3}, P^{(2)}_{i_2,4})}{d(P^{(2)}_{i_2,2}, P^{(2)}_{i_2,3})} b_{2,3} \tag{14}$$

$$b_{3,4} = \beta_2\beta_1 \frac{d(P^{(2)}_{i_2,3}, P^{(2)}_{i_2,4})}{d(P^{(2)}_{i_2,2}, P^{(2)}_{i_2,3})} \frac{d(P^{(1)}_{i_1,2}, P^{(1)}_{i_1,3})}{d(P^{(1)}_{i_1,1}, P^{(1)}_{i_1,2})} b_{1,2} \tag{15}$$

$$b_{k,k+1} = \beta_{k-1}\beta_{k-2}...\beta_1 \frac{d(P^{(k-1)}_{i_k,k}, P^{(k)}_{i_k,k+1})}{d(P^{(k-1)}_{i_k,k-1}, P^{(k-1)}_{i_k,k})} ... \frac{d(P^{(2)}_{i_2,3}, P^{(2)}_{i_2,4})}{d(P^{(2)}_{i_2,2}, P^{(2)}_{i_2,3})} \frac{d(P^{(1)}_{i_1,2}, P^{(1)}_{i_1,3})}{d(P^{(1)}_{i_1,1}, P^{(1)}_{i_1,2})} b_{1,2} \tag{16}$$

$k = 1, ..., N-1$. β_1, β_2, ... β_{k-1} are the approximation factors which are close to 1 and represent the approximation errors.

Due to the circular trajectory of the camera motion, we can estimate the ratio of the baselines between the first stereo pair and the last stereo pair from the feature matchings. Thus we add one more equation to the above set of linear equations and end up with an over-determined set of linear equations, which implicitly adds the constraint $\beta_N\beta_{N-1}\beta_{N-2}...\beta_1 = 1$. This constraint tends to force the individual approximation error factors β_N, β_{N-1}, β_{N-2} ... β_1 equal to 1.

3.6. Selection of good matching features by ratio fitting

The matching errors play a very significant role in the proposed stereo-based camera position estimation method. They also significantly influence the camera position estimation results using projection-theory approaches in computer vision.

Traditionally, the tri-view feature matching procedure is carried out in following fashion.[12] Assume we have three views I_1, I_2 and I_3. The matching feature points between I_1 and I_2 and between I_2 and I_3 are first detected, and then the matching relationships between I_1, I_2 and I_3 are constructed by a tracking technique. Usually, faulty matching relationships are unavoidable. Thus, the tri-view tensor is first estimated based on these matching points using methods such as RANSAC. Then, the good matchings are selected if the matching features respect the estimated tensor. The algorithm works well for most scenarios and has been widely adopted in computer vision. However, the above tensor is estimated from the matching feature pool with possibly many faulty matching relationships and it may not accurate. Moreover, the method will give very poor results in some scenarios where the epipolar constraints are not reliable, such as when the camera positions are too close to each other.

Under the special circumstance in this paper, we use the method we called ratio fitting to select good matching features from the matching feature pool. Because we know that the disparity ratio between two adjacent stereo pairs should be a constant, we select the majority of matching features that respect this constant ratio principle as good matching features. Our method is similar to the method to determine the codewords in the vector quantization coding scheme but implemented in one dimension.

In the proposed method, we can use the disparities between matching features, dense disparity maps or block-based disparities. Smooth depth variations are assumed when block-based disparities are used, which is the case for most practical environments.

4. PRE-PROCESSING OF THE PRE-CAPTURED IMAGES

In the above discussion, we assume that there are no vertical disparities between any two of the pre-captured images. However, it may not be the case in practice. The view rendering in Concentric Mosaics-based techniques is a column-based view interpolation. Thus, the vertical motions, and any other motions that deviate from the ideal ones, in the pre-capturing procedure will significantly affect the quality of the synthesized images.

In this section, we give a method to eliminate or reduce the possible vertical offsets and distortions in the pre-captured images. Because the distance between the camera positions where two adjacent images were taken is very short and the camera is shooting outward along the long bar, the two adjacent images are approximately captured in parallel directions. The view rectification in computer vision can be used.

In this paper, we apply affine transformations to reduce such offsets and distortions. The correspondent camera motions generally include rotations and vertical movements. Thus the approach is similar to what we proposed in ref. 13.

There are six parameters to define an affine transformation; we use a vector to represent them,

$$\boldsymbol{t} = \begin{bmatrix} t_{11} & t_{12} & t_{21} & t_{22} & t_{31} & t_{32} \end{bmatrix}^T \tag{17}$$

where t_{31} and t_{32} denote the translations along horizontal and vertical direction respectively. We use $\widetilde{I}(I, \boldsymbol{t})$ to represent the resulting image after applying the affine transformation with parameter $\boldsymbol{t}$ on the image I.

Assume that $I_1, I_2, ..., I_N$ are the pre-captured images. For any image I_i, an optimal global affine transformation is first found as,

$$\hat{\boldsymbol{t}}_i = \arg\min_{\boldsymbol{t}_i} \sum_{\boldsymbol{x}} |\widetilde{I}(I_i, \boldsymbol{t}_i)(\boldsymbol{x}) - \widetilde{I}_{i-1}(\boldsymbol{x})| \tag{18}$$

where, $\widetilde{I}_{i-1} = \widetilde{I}(I_{i-1}, \boldsymbol{t}'_{i-1})$ and $\widetilde{I}_1 = I_1$. The parameter vector $\boldsymbol{t}'_{i-1}$ is given by

$$\boldsymbol{t}'_{i-1} = \begin{bmatrix} t_{11}(i-1) & t_{12}(i-1) & t_{21}(i-1) & t_{22}(i-1) & 0 & t_{32}(i-1) \end{bmatrix}^T \tag{19}$$

if

$$\hat{\boldsymbol{t}}_{i-1} = \begin{bmatrix} t_{11}(i-1) & t_{12}(i-1) & t_{21}(i-1) & t_{22}(i-1) & t_{31}(i-1) & t_{32}(i-1) \end{bmatrix}^T. \quad (20)$$

The above pre-processing is applied for $i = 2, 3, ..., N$.

If the vertical disparities are the only issue of concern, the rotation matrix will be an identity matrix. In that case, we can use the vertical disparities that we obtained from equation (19) to perform re-sampling in the pre-processing.

5. RENDERING WITH IRREGULAR IMAGE SAMPLES

The capturing and rendering procedure of the Concentric Mosaics technique is illustrated in Fig. 3. The camera is mounted on one end of the long rotation bar CE. When CE rotates around C at a constant velocity, the video camera takes images. In the Concentric Mosaics technique, the pixels in one column of the pre-captured images are grouped into one condensed light ray, namely a sampled ray. The positions at which the images are captured on the camera path are sampled points. In Fig. 3, SP is a sampled point and SR is a sample ray, which corresponds to one column in the image I taken by the camera at the rotation angle σ. The sampled ray SR corresponds to a condensed light ray through angle β with CF. CF is perpendicular to image I and passes through its center.

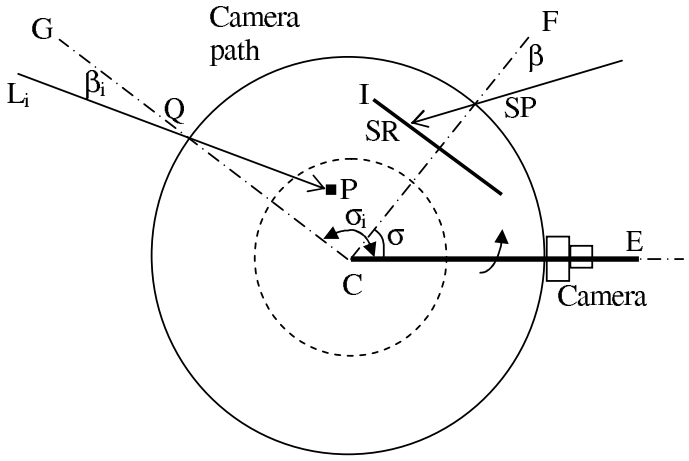

Figure 3. The capturing and rendering procedure

By stacking the pixels of each column into one element, the pre-captured image has a one-dimensional data structure. Thus the data structure of the whole set of pre-captured images can be represented in a σ-β plane as shown in Fig. 4. Each dot in Fig. 4 denotes a column within the entire set of pre-captured images. All dots in the same horizontal row correspond to one pre-captured image.

The rendering procedure can also be illustrated with Fig. 3. P is an arbitrary position within the navigation area (the dashed circle) and L_i is one condensed light ray toward P. An arbitrary view at position P is constructed by a set of condensed light rays when the viewing direction is given, just like putting a virtual camera at P.

The condensed light ray L_i is determined by two angles σ_i and β_i as shown in Fig. 3. If the intersection point Q of L_i with the camera path happens to be a sampled point and there is a sampled ray corresponding to β_i, that sample ray can be directly put into the final image. However, that is not generally true. For a general case, L_i is one point illustrated in Fig.4. L_i will be interpolated from its nearby sampled rays SR_1, SR_2, SR_3 and SR_4 in the figure as

$$L_i = \omega_1 SR_1 + \omega_2 SR_2 + \omega_3 SR_3 + \omega_4 SR_4, \quad (21)$$

where ω_1, ω_2,ω_3, and ω_4 are weights for interpolation.

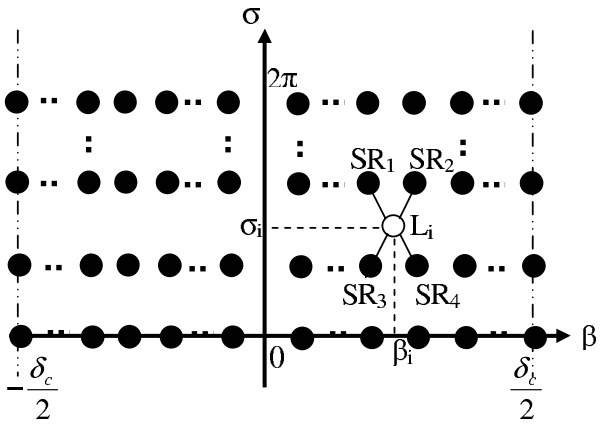

Figure 4. The data structure of pre-captured images in σ-β plane for the Concentric Mosaics technique

Depth information of the environment is required to correctly calculate the weights for interpolation, which is difficult to obtain. Thus the infinite depth assumption and the constant depth assumption are used; further details can be found in ref. 14. The nearest point approximation can also be classified in the above interpolation formula, with only one weight equal to one while the others are zero.

In the proposed SCOM technique, the data structure of image samples is very similar with the standard Concentric Mosaics technique and thus so are the rendering methods. The only difference is that the dots, which represent sample rays, are no longer uniformly distributed along the σ directions, but are irregularly distributed as shown in Fig.5. However, this will cause little difference in rendering.

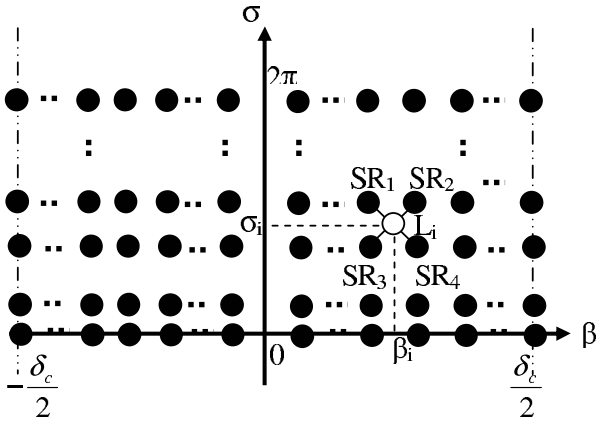

Figure 5. The data structure of pre-captured images in σ-β plane for the SCOM technique

6. EXPERIMENTAL RESULTS

The first two simulations to test the proposed camera position algorithm were carried out using Concentric Mosaics data (the 'kids' sequence in ref. 4) that was provided by Microsoft Research. In the Concentric Mosaics technique, the camera rotation is controlled by a motor and the camera positions are supposed to be uniformly distributed along the circle. We have no knowledge of the precision of the motor controller that was used by Microsoft Research. The third simulation involved rendering novel views using the proposed SCOM with non-uniformly distributed pre-captured image samples.

6.1. Simulation I: camera position estimation when cameras are uniformly distributed

In the first simulation, we tested the proposed camera position estimation algorithm in the uniform camera position scenario. In order to obtain reliable matching relationships between features and reduce the computations, the original sequence was sub-sampled. 297 images were selected from the image sequence and were equally spaced, except that the spacing between the first one and the last one is slightly different from others since the total number of images is 2967. The initial matching features were found using the publicly available PVT software.[15] The rotation angles for all camera positions where these images are supposed to be taken were estimated and the estimation results are shown in Fig. 6. The rotation angle, or the cumulative angle, is defined as the angle between the camera position where the current image was taken and the camera position where the first image was taken.

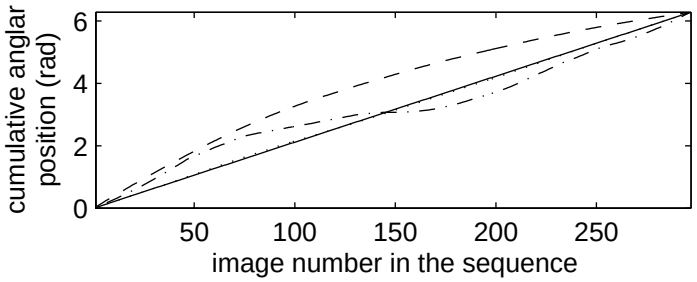

Figure 6. Estimate of cumulative angular position using three different methods for simulation I

In the figure, the solid straight line represents the assumed uniformly distributed angle along the circle. The dash-dot line is the rotation angles estimated by using the tensor-based matching features together with the closed-loop constraint. The dotted line shows the rotation angles estimated by using the matching features selected by the ratio fitting criterion together with the closed-loop constraint. Without the closed-loop constraint, the estimated rotation angles are shown in the dashed line. The method of minimizing mean square error to estimate baseline ratios from multiple matching features was used for all of the above situations. This result shows the importance of both the closed-loop constraint and the ratio fitting criterion.

6.2. Simulation II: camera position estimation when cameras are non-uniformly distributed

In order to test the proposed camera position estimation algorithm in the non-uniform camera position scenario, we selected pre-captured images from the 'kids' sequence in the following way. The first 50 images were spaced by 8 images, the second 50 images were spaced by 16 images, the third 50 images were spaced by 24 images, the remaining images were spaced by 8 images again. Assuming the positions in the original sequence were uniform, the angles between adjacent images in these sub-sequence was 0.017, 0.034, 0.051 and 0.017, respectively. Fig. 7 shows the simulation results using the proposed method with the closed-loop constraint and the ratio fitting criterion. The cumulative angles are shown in the top of the figure. The solid line is the ideal case, assuming the original pre-captured images are uniformly distributed, and the dashed line shows angles using the proposed camera angle estimation algorithm. The individual angles between two adjacent camera positions are shown in the bottom of the figure.

6.3. Simulation III: rendering with SCOM through non-uniformly distributed pre-captured images

In the last simulation, we tested the proposed rendering algorithm with non-uniformly distributed pre-captured images. The pre-captured images were captured by a CCD web-camera with selected resolution 320x240. The camera was mounted on one end of a long bar, with the other end of the long bar bolted to a desk at its center. The long bar was about 1.5 meters and could be manually rotated around the center of the desk. The camera was forced to rotate on a levelled circle because the long bar was held firmly against the desk surface. There were 2309 images taken during one loop. In Fig. 8, the cumulative angles for the estimated camera positions

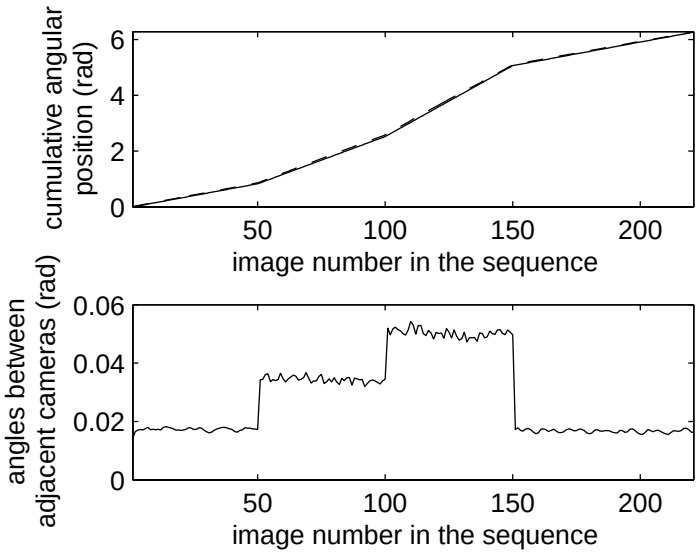

Figure 7. The results for simulation II using the proposed method

are shown, and one of the novel views generated by the proposed rendering algorithm under constant depth assumption is shown in Fig. 9.

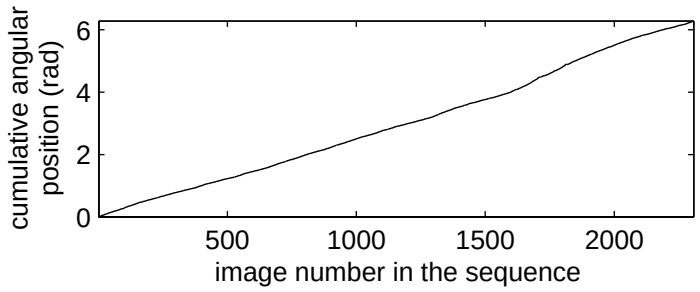

Figure 8. Results for simulation III: cumulative angular positions (rad)

Figure 9. Results for simulation III: one of the synthesized views

7. CONCLUSION AND FUTURE WORK

In this paper, a method to simplify the implementation of the Concentric Mosaics technique has been proposed. Compared with the conventional Concentric Mosaics technique for IBR, the implementation requirement is reduced in the proposed Simplified Concentric Mosaics. The method is oriented for an ordinary user to capture Concentric Mosaics data and put it into the Concentric Mosaics rendering framework. The key issue is to estimate the camera position from the pre-captured images.

From the simulation results, we can conclude that our method for camera position estimation works well either when the camera positions are uniformly distributed on a circle or when the camera positions are non-uniformly distributed. The closed-loop constraint and the method of using ratio-fitting to select good matching features are key elements for the proposed method to give good results. In addition, the synthesized novel views based on the proposed SCOM are of good quality. Currently, the camera position estimation errors are mainly caused by position errors between the matching features and work to reduce these errors is currently under way.

ACKNOWLEDGEMENT

This research was supported by Natural Sciences and Engineering Research Council of Canada under Strategic Grant STPGP 269997. The authors wish to thank Microsoft Research for providing the Concentric Mosaics data set. The authors would also like to thank Dr. Gerhard Roth for discussions on algorithms for matching feature points and Dr. Robert Laganière for his assistance in implementing the simulations.

REFERENCES

1. H.-Y. Shum, S. Kang, and S.-C. Chan, "Survey of image-based representations and compression techniques," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, pp. 1020–1037, November 2003.
2. S. E. Chen, "Quicktime VR — an image-based approach to virtual environment navigation," *Computer Graphics (SIGGRAPH'95)*, pp. 29–38, August 1995.
3. H.-Y. Shum and R. Szeliski, "Construction of panoramic image mosaics with global and local alignment," *International Journal of Computer Vision*, vol. 36, no. 2, pp. 101–130, 2000.
4. H.-Y. Shum and L. He, "Rendering with concentric mosaics," *Computer Graphics (SIGGRAPH'99)*, pp. 299–306, January 1999.
5. M. Levoy and P. Hanrahan, "Light field rendering," *Computer Graphics (SIGGRAPH'96)*, pp. 31–42, August 1996.
6. E. H. Adelson and J. R. Bergen, "The plenoptic function and the elements of early vision," *Computation Models of Visual Processing, MIT Press, Cambridge*, 1991.
7. S. Kang, M. Wu, Y. Li, and H.-Y. Shum, "Large environment rendering using plenoptic primitives," *IEEE Trans. Circuits Syst. Video Technol.*, pp. 1064–1073, 2003.
8. X. Sun and E. Dubois, "Scene sampling for the concentric mosaics technique," *Proc. 2002 IEEE Int. Conf. Image Processing*, pp. I–465 – I–468, September 2002.
9. J.-X. Chai, X. Tong, S.-C. Chan, and H.-Y. Shum, "Plenoptic sampling," *Computer Graphics (SIGGRAPH'2000)*, pp. 307–318, July 2000.
10. G. Roth and A. Whitehead, "Using projective vision to find camera positions in an image sequence," *Proc. Vision Interface 2000*, pp. 225–232, May 2000.
11. E. Izquierdo and J.-R. Ohm, "Image-based rendering and 3D modeling: A complete framework," *Signal Processing: Image Communication*, vol. 15, pp. 817–858, 2000.
12. A. Whitehead and G.Roth, "The projective vision toolkit," *Proc. Modelling and Simulation*, pp. 204–209, May 2000.
13. X. Sun and E. Dubois, "A novel algorithm to stitch multiple views in image mosaics," *Proc. 2004 IEEE Int. Conf. Acoustics Speech Signal Processing*, pp. III–481 – III–484, May 2004.
14. M. Wu, H. Sun, and H.-Y. Shum, "Real-time stereo rendering of concentric mosaics with linear interpolation," *Proc. SPIE, Visual Communications and Image Processing 2000*, vol. 4067, pp. 23–30, 2000.
15. G. Roth, "Projective vision toolkit," *http://www.cv.iit.nrc.ca/ ∼ gerhard/PVT/(visited March, 2005).*