April 2002

VideoPlus: A Method for Capturing the Structure and Appearance of Immersive Environments

Camillo J. Taylor
University of Pennsylvania, cjtaylor@cis.upenn.edu

VideoPlus: A Method for Capturing the Structure and Appearance of Immersive Environments

Abstract
This paper presents a simple approach to capturing the appearance and structure of immersive scenes based on the imagery acquired with an omnidirectional video camera. The scheme proceeds by combining techniques from structure-from-motion with ideas from image-based rendering. An interactive photogrammetric modeling scheme is used to recover the locations of a set of salient features in the scene (points and lines) from image measurements in a small set of keyframe images. The estimates obtained from this process are then used as a basis for estimating the position and orientation of the camera at every frame in the video clip. By augmenting the video sequence with pose information, we provide the end-user with the ability to index the video sequence spatially as opposed to temporally. This allows the user to explore the immersive scene by interactively selecting the desired viewpoint and viewing direction.

Keywords
Reconstruction, immersive environments, omnidirectional video, pose estimation

VideoPlus: A Method for Capturing the Structure and Appearance of Immersive Environments

Camillo J. Taylor, *Member*, *IEEE*

Abstract—This paper presents a simple approach to capturing the appearance and structure of immersive scenes based on the imagery acquired with an omnidirectional video camera. The scheme proceeds by combining techniques from structure from motion with ideas from image based rendering. An interactive photogrammetric modeling scheme is used to recover the locations of a set of salient features in the scene (points and lines) from image measurements in a small set of keyframe images. The estimates obtained from this process are then used as a basis for estimating the position and orientation of the camera at every frame in the video clip. By augmenting the video sequence with pose information we provide the end user with the ability to index the video sequence spatially as opposed to temporally. This allows the user to explore the immersive scene by interactively selecting the desired viewpoint and viewing direction.

Index Terms—Reconstruction, immersive environments, omnidirectional video, pose estimation.

1 Introduction

This paper presents a simple approach to capturing the appearance and structure of immersive scenes based on the imagery acquired with an omnidirectional video camera [15]. The scheme proceeds by combining techniques from structure from motion with ideas from image-based rendering. An interactive photogrammetric modeling scheme is used to recover the locations of a set of salient features in the scene (points and lines) from image correspondences in a small set of keyframe images. In the current implementation, these correspondences are specified manually by allowing the user to select salient image features in the keyframes. The main contribution of this work is a novel structure from motion algorithm which is used to automatically recover the camera locations and feature positions from these measurements. The estimates obtained from this process are then used as a basis for estimating the position and orientation of the camera at every frame in the video clip.

By augmenting the video sequence with pose information, we provide the end user with the ability to index the video sequence spatially as opposed to temporally. This allows the user to explore the immersive scene by interactively selecting the desired viewpoint and viewing direction. This technology could be used to implement "Museum in a box" applications, where an individual content creator could acquire omnidirectional imagery of an environment of interest which other users could then access over the internet or on storage media (such as DVD ROMs) and explore interactively.

The proposed method offers a number of advantages. First, since the scheme is based entirely on image data, it does not require the use of a secondary positioning system. This is an advantage since it allows us to avoid the limitations, complications, and expense associated with most position sensing technologies. Global Positioning systems, for example, cannot be employed in indoor environments where the satellite signals are occluded. This is an unfortunate limitation since indoor scenes are a natural target for immersive exploration.

Odometry systems suffer from odometric drift and require fairly complex, instrumented mechanical platforms that often limit the freedom of movement of the camera. In contrast, the proposed scheme can be applied to sequences acquired with a hand held camera.

Since the proposed technique is based on image measurements, the results obtained are "pixel accurate," that is, the estimates for the camera positions and 3D feature locations are in good agreement with the observed image features by design. It would be very difficult to achieve a similar level of accuracy with an external positioning technology without significant calibration effort.

Another advantage of the technique is that it does not require the use of artificial fiducials, beacons, or other instrumentation of the environment; naturally occurring features prove to be quite adequate. This fact significantly simplifies the process of deploying the scheme in actual environments.

1.1 Related Work

The idea of using omnidirectional camera system for reconstructing environments from video imagery has been explored by Yagi et al. [24] and Ishiguro et al. [8], [7], [9]. These authors presented an omnidirectional camera system

• *The author is with the GRASP Laboratory, CIS Department, University of Pennsylvania, 3401 Walnut St., Rm. 335C, Philadelphia, PA 19104-6229. E-mail: cjtaylor@central.cis.upenn.edu.*

Manuscript received 3 Aug. 2000; revised 5 Mar. 2001; accepted 27 June 2001.
For information on obtaining reprints of this article, please send e-mail to: tvcg@computer.org, and reference IEEECS Log Number 112640.

based on a conical mirror and described how the measurements obtained from the video imagery acquired with their camera system could be combined with odometry measurements from a robot platform to construct maps of the robots environment. The techniques described in this paper do not require odometry information, which means that they can be employed on simpler platforms, like the one shown in Fig. 13, which are not equipped with odometers. It also simplifies the data acquisition process since we do not have to calibrate the relationship between the camera system and the robots odometry system.

Szeliski and Shum [18] describe an interactive approach to reconstructing scenes from panoramic imagery. The panoramic images are constructed by stitching together video frames that are acquired as a camera is spun around its center of projection. Coorg and Teller [3] describe a system which is able to automatically extract building models from a data set of panoramic images augmented with pose information, which they refer to as pose imagery.

The process of acquiring omnidirectional video imagery of an immersive environment is much simpler than the process of acquiring panoramic images. One would not really consider constructing a sequence of tightly spaced panoramic images of an environment because of the time required to acquire the imagery and stitch it together. However, this is precisely the type of data contained in an omnidirectional video sequence. By estimating the pose at every location in the sequence, the Video Plus system is able to fully exploit the range of viewpoints represented in the image sequence.

Boult [1] describes an interesting system which allows a user to experience remote environments by viewing video imagery acquired with an omnidirectional camera. During playback, the user can control the direction from which she views the scene interactively. The VideoPlus system described in this paper provides the end user with the ability to control her viewing position as well as her viewing direction. This flexibility is made possible by the fact that the video imagery is augmented with pose information which allows the user to navigate the sequence in an order that is completely different from the temporal ordering of the original sequence.

The VideoPlus system in similar in spirit to the Movie Map system described by Lippman [13] and to the QuickTime VR system developed by Chen [2] in that the end result of the analysis is a set of omnidirectional images annotated with position. The user is able to navigate through the scene by jumping from one image to another. The contribution of this work is to propose a simple and effective way of recovering the positions of the omnidirectional views from image measurements without having to place artificial fiducials in the environment or requiring a separate pose estimation system.

Shum and He [16] describe an innovative approach to generating novel views of an environment based on a set of images acquired while the camera is rotated around a set of concentric circles. Takahashi et al. [19] describe a similar system that uses as input a set of omnidirectional images captured at evenly spaced locations along a linear trajectory. Both of these systems build on the plenoptic sampling ideas described by Levoy and Hanrahan [11] and Gortler et al. [6].

Unlike these image-based rendering schemes, the Video Plus system does *not* address the problem of synthesizing images from novel vantage points. What it does provide is a mechanism for estimating the camera's trajectory so that the user can navigate through the viewpoints contained in the input video sequence. This approach offers many of the advantages of image-based rendering in that it can be used to explore arbitrarily complex environments without requiring models for the geometric and photometric properties of the surfaces in the scene. It can also be used to capture the appearance of extended environments, such as office complexes, which involve walls and other occluding surfaces that are not currently handled by plenoptic sampling schemes.

2 Method

This section describes the proposed method for estimating the trajectory of a moving camera and the locations of a set of selected scene features from image data. The basic approach, which is outlined below and in Fig. 1, is similar in spirit to the reconstruction schemes described in [23] and [4].

1. Acquire an omnidirectional video sequence in the environment of interest.
2. Select a small set of keyframes from the sequence.
3. Select a set of point and line features in the scene and indicate where they appear in the images through a simple point and click interface.
4. Apply the reconstruction algorithm, which automatically constructs estimates for the positions of the features and the locations from which the keyframes were taken.
5. Based on these estimates for keyframe and feature locations, the system then estimates the position of the camera for every frame in the video sequence.

The problem of recovering the feature and keyframe locations is posed as an optimization problem where the goal is to minimize an objective function which indicates the discrepancy between the predicted image features and the observed image features as a function of the model parameters and the camera locations. This objective function is described in more detail in Section 2.2.

2.1 Omnidirectional Imaging Model

In order to carry out this procedure, it is important to understand the relationship between the locations of features in the world and the coordinates of the corresponding image features in the omnidirectional imagery. The catadioptric camera system proposed by Nayar [15] consists of a parabolic mirror imaged by an orthographic lens. With this imaging model there is an effective single point of projection located at the focus of the parabola, as shown in Fig. 2.

Given a point with coordinates (u, v) in the omnidirectional image we can construct a vector, υ, which is aligned with the ray connecting the imaged point and the center of projection of the camera system.

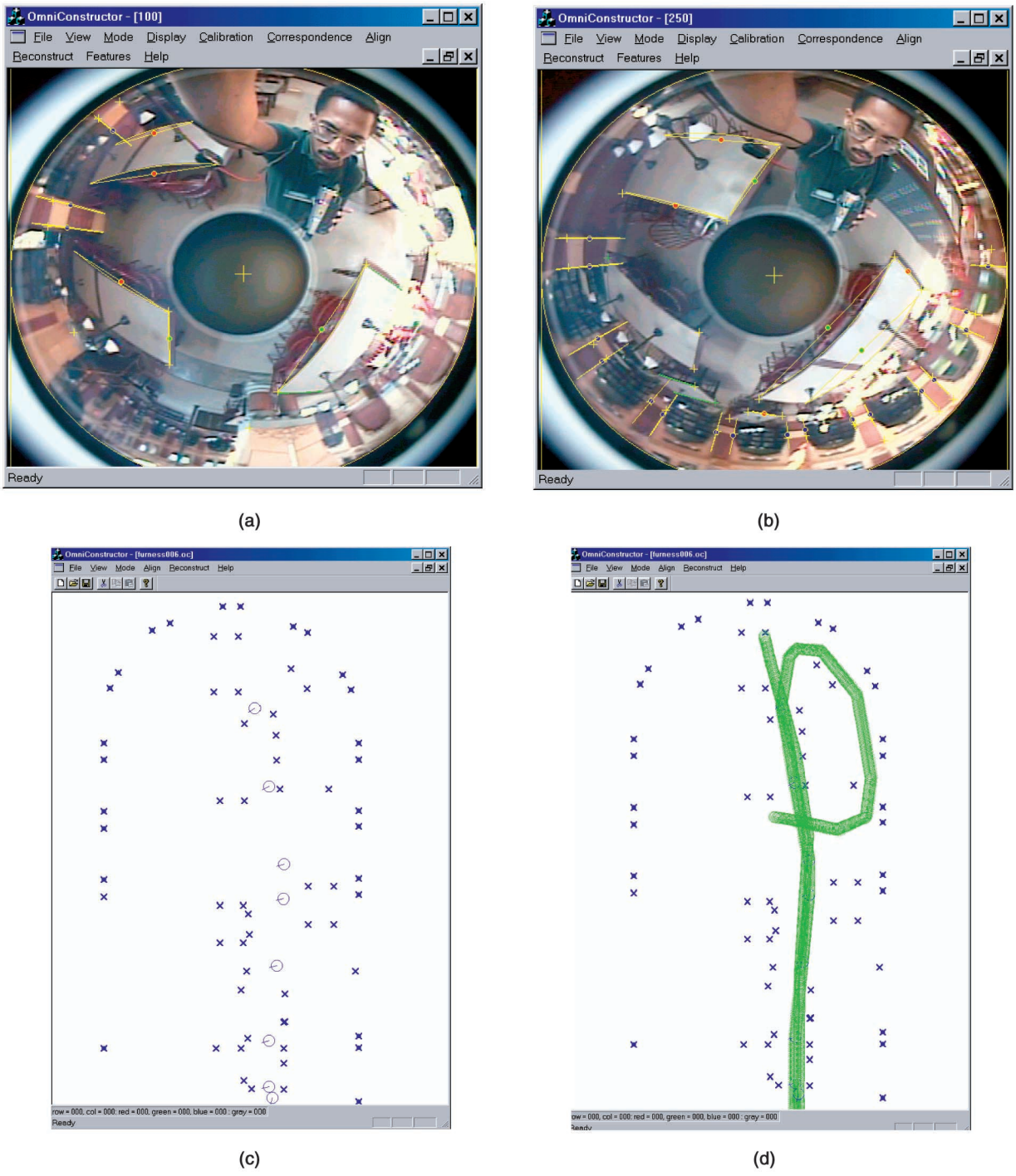

Fig. 1. The reconstruction algorithm takes as input a small set of keyframes from the video sequence. The user selects a set of point and line features in the scene and indicates where these appear in the images through a simple point and click interface; in (a) and (b) these correspondences are indicated as yellow points and arcs. Part (c) shows a 2D projection of the 3D model produced by the reconstruction algorithm from the image measurements. The dots and crosses in (c) correspond to selected features in the scene, while the circles correspond to the keyframe locations. Based on these estimates, the system then constructs estimates for the location of the camera at every frame in the video sequence (d).

$$v = \begin{pmatrix} s_x(\mathrm{u} - c_x) \\ s_y(\mathrm{v} - c_y) \\ (s_x(\mathrm{u} - c_x))^2 + (s_y(\mathrm{v} - c_y))^2 - 1 \end{pmatrix}. \quad (1)$$

This vector is expressed in terms of a coordinate frame of reference with its origin at the center of projection and with the z-axis aligned with the optical axis of the device, as shown in Fig. 2.

The calibration parameters, s_x, s_y, c_x, and c_y associated with the imagery can be obtained in a separate calibration procedure [5]. It is assumed that these calibration parameters remain constant throughout the video sequence.

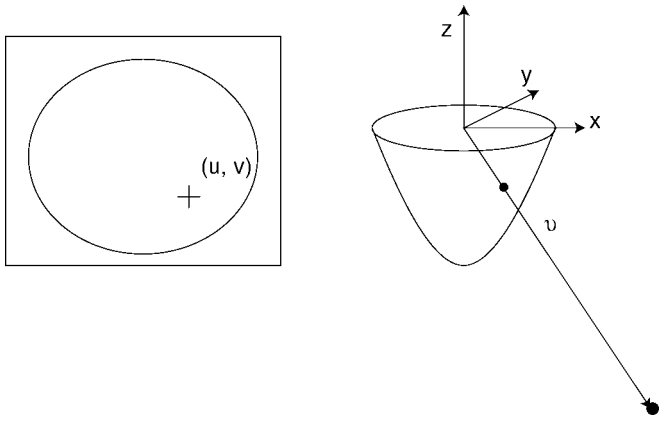

Fig. 2. The relationship between a point feature in the omnidirectional image and the ray between the center of projection and the imaged point.

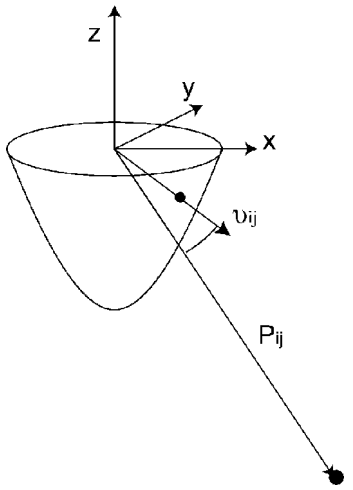

Fig. 3. Given a correspondence between a point feature in the omnidirectional image and a point feature in the model, we can construct an objective function by considering the disparity between the predicted ray between the camera center and the point feature, $\mathbf{P}_{ij}$, and the vector, v_{ij}, computed from the image measurement.

Note that, since the catadioptric camera system has a single point of projection, it is possible to resample the resulting imagery to produce "normal" perspective with arbitrary viewing directions [15]. The current system exploits this capability by providing a mechanism which allows the user to create a virtual viewpoint which she can pan and tilt interactively.

2.2 Constructing an Objective Function

The current implementation of the reconstruction system allows the user to model two types of features: point features and straight lines aligned with one of the vertical or horizontal axes of the global frame of reference. These types of features were chosen because they are particularly prevalent and salient in manmade immersive environments, but other types of features, such as lines at arbitrary orientations, could easily be included. The locations of point features can be represented in the usual manner by vectors in $\mathbb{R}^3$, (X_i, Y_i, Z_i).[1] The locations of the straight lines can be denoted with only two parameters. For example, the location of a vertical line can be specified by parameterizing the location of its intercept with the xy-plane, (X_i, Y_i), since the vertical axis corresponds to the z-axis of the global coordinate frame. Note that, for purposes of reconstruction, the lines are considered to have infinite length, so no attempt is made to represent their endpoints.

The position and orientation of the camera with respect to the world frame of reference during frame j of the sequence is captured by two parameters, a rotation $R_j \in SO(3)$ and a translation $\mathbf{T}_j \in \mathbb{R}^3$. This means that, given the coordinates of a point in the global coordinate frame, $\mathbf{P}_{iw} \in \mathbb{R}^3$, we can compute its coordinates with respect to camera frame j, $\mathbf{P}_{ij}$, from the following expression.

$$\mathbf{P}_{ij} = R_j(\mathbf{P}_{iw} - \mathbf{T}_j). \tag{2}$$

The reconstruction program takes as input a set of correspondences between features in the omnidirectional imagery and features in the model. For correspondences between point features in the image and point features in the model we can construct an expression which measures the discrepancy between the predicted projection of the point and the vector obtained from the image measurement, v_{ij}, where $\mathbf{P}_{ij}$ is computed from (2).

$$\|(v_{ij} \times \mathbf{P}_{ij})\|^2 / (\|\mathbf{P}_{ij}\|^2 \|v_{ij}\|^2). \tag{3}$$

This expression yields a result equivalent to the square of the sine of the angle between the two vectors, v_{ij} and $\mathbf{P}_{ij}$, shown in Fig. 3.

For correspondences between point features in the image and line features in the model, we consider the plane containing the line and the center of projection of the image. The normal to this plane, $\mathbf{m}_{ij}$ can be computed from the following expression:

$$\mathbf{m}_{ij} = R_j(\mathbf{v}_i \times (\mathbf{d}_i - \mathbf{T}_j)), \tag{4}$$

where the vector $\mathbf{v}_i$ denotes the direction of the line in space and the vector $\mathbf{d}_i$ denotes an arbitrary point on the line. As an example, for vertical lines, the vector $\mathbf{v}_i$ will be aligned with the z axis $(0, 0, 1)^T$ and the vector $\mathbf{d}_i$ will have the form $(X_i, Y_i, 0)^T$.

The following expression measures the extent to which the vector obtained from the point feature in the omnidirectional imagery, v_{ij}, deviates from the plane defined by the vector $\mathbf{m}_{ij}$ as shown in Fig. 4:

$$(\mathbf{m}^T v_{ij})^2 / (\|\mathbf{m}_{ij}\|^2 \|v_{ij}\|^2). \tag{5}$$

A global objective function is constructed by considering all of the correspondences in the data set and summing the resulting expressions together. Estimates for the structure of the scene and the locations of the cameras are obtained by minimizing this objective function with respect to the unknown parameters, R_j, $\mathbf{T}_j$, X_i, Y_i, and Z_i. This minimization is carried out using a variant of the Newton-Raphson method [22], [23], [10]. For the scenes described in Section 3, which involve on the order of 20 keyframes and 50 model features, the optimization procedure requires less

1. The subscript i serves to remind us that these parameters describe the position of the ith feature in the model.

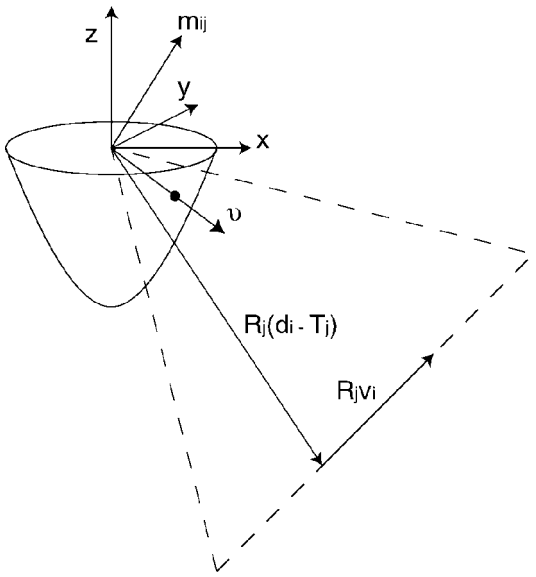

Fig. 4. Given a correspondence between a point feature in the omnidirectional image and a line feature in the model, we can construct an objective function by considering the disparity between the predicted normal vector to the plane containing the center of projection and the model line, $\mathbf{m}_{ij}$, and the vector, v_{ij}, computed from the image measurement.

than a minute of compute time on a Pentium-based laptop computer.

As with any structure from motion algorithm, the proposed scheme recovers the geometry of the scene up to an unknown scale factor since all of the dimensions can be multiplied by an arbitrary positive number without affecting the image measurements on which the reconstruction is based.

Note that it is *not* necessary to have a correspondence for every scene feature in every keyframe. This is particularly advantageous in complicated indoor environments where features are often occluded by intervening structures.

2.3 Obtaining Initial Estimates

An initial estimate for the orientation of the camera frames, R_j, can be obtained by considering the lines in the scene with known orientation, such as lines parallel to the x, y, or z axes of the environment. If v_1 and v_2 represent the vectors corresponding to two points along the projection of a line in the image plane, then the normal to the plane between them in the cameras frame of reference can be computed as follows: $\mathbf{n} = v_1 \times v_2$. If R_j represents the rotation of the camera frame and $\mathbf{v}$ represents the direction of the line in world coordinates, then the following objective function represents the fact that the normal to the plane should be perpendicular to the direction of the line in the coordinates of the camera frame:

$$(\mathbf{n}^T R_j \mathbf{v})^2. \tag{6}$$

An objective function can be created by considering all such lines in an image and summing these penalty terms. The obvious advantage of this expression is that the only unknown parameter is the camera rotation, R_j, which means that we can minimize the expression with respect to this parameter in isolation to obtain an initial estimate for the camera orientation.

2.4 Expressing Constraints

The current implementation of the reconstruction system also allows the user to specify constraints that relate the features in the model. For example, the user would be able to specify that two or more features share the same z-coordinate, which would force them to lie on the same horizontal plane. This constraint is maintained by reparameterizing the reconstruction problem such that the z-coordinates of the points in question all refer to the same variable in the parameter vector.

The ability to specify these relationships is particularly useful in indoor environments since it allows the user to exploit common constraints among features such as two features belonging to the same wall or multiple features lying on a ground plane. These constraints reduce the number of free parameters that the system must recover and improve the coherence of the model when the camera moves large distances in the world.

2.5 Guidelines for Selecting Features

The reconstruction procedure described in the previous sections is fairly robust to the choice of features used. There are, however, some guidelines that should be observed. The most important thing is to choose a set of features that are widely distributed throughout the scene as opposed to being clustered in a small section of the image. It is also a good idea to choose subsets of features that are linked by some constraint, such as coplanarity, since these constraints can be effectively exploited by the reconstruction system.

For most nontrivial scenes, it is usually impossible to find features that are visible in all of the images in the sequence, but one should strive to choose features that are visible in at least two or three of the keyframes used for reconstruction. Another thing to keep in mind is that extended features like lines can often be localized more accurately in the imagery than point features.

2.6 Estimating the Camera Trajectory

Once the locations of a set of model features have been reconstructed using the image measurements obtained from a set of keyframes in the sequence, these features can then be used as fiducials to recover the pose of the camera at other frames in the sequence.

For example, if frame number 1,000 and frame number 1,500 were used as keyframes in the reconstruction process, then we know where a subset of the model features appears in these frames. Correspondences between features in the intervening images and features in the model can be obtained by applying applying standard feature tracking algorithms to the data set. The current system employs a variant of the Lucas and Kanade [14] algorithm to localize and track feature points through intervening frames.

Based on these correspondences, the pose of the camera during these intermediate frames can be estimated by simply minimizing the objective function described previously with respect to the pose parameters of the camera. The locations of the feature points are held constant during this pose estimation step. Initial estimates for the camera pose can be obtained from the estimates for the locations of

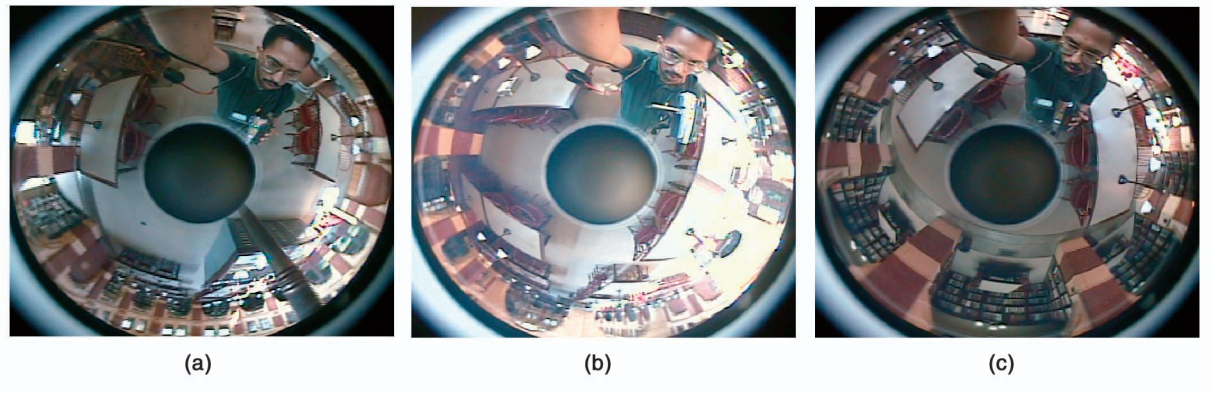

Fig. 5. Three images taken from a video sequence obtained as the camera is moved through the library.

Fig. 6. Images of the Fine Arts Library at the University of Pennsylvania. The building was designed by Frank Furness in 1891 and remains one of the most distinctive and most photographed buildings on campus.

the keyframes that were produced during the reconstruction process.

Another approach to estimating the pose of the camera during the intervening frames is to simply interpolate the pose parameters through the frames of the subsequence. That is, given that the camera pose in frames 1,000 and 1,500 is known, we could simply estimate the roll, pitch, and yaw angles of the intervening frames along with the translational position by interpolating these parameter values linearly. This approach is most appropriate in situations where the camera is moving with an approximately constant translational and angular velocity between keyframes.

2.7 Interactive Exploration of the Video Sequence

Once the video sequence has been fully annotated with camera pose information, the user is able to index the data set *spatially* as well as temporally. In the current implementation, the user is able to navigate through an immersive environment, such as the office complex shown in Fig. 5, in a natural manner by panning and tilting his virtual viewpoint and moving forward and backward. As the user changes the location of her viewpoint, the system simply selects the closest view in the omnidirectional video sequence and generates an image in the appropriate viewing direction.

The current implementation also allows the user to generate movies by specifying camera trajectories that pass through the original video sequence. The system can then automatically generate frames corresponding to the desired trajectory by resampling the original imagery. In this way, the user can reshoot the scene with a temporal order that differs from the ordering of the original video sequence.

3 Results

In order to illustrate what can be achieved with the proposed techniques, we present results obtained from three different immersive environments.

3.1 Furness Library

Fig. 5 shows three images taken from a video sequence acquired in the Fine Arts Library at the University of Pennsylvania. This building was designed by Frank Furness in 1891 and refurbished on its centenary in 1991; images of the interior and exterior of the building are shown in Fig. 6.

The reconstruction of this environment was carried out using approximately 100 model features viewed in nine frames of the video sequence. Fig. 7a shows a floor plan view of the resulting reconstruction. The reconstructed feature locations were then used as fiducials to recover the position of 15 other frames in the sequence. Pose interpolation was employed to estimate the position and orientation of the camera during intervening frames. Fig. 7b shows the resulting estimates for the camera position during the entire sequence. The original video sequence was 55 seconds long and consisted of 550 frames. During the sequence, the camera traveled a distance of approximately 150 feet. Fig. 8 shows viewpoints generated by the system as the user conducts a virtual tour of this environment.

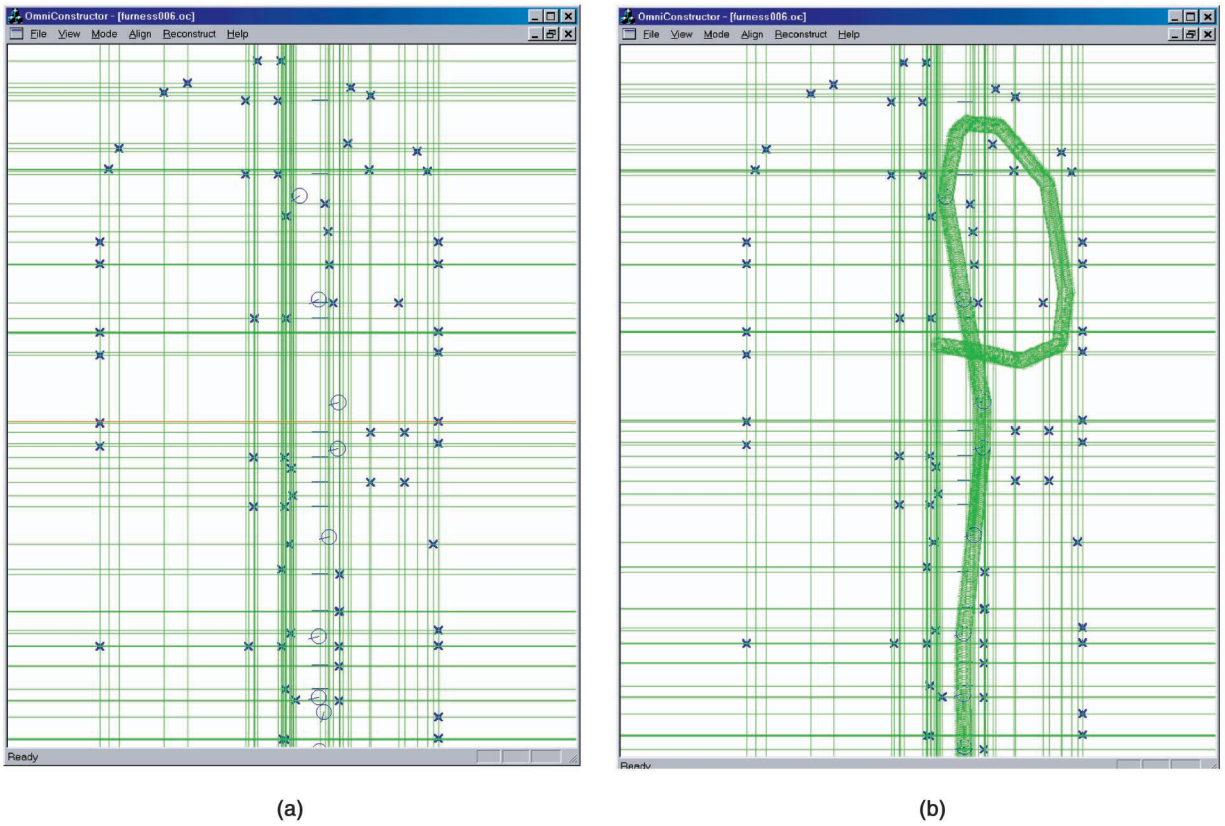

Fig. 7. (a) A floor plan view of the library showing the locations of the features recovered from nine keyframes in the video sequence. The circles correspond to the recovered camera positions, while the dots and crosses correspond to line and point features. (b) Based on these fiducials, the system is able to estimate the location of the camera for all the intervening frames.

Fig. 8. Views generated by the system as the user conducts a virtual tour of the library.

3.2 GRASP Laboratory

Fig. 9 shows three images taken from a video sequence acquired in the GRASP laboratory at the University of Pennsylvania; snapshots of the lab are shown in Fig. 10. In this case, the video imagery was obtained in a sequence of short segments as the camera was moved through various sections of the laboratory. The entire video sequence was 154 seconds long and consisted of 4,646 frames. The approximate dimensions of the region of the laboratory explored are 36 feet by 56 feet and the camera moved over 250 feet during the exploration. The reconstruction of this scene was carried out using approximately 50 model features viewed in 16 images of the sequence. The resulting model is shown in Fig. 11a; Fig. 11b shows the result of applying pose estimation and interpolation to the rest of the video sequence. Fig. 12 shows some samples of images created as the user explores this environment interactively. Notice that the user can freely enter and exit various rooms and alcoves in the laboratory.

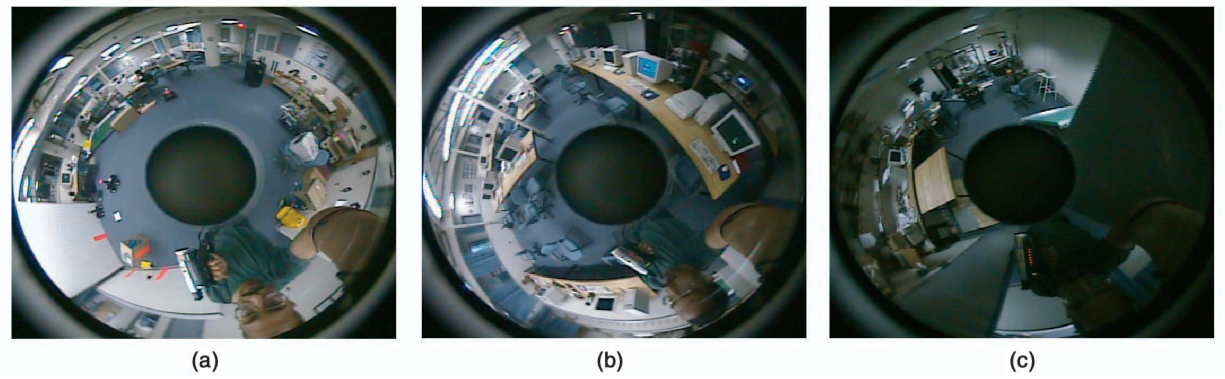

Fig. 9. Three images taken from a video sequence obtained as the camera is moved through the GRASP laboratory.

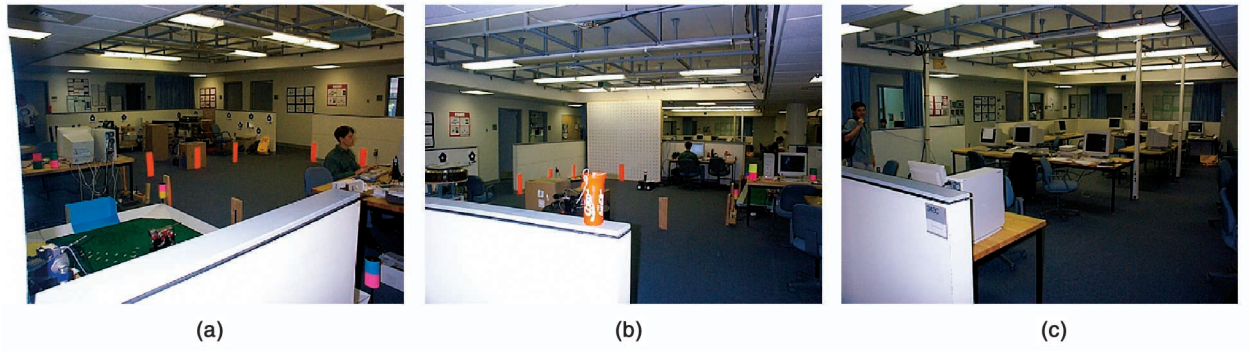

Fig. 10. Images of the GRASP laboratory at the University of Pennsylvania.

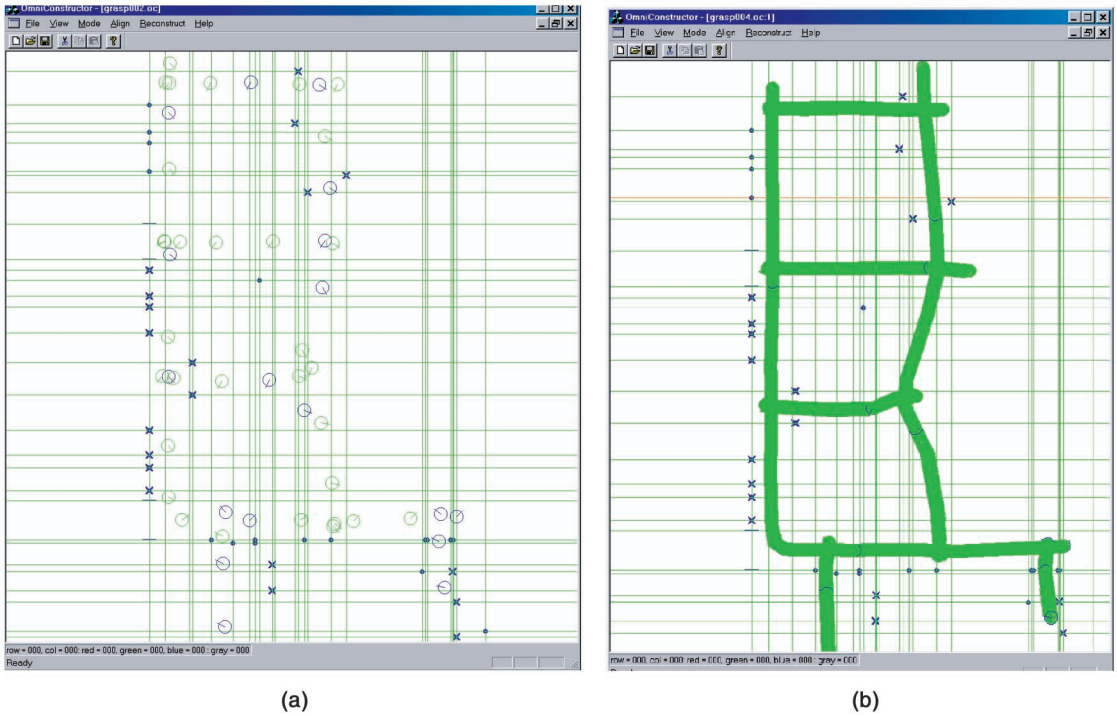

Fig. 11. (a) A floor plan view of the laboratory showing the locations of the features recovered from 17 keyframes in the video sequence. The circles correspond to the recovered camera positions, while the dots and crosses correspond to line and point features. (b) Based on these fiducials, the system is able to estimate the location of the camera for all the intervening frames. Notice that, during the exploration, the camera is moved into two side rooms that are accessed from the corridor surrounding the laboratory; these are represented by the two excursions at the bottom of this figure.

In order to gauge the accuracy of the reconstruction procedure, we compared 17 of the scene dimensions recovered by the program to measurements obtained with a ruler. Recall that the structure from motion algorithm provides reconstructions up to an unknown scale factor. Once this scale factor was accounted for, the mean disparity

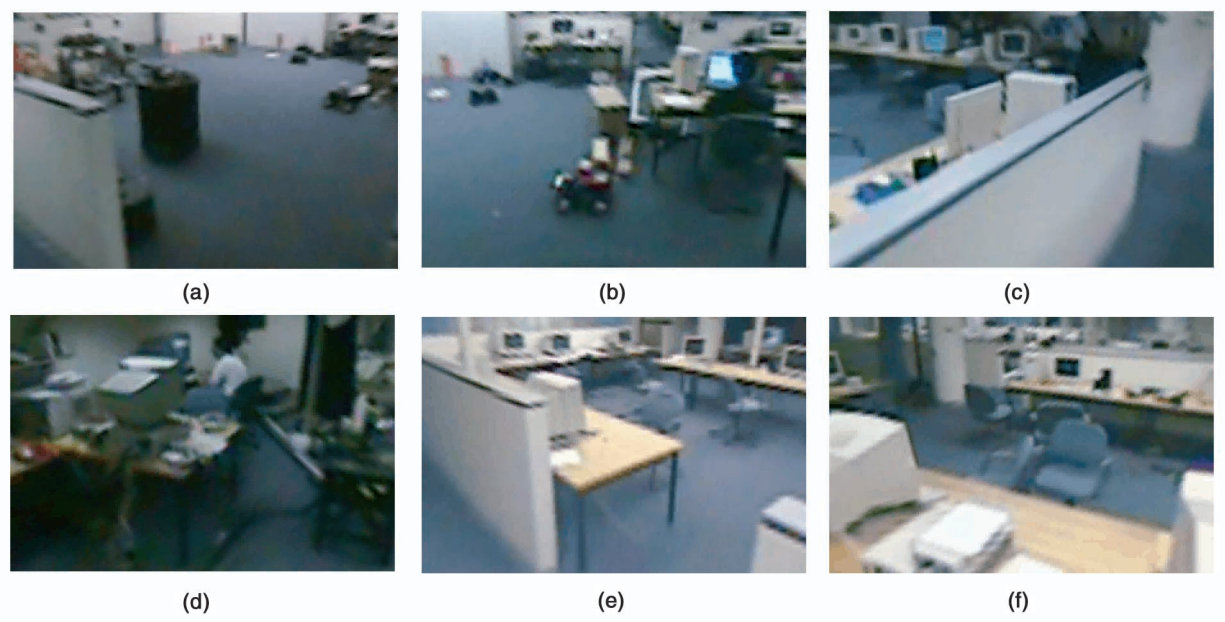

Fig. 12. Views generated by the system as the user conducts a virtual tour of the library.

between the estimated dimensions and measured dimensions was 3.5 inches. Expressed as a percentage, the estimated dimensions agreed with the measured dimensions to within 1.52 percent on average.

3.3 Hospital Interior

Fig. 15 shows the results of applying the reconstruction procedure to 14 images acquired from a sequence taken inside an abandoned hospital building. This figure demonstrates the capability of constructing polyhedral models from the recovered model features.

Using the procedure outlined above, we were able to reconstruct the model shown in Fig. 15 from 14 images taken from a video sequence of an indoor scene.

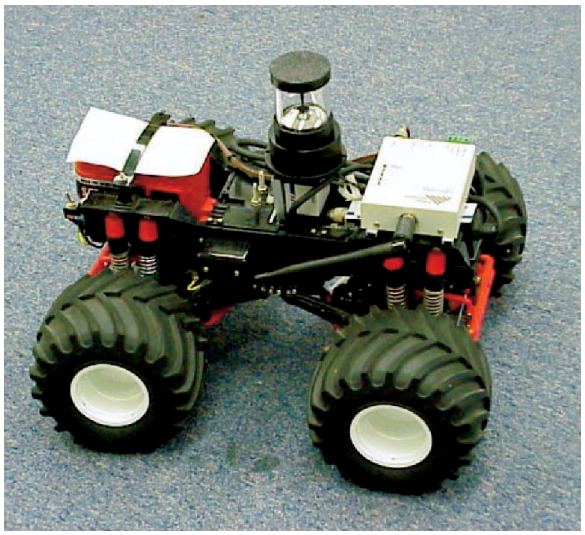

Fig. 13. Mobile platform equipped with an omnidirectional camera system that was used to acquire video imagery of an indoor environment.

The polyhedral model was constructed by manually attaching surfaces to the reconstructed features. Texture maps for these surfaces were obtained by sampling the original imagery.

The fact that the reconstruction process can be carried out entirely from the video sequence simplifies the process of data collection. Fig. 13 shows a mobile platform outfitted with an omnidirectional camera system produced by Remote Reality inc.. This system was used to acquire the imagery that was used to construct the model shown in Fig. 15. Note that the only sensor carried by this robot is the omnidirectional camera; it does not have any odometry or range sensors. During the data collection process the system was piloted by a remote operator using an RC link.

The video data that was used to construct the models shown in Figs. 7 and 11 was collected with a handheld omnidirectional camera system, as shown in Fig. 16. In both cases, the video data was captured on a Sony Digital camcorder and transferred to a PC for processing using an IEEE 1394 Firewire link. The images were digitized at a resolution of 720×480 at 24 bits per pixel. During the data collection process, the system was piloted by a remote operator using an RC link. Samples of the imagery acquired with the robot are shown in Fig. 14.

4 Conclusions

This paper presents an effective scheme for estimating the trajectory of a moving camera and the locations of a selected set of point and line features from image correspondences in an omnidirectional image sequence. An important practical advantage of using omnidirectional imagery in this application is that the 3D structure can be recovered from a smaller number of images since the features of interest are more likely to remain in view as the camera moves from one location to another.

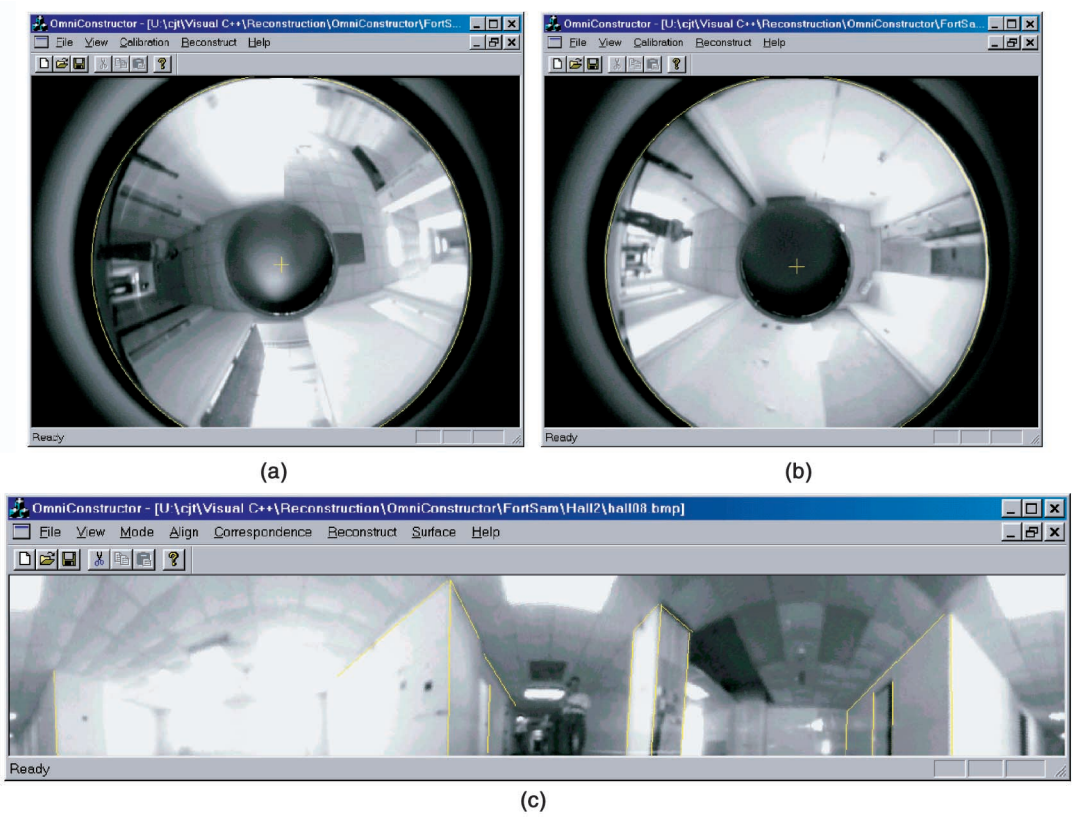

Fig. 14. Two of the omnidirectional images from a set of 14 keyframes are shown in (a) and (b). A panoramic version of another keyframe is shown in (c).

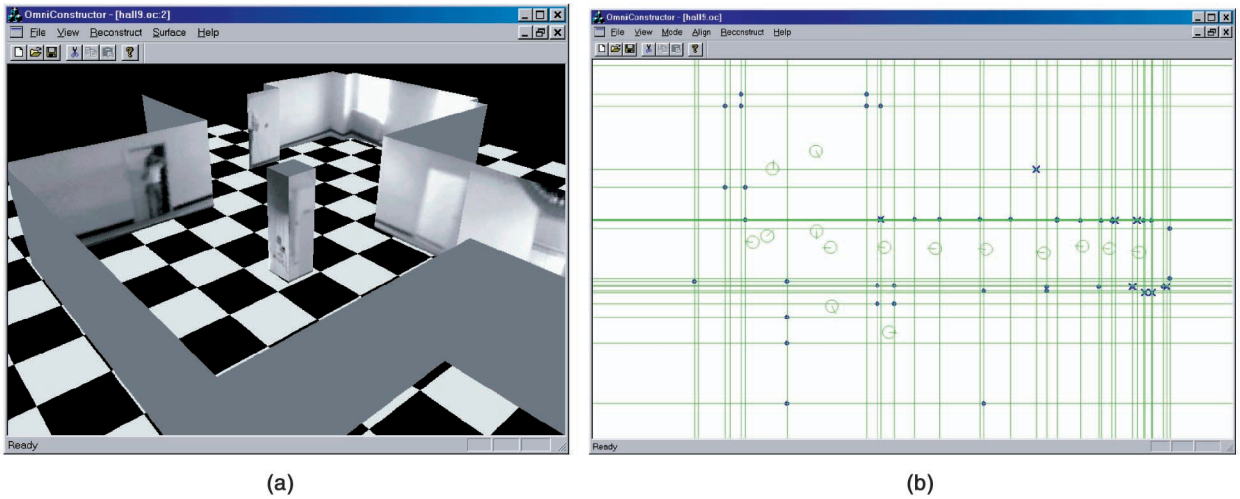

Fig. 15. (a) 3D model of the environment constructed from the data set shown in Fig. 14. (b) Floor plan view showing the estimated location of all the images and an overhead view of the feature locations. The circles correspond to the recovered camera positions, while the dots and crosses correspond to vertical line and point features.

By augmenting the video sequence with pose information, we provide the end user with the capability of indexing the video sequence spatially as opposed to temporally. This means that the user can explore the image sequence in ways that were not envisioned when the sequence was initially collected.

The cost of augmenting the video sequence with pose information is very slight since it only involves storing six numbers per frame. The hardware requirements of the proposed scheme are also quite modest since the reconstruction is performed entirely from the image data. It does not involve a specific camera trajectory or a separate sensor for measuring the camera position. As such, the method is particularly appropriate for immersive manmade structures where GPS data is often unavailable.

We envision that this system could be used to acquire representations of immersive environments, like museums, that users could then explore interactively. It might also be

Fig. 16. (a) The video imagery used to produce the reconstructions of the library and the laboratory environments was acquired using a handheld omnidirectional camera system (b) The equipment used to acquire the data.

appropriate for acquiring immersive backgrounds for video games or training simulators.

4.1 Future Work

Several projects currently underway seek to improve upon various aspects of the scheme presented in this manuscript. One such effort aims at improving the accuracy of the method for estimating the camera trajectory by incorporating measurements from a set of accelerometers mounted on the omnidirectional video camera.

The method used to generate views of an environment during a walkthrough is also a target for improvement. Currently, the system simply selects the omnidirectional image that is closest to the users desired viewpoint and generates an image with the appropriate viewing direction. The obvious limitation of this approach is that the viewing position is restricted to locations which were imaged in the original video sequence.

One approach to generating novel views that is currently being pursued involves finding correspondences between salient image features in neighboring omnidirectional images in the original sequence. These correspondences can then be used to construct warping functions which map pixels from the original images to the virtual viewpoint [12].

The success of any view generation technique will depend upon having a set of images taken from a sufficiently representative set of viewpoints. A better understanding of how to go about capturing such a data set taking into account the structure of the scene and the viewpoints that are likely to be of most interest is needed. The ultimate goal would be to produce a system where the user could arbitrarily select the desired viewpoint and viewing direction so as to explore the environment in an unconstrained manner.

The largest drawback to using omnidirectional video imagery is the reduced image resolution. This effect can be mitigated by employing higher resolution video cameras. One of the trade-offs that is currently being explored is the possibility of acquiring higher resolution imagery at a lower frame rate. This would allow us to produce sharper images of the scene, but would either slow down the data acquisition process or require better view interpolation strategies.

Acknowledgments

This material is based upon work supported by the US National Science Foundation under a CAREER Grant (Grant No. 9875867).

References

[1] T.E. Boult, "Remote Reality via Omni-Directional Imaging," *Conference Abstracts and Applications: SIGGRAPH '98, Computer Graphics,* S. Grisson, J. McAndless, O. Ahmad, C. Stapleton, A. Newton, C. Pearce, R. Ulyate, and R. Parent, eds., pp. 253–253, July 1998.

[2] S.E. Chen, "Quicktime VR—An Image-Based Approach to Virtual Environment Navigation," *Proc. SIGGRAPH,* pp. 29-38, Aug. 1995.

[3] S. Coorg and S. Teller, "Automatic Extraction of Textured Vertical Facades from Pose Imagery," technical report, MIT Computer Graphics Group, Jan. 1998.

[4] P.E. Debevec, C.J. Taylor, and J. Malik, "Modeling and Rendering Architecture from Photographs: A Hybrid Geometry- and Image-Based Approach," *Proc. SIGGRAPH '96,* pp. 11-21, Aug. 1996.

[5] C. Geyer and K. Daniilidis, "Catadioptric Camera Calibration," *Proc. Int'l Conf. Computer Vision,* pp. 398-404, 1999.

[6] S.J. Gortler, R. Grzeszczuk, R. Szeliski, and M. Cohen, "The Lumigraph," *Proc. SIGGRAPH '96,* pp. 31-43, Aug. 1996.

[7] H. Ishiguro, T. Maeda, T. Miyashita, and S. Tsuji, "A Strategy for Acquiring an Environmental Model with Panoramic Sensing by a Mobile Robot," *Proc. IEEE Int'l Conf. Robotics and Automation,* pp. 724-729, 1994.

[8] H. Ishiguro, K. Ueda, and S. Tsuji, "Omnidirectional Visual Information for Navigating a Mobile Robot," *Proc. IEEE Int'l Conf. Robotics and Automation,* pp. 799-804, 1993.

[9] H. Ishiguro, M. Yamamoto, and S. Tsuji, "Omni-Directional Stereo," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 14, no. 2, pp. 257-262, Feb. 1992.

[10] J.E. Dennis Jr. and R.B. Schnabel, *Numerical Methods for Unconstrained Optimization and Nonlinear Equations.* Prentice Hall, 1983.

[11] M. Levoy and P. Hanrahan, "Light Field Rendering," *Proc. SIGGRAPH '96,* pp. 31-43, Aug. 1996

[12] M. Lhuillier and L. Quan, "Image Interpolation by Joint View Triangulation," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* vol. 2, pp. 139-145, June 1999.

[13] A. Lippman, "Movie Maps: An Application of the Optical Video-Disc to Computer Graphics," *Proc. SIGGRAPH,* pp. 32-42, July 1980.

[14] B.D. Lucas and T. Kanade, "An Iterative Image Registration Technique with an Application to Stereo Vision," *Proc. Seventh Int'l Joint Conf. Artificial Intelligence,* 1981.

[15] S. Nayar, "Catadioptric Omnidirectional Camera," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* 1997.

[16] H.-Y. Shum and L.-W. He, "Rendering with Concentric Mosaics," *Proc. SIGGRAPH,* pp. 299-306, Aug. 1999.

[17] T. Svoboda, T. Pajdla, and V. Hlavac, "Epipolar Geometry for Panoramic Cameras," *Proc. European Conf. Computer Vision,* pp. 218-232, 1998.

[18] R. Szeliski and H.Y. Shum, "Creating Full View Panoramic Image Mosaics and Texture-Mapped Models," *Proc. SIGGRAPH,* pp. 251-258, Aug. 1997.

[19] T. Takahashi, H. Kawasaki, K. Ikeuchi, and M. Sakauchi, "Arbitrary View Position and Direction Rendering for Large-Scale Scenes," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* pp. 296-303, 2000.

[20] C.J. Taylor, "Video Plus," *Proc. IEEE Workshop Omnidirectional Vision,* K. Daniilidis, ed., pp. 3-11, June 2000.

[21] C.J. Taylor, "Video Plus: A Method for Capturing the Structure and Appearance of Immersive Environments," *Proc. Second European Workshop 3D Structure from Multiple Images of Large-Scale Environments,* M. Pollefeys, L. van Gool, A. Zisserman, and A. Fitzgibbon, eds., pp. 187-204, July 2000.

[22] C.J. Taylor and D.J. Kriegman, "Minimization on the Lie Group SO(3) and Related Manifolds," Technical Report 9405, Center for Systems Science, Dept. of Electrical Eng., Yale Univ., New Haven, Conn., Apr. 1994.

[23] C.J. Taylor and D.J. Kriegman, "Structure and Motion from Line Segments in Multiple Images," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 17, no. 11, Nov. 1995.

[24] Y. Yagi, S. Kawato, and S. Tsuji, "Real-Time Omnidirectional Image Sensor (copis) for Vision-Guided Navigation," *IEEE J. Robotics and Automation,* vol. 10, no. 1, pp. 11-21, Feb. 1994.

Camillo J. Taylor received the AB degree in electrical computer and systems engineering from Harvard College in 1988. He received the MS and PhD degrees from Yale University in 1990 and 1994, respectively. He is currently an assistant professor in the Computer Information Science Department at the University of Pennsylvania, where he has been since September 1997. He has carried out research on several problems in computer vision and robotics including: reconstruction of 3D models from images, automatic control of vision-guided motor vehicles, mobile robot navigation, and multirobot coordination. He was the Jamaica Scholar in 1984, a member of the Harvard chapter of Phi Beta Kappa, and held a Harvard College Scholarship from 1986-1988. From 1994 to 1997, Dr. Taylor was a postdoctoral researcher and lecturer with the Department of Electrical Engineering and Computer Science at the University of California at Berkeley. He received a US National Science Foundation CAREER award in 1998 and the Lindback Minority Junior Faculty Award in 2001. He is a member of the IEEE and the IEEE Computer Society.