3-D Scene Data Recovery using Omnidirectional Multibaseline Stereo

Sing Bing Kang

Digital Equipment Corporation
Cambridge Research Lab
One Kendall Square, Bldg. 700
Cambridge, MA 02139
sbk@crl.dec.com

Richard Szeliski

Microsoft Corporation
One Microsoft Way
Redmond, WA 98052-6399
szeliski@microsoft.com

Abstract

A traditional approach to extracting geometric information from a large scene is to compute multiple 3-D depth maps from stereo pairs or direct range finders, and then to merge the 3-D data. However, the resulting merged depth maps may be subject to merging errors if the relative poses between depth maps are not known exactly. In addition, the 3-D data may also have to be resampled before merging, which adds additional complexity and potential sources of errors.

This paper provides a means of directly extracting 3-D data covering a very wide field of view, thus by-passing the need for numerous depth map merging. In our work, cylindrical images are first composited from sequences of images taken while the camera is rotated 360° about a vertical axis. By taking such image panoramas at different camera locations, we can recover 3-D data of the scene using a set of simple techniques: feature tracking, an 8-point structure from motion algorithm, and multibaseline stereo. We also investigate the effect of median filtering on the recovered 3-D point distributions, and show the results of our approach applied to both synthetic and real scenes.

1 Introduction

A traditional approach to extracting geometric information from a large scene is to compute multiple (possibly numerous) 3-D depth maps from stereo pairs, and then to merge the 3-D data [3, 5, 16, 19]. This is not only computationally intensive, but the resulting merged depth maps may be subject to merging errors, especially if the relative poses between depth maps are not known exactly. The 3-D data may also have to be resampled before merging, which adds additional complexity and potential sources of errors.

This paper provides a means of directly extracting 3-D data covering a 360° horizontal field of view, thus bypassing the need for numerous depth map merging. Cylindrical images are first composed from sequences of images taken while the camera is rotated 360° about a vertical axis. By taking such image panoramas at different camera locations, we can recover 3-D data of the scene using a set of simple techniques: feature tracking, an 8-point structure from motion algorithm, and multibaseline stereo.

There are several advantages to this approach. First, the cylindrical image mosaics can be built quite accurately, since the camera motion is very restricted. Second, the relative pose of the various camera locations can be determined with much greater accuracy than with regular structure from motion applied to images with narrower fields of view. Third, there is no need to build or purchase a specialized stereo camera whose calibration may be sensitive to drift over time—any conventional video camera on a tripod will suffice. Our approach can be used to construct models of building interiors, both for virtual reality applications (games, home sales, architectural remodeling), and for robotics applications (navigation).

2 Relevant work

There is a significant body of work on range image recovery using stereo (the most recent comprehensive survey is given in [2]). Most work on stereo uses images with limited fields of view. One of the earliest work to use panoramic images is the omnidirectional stereo system of Ishigura [6], which uses two panoramic views. One of the disadvantages of this method is the slow data accumulation, which takes about 10 mins.

Murray [14] generalizes Ishigura *et al.*'s approach by using all the vertical slits of the image (except in the paper, he uses a single image raster). This would be equivalent to structure from known motion or motion stereo. The analysis involved in this work is similar to Bolles *et al.*'s [1] spatio-temporal epipolar analysis.

Another related work is that of plenoptic modeling [13]. The idea is to compose rotated camera views into panoramas, and based on two cylindrical panoramas, project disparity values between these locations to a given viewing position. However, there is no explicit 3-D reconstruction.

Our approach is similar to that of [13] in that we compose rotated camera views to panoramas as well. However, we are going a step further in reconstructing 3-D feature points and modeling the scene based upon the recovered points. We use multiple panoramas for more accurate 3-D reconstruction.

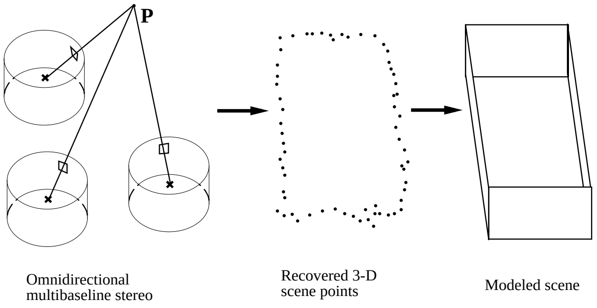

P
Omnidirectional multibaseline stereo
Recovered 3-D scene points
Modeled scene

Figure 1: Generating scene model from multiple 360° panoramic views.

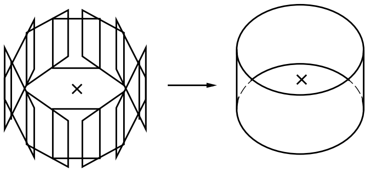


Figure 2: Compositing multiple rotated camera views into a panorama. The '×' mark indicate the locations of the camera optical and rotation center.

3 Basic idea

Our ultimate goal is to generate a photorealistic model to be used in a variety of scenarios. We are interested in providing a simple means of generating such models. In our case, we use a workstation with framegrabber (real-time image digitizer) and a commercially available 8-mm camcorder.

Our approach is straightforward: at each camera location in the scene, capture sequences of images while rotating the camera about the vertical axis passing through the camera optical center. Compose each set of images to produce panoramas at each location. Use stereo to extract 3-D data from the scene. Finally, model the scene using these 3-D data and render it with the texture provided by the input 2-D images. This approach is summarized in Figure 1.

Using multiple camera locations in stereo analysis significantly reduces the number of ambiguous matches and also has the effect of reducing errors by averaging [15, 9]. This is especially important for images with very wide fields of view, because depth recovery is unreliable near the epipoles, where the looming effect takes place, resulting in very poor depth cues.

4 Extraction of panoramic images

A panoramic image is created by composing a series of rotated camera image images, as shown in Figure 2. In order to create this panoramic image, we first have to ensure that the camera is rotating about an axis passing through its optical center, i.e., we must eliminate motion parallax when panning the camera around. To achieve this, we manually adjust the position of camera relative to an X-Y precision stage (mounted on the tripod) such that the motion parallax effect disappears when the camera is rotated back and forth about the vertical axis [20].

Figure 3: Image sequence of an office.

Prior to image capture of the scene, we calibrate the camera to compute its intrinsic camera parameters (specifically its focal length f, aspect ratio r, and radial distortion coefficient κ). The camera is calibrated by taking multiple snapshots of a planar dot pattern grid with known depth separation between successive snapshots. We use an iterative least-squares algorithm (Levenberg-Marquardt) to estimate camera intrinsic and extrinsic parameters (except for κ) [23]. κ is determined using 1-D search (Brent's parabolic interpolation in 1-D [17]) with the least-squares algorithm as the black box.

The steps involved in extracting a panoramic scene are as follow:

- At each camera location, capture sequence while panning camera around 360°.
- Using the intrinsic camera parameters, correct the image sequence for r, the aspect ratio, and κ, the radial distortion coefficient.
- Convert the (r, κ)-corrected 2-D flat image sequence to cylindrical coordinates, with the focal length f as its cross-sectional radius [24]. An example of a sequence of corrected images (of an office) is shown in Figure 3.
- Compose the images to yield the desired panorama [21]. The relative displacement of one frame to the next is coarsely determined by using phase correlation [11]. Subsequently, the image translation is refined using local image registration by directly comparing the overlapped regions between the two images [21].
- Correct for slight errors in the resulting length (which in theory equals $2\pi f$) by propagating residual displacement error equally across all images and recomposing. The error in length is usually within a percent of the expected length.

An example of a panoramic image created from the office scene in Figure 3 is shown in Figure 4.

5 Recovery of epipolar geometry

In order to extract 3-D data from a given set of panoramic images, we have to first know the relative positions of the

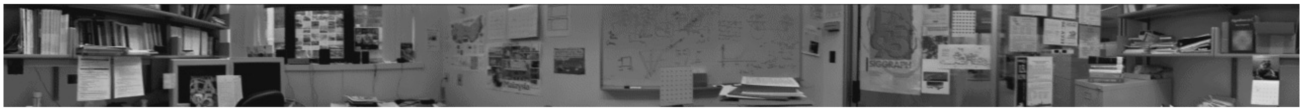


Figure 4: Panorama of office scene after composing.

camera corresponding to the panoramic images. For a calibrated camera, this is equivalent to determining the epipolar geometry between a reference panoramic image and every other panoramic image.

5.1 Using the 8-point algorithm

We use the 8-point algorithm [12, 4] to extract what is called the *essential matrix*, which yields both the relative camera placement and epipolar geometry. This is done pairwise, namely between a reference panoramic image and another panoramic image. There are, however, four possible solutions [4]. The solution that yields the most *positive* projections (i.e., projections away from the camera optical centers) is chosen.

After recovering the essential matrix, we can then determine both the orientation and translation (up to a scale). In our work, the scale is determined from the measured distance between camera positions, though this is not critical [8].

If the number of input points is small and not well distributed in the image, the output of algorithm is sensitive to noise. On the other hand, it turns out that *normalizing* the 3-D point location vector on the cylindrical image reduces sensitivity of the 8-point algorithm to noise. This is similar in spirit to Hartley's application of isotropic scaling [4] prior to using the 8-point algorithm. The 3-D cylindrical points are normalized according to the relation

$$\mathbf{u} = (f \sin\theta, y, f\cos\theta) \rightarrow \hat{\mathbf{u}} = \mathbf{u}/|\mathbf{u}| \qquad (1)$$

5.2 Tracking features for 8-point algorithm

The 8-point algorithm assumes that feature point correspondences are available. Feature tracking is difficult since purely local tracking fails because the displacement can be large (of the order of about 100 pixels, in the direction of camera motion). The approach that we have adopted comprises spline-based tracking [22, 25], which attempts to globally minimize the image intensity differences. This yields estimates of optic flow, which in turn are used by a local tracker [18] to refine the amount of feature displacement.

The approach we have developed for object tracking can be thought of as a "fine-to-finer" tracking approach. In addition to feature displacements, the measure of reliability of tracks is available (according to match errors and local texturedness, the latter indicated by the minimum eigenvalue of the local Hessian [18, 25]). As we'll see later in Section 8.1, this is used to cull possibly bad tracks and improve 3-D estimates.

Once we have extracted point feature tracks, we can then proceed to recover 3-D positions corresponding to these feature tracks. 3-D data recovery is based on the simple notion of stereo.

6 Omnidirectional multibaseline stereo

The idea of extracting 3-D data simultaneously from more than the theoretically sufficient number of two camera views is based on two simple tenets: statistical robustness from redundancy and disambiguation of matches due to overconstraints [15, 9]. The notion of using multiple camera views is even more critical when using panoramic images taken at the same vertical height, which results in the epipoles falling *within* the images. If only two panoramic images are used, points that are close to the epipoles will not be reliable. It is also important to note that this problem will persist if all the multiple panoramic images are taken at camera positions that are colinear. In the experiments described in Section 8, the camera positions are deliberately arranged such that all the positions are *not* colinear. In addition, all the images are taken at the same vertical height to maximize view overlap between panoramic images.

We use three related approaches to reconstruct 3-D from multiple panoramic images. 3-D data recovery is done either by (1) using just the 8-point algorithm on the tracks and directly recovering the 3-D points, or (2) proceeding with an iterative least-squares method to refine both camera pose and 3-D feature location, or (3) going a step further to impose epipolar constraints in performing a full multiframe stereo reconstruction. The first approach is termed as *unconstrained tracking and 3-D data merging* while the second approach is *iterative structure from motion*. The third approach is named *constrained depth recovery using epipolar geometry*.

6.1 Reconstruction Method 1: Unconstrained feature tracking and 8-point data merging

In this approach, we use the tracked feature points across all panoramic images and apply the 8-point algorithm. From the extracted essential matrix and camera relative poses, we can then estimate directly the 3-D positions.

The sets of 2-D image data are used to determine (pairwise) the essential matrix. The recovery of the essential matrix turns out to be reasonably stable, and this is due to the large (360°) field of view. We have found that extracting the essential matrix using the 8-point algorithm is stable when the points are well distributed over the field of view.

In this approach, we use the same set of data to recover Euclidean shape. In theory, the recovered positions are only true up to a scale. Since the distance between camera locations are known and measured, we are able to get the true scale of the recovered shape. Note, however, that the camera distances need not be known [8].

Let $\mathbf{u}_{ik}$ be the ith point of image k, $\hat{\mathbf{v}}_{ik}$ be the unit vector from the optical center to the panoramic image point in 3-D space, Λ_{ik} be the corresponding line passing through both the optical center and panoramic image point in space, and $\mathbf{t}_k$ be the camera translation associated with the kth panoramic image (note that $\mathbf{t}_1 = \mathbf{0}$). The equation of line Λ_{ik} is then $\mathbf{r}_{ik} = \lambda_{ik}\hat{\mathbf{v}}_{ik} + \mathbf{t}_k$. Thus, for each point i (that is constrained to lie on line Λ_{i1}), we minimize the error function

$$\mathcal{E}_i = \sum_{k=2}^{N} ||\mathbf{r}_{i1} - \mathbf{r}_{ik}||^2 \tag{2}$$

where N is the number of panoramic images. By taking the partial derivatives of $\mathcal{E}_i$ with respect to λ_{ij}, j = 1, ..., N, equating them to zero, and solving, we get

$$\lambda_{i1,\mathrm{opt}} = \frac{\sum_{k=2}^{N} \mathbf{t}_k^{\mathrm{T}} \left(\hat{\mathbf{v}}_{i1} - \left(\hat{\mathbf{v}}_{i1}^{\mathrm{T}} \hat{\mathbf{v}}_{ik} \right) \hat{\mathbf{v}}_{ik} \right)}{\sum_{k=2}^{N} \left(1 - \left(\hat{\mathbf{v}}_{i1}^{\mathrm{T}} \hat{\mathbf{v}}_{ik} \right)^2 \right)}, \tag{3}$$

from which the reconstructed 3-D point is calculated using the relation $\mathbf{p}_{i1,\mathrm{opt}} = \lambda_{i1,\mathrm{opt}}\hat{\mathbf{v}}_{i1}$. Note that a more optimal manner of estimating the 3-D point is to minimize the expression

$$\mathcal{E}_i = \sum_{k=1}^{N} ||\mathbf{p}_{i1,\mathrm{opt}} - \mathbf{r}_{ik}||^2 \tag{4}$$

However, due to the practical consideration of texture-mapping the recovered 3-D mesh of the estimated point distribution, the projection of the estimated 3-D point has to coincide with the 2-D image location in the reference image. This can be justified by saying that since the feature tracks originate from the reference image, it is reasonable to assume that there is no uncertainty in feature location in the reference image.

6.2 Reconstruction Method 2: Iterative panoramic structure from motion

The 8-point algorithm recovers the camera motion parameters directly from the panoramic tracks, from which the corresponding 3-D points can be computed. However, the camera motion parameters may not be optimally recovered, even though experiments by Hartley using narrow view images indicate that the motion parameters are close to optimal [4]. Using the output of the 8-point algorithm and the recovered 3-D data, we can apply an iterative least-squares minimization to refine both camera motion and 3-D positions *simultaneously*. This is similar to work done by Szeliski and Kang on structure from motion using multiple narrow camera views [23]. It turns out that applying iterative least-squares minimization does not significantly improve the accuracy of the recovered 3-D stereo data. Interested readers are referred to [8] for details.

6.3 Reconstruction Method 3: Constrained depth recovery using epipolar geometry

As a result of the first reconstruction method's reliance on tracking, it suffers from the aperture problem and hence limited number of reliable points. The approach of using the epipolar geometry to limit the search is designed to reduce the severity of this problem. Given the epipolar geometry, for each image point in the reference panoramic image, a constrained search is performed along the line of sight through the image point. Subsequently, the position along this line which results in minimum match error at projected image coordinates corresponding to other viewpoints is chosen. Using this approach results in a denser depth map, due to the epipolar constraint. The principle is the same as that described in [9].

While this approach mitigates the problem of the aperture problem, it incurs a much higher computational demand. In addition, the recovered epipolar geometry is still dependent on the output quality of the 8-point algorithm (which in turn depends on the quality of tracking). The user has to also specify minimum and maximum depths as well as the resolution of depth search.

7 Stereo data segmentation and modeling

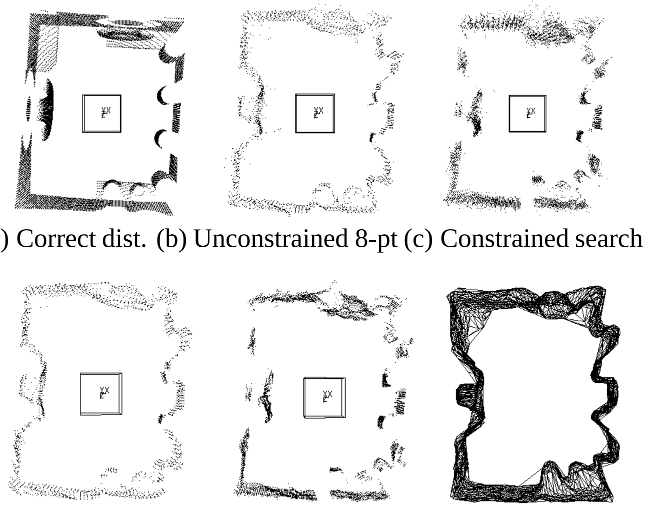
Once the 3-D stereo data has been extracted, we can then model them with a 3-D mesh and texture-map each face with the associated part of the 2-D image panorama. We have done work to reduce the complexity of the resulting 3-D mesh by planar patch fitting and boundary simplification. The displayed models shown in this paper are rendered using our modeling system. A more detailed description of model extraction from range data is given in [7].

8 Experimental results

In this section, we present the results of applying our approach to recover 3-D data from multiple panoramic images. We have used both synthetic and real images to test our approach. As mentioned earlier, in the experiments described in this section, the camera positions are deliberately arranged so that all of the positions are not colinear. In addition, all the images are taken at the same vertical height to maximize the overlap between panoramic images.

8.1 Synthetic scene

The synthetic scene is a room comprising objects such as tables, tori, cylinders, and vases. One half of the room is tex-



(a) Correct dist. (b) Unconstrained 8-pt (c) Constrained search

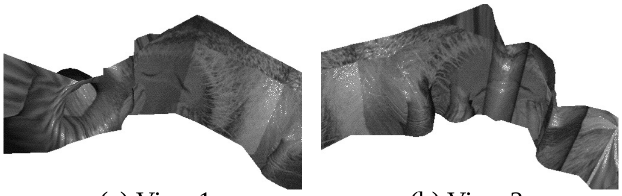
(d) Median filtered 8-pt (e) Median filtered constrained (f) Top view of 3-D mesh of (d)

Figure 6: Comparison of 3-D points recovered of synthetic room. The results for the iterative method are very similar to those for unconstrained 8-point.

tured with a mandrill image while the other is textured with a regular Brodatz pattern. The synthetic objects and images are created using Rayshade, which is a program for creating ray-traced color images [10]. The omnidirectional synthetic depth map of the entire room is created by merging the depth maps associated with the multiple views taken inside the room.

The composite panoramic view of the synthetic room from its center is shown in Figure 5. The results of applying both reconstruction methods (i.e., unconstrained search with 8-point and constrained search using epipolar geometry) can be seen in Figure 6. We get many more points using constrained search (about 3 times more), but the quality of the 3-D reconstruction appears more degraded (compare Figure 6(b) with (c)). The dimensions of the synthetic room are 10(length) × 8(width) × 6(height), and the resolution for the depth search in the multibaseline stereo algorithm is 0.01. What is interesting is that the quality of the recovered 3-D data looks better after applying a 3-D median filter. However, the median filter also has the effect of rounding off corners.

The mesh in Figure 6(f) and the two views in Figure 7 are generated by our rendering system described in [7]. As can be seen from these figures, the 3-D recovered points and the subsequent model based on these points basically preserved the shape of the synthetic room.



(a) View 1 (b) View 2

Figure 7: Two views of modeled synthetic room.

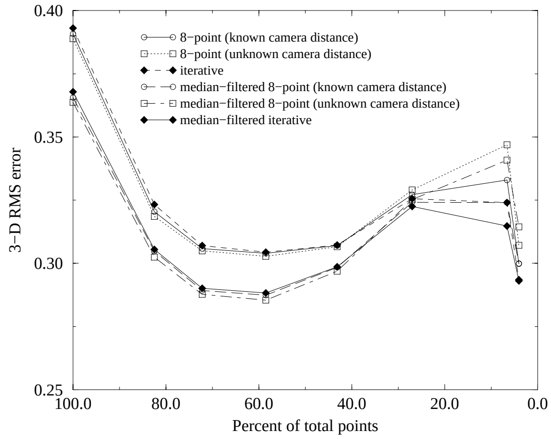


Figure 8: 3-D RMS error vs. number of points. The original number of points (corresponding to 100%) is 3057. The dimensions of the synthetic room are 10(length) × 8(width) × 6(height).

In addition, we performed a series of experiments to examine the effect of both "bad" track removal and median filtering on the quality of recovered depth information of the synthetic room. The feature tracks are sorted in increasing order according to the error in matching. We continually remove tracks that have the worst amount of match error, recovering the 3-D point distribution at each instant.

From the graph in Figure 8, we see an interesting result: as more tracks are taken out, retaining the better ones, the quality of 3-D point recovery improves—up to a point. The improvement in the accuracy is not surprising, since the worse tracks, which are more likely to result in worse 3-D estimates, are removed. However, as more and more tracks are removed, the gap between the amount of accuracy demanded of the tracks, given an increasingly smaller number of available tracks, and the track accuracy available, grows. This results in generally worse estimates of the epipolar geometry, and hence 3-D data. Reducing the number of points degrades the quality of both epipolar geometry (in the form of the essential matrix) and 3-D data. This is evidenced by the fluctuation of the curves at the lower end of the graph. Another interesting result that can be observed is that the 3-D point distribution that has been median filtered have lower errors, especially for higher numbers of recovered 3-D points.

As indicated by the graph in Figure 8, the accuracy of the point distribution derived from just the 8-point algorithm is almost equivalent that that of using an iterative

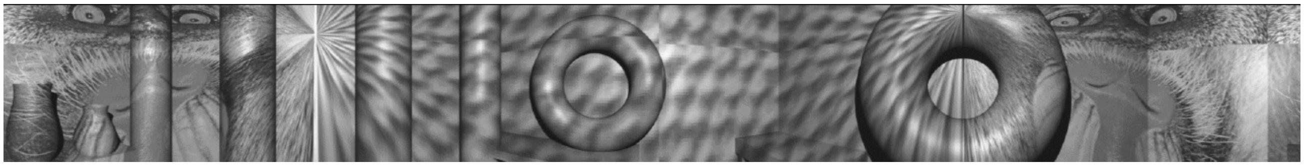


Figure 5: Panorama of synthetic room after composing.

least-squares (Levenberg-Marquardt) minimization, which is statistically optimal near the true solution. This result is in agreement with Hartley's application of the 8-point algorithm to narrow-angle images [4]. It is also worth noting that the accuracy of the iterative algorithm is best at smaller numbers of input points, suggesting that it is more stable given a smaller number of input data.

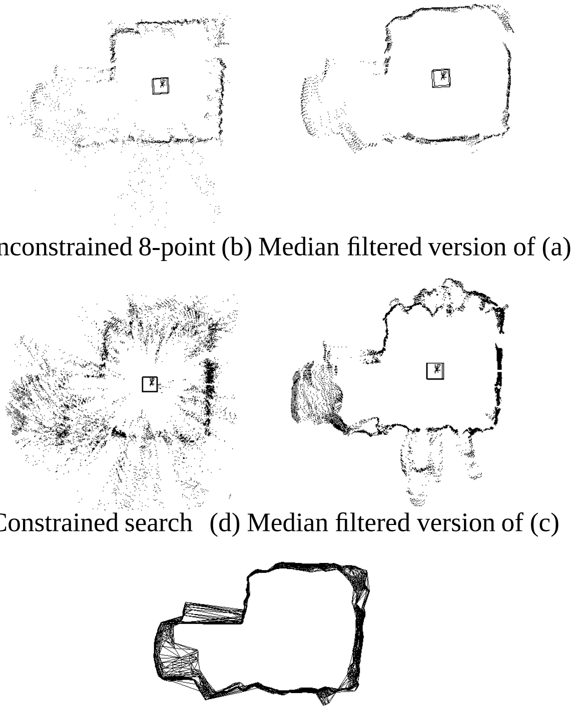
8.2 A real scene

The setup that we used to record our image sequences consists of a DEC Alpha workstation with a J300 framegrabber, and a camcorder mounted on an X-Y position stage affixed on a tripod stand.

We recorded image sequences of a lab scene. A panoramic image of the lab scene is shown in Figure 9. A total of eight panoramas at eight different locations (about 3 inches apart, ordered roughly in a zig-zag fashion) in the lab are extracted. The longest dimensions of the L-shaped lab is about 15 feet by 22.5 feet. The 3-D point distribution is shown in Figure 10 while Figure 11 shows two views of the recovered model of the lab. As can be seen, the shape of the lab has been reasonably well recovered; the "noise" points at the bottom of Figure 10(a) corresponds to the positions *outside* the laboratory, since there are parts of the transparent laboratory window that are not covered. This reveals one of the weaknesses of any correlation-based algorithm (namely all stereo algorithms): they do not work well with image reflections and transparent material. Again, we observe that the points recovered using constrained search are worse.

9 Discussion

We have shown that omnidirectional depth data (whose denseness depends on the amount of local texture) can be extracted using a set of simple techniques: camera calibration, image composing, feature tracking, the 8-point algorithm, and constrained search using the recovered epipolar geometry. The advantage of our work is that we are able to extract depth data within a wide field of view simultaneously, which removes many of the traditional problems associated with recovering camera pose and narrow-baseline stereo. Despite the practical problems caused by using unsophisticated equipment which result in slightly incorrect panoramas, we are still able to extract reasonable 3-D data. Thus far, the best real data results come from using unconstrained tracking and 8-point algorithm.



(a) Unconstrained 8-point (b) Median filtered version of (a)

(c) Constrained search (d) Median filtered version of (c)

(e) 3-D mesh of (b)

Figure 10: Extracted 3-D points and mesh of laboratory scene. The results for the iterative method are very similar to those for unconstrained 8-point.

10 Summary

We have described an approach to extracting omnidirectional 3-D data from multiple panoramas taken at arbitrary different locations. This reduces the need for numerous multiple merging of disparate depth maps corresponding to different camera views of the same scene. Results indicate that the application of 3-D median filtering improves both the accuracy and appearance of stereo-computed 3-D point distribution.

Currently, the omnidirectional data, while obtained through a 360° view, has limited vertical view. We plan to extend this work by incorporating *tilted* (i.e., rotated about a horizontal axis) camera views to increase the vertical panoramic extent. This would enable scene reconstruction of a building floor involving multiple rooms with good vertical view. In addition, we are also currently characterizing the effects of misestimated intrinsic camera parameters (focal length, aspect ratio, and the radial distortion factor)

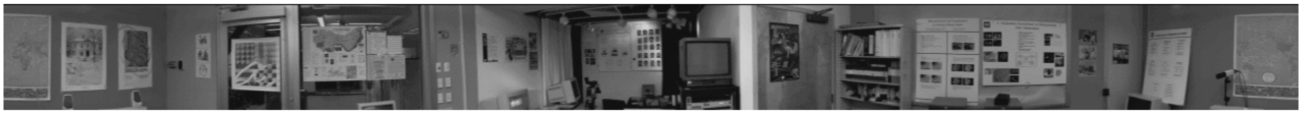
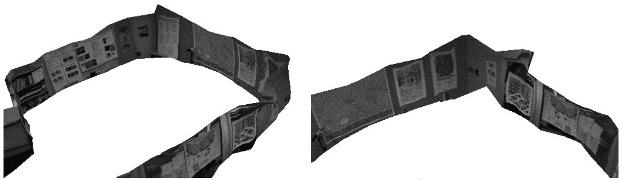


Figure 9: Panorama of laboratory after composing.



(a) View 1

(b) View 2

Figure 11: Two views of modeled laboratory scene

on the accuracy of the recovered 3-D data.

Acknowledgements

We would like to thank Andrew Johnson for the use of his 3-D modeling and rendering program and Richard Weiss for helpful discussions.

References

[1] R. C. Bolles, H. H. Baker, and D. H. Marimont. Epipolar-plane image analysis: An approach to determining structure from motion. *International Journal of Computer Vision*, 1:7–55, 1987.

[2] U. R. Dhond and J. K. Aggarwal. Structure from stereo—A review. *IEEE Transactions on Systems, Man, and Cybernetics*, 19(6):1489–1510, November/December 1989.

[3] F.P. Ferrie and M.D. Levine. Integrating information from multiple views. In *IEEE Workshop on Computer Vision*, pages 117–122. IEEE Computer Society, 1987.

[4] R. Hartley. In defence of the 8-point algorithm. In *Fifth International Conference on Computer Vision (ICCV'95)*, pages 1064–1070, Cambridge, Massachusetts, June 1995. IEEE Computer Society Press.

[5] K. Higuchi, M. Hebert, and K. Ikeuchi. Building 3-D models from unregistered range images. Technical Report CMU-CS-93-214, Carnegie Mellon University, November 1993.

[6] H. Ishigura, M. Yamamoto, and S. Tsuji. Omni-directional stereo. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14:257–262, 1992.

[7] S. B. Kang, A. Johnson, and R. Szeliski. Extraction of concise and realistic 3-D models from real data. Technical Report 95/7, Digital Equipment Corporation, Cambridge Research Lab, October 1995.

[8] S. B. Kang and R. Szeliski. 3-D scene data recovery using omnidirectional multibaseline stereo. Technical Report 95/6, Digital Equipment Corporation, Cambridge Research Lab, October 1995.

[9] S. B. Kang, J. Webb, L. Zitnick, and T. Kanade. A multibaseline stereo system with active illumination and real-time image acquisition. In *Fifth International Conference on Computer Vision (ICCV'95)*, pages 88–93, Cambridge, Massachusetts, June 1995.

[10] C. E. Kolb. Rayshade user's guide and reference manual. August 1994.

[11] C. D. Kuglin and D. C. Hines. The phase correlation image alignment method. In *IEEE 1975 Conference on Cybernetics and Society*, pages 163–165, New York, September 1975.

[12] H. C. Longuet-Higgins. A computer algorithm for reconstructing a scene from two projections. *Nature*, 293:133–135, 1981.

[13] L. McMillan and G. Bishop. Plenoptic modeling: An image-based rendering system. *Computer Graphics (SIGGRAPH'95)*, pages 39–46, Aug. 1995.

[14] D.W. Murray. Recovering range using virtual multicamera stereo. *Computer Vision and Image Understanding*, 61(2):285–291, 1995.

[15] M. Okutomi and T. Kanade. A multiple baseline stereo. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(4):353–363, April 1993.

[16] B. Parvin and G. Medioni. B-rep from unregistered multiple range images. In *IEEE Int'l Conference on Robotics and Automation*, pages 1602–1607. IEEE Society, May 1992.

[17] W. H. Press, B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling. *Numerical Recipes in C: The Art of Scientific Computing*. Cambridge University Press, Cambridge, England, second edition, 1992.

[18] J. Shi and C. Tomasi. Good features to track. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'94)*, pages 593–600, Seattle, Washington, June 1994. IEEE Computer Society.

[19] H.-Y. Shum, K. Ikeuchi, and R. Reddy. Principal component analysis with missing data and its application to object modeling. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'94)*, pages 560–565, Seattle, Washington, June 1994. IEEE Computer Society.

[20] G. Stein. Accurate internal camera calibration using rotation, with analysis of sources of error. In *Fifth International Conference on Computer Vision (ICCV'95)*, pages 230–236, Cambridge, Massachusetts, June 1995.

[21] R. Szeliski. Image mosaicing for tele-reality applications. Technical Report 94/2, Digital Equipment Corporation, Cambridge Research Lab, June 1994.

[22] R. Szeliski and J. Coughlan. Hierarchical spline-based image registration. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'94)*, pages 194–201, Seattle, Washington, June 1994. IEEE Computer Society.

[23] R. Szeliski and S. B. Kang. Recovering 3D shape and motion from image streams using nonlinear least squares. *Journal of Visual Communication and Image Representation*, 5(1):10–28, March 1994.

[24] R. Szeliski and S. B. Kang. Direct methods for visual scene reconstruction. In *IEEE Workshop on Representations of Visual Scenes*, Cambridge, Massachusetts, June 1995.

[25] R. Szeliski, S. B. Kang, and H.-Y. Shum. A parallel feature tracker for extended image sequences. In *IEEE International Symposium on Computer Vision*, pages 241–246, Coral Gables, Florida, November 1995.