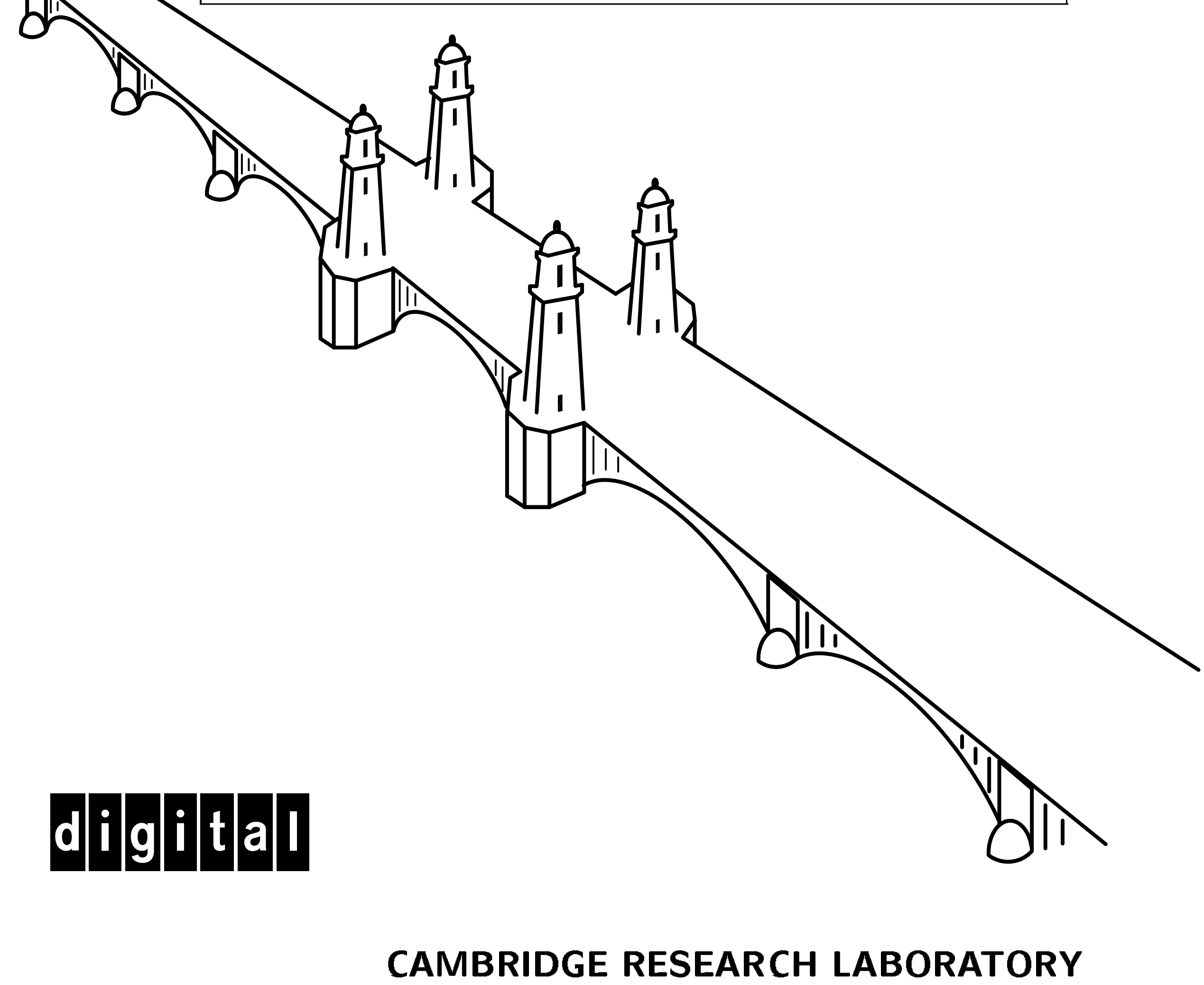
3-D Scene Data Recovery using Omnidirectional Multibaseline Stereo

Sing Bing Kang and Richard Szeliski

Digital Equipment Corporation
Cambridge Research Lab

CRL 95/6 October, 1995



digital

CAMBRIDGE RESEARCH LABORATORY
Technical Report Series

Cambridge Research Laboratory
One Kendall Square
Cambridge, Massachusetts 02139

3-D Scene Data Recovery using Omnidirectional Multibaseline Stereo

Sing Bing Kang and Richard Szeliski[1]

Digital Equipment Corporation
Cambridge Research Lab

CRL 95/6 October, 1995

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 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 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.

Keywords: Omnidirectional multibaseline stereo, 8-point algorithm, 3-D modeling.

[1]Microsoft Corporation, One Microsoft Way, Redmond, WA 98052-6399

Contents

List of Figures

List of Tables

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 [Ferrie and Levine, 1987; Higuchi *et al.*, 1993; Parvin and Medioni, 1992; Shum *et al.*, 1994]. 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 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, 8-point direct and iterative structure from motion algorithms, 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).

In this paper, we describe our approach to generate 3-D data corresponding to a very wide field of view (specifically 360°), and show results of our approach on both synthetic and real scenes. We first review relevant work in Section 2 before delineating our basic approach in Section 3. The method to extract wide-angle images (i.e., *panoramic images*) is described in Section 4. Section 5 reviews the 8-point algorithm and shows how it can be applied for cylindrical panoramic images. Section 6 describes two methods of extracting 3-D point data: the first relies on unconstrained tracking and using 8-point data input, while the second constrains the search for feature correspondences to epipolar lines. We briefly outline our approach in modeling the data in Section 7—details of this is given elsewhere [Kang *et al.*, 1995a]. Finally, we show results of our approach in Section 8 and close with a discussion and conclusions.

2 Relevant work

There is a significant body of work on range image recovery using stereo (a comprehensive survey is given in [Barnard and Fischler, 1982]). 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 [Ishigura *et al.*, 1992], which uses two panoramic views. Each panoramic view is created by one of the two vertical slits of the camera image sweeping around 360°; the cameras (which are displaced in front of the rotation center) are rotated by very small angles, typically about 0.4°. One of the disadvantages of this method is the slow data accumulation, which takes about 10 mins. The camera angular increments must be approximately $1/f$ radians, and are assumed to be known *a priori*.

Murray [Murray, 1995] 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 advantage is more efficient data acquisition, done at lower angular resolution. The analysis involved in this work is similar to Bolles *et al.*'s [Bolles *et al.*, 1987] spatio-temporal epipolar analysis, except that the temporal dimension is replaced by that of angular displacement.

Another related work is that of plenoptic modeling [McMillan and Bishop, 1995]. The idea is to composite 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 [McMillan and Bishop, 1995] in that we composite 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.

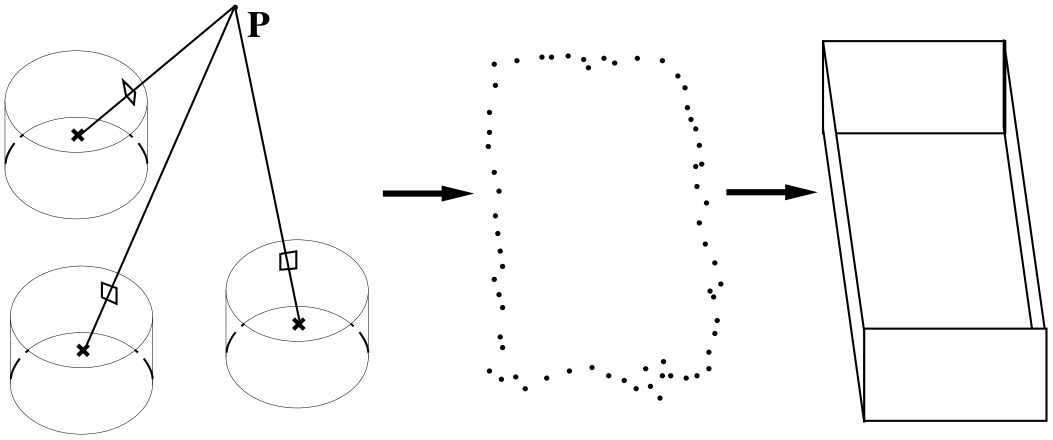


Figure 1: Generating scene model from multiple 360° panoramic views.

3 Overview of approach

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. We also wish to minimize the use of CAD packages as a means of 3-D model generation, since such an effort is labor-intensive. In addition, we impose the requirement that the means of generating models from real scene be done using commercially available equipment. 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. Composite each set of images to produce panoramas at each camera location. Use stereo to extract 3-D data of the scene. Finally, model the scene using these 3-D data input and render it with the texture provided by the input 2-D image. This approach is summarized in Figure 1.

By using panoramic images, we can extract 3-D data covering a very wide field of view, thus by-passing the need for numerous depth map merging. Multiple depth map merging 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

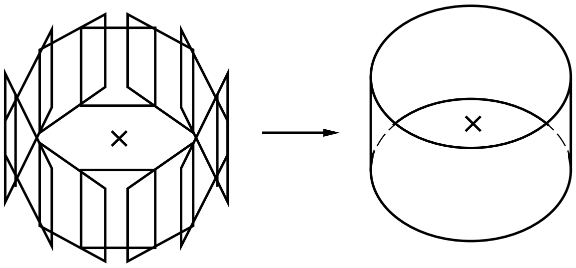


Figure 2: Compositing multiple rotated camera views into a panorama. The '$\times$' marks indicate the locations of the camera optical and rotation center.

errors.

Using multiple camera locations in stereo analysis significantly reduces the number of ambiguous matches and also has the effect of reducing errors by averaging [Okutomi and Kanade, 1993; Kang *et al.*, 1995b]. This is especially important for images with very wide fields of view, because depth recovery is unreliable near the epipoles[1], where the looming effect takes place, resulting in very poor depth cues.

4 Extraction of panoramic images

A panoramic image is created by compositing 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 [Stein, 1995].

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-

[1] For a pair of images taken at two different locations, the epipoles are the location on the image planes which are the intersection between these image planes and the line joining the two camera optical centers. An excellent description of the stereo vision is given in [Faugeras, 1993].

Image 1 Image 2 . . . Image (N-1) Image N

Figure 3: Example undistorted image sequence (of an office).

Marquardt) to estimate camera intrinsic and extrinsic parameters (except for κ) [Szeliski and Kang, 1994]. κ is determined using 1-D search (Brent's parabolic interpolation in 1-D [Press *et al.*, 1992]) 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. An example of a sequence of corrected images (of an office) is shown in Figure 3.
- Composite the images (with only x-directional DOF, which is equivalent to motion in the angular dimension of cylindrical image space) to yield the desired panorama [Szeliski, 1994]. The relative displacement of one frame to the next is coarsely determined by using phase correlation [Kuglin and Hines, 1975]. This technique estimates the 2-D translation between a pair of images by taking 2-D Fourier transforms of both images, computing the phase difference at each frequency, performing an inverse Fourier transform, and searching for a peak in the magnitude image. Subsequently, the image translation is refined using local image registration by directly comparing the overlapped regions between the two images [Szeliski, 1994].
- 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 recompositing. The error in length is usually within a percent of the expected length.

Figure 4: Panorama of office scene after compositing.

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 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.

The epipolar geometry dictates the *epipolar constraint*, which refers to the locus of possible image projections in one image given an image point in another image. For planar image planes, the epipolar constraint is in the form of straight lines. The interested reader is referred to [Faugeras, 1993] for details.

We use the 8-point algorithm [Longuet-Higgins, 1981; Hartley, 1995] 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 [Hartley, 1995]. The solution that yields the most *positive* projections (i.e., projections away from the camera optical centers) is chosen.

5.1 8-point algorithm: Basics

We briefly review the 8-point algorithm here: If the camera is calibrated (i.e., its intrinsic parameters are known), then for any two corresponding image points (at two different camera placements) $(u, v, w)^{\mathrm{T}}$ and $(u', v', w')^{\mathrm{T}}$ in 3-D, we have

$$(u', v', w')\mathbf{E}\begin{pmatrix} u \\ v \\ w \end{pmatrix} = 0 \qquad (1)$$

The matrix E is called the *essential matrix*, and is of the form $E = [\mathbf{t}]_\times R$, where R and **t** are the rotation matrix and translation vectors, respectively, and $[\mathbf{t}]_\times$ is the matrix form of the cross product with **t**.

If the camera is not calibrated, we have a more general relation between two corresponding image points (on the image plane) $(u, v, 1)^{\mathrm{T}}$ and $(u', v', 1)^{\mathrm{T}}$, namely

$$(u', v', 1)\mathbf{F}\begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = 0 \tag{2}$$

F is called the fundamental matrix and is also of rank 2, $F = [\mathbf{t}]_\times A$, where A is an arbitrary 3×3 matrix. The fundamental matrix is the generalization of the essential matrix E, and is usually employed to establish the epipolar geometry and to recover projective depth [Faugeras, 1992; Shashua, 1994].

In our case, since we know the camera parameters, we can recover E. Let **e** be the vector comprising e_{ij}, where e_{ij} is the (i,j)th element of E. Then for all the point matches, we have from (1)

$$uu'e_{11} + uv'e_{21} + uw'e_{31} + vu'e_{12} + vv'e_{22} + vw'e_{32} + wu'e_{13} + wv'e_{23} + ww'e_{33} = 0, \tag{3}$$

from which we get a set of linear equations of the form

$$\mathcal{A}\mathbf{e} = 0. \tag{4}$$

If the number of input points is small, 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 [Hartley, 1995] prior to using the 8-point algorithm. The 3-D cylindrical points are normalized according to the relation

$$\mathbf{u} = (f \sin\theta, y, f\cos\theta) \rightarrow \hat{\mathbf{u}} = \mathbf{u}/|\mathbf{u}| \tag{5}$$

With N panoramic images, we solve for $(N-1)$ sets of linear equations of the form (4). The kth set corresponds to the panoramic image pair 1 and $(k+1)$. Notice that the solution of **e** is defined only up to an unknown scale. In our work, we measure the distance between camera positions; this enable us to recover the scale. However, we can relax this assumption by carrying out the following steps:

- Fix camera distance of first pair (pair 1), to, say unit distance. Assign camera distances for all the other pairs to be the same as the first.
- Calculate the essential matrices for all the pairs of panoramic images, assuming unit camera distances.
- For each pair, compute the 3-D points.
- To estimate the relative distances between of camera positions for pair $j \neq 1$ (i.e., not the first pair), find the scale of the 3-D points corresponding to pair j that minimizes the distance error to those corresponding to pair 1. Robust statistics is used to reject outliers; specifically, only the best 50% are used.

5.2 Tracking features for 8-point algorithm

The 8-point algorithm assumes that feature point correspondences are available. Feature tracking is a challenge in that 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, which attempts to globally minimize the image intensity differences. This yields estimates of optic flow, which in turn is used by a local tracker to refine the amount of feature displacement.

The optic flow between a pair of cylindrical panoramic images is first estimated using spline-based image registration between the pair [Szeliski and Coughlan, 1994; Szeliski *et al.*, 1995]. In this image registration approach, the displacement fields $u(x, y)$ and $v(x, y)$ (i.e., displacements in the x- and y- directions as functions of the pixel location) are represented as two-dimensional *splines* controlled by a smaller number of displacement estimates which lie on a coarser *spline control grid*.

Once the initial optic flow has been found, the best candidates for tracking are then chosen. The choice is based on the minimum eigenvalue of the local Hessian, which is an indication of local image texturedness. Subsequently, using the initial optic flow as an estimate displacement field, we use the Shi-Tomasi tracker [Shi and Tomasi, 1994] with a window of size 25 pixels $\times$ 25 pixels to further refine the displacements of the chosen point features.

Why did we use the approach of applying the spline-based tracker before using the Shi-Tomasi tracker? This approach is used to take advantage of the complementary characteristics of these two trackers, namely:

1. the spline-based image registration technique is capable of tracking features with larger displacements. This is done through coarse-to-fine image registration; in our work, we use 6 levels of resolution. While this technique generally results in good tracks (sub-pixel accuracy) [Szeliski *et al.*, 1995], poor tracks may result in areas in the vicinity of object occlusions/disocclusions.

2. the Shi-Tomasi tracker is a local tracker that fails at large displacements. It performs better for a small number of frames and for relatively small displacements, but deteriorates at large numbers of frames and in the presence of rotation on the image plane [Szeliski *et al.*, 1995]. We are considering a small number of frames at a time, and image warping due to local image plane rotation is not expected. The Shi-Tomasi tracker is also capable of sub-pixel accuracy.

The approach that we have undertaken 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 [Shi and Tomasi, 1994; Szeliski *et al.*, 1995]). 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 founded on two simple tenets: statistical robustness from redundancy and disambiguation of matches due to overconstraints [Okutomi and Kanade, 1993; Kang *et al.*, 1995b]. 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 collinear. In the experiments described in Section 8, the camera positions are deliberately arranged such that all the positions are *not* collinear. 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 3-D 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 directly estimate 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; this is due to the large (360°) field of view. A problem with the 8-point algorithm is that optimization occurs in function space and not image space, i.e., it is not minimizing error in distance between 2-D image point and corresponding epipolar line. Deriche *et al.* [Deriche *et al.*, 1994] use a robust regression method called *least-median-of-squares* to minimize distance error between expected (from the estimated fundamental matrix) and given 2-D image points. We have found that extracting the essential matrix using the 8-point algorithm is relatively stable as long as (1) the number of points is large (at least in the hundreds), and (2) 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 this approach is not critical upon knowing the camera distances, as indicated in Section 5.1.

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

$$\mathcal{E}_i = \sum_{k=2}^{N} \|\mathbf{r}_{i1} - \mathbf{r}_{ik}\|^2 \tag{6}$$

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

$$\lambda_{i1,\mathrm{opt}} = \frac{\sum_{k=2}^{N} \mathbf{t}_k^{\mathrm{T}} \left(\hat{\mathbf{v}}_{i1} - \left(\hat{\mathbf{v}}_{i1}^{\mathrm{T}} \hat{\mathbf{v}}_{ik} \right) \hat{\mathbf{v}}_{ik} \right)}{\sum_{k=2}^{N} \left(1 - \left(\hat{\mathbf{v}}_{i1}^{\mathrm{T}} \hat{\mathbf{v}}_{ik} \right)^2 \right)}, \tag{7}$$

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

$$\mathcal{E}_i = \sum_{k=1}^{N} \|\mathbf{p}_{i1,\mathrm{opt}} - \mathbf{r}_{ik}\|^2 \tag{8}$$

A detailed derivation involving (8) is given in Appendix A. 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.

An immediate problem with the approach of feature tracking and data merging is its reliance on tracking, which makes it relatively sensitive to tracking errors. It inherits the problems associated with tracking, such as the aperture problem and sensitivity to changing amounts of object distortion at different viewpoints. However, this problem is mitigated if the number of sampled points is large. In addition, the advantage is that there is no need to specify minimum and maximum depths and resolution associated with multibaseline stereo depth search (e.g., see [Okutomi and Kanade, 1993; Kang *et al.*, 1995b]). This is because the points are extracted directly analytically once the correspondence is established.

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 [Hartley, 1995]. 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 [Szeliski and Kang, 1994].

As input to our reconstruction method, we use 3-D *normalized* locations of cylindrical image point. The equation linking a 3-D normalized cylindrical image position $\mathbf{u}_{ij}$ in frame j to its 3-D position $\mathbf{p}_i$, where i is the track index, is

$$\mathbf{u}_{ij} = \mathcal{P}\left(\mathbf{R}_j^{(k)}\mathbf{p}_i + \mathbf{t}_j^{(k)}\right) = \mathcal{F}\left(\mathbf{p}_i, \mathbf{q}_j, \mathbf{t}_j\right) \tag{9}$$

where $\mathcal{P}()$ is the projection transformation; $\mathbf{R}_j^{(k)}$ and $\mathbf{t}_j^{(k)}$ are the rotation matrix and translation vector, respectively, associated with the relative pose of the jth camera. We represent each rotation by a quaternion $\mathbf{q} = [w, (q_0, q_1, q_2)]$ with a corresponding rotation matrix

$$\mathbf{R}(\mathbf{q}) = \begin{pmatrix} 1 - 2q_1^2 - 2q_2^2 & 2q_0q_1 - 2wq_2 & 2q_0q_2 + 2wq_1 \\ 2q_0q_1 + 2wq_2 & 1 - 2q_0^2 - 2q_2^2 & 2q_1q_2 - 2wq_0 \\ 2q_0q_2 - 2wq_1 & 2q_1q_2 + 2wq_0 & 1 - 2q_0^2 - 2q_1^2 \end{pmatrix} \tag{10}$$

(alternative representations for rotations are discussed in [Ayache, 1991]).

The projection equation is given simply by

$$\begin{pmatrix} u \\ v \\ w \end{pmatrix} = \mathcal{P}\begin{pmatrix} x \\ y \\ z \end{pmatrix} \equiv \frac{1}{\sqrt{x^2 + y^2 + z^2}}\begin{pmatrix} x \\ y \\ z \end{pmatrix} \tag{11}$$

In other words, all the 3-D points are projected onto the surface of a 3-D unit sphere.

To solve for the structure and motion parameters simultaneously, we use the iterative Levenberg-Marquardt algorithm. The Levenberg-Marquardt method is a standard non-linear least squares technique [Press *et al.*, 1992] that works well in a wide range of situations. It provides a way to vary smoothly between the inverse-Hessian method and the steepest descent method.

The merit or objective function that we minimize is

$$\mathcal{C}(\mathbf{a}) = \sum_i \sum_j c_{ij} \left|\mathbf{u}_{ij} - \mathcal{F}\left(\mathbf{a}_{ij}\right)\right|^2, \tag{12}$$

where $\mathcal{F}()$ is given in (9) and

$$\mathbf{a}_{ij} = \left(\mathbf{p}_i^{\mathrm{T}}, \mathbf{q}_j^{\mathrm{T}}, \mathbf{t}_j^{\mathrm{T}}\right)^{\mathrm{T}} \tag{13}$$

is the vector of structure and motion parameters which determine the image of point i in frame j. The weight c_{ij} in (12) describes our confidence in measurement $\mathbf{u}_{ij}$, and is normally set to the inverse variance σ_{ij}^{-2}. We set $c_{ij} = 1$.

The Levenberg-Marquardt algorithm first forms the approximate Hessian matrix

$$\mathbf{A} = \sum_i \sum_j c_{ij} \left(\frac{\partial \mathcal{F}(\mathbf{a}_{ij})}{\partial \mathbf{a}}\right)^{\mathrm{T}} \frac{\partial \mathcal{F}(\mathbf{a}_{ij})}{\partial \mathbf{a}} \tag{14}$$

and the weighted gradient vector

$$\mathbf{b} = -\sum_i \sum_j c_{ij} \left(\frac{\partial \mathcal{F}(\mathbf{a}_{ij})}{\partial \mathbf{a}}\right)^{\mathrm{T}} \mathbf{e}_{ij}, \tag{15}$$

where $\mathbf{e}_{ij} = \mathbf{u}_{ij} - \mathcal{F}(\mathbf{a}_{ij})$ is the image plane error of point i in frame j. Given a current estimate of $\mathbf{a}$, it computes an increment $\delta\mathbf{a}$ towards the local minimum by solving

$$(\mathbf{A} + \lambda\mathbf{I})\delta\mathbf{a} = -\mathbf{b}, \tag{16}$$

where λ is a stabilizing factor which varies over time [Press *et al.*, 1992]. Note that the matrix $\mathbf{A}$ is an approximation to the Hessian matrix, as the second-derivative terms are left out. As mentioned in [Press *et al.*, 1992], inclusion of these terms can be destabilizing if the model fits badly or is contaminated by outlier points.

To compute the required derivatives for (14) and (15), we compute derivatives with respect to each of the fundamental operations (perspective projection, rotation, translation) and apply the chain rule. The equations for each of the basic derivatives are given in Appendix B. The derivation is exactly the same as in [Szeliski and Kang, 1994], except for the projection equation.

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. This constrain reduces the aperture problem during search (which theoretically only occurs if the direction of ambiguity is along the epipolar line of interest). The principle is the same as that described in [Kang *et al.*, 1995b].

While this approach mitigates the problem of the aperture problem, it suffers from 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 resolution of depth search.

An alternative to working in cylindrical coordinates is to project sections of cylinder to a tangential rectilinear image plane, rectify it, and use the rectified planes for multibaseline stereo. This mitigates the computational demand as search is restricted to horizontal scanlines in the rectified images. However, there is a major problem with this scheme: reprojecting to rectilinear coordinates and rectifying is problematical due to the increasing distortion away from the new center of projection. This creates a problem with matching using a window of a fixed size. As a result, this scheme of reprojecting to rectilinear coordinates and rectifying is not used.

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 [Kang *et al.*, 1995a].

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

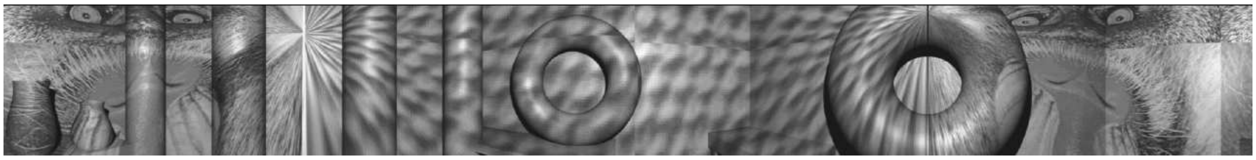


Figure 5: Panorama of synthetic room after compositing.

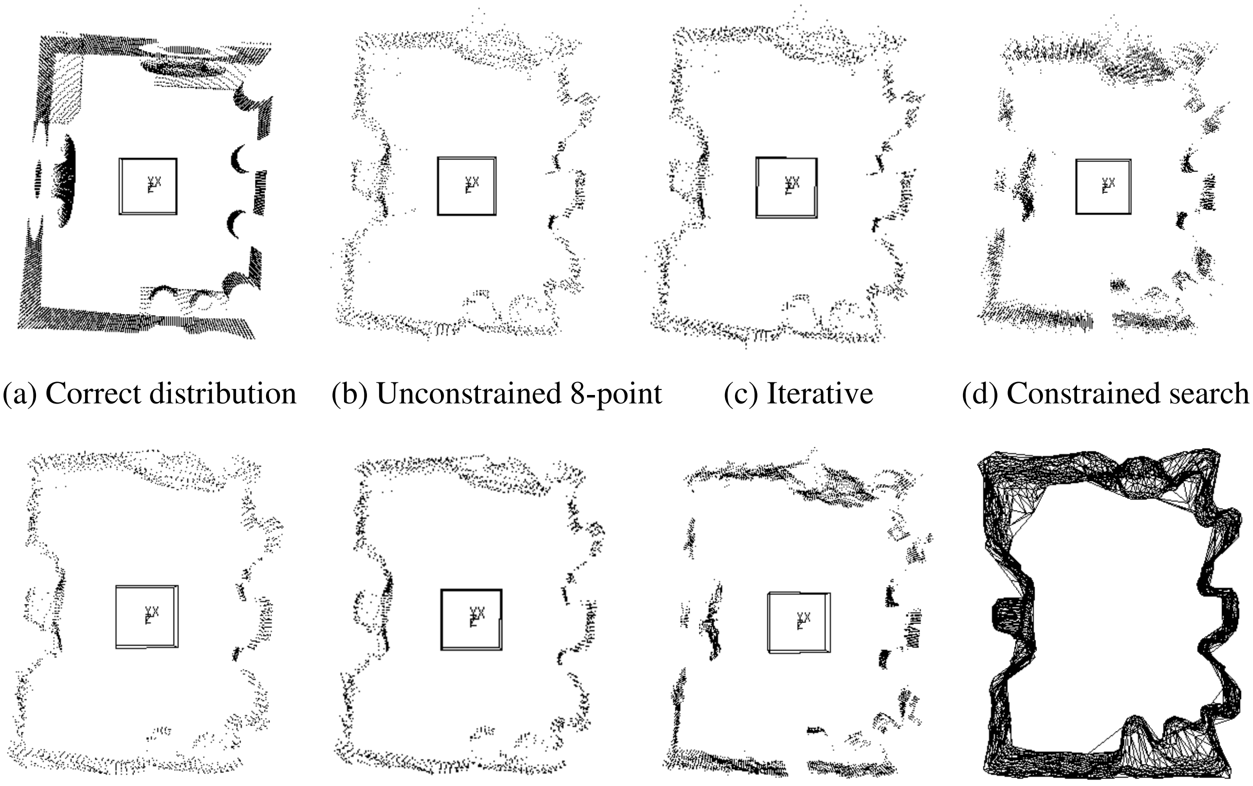
earlier, in the experiments described in this section, the camera positions are deliberately arranged so that all of the positions are not collinear. In addition, all the images are taken at the same vertical height to maximize overlap between panoramic images.

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 textured 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 [Kolb, 1994]. The synthetic images created are free from any radial distortion, since Rayshade is currently unable to model this camera characteristic. The omnidirectional synthetic depth map of the entire room is created by merging the depth maps associated with the multiple views taken around inside the room.

The composite panoramic view of the synthetic room from its center is shown in Figure 5. From left to right, we can observe the vases resting on a table, vertical cylinders, a torus resting on a table, and a larger torus. 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)). This is in part due to matching occurring at integral values of pixel positions, limiting its depth resolution. The dimensions of the synthetic room are 10(length) $\times$ 8(width) $\times$ 6(height), and the specified resolution is 0.01. The quality of the recovered 3-D data appears to be enhanced by applying a 3-D median filter [2]. However, the median

[2] The median filter works in the following manner: For each feature point in the cylindrical panoramic image, find other feature points within a certain neighborhood radius (20 in our case). Then sort the 3-D depths associated with the neighborhood feature points, find the median depth, and *rescale* the depth associated with the current feature point such that the new depth is the median depth. As an illustration, suppose the original 3-D feature location is $\mathbf{v}_i = d_i \hat{\mathbf{v}}_i$, where d_i is the original depth and $\hat{\mathbf{v}}_i$ is the 3-D unit vector from the camera center in the direction of the image point. If d_{med}



(a) Correct distribution (b) Unconstrained 8-point (c) Iterative (d) Constrained search

(e) Median-filtered 8-point (f) Median-filtered iterative (g) Median-filtered constrained (h) Top view of 3-D mesh of (e)

Figure 6: Comparison of 3-D points recovered of synthetic room.

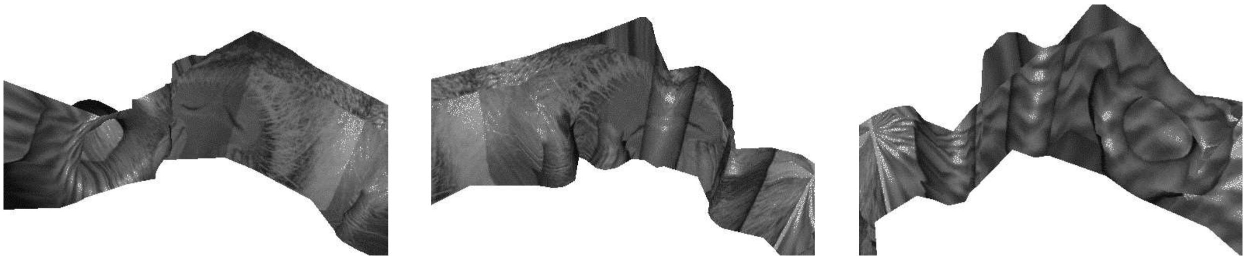
filter also has the effect of rounding off corners.

The mesh in Figure 6(f) and the three views in Figure 7 are generated by our 3-D modeling system described in [Kang *et al.*, 1995a]. 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.

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[3]. We continually

is the median depth within its neighborhood, then the filtered 3-D feature location is given by $\mathbf{v}'_i = (d_{\mathrm{med}}/d_i)\mathbf{v}_i = d_{\mathrm{med}}\hat{\mathbf{v}}_i$.

[3]Note that in general, a "worse" track in this sense need not necessarily translate to a worse 3-D estimate. A high



(a) View 1 (b) View 2 (b) View 3

Figure 7: Three views of modeled synthetic room of Figure 6(h).

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. Concomitant to the reduction of the number of points is the sensitivity of the recovery 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 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 [Hartley, 1995]. 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.

Table 1 lists the 3-D errors of both constrained and unconstrained (8-point only) methods for the synthetic scenes. It appears from this result that the constrained method yields better results (after

match error may be due to apparent object distortion at different viewpoints.

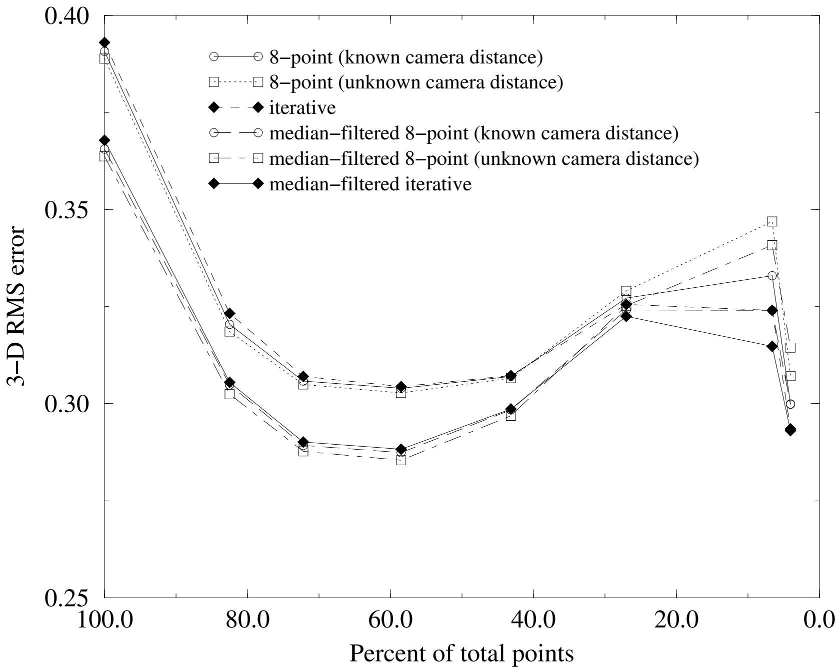


Figure 8: 3-D RMS error vs. number of points. The original number of points (corresponding to 100%) is 3057. The dimensions of the synthetic room are 10(length) $\times$ 8(width) $\times$ 6(height).

	constrained(n=10040)	8-point(n=3057)	8-point(n=1788)
original	0.315039	0.393777	0.302287
median-filtered	0.266600	0.364889	0.288079

Table 1: Comparison of 3-D RMS error between unconstrained and constrained stereo results (n is the number of points).

median filtered) and more points (a result of reducing the aperture problem). In practice, as we shall see in the next section, problems due to misestimation of camera intrinsic parameters (specifically focal length, aspect ratio and radial distortion coefficient) causes 3-D reconstruction from real images to be worse. This is a subject of on-going research.

8.2 Real scenes

The setup that we used to record our image sequences consists of a DEC Alpha workstation with a J300 framegrabber, and a camcorder (Sony Handycam CCD-TR81) mounted on an X-Y position stage affixed on a tripod stand. The camcorder settings are made such that its field of view is maximized (at about 43°).

To reiterate, our method of generating the panoramic images are as follows:

- Calibrate camcorder using an iterative Levenberg-Marquardt least-squares algorithm [Szeliski and Kang, 1994].
- Adjust the X-Y position stage while panning the camera left and right to remove the effect of motion parallax; this ensures that the camera is then rotated about its optical center.
- At each camera location, record onto tape an image sequence while rotating the camera, and then digitize the image sequence using the framegrabber.
- Using the recovered camera intrinsic parameters (focal length, aspect ratio, radial distortion factor), undistort each image.
- Project each image, which is in rectilinear image coordinates, into cylindrical coordinates (whose cross-sectional radius is the camera focal length).
- Composite the frames into a panoramic image. The number of frames used to extract a panoramic image in our experiments is typically about 50.

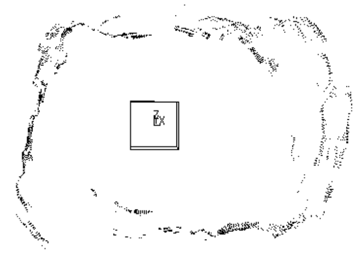
We recorded image sequences of two scenes, namely an office scene and a lab scene. A panoramic image of the office scene is shown in Figure 4. We extracted four panoramic images corresponding to four different locations in the office. (The spacing between these locations is about 6 inches and the locations are roughly at the corners of a square. The size of the office is about 10 feet by 15 feet.) The results of 3-D point recovery of the office scene is shown in Figure 9, with three sample views of its model shown in Figure 10. As can be seen from Figure 9, the results due to the constrained search approach looks much worse. This may be directly attributed to the inaccuracy of the extracted intrinsic camera parameters. As a consequence, the composited panoramas may actually be not exactly physically correct. In fact, as the matching (with epipolar constraint) is in progress, it has been observed that the actual correct matches are not exactly along the epipolar lines; there are slight vertical drifts, generally of the order of about one or two pixels.

Another example of real scene is shown in Figure 11. 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 12 while Figure 13 shows three 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 12(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 is worse.

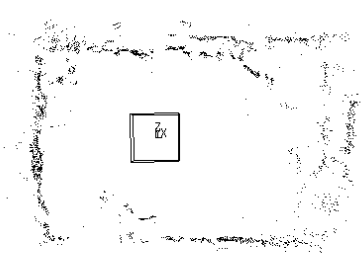
The errors that were observed with the real scene images, especially with constrained search, are due to the following practical problems:

- The auto-iris feature of the camcorder used cannot be deactivated (even though the focal length was kept constant). As a result, there may be in fact slight variations in focal length as the camera was rotated.
- The camera may not be rotating exactly about its optical center, since the adjustment of the X-Y position stage is done manually and there may be human error in judging the absence of motion parallax.
- The camera may not be rotating about a unique axis all the way around (assumed to be vertical) due to some play or unevenness of the tripod.
- There were digitization problems. The images digitized from tape (i.e., while the camcorder is playing the tape) contain scan lines that are occasionally horizontally shifted; this is probably caused by the degraded blanking signal not properly detected by the framegrabber. However, compositing many images averages out most of these artifacts.

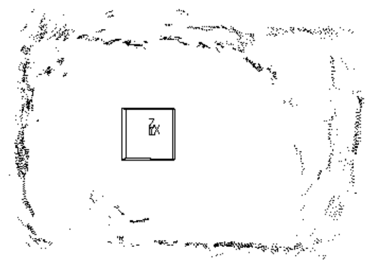
(a) Unconstrained 8-point



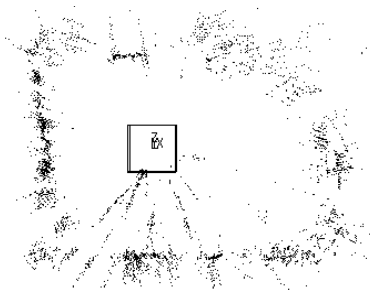
(b) Median-filtered version of (a)



(c) Iterative

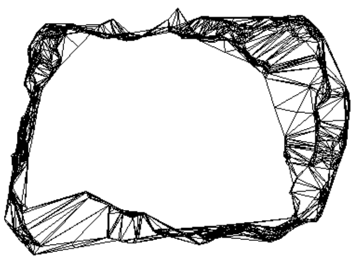


(d) Median-filtered version of (c)



(e) Constrained search

(f) Median-filtered version of (e)



(g) 3-D mesh of (b)

Figure 9: Extracted 3-D points and mesh of office scene. Notice that the recovered distributions shown in (c) and (d) appear more rectangular than those shown in (a) and (b).

(a) View 1 (b) View 2 (b) View 3

Figure 10: Three views of modeled office scene of Figure 9(g)

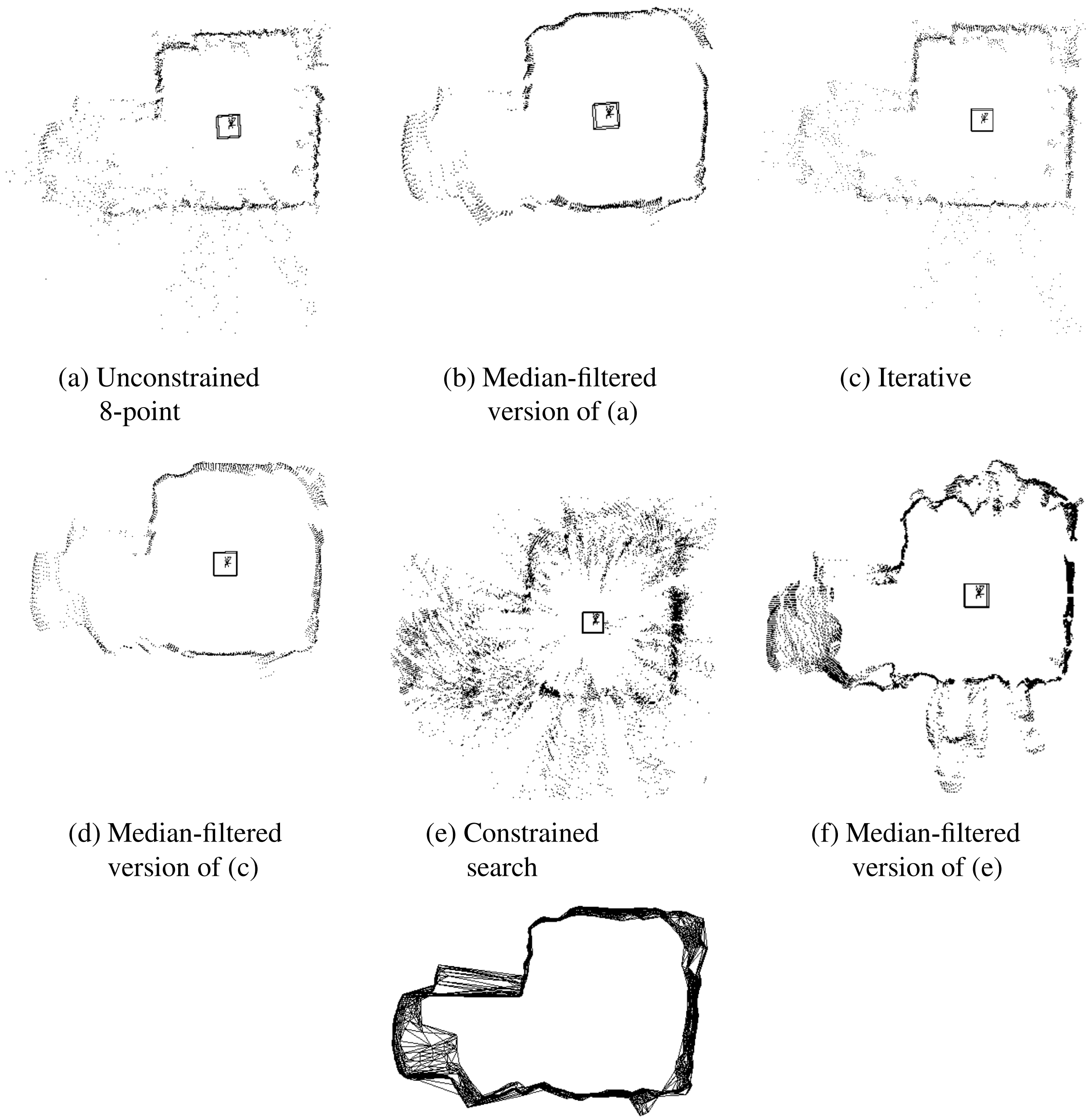
Figure 11: Panorama of laboratory after compositing.

- The extracted camera intrinsic parameters may not be very precise.

As a result of the problems encountered, the resulting composited panorama may not be physically correct. This especially causes problems with constrained search given the estimated epipolar geometry (through the essential matrix). We actually widened the search a little by allowing search as much as a couple of pixels away from the epipolar line; however, this further significantly increases the computational demand and has the effect of loosening the constraints, making this approach less attractive.

9 Discussion and conclusions

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 compositing, 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 the 8-point al-



(a) Unconstrained 8-point

(b) Median-filtered version of (a)

(c) Iterative

(d) Median-filtered version of (c)

(e) Constrained search

(f) Median-filtered version of (e)

(g) 3-D mesh of (b)

Figure 12: Extracted 3-D points and mesh of laboratory scene.

(a) View 1 (b) View 2 (b) View 3

Figure 13: Three views of modeled laboratory scene of Figure 12(g)

gorithm (both direct and iterative structure from motion). Results also indicate that the application of 3-D median filtering improves both the accuracy and appearance of stereo-computed 3-D point distribution.

To expedite the panorama image production in critical applications that require close to real-time modeling, special camera equipment may be called for. One such possible specialized equipment is Ahuja's camera system (as reported in [Freedman, 1995]), in which the lens can be rotated relative to the imaging plane. However, we are currently putting our emphasis on the use of commercially available equipment such as a cheap camcorder.

Even if all the practical problems associated with imperfect data acquisition were solved, we still have the fundamental problem of stereo—that of the inability to match and extract 3-D data in textureless regions. In scenes that involve mostly textureless components such as bare walls and objects, special pattern projectors may need to be used in conjunction with the camera [Kang *et al.*, 1995b].

Currently, the omnidirectional data, while obtained through a 360° view, has limited vertical view. We plan to extend this work by merging multiple omnidirectional data obtained at both different heights and at different locations. We will also look into the possibility of extracting panoramas of larger height extents by incorporating *tilted* (i.e., rotated about a horizontal axis) camera views. This would enable scene reconstruction of a building floor involving multiple rooms with good vertical view. We are currently characterizing the effects of misestimated intrinsic camera parameters (focal length, aspect ratio, and the radial distortion factor) on the accuracy of the recovered 3-D data.

In summary, our set of methods for reconstructing 3-D scene points within a wide field of view has been shown to be quite robust and accurate. Wide-angle reconstruction of 3-D scenes is conventionally achieved by merging multiple range images; our methods have been demonstrated to be a very attractive alternative in wide-angle 3-D scene model recovery. In addition, these methods do not require specialized camera equipment, thus making commercialization of this technology easier and more direct. We strongly feel that this development is a significant one toward attaining the goal of creating photorealistic 3-D scenes with minimum human intervention.

Acknowledgments

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

A Optimal point intersection

In order to find the point closest to all of the rays whose line equations are of the form $\mathbf{r} = \mathbf{t}_k + \lambda_k \hat{\mathbf{v}}_k$, we minimize the expression

$$\mathcal{E} = \sum_k \|\mathbf{p} - (\mathbf{t}_k + \lambda_k \hat{\mathbf{v}}_k)\|^2 \tag{17}$$

where $\mathbf{p}$ is the optimal point of intersection to be determined. Taking the partials of $\mathcal{E}$ with respect to λ_k and $\mathbf{p}$ and equating them to zero, we have

$$\begin{aligned} \frac{\partial \mathcal{E}}{\partial \lambda_k} &= 2\hat{\mathbf{v}}_k^T(\mathbf{t}_k + \lambda_k \hat{\mathbf{v}}_k - \mathbf{p}) = 0 && (18) \\ \frac{\partial \mathcal{E}}{\partial \mathbf{p}} &= -2\sum_k (\mathbf{t}_k + \lambda_k \hat{\mathbf{v}}_k - \mathbf{p}) = 0. && (19) \end{aligned}$$

Solving for λ_k in (18), noting that $\hat{\mathbf{v}}_k^T \hat{\mathbf{v}}_k = 1$, and substituting λ_k in (19) yields

$$\sum_k \left(\mathbf{t}_k - \hat{\mathbf{v}}_k(\hat{\mathbf{v}}_k^T \mathbf{t}_k) - \mathbf{p} + \hat{\mathbf{v}}_k(\hat{\mathbf{v}}_k^T \mathbf{p})\right) = 0,$$

from which

$$\mathbf{p} = \left[\sum_k \mathbf{A}_k\right]^{-1} \left[\sum_k \mathbf{A}_k \mathbf{t}_k\right] = \left[\sum_k \mathbf{A}_k\right]^{-1} \left[\sum_k \mathbf{p}_k^*\right], \tag{20}$$

where

$$\mathbf{A}_k = \mathbf{I} - \hat{\mathbf{v}}_k \hat{\mathbf{v}}_k^T$$

is the perpendicular projection operator for ray $\hat{\mathbf{v}}_k$, and

$$\mathbf{p}_k^* = \mathbf{t}_k - \hat{\mathbf{v}}_k(\hat{\mathbf{v}}_k^T \mathbf{t}_k) = \mathbf{A}_k \mathbf{t}_k$$

is the point along the viewing ray $\mathbf{r} = \mathbf{t}_k + \lambda_k \hat{\mathbf{v}}_k$ closest to the origin.

Thus, the optimal intersection point for a bundle of rays can be computed as a weighted sum of adjusted camera centers (indicated by $\mathbf{t}_k$'s), where the weighting is in the direction perpendicular to the viewing ray.

A more "optimal" estimate can be found by minimizing the formula

$$\mathcal{E} = \sum_k \lambda_k^{-2} \|\mathbf{p} - (\mathbf{t}_k + \lambda_k \hat{\mathbf{v}}_k)\|^2 \tag{21}$$

with respect to $\mathbf{p}$ and λ_k's. Here, by weighting each squared perpendicular distance by λ_k^{-2}, we are downweighting points further away from the camera. The justification for this formula is that the uncertainty in $\hat{\mathbf{v}}_k$ direction defines a *conical* region of uncertainty in space centered at the camera, i.e., the uncertainty in point location (and hence the inverse weight) grows linearly with λ_k. However, implementing this minimization requires an interative non-linear solver.

B Elemental transform derivatives

The derivative of the projection function (11) with respect to its 3-D arguments and internal parameters is straightforward:

$$\frac{\partial \mathcal{P}(\mathbf{x})}{\partial \mathbf{x}} = \frac{1}{D} \begin{pmatrix} y^2 + z^2 & -xy & -xz \\ -xy & x^2 + z^2 & -yz \\ -xz & -yz & x^2 + y^2 \end{pmatrix},$$

where

$$D = \left(x^2 + y^2 + z^2\right)^{\frac{3}{2}}$$

The derivatives of an elemental rigid transformation (9)

$$\mathbf{x}' = \mathbf{R}\mathbf{x} + \mathbf{t}$$

are

$$\frac{\partial \mathbf{x}'}{\partial \mathbf{x}} = \mathbf{R}, \quad \frac{\partial \mathbf{x}'}{\partial \mathbf{t}} = \mathbf{I}, \quad \text{and} \; \frac{\partial \mathbf{x}'}{\partial \mathbf{q}} = -\mathbf{R}\mathbf{C}(\mathbf{x})\mathbf{G}(\mathbf{q}),$$

where

$$\mathbf{C}(\mathbf{x}) = \begin{pmatrix} 0 & -z & y \\ z & 0 & -x \\ -y & x & 0 \end{pmatrix} \; \text{and} \; \mathbf{G}(\mathbf{q}) = 2 \begin{pmatrix} -q_0 & w & q_2 & -q_1 \\ -q_1 & -q_2 & w & q_0 \\ -q_2 & q_1 & -q_0 & w \end{pmatrix}$$

(see [Shabana, 1989]). The derivatives of a screen coordinate with respect to any motion or structure parameter can be computed by applying the chain rule and the above set of equations.

References

[Ayache, 1991] N. Ayache. *Artificial Vision for Mobile Robots: Stereo Vision and Multisensory Perception*. MIT Press, Cambridge, Massachusetts, 1991.

[Barnard and Fischler, 1982] S. T. Barnard and M. A. Fischler. Computational stereo. *Computing Surveys*, 14(4):553–572, December 1982.

[Bolles *et al.*, 1987] 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.

[Deriche *et al.*, 1994] R. Deriche, Z. Zhang, Q.-T. Luong, and O. Faugeras. Robust recovery of the epipolar geometry for an uncalibrated stereo rig. In *Third European Conference on Computer Vision (ECCV'94)*, pages 567–576, Springer-Verlag, Stockholm, Sweden, May 1994.

[Faugeras, 1993] O. Faugeras. *Three-dimensional computer vision: A geometric viewpoint*. MIT Press, Cambridge, Massachusetts, 1993.

[Faugeras, 1992] O. D. Faugeras. What can be seen in three dimensions with an uncalibrated stereo rig? In *Second European Conference on Computer Vision (ECCV'92)*, pages 563–578, Springer-Verlag, Santa Margherita Liguere, Italy, May 1992.

[Ferrie and Levine, 1987] 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.

[Freedman, 1995] D.H. Freedman. A camera for near, far, and wide. *Discover*, 16(48):48, November 1995.

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

[Higuchi *et al.*, 1993] 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.

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

[Kang *et al.*, 1995a] 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.

[Kang *et al.*, 1995b] 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.

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

[Kuglin and Hines, 1975] 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.

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

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

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

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

[Parvin and Medioni, 1992] 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.

[Press *et al.*, 1992] 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.

[Shabana, 1989] A. A. Shabana. *Dynamics of Multibody Systems*. J. Wiley, New York, 1989.

[Shashua, 1994] A. Shashua. Projective structure from uncalibrated images: Structure from motion and recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(8):778–790, August 1994.

[Shi and Tomasi, 1994] 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, IEEE Computer Society, Seattle, Washington, June 1994.

[Shum *et al.*, 1994] 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, IEEE Computer Society, Seattle, Washington, June 1994.

[Stein, 1995] 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.

[Szeliski, 1994] R. Szeliski. *Image Mosaicing for Tele-Reality Applications*. Technical Report 94/2, Digital Equipment Corporation, Cambridge Research Lab, June 1994.

[Szeliski and Coughlan, 1994] 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, IEEE Computer Society, Seattle, Washington, June 1994.

[Szeliski and Kang, 1994] 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.

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