

# Extending Shape-from-Motion to Noncentral Omnidirectional Cameras

Dennis Strelow, Jeff Mishler, Sanjiv Singh, and Herman Herman

*Carnegie Mellon University*

{dstrelow, jmishler, ssingh, herman}@cs.cmu.edu

## Abstract

*Algorithms for shape-from-motion simultaneously estimate camera motion and scene structure. When extended to omnidirectional cameras, shape-from-motion algorithms are likely to provide robust motion estimates, in particular, because of the camera's wide field of view. In this paper, we describe both batch and online shape-from-motion algorithms for omnidirectional cameras, and a precise calibration technique that improves the accuracy of both methods.*

*The shape-from-motion and calibration methods are general and handle a wide variety of omnidirectional camera geometries. In particular, the methods do not require that the camera-mirror combination have a single center of projection. We describe a noncentral camera that we have developed, and show experimentally that combining shape-from-motion with this design produces highly accurate motion estimates.*

## 1. Introduction

Cameras are attractive sensors for autonomous vehicles because they are inexpensive and lightweight; because the information provided by cameras compliments that provided by other sensors; and because unlike laser range finders or sonar, cameras are not inherently limited in range. Algorithms for shape-from-motion exploit on-board cameras by simultaneously estimating camera motion and scene structure. When extended to omnidirectional cameras, shape-from-motion algorithms are likely to provide robust motion estimates, in particular, because of the camera's wide field of view. In this paper we consider omnidirectional shape-from-motion for motion estimation of autonomous ground and air vehicles.

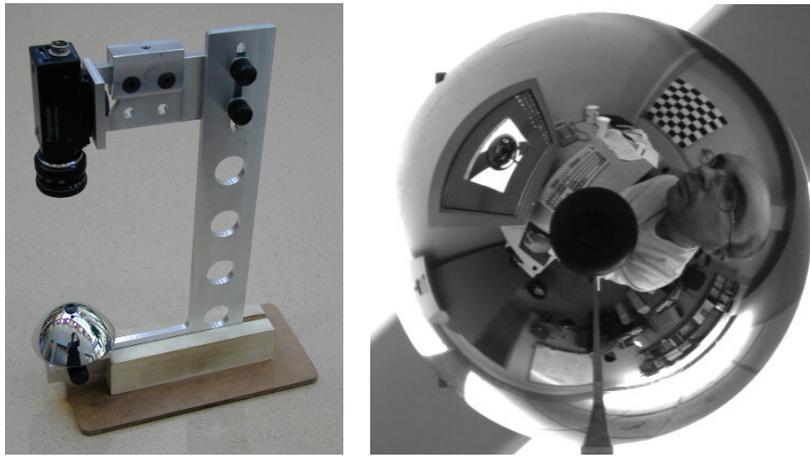
Recent omnidirectional cameras combine a conventional camera with a mirror that multiplies the camera's field of view. An omnidirectional camera developed at Carnegie Mellon is shown in Figure 1, along with a typical output image. Besides producing more robust motion estimates, omnidirectional cameras are well-suited to autonomous navigation because they can see the entire scene without the use of an additional and potentially heavy pan-tilt mechanism.

In this paper we present two shape-from-motion algorithms for omnidirectional cameras. The first is a batch method that determines shape and motion estimates using all of the image observations simultaneously. This method produces optimal estimates and is a useful tool for studying fundamental aspects of the problem, including the sensitivity of solutions. The second method is an online method, which processes each image as it becomes available. Because this method does not require that all of the image data be available before execution begins, it is suitable for use on infinite image sequences, and is intended for deployment on autonomous ground and air vehicles. The accuracy of both methods is improved by a precise calibration technique for omnidirectional cameras that we have developed. This calibration method is also described briefly.

The remaining sections are organized as follows. Section 2 summarizes previous related work on shape-from-motion and omnidirectional cameras. Section 3 describes the relevant properties of the omnidirectional camera design, our calibration method, and the two shape-from-motion algorithms. Section 4 presents experimental results and compares recovered motion estimates with ground truth. The final section offers conclusions and outlines planned future work.

## 2. Related Work

Approaches to shape-from-motion for conventional cameras that can be generalized to handle omnidirectional cameras can be categorized as either *batch* or *online* methods. Batch methods, such as Szeliski and Kang[19], typically apply general nonlinear minimization methods to all observations simultaneously, producing optimal estimates for the shape and motion. Bundle adjustment[20], which has long been used in photogrammetry for mapping from aerial photographs, falls into this category. Online methods, such as Broida, *et al.*[3] and Azarbayejani and Pentland[1], use a Kalman filter to refine the motion and shape estimates as each image arrives. The time complexity of online methods is linear in the number of images, whereas batch methods are typically cubic in the number of images. We have developed omnidirectional shape-from-motion algorithms based on both of these paradigms, which we describe in the fol-



**Figure 1. The omnidirectional camera used in our experiments and a typical output image.**

lowing sections.

Recently, researchers have been highly active in the design of omnidirectional cameras and their application; [7] contains a sampling of recent work in this area. Nayar’s design[15], which combines an orthographic camera with a parabolic mirror, is the best known. This camera achieves a single viewpoint, i.e., rays that are reflected from the mirror onto the sensor all intersect in a single point inside the mirror. This property allows portions of the omnidirectional image to be remapped into conventional perspective images; allows the image of a world point to be computed easily; and allows the use of computer vision techniques that depend on the existence of a camera center, such as epipolar geometry[18]. However, this system has lower resolution in the center of the image than in the periphery. An alternative design which sacrifices the single viewpoint for approximately uniform resolution is described by Chahl and Srinivasan[4]. Our own mirror[16], which is used in the experiments described here and is described in more detail in Section 3, is a refinement of this design that provides exact uniformity.

There has already been some work on the use of omnidirectional cameras for motion estimation and autonomous navigation. Gluckman and Nayer[9] extended three algorithms for ego-motion estimation with conventional cameras to the single viewpoint omnidirectional case. Gluckman and Nayar’s work recovers camera motion independent of scene structure using the differential form of the epipolar constraint; similarly, work by Peng and Hebert[6] recovers the motion using the normal form of the epipolar constraint, which is more suitable to large interframe motions. Both of these methods differ from ours in that we recover scene structure and motion simultaneously. Work by Deans and Hebert[8] in Simultaneous Localization and Mapping (SLAM) uses an omnidirectional camera and odometry to simultaneously determine the motion of a rover and the location of landmarks in the rover’s environment. Because

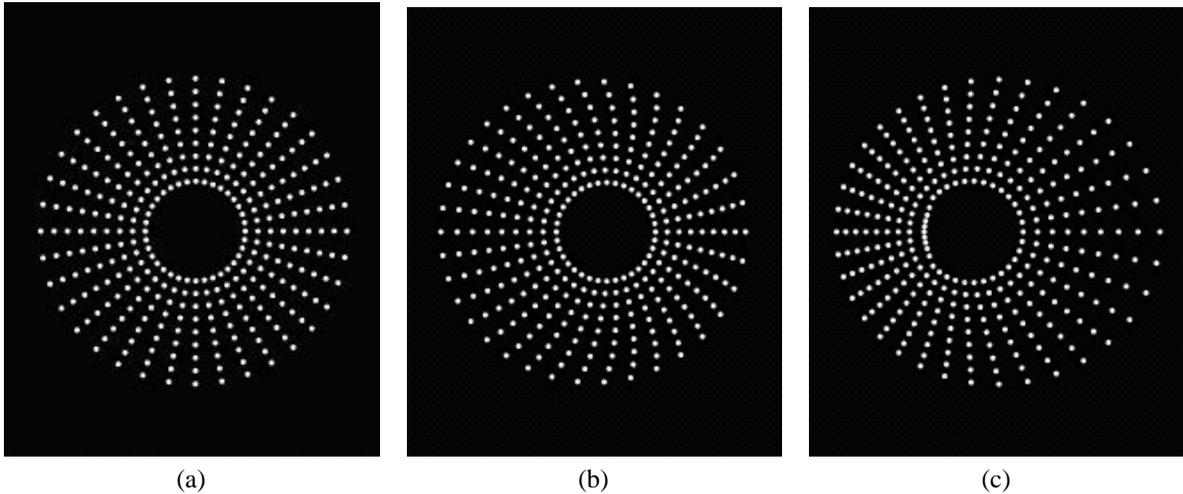
this work is targeted at recovering motion and landmark locations in a plane, it uses only the bearing, i.e., the radial direction, of landmark points in the omnidirectional image. In contrast, our method uses both components of the image location but no odometry because it is targeted in part at the recovery of six degree of freedom motion for air vehicles. Systems that recover a dense description of the environment from omnidirectional images include Chahl and Srinivasan[5], which uses the known motion of an omnidirectional camera to determine the approximate range to obstacles in the camera’s vicinity; and Kang and Szeliski[11] which constructs dense depth maps from panoramic views that are created by rotating a conventional camera.

### 3. Method

#### 3.1 Shape-from-motion overview

A two-stage approach is employed when applying shape-from-motion to video. In the first stage, sparse, two-dimensional point features are extracted and tracked through the image sequence. In our experiments we have used the Lucas-Kanade tracker[12][2], which has shown good performance on conventional sequences in comparative studies. Lucas-Kanade is an iterative method that tracks to sub-pixel resolution; one-tenth of a pixel is the accuracy typically cited. In our experience the method performs well on omnidirectional image sequences, which contain larger rotational motion than is typical in conventional sequences, although like all trackers, the method has difficulties when tracked points become occluded or change appearance.

In the second stage, the resulting feature tracks are used to recover the camera’s motion and the three-dimensional locations of the tracked points by minimizing the error between the tracked point locations and the image locations predicted by the shape and motion estimates. Because we recover a three-parameter rotation and a three-dimensional



**Figure 2.** When the equiangular camera’s mirror axis and camera axis are identical, vertical lines in the environment appear as radial lines in the image, as shown in (a). When these axes are not identical, the resulting image is distorted. Images (b) and (c) show the distortion that occurs if the mirror is translated approximately 2 cm along its  $x$  axis, or rotated approximately 0.2 radians about its  $y$  axis, respectively.

translation for each image, and a three-dimensional position for each point, the total number of estimated parameters is  $6f + 3p$ , where  $f$  is the number of images and  $p$  is the number of points. In the experiments described here, the number of images used varies from forty-one to 150, and the number of points tracked varies from thirty-two to 112, but even wider variability can be found in typical shape-from-motion applications.

Shape-from-motion recovers shape and motion estimates only up to a scaled rigid transformation. That is, applying the same scaling and rigid transformation to all of the camera and point estimates produces new estimates that explain the observed data as well as the originals. In many applications it is not necessary to remove this ambiguity, but for autonomous navigation, this ambiguity must be removed by augmenting shape-from-motion with appropriate additional sensors. We will address this issue in future work.

### 3.2 Equiangular omnidirectional cameras

Suppose that an omnidirectional camera is placed at the center of a sphere, and that points placed along a longitudinal line of the sphere, and separated by a uniform angular distance  $\Delta\phi$ , are imaged. If the camera’s resolution is uniform from the center of the image to the periphery, then the resulting image points are separated by some uniform distance  $\Delta r$  along the image’s radial lines, as shown in Figure 2(a). We call such a camera *equiangular*. We have designed and fabricated a number of equiangular cameras, and the camera used for the experiments described in this paper is shown in Figure 1.

Uniform resolution is an advantage of equiangular cameras over single viewpoint designs[15], but computing the

projection (i.e., the image location) of a three-dimensional point is more difficult with an equiangular camera than with a single viewpoint camera. This problem reduces to finding the height  $z$  and azimuth  $\theta$  of the point  $m$  on the mirror at which the angle of incidence (i.e., the angle to the three-dimensional point  $p$ ) equals the angle of reflection (i.e., the angle to the camera’s center  $c$ ). These constraints produce a system of two nonlinear equations in  $z$  and  $\theta$  which we solve numerically:

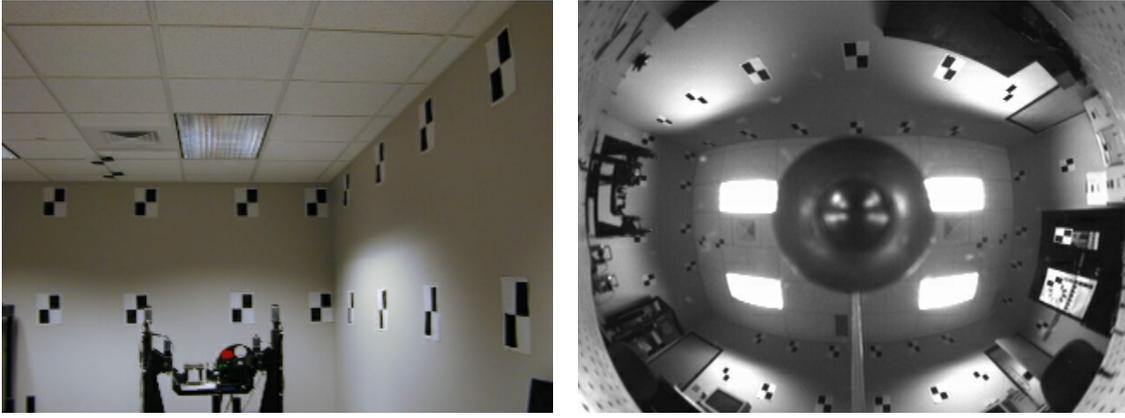
$$\frac{b_1 \cdot dp}{b_3 \cdot dp} = \frac{-b_1 \cdot dc}{b_3 \cdot dc} \quad \frac{b_2 \cdot dp}{b_3 \cdot dp} = \frac{-b_2 \cdot dc}{b_3 \cdot dc} \quad (1)$$

Here,  $B = \{b_1, b_2, b_3\}$  is an orthogonal basis for a local coordinate system on the mirror, relative to the global mirror coordinate system.  $B$  is centered on the mirror point  $m$ , with  $b_1$  and  $b_2$  spanning a plane tangent to the mirror at  $m$ .  $dp$  and  $dc$  are the directions from  $m$  to the three-dimensional point  $p$  and the camera center  $c$ , respectively, expressed in the global mirror coordinate system.  $B$ ,  $dp$ , and  $dc$  are all functions of the mirror point  $m$ . Once  $m$  has been determined, the projection can be determined from  $m$  using the conventional projection model.

Our shape-from-motion algorithms also require the derivative of a projection with respect to the components  $(x_p, y_p, z_p)$  of the three-dimensional point  $p$ . Differentiating the equations (1) in  $z$  and  $\theta$  with respect to  $x_p$ ,  $y_p$ , and  $z_p$  produces three  $2 \times 2$  linear systems that determine columns of the derivative:

$$\frac{\partial m}{\partial p} = \begin{bmatrix} \frac{\partial z}{\partial x_p} & \frac{\partial z}{\partial y_p} & \frac{\partial z}{\partial z_p} \\ \frac{\partial \theta}{\partial x_p} & \frac{\partial \theta}{\partial y_p} & \frac{\partial \theta}{\partial z_p} \end{bmatrix} \quad (2)$$

With these derivatives in hand the derivatives of the projection with respect to  $p$  can be found using the chain rule.



**Figure 3. Targets on the wall and ceiling of the calibrated imaging lab (left) and an omnidirectional image of the targets used for calibrating the mirror-to-camera transformation (right).**

### 3.3 Calibration

Our camera design produces equiangular projections if the mirror axis is identical to the camera axis. As shown in Figures 2(b) and 2(c), translations or rotations of the mirror away from this ideal position produce distortions in the projections. The projection equations described in Section 3.2 accommodate such misalignment, but the transformation from the mirror coordinate system to the camera coordinate system must be known.

A precise calibration of our camera therefore requires that we determine the mirror-to-camera transformation in addition to the conventional intrinsic parameters (e.g., the focal length and radial distortion) of the camera. We determine the mirror-to-camera transformation by imaging a set of known three-dimensional points (as shown in Figure 3), and then minimizing the error between the observed and predicted projections with respect to the mirror-to-camera parameters. Because the mirror is a surface of revolution, the mirror-to-camera transformation is parameterized by five rather than six parameters. Since we do not assume the position of the camera is known with respect to the target locations, we must determine not only the position of the mirror with respect to the camera, but the position of the camera with respect to the target locations, which introduces an additional six parameters. So, we minimize with respect to eleven parameters total, although only five (the mirror-to-camera parameters) are subsequently needed.

The intrinsics (e.g., focal length and radial distortion coefficients) of the conventional camera that views the mirror can be determined using existing techniques. For these experiments we have used the method implemented in [13].

### 3.4 Batch shape-from-motion

Our batch shape-from-motion algorithm uses Levenberg-Marquardt to minimize the error between

observed projections (i.e., feature tracks) and predicted projections with respect to the camera and point estimates. In this section we briefly describe our error function, and refer the reader to [17] for the details of Levenberg-Marquardt, which is widely used.

Suppose we have  $f$  images and  $p$  points, and that the image location of feature  $i \in \{1, \dots, p\}$  in image  $j \in \{1, \dots, f\}$  is the two-vector  $x_{ij}$ . Let point  $i$  be represented by the three-vector  $X_i$  giving the point's location in a world coordinate system. We define a camera coordinate system for each image, and let camera position  $j$  be represented by the rotation  $R_j(\alpha_j, \beta_j, \gamma_j)$  and translation  $t_j = (t_{jx}, t_{jy}, t_{jz})$  of the world-to-camera coordinate system transformation for image  $j$ , where  $\alpha_j$ ,  $\beta_j$ , and  $\gamma_j$  are the Euler angles specifying the rotation.

Let  $\Pi : \mathcal{R}^3 \mapsto \mathcal{R}^2$  be the projection described in the Section 3.2, which gives the image location for a three-dimensional point.  $\Pi$  depends on the camera intrinsics, the mirror-to-camera transformation, and the mirror shape, which are all assumed known. Since  $\Pi$  operates on points specified in the camera coordinate system, the projection of a point  $X_i$  specified in the world system is  $\Pi(R_j X_i + t_j)$ .

The function we minimize is then:

$$\chi^2 = \sum \| x_{ij} - \Pi(R_j X_i + t_j) \|^2 \quad (3)$$

where the summation is over all  $i, j$  such that point  $i$  was observed in image  $j$ . Minimizing this function provides maximum likelihood estimates assuming uniform, isotropic Gaussian noise in the observations. If confidences or covariances are available for the observations, e.g., from the tracker, then this method can easily be modified to incorporate this information. Approximate covariances on the estimated parameters are given by the inverse of the Levenberg-Marquardt Hessian matrix when the method has converged.

To guarantee convergence, the initial estimate provided to the method must be sufficiently close to the solution. We currently use the estimate provided by the online method,

described in the next section, as the initial estimate to the batch method.

### 3.5 Online shape-from-motion

Our online shape-from-motion method uses an extended Kalman filter to estimate shape and motion and their covariances. Our method is similar to [3], but replaces the perspective camera model used in that method with the omnidirectional camera model described in the Section 3.2. Here, we describe the specifics of our method without describing the Kalman filter in general. See [14] for details on the Kalman filter in general, or [3] and [1] for more detailed information on Kalman filtering for conventional shape-from-motion.

When new observations are supplied to a Kalman filter, the filter performs a propagation step and a measurement step to determine a new estimate of the state and its covariances. In our application, the observations are the projection data for the current image, and the state estimate consists of a six degree of freedom camera position and a three-dimensional position for each point. So, the total size of the state estimate is  $6 + 3p$ , where  $p$  is the number of tracked points.

The propagation step of a Kalman filter uses a model to estimate the change in the state and its covariance since the last observations, without reference to the new observations. For instance, an airplane’s estimated position might be updated based on the position, velocity, and acceleration estimates at the last time step, and on the length of time between updates. In our current formulation we assume that three-dimensional points in the scene are static, but make no explicit assumptions about the motion of the camera. Therefore, our propagation step leaves the point and camera estimates unmodified, but adds a large uncertainty  $\alpha$  to each of the camera parameter estimates. With this simple model, an implicit assumption is made that the camera motion between observations is small, but this assumption is made weaker as  $\alpha$  is increased.

The measurement step of a Kalman filter uses a measurement equation and the new observations to refine the state estimates and covariances produced by the propagation step. For our application, the measurement equation is the projection equation  $x_{ij} = \Pi(R_j X_i + t_j)$ . This measurement equation is nonlinear in the estimated parameters, so it is necessary to use the extended Kalman filter, which linearizes the measurement equation about the current estimate, rather than the standard filter. In our filter essentially all of the computation occurs in the measurement step, which adjusts the camera and point estimates to fit the new observations, since the propagation step does not modify the estimates.

As described, the filter is susceptible to errors in the two-dimensional tracking. To improve performance in the face of mistracking, we discard the point with highest residual after the measurement step if the residual is over some

threshold (e.g., two pixels). The measurement step is then re-executed from the propagation step estimate, and this process is repeated until no points have a residual greater than the threshold. We have found this to be an effective method for identifying points that are mistracked, become occluded, or are on independently moving objects in the scene.

We initialize the filter’s point estimates by backprojecting the projections in the first image onto a sphere centered on the camera, and setting the point uncertainties to be very high in the backprojection direction, and low in the perpendicular directions. While this initialization method would be inappropriate for scenes in which the distance to points varies by many orders of magnitude, this method has worked well for the sequences to which we have applied it.

## 4. Results

In the following subsections, we present three initial tests of our omnidirectional shape-from-motion algorithms. Ground truth measurements for the camera motion are available in all three tests; in the calibrated imaging lab test, ground truth measurements for the three-dimensional points are also available, although motion rather than shape recovery is our emphasis here.

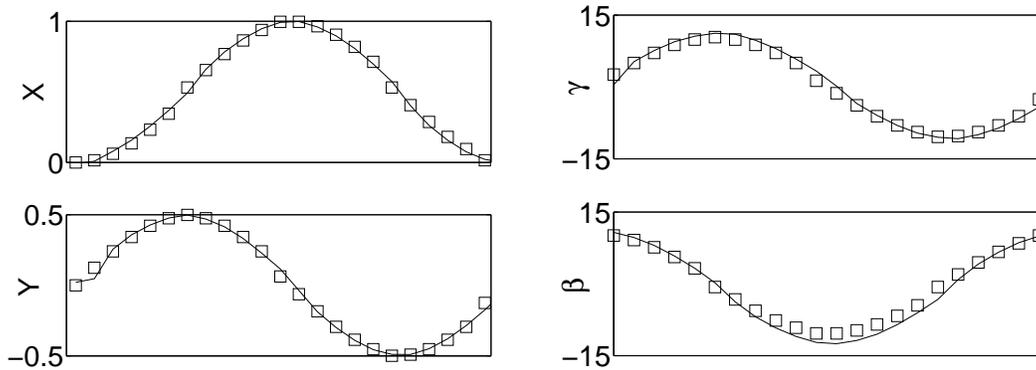
In each test we have used the Lucas-Kanade algorithm to perform feature extraction and tracking. The online shape-from-motion method is then used to construct shape and motion estimates and to identify mistracked points, and the estimates are refined by the batch algorithm. Only those projections identified as inliers by the online method are used as input to the batch minimization.

As mentioned in Section 3.1, camera and point estimates from shape-from-motion are correct only up to a scaled rigid transformation. Therefore, before computing errors we have transformed the camera and point estimates using the scaled rigid transformation that best aligns the estimates with the ground truth values. This transformation can be found in closed form using the algorithm described by Horn[10].

### 4.1 Calibrated imaging lab

As described in Section 3.3 the locations of forty-four calibration targets on the walls and ceiling of Carnegie Mellon’s calibrated imaging lab have been precisely surveyed. In our first experiment, we have taken a sequence of forty-one images of these targets by moving the camera in a known pattern on the lab’s optical bench, producing a sequence with ground-truth values for both the motion and the shape. The motion was a “U” shape along three edges of the optical bench, so that the camera path contained some motion parallel to each wall. One image from the sequence is shown in Figure 3.

The locations of forty-two of the targets were extracted by hand and tracked using Lucas-Kanade; tracking errors



**Figure 4. Ground truth (squares) and estimated (lines) motion for the nonzero motion components in the crane experiment. The unit of measurement for the vertical axes is meters for  $x$  and  $y$  and degrees for pitch  $\beta$  and roll  $\gamma$ . The horizontal axis of each plot spans the image sequence of 140 images.**

were detected and fixed by hand. The average absolute depth error in the recovered points, as measured from the centermost camera, was 2.4 cm; this is an average relative depth error of 0.8%. The average camera translation error was 0.48 cm, which is a 0.12% error measured relative to the total distance traveled (4.08 m). Although we don't know the ground truth camera rotation, we know that all cameras had the same rotation; the average difference of the rotation estimates from the mean rotation was 0.40 degrees.

## 4.2 Robocrane

The Robocrane is a twenty-foot high robotic crane (more specifically, an inverted Stewart platform) built by the National Institute of Standards and Technology (NIST). The Robocrane features a large triangular platform whose six degree of freedom position within the  $10 \times 10 \times 10$  foot workspace can be specified by computer. Because the Robocrane's platform can be moved to known positions in such a large space above the ground, it is ideal for creating simulated air vehicle motions with ground truth.

The results of one Robocrane experiment are shown in Figure 4. In this experiment, the platform and the attached camera were translated in a horizontal circle, with the camera's pitch and roll chosen to maintain a gaze on a single ground point. In a sequence of 140 images, thirty-two points were tracked, of which ten were eventually mis-tracked by Lucas-Kanade and detected as outliers by the Kalman shape-from-motion method. The average translation error in this example is 2.4 cm, which is a 0.79% error relative to the total distance traveled (3.08 m); the average rotational error is 1.61 degrees.

## 4.3 Ground rover

As an initial test of our algorithm's suitability for autonomous ground vehicle navigation, we have attached the

omnidirectional camera to our ground rover. The distances and attitudes provided by the rover's on-board encoders and fiber optic gyro, respectively, can be integrated to provide a dead reckoning estimate of the rover's motion, which we take as the ground truth motion.

The results are shown in Figure 5. In this experiment, 112 points were tracked in a sequence of 150 images. Seven points were identified as outliers by the Kalman shape-from-motion method. The average three-dimensional translation error in this example is 4.00 cm, which is a 1.02% error relative to the total distance traveled (3.91 m). Because the rover moves in a plane, we assume that the  $z$  component of the ground truth motion is zero; the average absolute  $z$  of the estimated positions is 0.8 cm. The average error in yaw is 0.48 degrees.

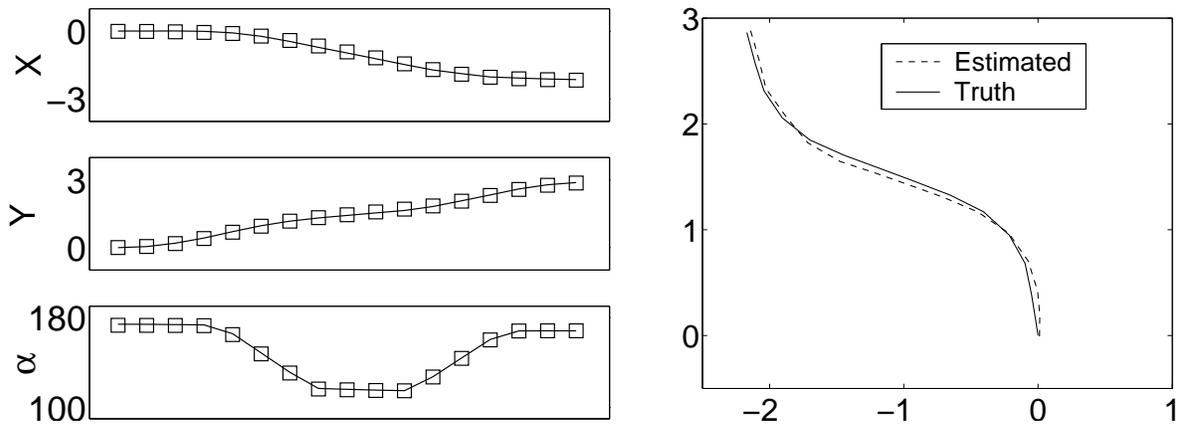
## 5. Conclusions

We have shown that it is possible to determine precise motion using shape-from-motion with omnidirectional cameras, even in the presence of mistracking. In the future, we plan to investigate: the integration of additional sensors, such as inertial motion sensors, into this framework; the reacquisition of features in long traverses; and particularly, the recovery of dense depth for obstacle detection, using the known camera positions recovered by our current method.

We have placed additional materials related to this paper on the web at <http://www.frc.ri.cmu.edu/~jmishler/iros01/>. These materials include movies of the sequences for our tests, and VRML files showing the resulting shape and motion estimates.

## Acknowledgements

Special thanks to Mark Ollis, who designed the mirror used in our experiments and supplied a precise numerical



**Figure 5. Left: ground truth (squares) and estimated (lines) motion for the nonzero motion components in the rover experiment. The unit of measurement for the vertical axes of the plots is meters for  $x$  and  $y$  and degrees for yaw. The horizontal axis of each plot spans the image sequence of 150 images. Right: the ground truth (solid line) and estimated (dashed line)  $xy$  motion as seen from above; the units on both axes are meters.**

description of the mirror's shape; Ray Russell, who fabricated our omnidirectional camera mount; Tim Hegadorn, who surveyed the three-dimensional locations of the calibration targets in the calibrated imaging lab; and Trey Smith and Dave Hershberger, whose expertise and patience made our Robocrane and rover experiments possible.

## References

- [1] A. Azarbayejani and A. Pentland. Recursive estimation of motion, structure, and focal length. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(6):562–575, June 1995.
- [2] S. Birchfield. <http://vision.stanford.edu/~birch/klt/>.
- [3] T. J. Broida, S. Chandrashekar, and R. Chellappa. Recursive 3-D motion estimation from a monocular image sequence. *IEEE Transactions on Aerospace and Electronic Systems*, 26(4):639–656, July 1990.
- [4] J. Chahl and M. Srinivasan. Reflective surfaces for panoramic imaging. *Applied Optics*, 36(31):8275–8285, 1997.
- [5] J. Chahl and M. Srinivasan. A complete panoramic vision system, incorporating imaging, ranging, and three-dimensional navigation. In Daniilidis [7].
- [6] P. Chang and M. Hebert. Omni-directional structure from motion. In Daniilidis [7].
- [7] K. Daniilidis, editor. *IEEE Workshop on Omnidirectional Vision*, Hilton Head Island, South Carolina, June 2000.
- [8] M. Deans and M. Hebert. Experimental comparison of techniques for localization and mapping using a bearing-only sensor. In *International Conference on Experimental Robotics*, Honolulu, Hawaii, December 2000.
- [9] J. Gluckman and S. K. Nayar. Ego-motion and omnidirectional cameras. In *IEEE International Conference on Computer Vision*, pages 999–1005, Bombay, 1998.
- [10] B. K. P. Horn, H. M. Hilden, and S. Negahdaripour. Closed-form solution of absolute orientation using orthonormal matrices. *Journal of the Optical Society of America A*, 5(7), July 1988.
- [11] S. B. Kang and R. Szeliski. 3-D scene data recovery using omnidirectional multibaseline stereo. In *IEEE Computer Vision and Pattern Recognition*, pages 364–370, 1996.
- [12] B. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. In *Proceedings of the Seventh International Joint Conference on Artificial Intelligence*, 1981.
- [13] Intel Corporation. Open source computer vision library. <http://www.intel.com/research/mrl/research/cvlib/>.
- [14] The Analytic Sciences Corporation. *Applied Optimal Estimation*. MIT Press, Cambridge, Massachusetts, 1974.
- [15] S. K. Nayar. Catadioptric omnidirectional camera. In *IEEE Computer Vision and Pattern Recognition*, pages 482–488, 1997.
- [16] M. Ollis, H. Herman, and S. Singh. Analysis and design of panoramic stereo vision using equi-angular pixel cameras. Technical Report CMU-RI-TR-99-04, Carnegie Mellon University Robotics Institute, Pittsburgh, PA, 1999.
- [17] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery. *Numerical Recipes in C: The Art of Scientific Computing*. Cambridge University Press, Cambridge, 1992.
- [18] T. Svoboda and T. P. V. Hlavac. Epipolar geometry for panoramic cameras. In *Fifth European Conference on Computer Vision*, pages 218–232, Freiburg, Germany, June 1998.
- [19] 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.
- [20] P. R. Wolf. *Elements of Photogrammetry*. McGraw-Hill, New York, 1983.