Omnidirectional Visual Odometry for a Planetary Rover

Peter Corke*† and Dennis Strelow† and Sanjiv Singh†
* CSIRO ICT Centre, Queensland, Australia
peter.corke@csiro.au
† Robotics Institute, Carnegie-Mellon University, Pittsburgh, USA
{dstrelow,ssingh}@ri.cmu.edu

***Abstract*— Position estimation for planetary rovers has been typically limited to odometry based on proprioceptive measurements such as the integration of distance traveled and measurement of heading change. Here we present and compare two methods of online visual odometry suited for planetary rovers. Both methods use omnidirectional imagery to estimate motion of the rover. One method is based on robust estimation of optical flow and subsequent integration of the flow. The second method is a full structure-from-motion solution. To make the comparison meaningful we use the same set of raw corresponding visual features for each method. The dataset is an sequence of 2000 images taken during a field experiment in the Atacama desert, for which high resolution GPS ground truth is available.**

I. Introduction

Since GPS is not available on Mars, estimating rover motion over long distances by integrating odometry measurements inevitably produces estimates that drift due to odometry measurement noise and wheel slippage. Recent experiments have shown that in planetary analog environments such as the Atacama Desert in Chile, odometric error is approximately 5 percent of distance traveled. Such error can increase further in loose soil because wheels can slip considerably. We would like a method that compensates for such error.

Much has been written in the biological literature about estimation of motion using sequences of visual images and several research efforts have attempted to use these concepts for robotics (e.g. [1]). This type of work seeks inspiration from a number of different ways in which insects use cues derived from optical flow for navigational purposes, such as safe landing, obstacle avoidance and dead reckoning. We have similar motivations but seek analytical methods with high accuracy.

Motion estimation from imagery taken from onboard cameras has the potential to greatly increase the accuracy of rover motion estimation because images and the rover's motion can be used together to establish the three-dimensional positions of environmental features relative to the rover, and because the rover's position can in turn be estimated with respect to these external landmarks over the subsequent motion. While visual odometry has its own drift due to discretization and mistracking of visual features, the advantage is that it is not correlated with the errors associated with wheel and gyro based odometry.

Fig. 1
Hyperion is a solar power robot developed at Carnegie Mellon University intended for autonomous navigation in planetary analog environments.

Relative to conventional cameras, omnidirectional (panospheric) cameras trade resolution for an increased field of view. In our experience this tradeoff is beneficial for motion estimation, and as others have shown, estimating camera motion from omnidirectional images does not suffer from some ambiguities that conventional image motion estimation suffers from [2]. This is primarily because in an environment with sufficient optical texture, motion in any direction produces good optical flow. This is in contrast to conventional cameras that require that cameras be pointed orthogonal to the direction of motion. In addition, as the camera moves through the environment, environmental features whose three-dimensional positions are established are retained longer in the wide field of view of an omnidirectional camera than in a conventional camera's field of view, providing a stronger reference for motion estimation over long intervals.

Our approach uses a single camera rather than stereo cameras for motion estimation. An advantage of this method is that the range of external points whose three-dimensional positions can be established is larger than the

range of external points whose three-dimensional positions can be established by stereo cameras since the baseline over which points can be estimated in the former method can be much larger. A strategy that we have not investigated, but that has been examined by some other researchers (e.g., [3]) integrates both stereo pairs and feature tracking over time, and this is a promising approach for the future.

This paper compares two methods of online (not batch) visual odometry. The first method is based on robust optical flow from salient visual features tracked between pairs of images. The terrain around the robot is approximated to be a plane and a displacement is computed for each frame in the image using an optimization method that also computes camera intrinsics and extrinsics at every step. Motion estimation is done by integrating the three-DOF displacement found at each step. The second method, implemented as an iterated extended Kalman filter estimates both the motion of the camera as well as the three dimensional location of visual features in the environment. This method makes no assumption on the planarity of visual features and tracks these features over many successive images. In this case the six-DOF pose of the camera as well as the three-DOF position of the feature points are extracted. We report comparisons of visual odometry generated by these two methods on a sequence of 2000 images taken during a desert traverse. To make the comparison meaningful we use the same set of raw corresponding visual features for each method.

II. Experimental platform

A. Hyperion

Carnegie Mellon's Hyperion, shown in Figure 1, is a solar powered rover testbed for the development of science and autonomy techniques suitable for large-scale explorations of life in planetary analogs such as the Atacama Desert in Chile. Hyperion's measurement and exploration technique combines long traverses, sampling measurements on a regional scale, and detailed measurements of individual targets. Because Hyperion seeks to emulate the long communication delays between Earth and Mars, it must be able to perform much of this exploration autonomously, including the estimation of the rover's position without GPS. Having been demonstrated to autonomously navigate over extended periods of time in the Arctic Circle during the summer of 2001, Hyperion was recently used in field tests in Chile's Atacama Desert on April 5-28, 2003.

B. Omnidirectional camera

Recent omnidirectional camera designs combine a conventional camera with a convex mirror that greatly expands the camera's field of view, typically to 360 degrees in azimuth and 90-140 degrees in elevation. On five days during Hyperion's field test, the rover carried an omnidirectional camera developed at Carnegie Mellon and logged high-resolution color images from the camera for visualization and for motion estimation experiments. This camera is shown in Figure 3.

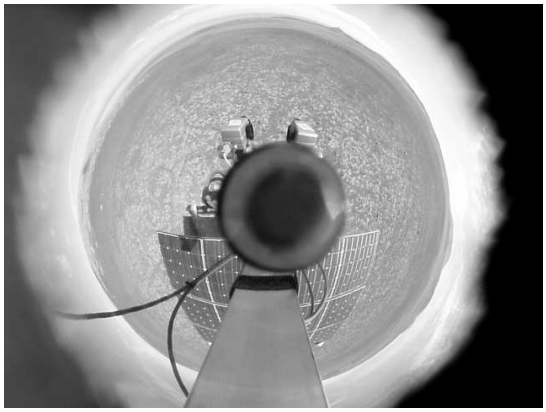

Fig. 2
An example omnidirectional image from our sequence, taken by Hyperion in the Atacama desert.

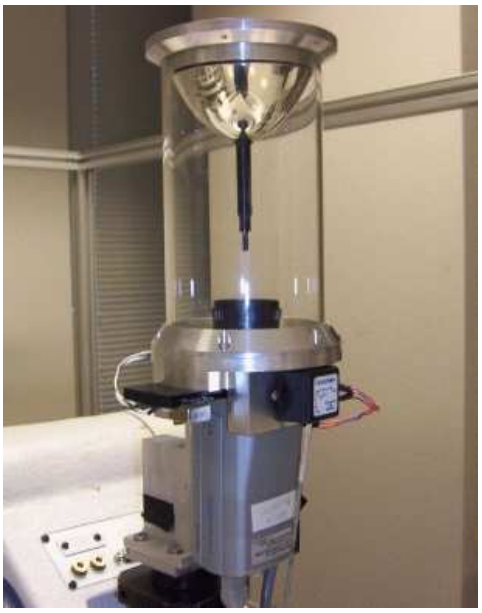

Fig. 3
The omnidirectional camera used in our experiments. The mirror used has a profile that produces equi-angular resolution. That is, each pixel in the radial direction has exactly the same vertical field of view. This camera was designed and fabricated at Carnegie Mellon University.

An example image taken from the omnidirectional camera while mounted on Hyperion is shown in 2. The dark circle in the center of the image is the center of the mirror, while the rover solar panel is visible in the bottom central part of the image. The ragged border around the outside of the image is an *ad-hoc* iris constructed in the field to prevent the sun from being captured in and saturating the images.

The camera design is described by [4] and is summarized in Figure 5. The mirror has the property that the angle of the outgoing ray from vertical is proportional to the angle of the ray from the camera to the mirror, see Figure 5. The

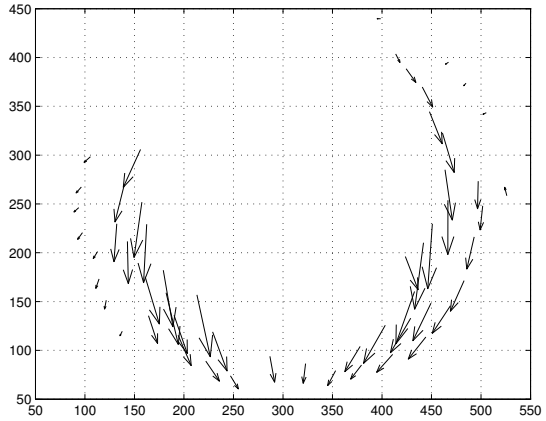

Fig. 4
TYPICAL FEATURE FLOW BETWEEN CONSECUTIVE FRAMES.

scale factor α is the elevation gain and is approximately 10.

III. FEATURE TRACKING

In this work we have investigated two approaches to feature tracking. The first is based on independently extracting salient features in image pairs and using correlation to establish correspondence. The search problem can be greatly reduced by using first-order image-plane feature motion prediction and constraints on possible inter-frame motion. This strategy involves no history or long term feature tracking. An example of this approach is [5] which uses the Harris corner extractor to identify salient feature points followed by zero-mean normalized cross correlation to establish correspondence.

An alternate strategy is to extract features in one image, and then use some variant of correlation to find the feature's position in the second image. Using this approach, the feature's location can be identified not only in the second image, but in every subsequent image where it is visible, and this advantage can be exploited to improve the estimates of both the point's position and the rover's motion by algorithms such as the online shape-from-motion algorithm described in section V.

One method for performing the best correlated feature location in the second and subsequent images is Lucas-Kanade [6], which uses Gauss-Newton minimization to minimize the sum of squared intensity errors between the intensity mask of the feature being tracked and the intensities visible in the current image, with respect to the location of the feature in the new image. Coupled with bilinear interpolation for computing intensities at non-integer location in the current image, Lucas-Kanade is capable of tracking features to subpixel resolution, and one-tenth of a pixel is an accuracy that is commonly cited.

One method for extracting features suitable for tracking with Lucas-Kanade chooses features in the original image that provide the best conditioning for the system that Lucas-Kanade's Gauss-Newton minimization solves on each iteration. We have used this method in our experiment, but in practice any sufficiently textured region of the image can be tracked using Lucas-Kanade.

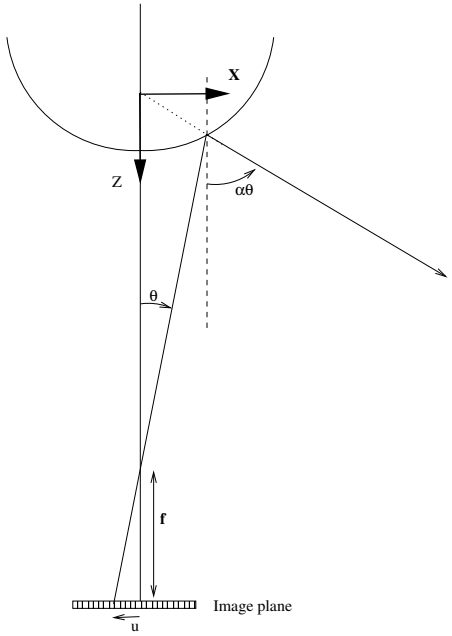

Fig. 5
PANORAMIC CAMERA NOTATION.

In this paper we have adopted the second of these paradigms, and used Lucas-Kanade to track features through the image sequence as long as they are visible. Although only pairwise correspondences are required by the robust optical flow approach described in Section IV, correspondences through multiple images are required for the online shape-from-motion approach that we describe in Section V. So, adopting this approach allows us to perform an meaningful comparison between the two methods using the same tracking data. A typical inter-frame feature flow pattern is shown in Figure 4.

IV. ROBUST OPTICAL FLOW METHOD

A. Algorithm

For each visual feature, (u,v), we can compute a ray in space as shown in Figure 5. From similar triangles we can write $\tan\theta = u/f$. We will approximate the origin to be at the center of the mirror so an arbitrary ray can be written in parametric form as

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \lambda \begin{bmatrix} a \\ b \\ 1 \end{bmatrix}$$

where $a = \tan\{\alpha \tan^{-1} u/f\}\cos\beta$ and $b = \tan\{\alpha \tan^{-1} v/f\}\sin\beta$. We will further assume that the ground is an arbitrary plane $Ax + By + Cz = 1$ which the ray intersects at the point on the line, λ, where

$$\lambda \begin{bmatrix} A & B & C \end{bmatrix} \begin{bmatrix} a \\ b \\ 1 \end{bmatrix} = 1$$

Thus an image-plane point (u,v) is projected onto the ground plane at (x,y). If the robot moves by $(\Delta x, \Delta y, \Delta\theta)$ that point becomes (x', y') which can be mapped back to the image plane as (u', v') from which we can compute the image-plane displacement or optical flow

$$(\hat{du}, \hat{dv}) = \mathcal{P}(u, v, \{u_0, v_0, f, \alpha\}, \{\Delta x, \Delta y, \Delta\theta\}) \quad (1)$$

which is a function of the feature coordinate, the camera intrinsic parameters (principle point (u_0, v_0), focal length f, and elevation gain α) and the vehicle motion. We assume that camera height, h, is known. Our observation is the displacement at a number of image coordinates, and our cost function is based on the median of the error norm between the estimated and observed displacement

$$e_1 = med\sqrt{(du_i - \hat{du}_i)^2 + (dv_i - \hat{dv}_i)^2}$$

We optimize over the intrinsic, (α, h, f, u_0, v_0), and motion parameters, $(\Delta x, \Delta y, \Delta\theta)$, to minimize e_1 and find the best fit to the observed data. We use the median statistic rather than the summation since it provides robustness to outliers albeit at greater computational expense. Our feature matching step also yields a confidence measure which could be used to weight the corresponding error but this is not currently implemented.

Optimization is currently achieved using Matlab's `fmincon()` function. Imposing constraints on the minimization was found to greatly improve the reliability of achieving a solution. Explicit gradients are not used, though (1) could be differentiated symbolically.

B. Results

Results are shown in Figure 6. The x-axis (forward) velocity of Hyperion is mostly positive, with some reversing early in the path and a stop around t=1400s. The parameter estimates are somewhat noisy but the median over the path, given below, agrees closely with those determined using a laboratory calibration method. The calibration procedure identifies separate focal lengths in x- and y-directions, but these are similar to within a few percent and represented here by a single focal length parameter.

Parameter	Value	True	Units
u_0	247.4	247	pix
v_0	199.1	199	pix
α	10.6		
f	1000.0	999	pix

The estimated angular rotation can be integrated to determine the heading angle of the vehicle. The relationship between the observed robot frame velocity and world-frame velocity from optical flow is given by

$$\begin{bmatrix} {}^R\dot{x} \\ {}^R\dot{y} \end{bmatrix} = R_Z(\theta) \begin{bmatrix} {}^W\dot{x} \\ {}^W\dot{y} \end{bmatrix}$$

which, given θ, allows us to solve for world-frame velocity which can then be integrated. Figure 7 compares the path due to visual odometry with GPS ground truth. The general form of the path is a qualitatively good match, but with some event that introduces a significant heading error near the point $(100, -100)$ that is not yet fully understood. Zooming in on the region at the start of the path where there is turning and reversing we note that the agreement is less good.

Interestingly we notice that the pose recovered by this method is very dependent on the feature tracker used. Using the Harris corner detector [5], which generates a greater number of corresponding features, we find that the accuracy is improved in the short term through the reversing and turning phase but is worse over the longer path.

V. Structure from motion method

A. Algorithm

Our method for online shape-from-motion using omnidirectional cameras is an iterated extended Kalman filter (IEKF), and is a major refinement of the online shape-from-motion method that we described in [7]. In this section we give a concise overview of this method, without describing Kalman filtering in detail. See [8] for details on the Kalman filter in general, or [9] and [10] for detailed information on Kalman filtering for conventional shape-from-motion.

A Kalman filter maintains a Gaussian state estimate distribution, and refines this distribution over time as relevant new observations arrive by applying a propagation step and a measurement step. 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 is $6 + 3p$, where p is the number of tracked points visible in the current image.

The propagation step of a Kalman filter uses a model to estimate the change in the state distribution since the previous 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 α 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 α is increased.

The measurement step of a Kalman filter uses the new observations and a measurement model that relates them to the state to find a new state estimate distribution that is most consistent with both the new observations and the state distribution produced by the propagation step. For our application, the measurement model assumes that the tracked feature positions visible in the new image are the re-projections of the three-dimensional point positions projected onto the camera position, plus Gaussian noise.

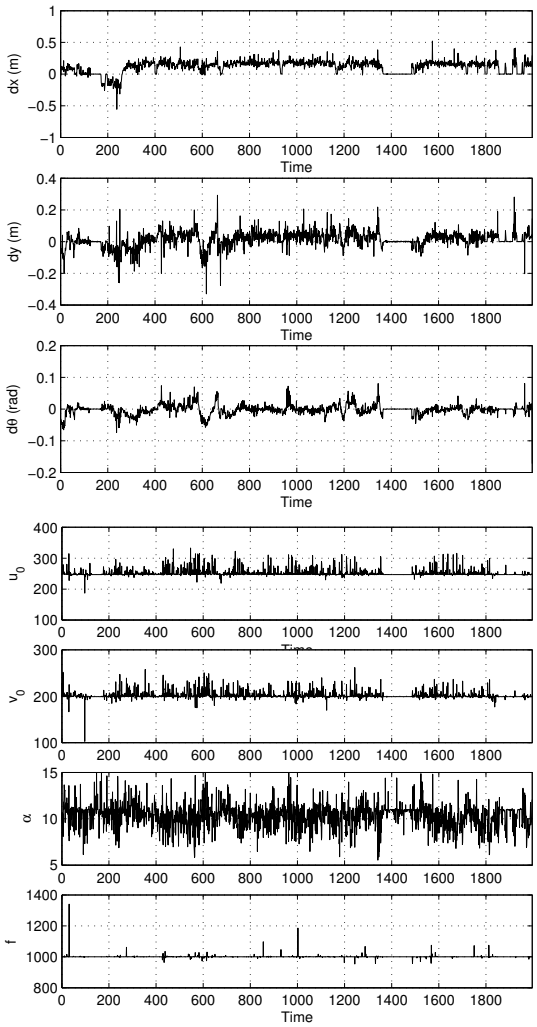

Fig. 6

RESULTS OF SIMULTANEOUS FIT TO MOTION (TOP) AND CAMERA INTRINSICS (BOTTOM). dx, dy, AND $d\theta$ ARE THE ESTIMATED INCREMENTAL MOTION BETWEEN FRAMES. (u_0, v_0) ARE THE COORDINATES OF THE PRINCIPLE POINT, α THE MIRROR'S ELEVATION GAIN AND f THE FOCAL LENGTH.

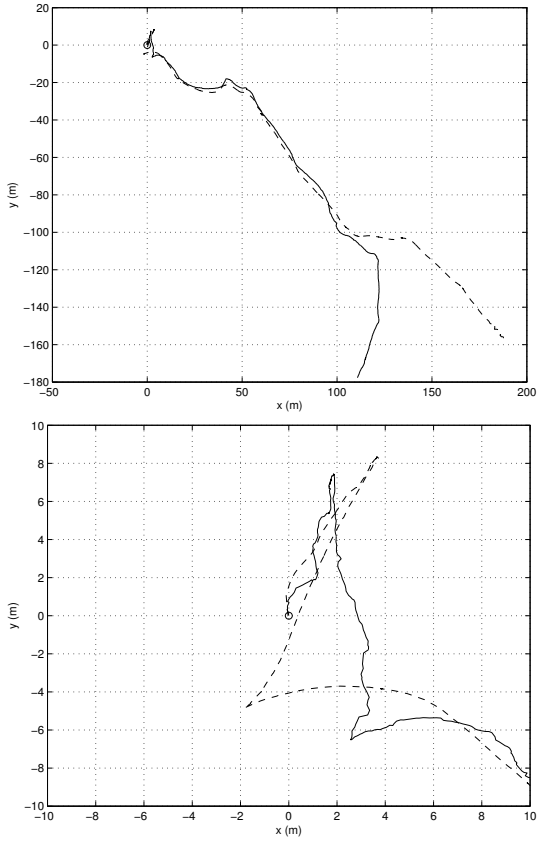

Fig. 7

COMPARISON OF PATH FROM INTEGRATED VELOCITY (SOLID) WITH GROUND TRUTH FROM GPS (DASHED). TOP IS ALL 2000 FRAMES, BOTTOM IS THE REGION AROUND THE STARTING POINT.

The re-projection of point j is:

$$x_j = \Pi(R(\rho)^{\mathrm{T}} X_j + t) \tag{2}$$

Here, ρ and t are the camera-to-world rotation Euler angles and translation of the camera, $R(\rho)$ is the rotation matrix described by ρ, and X_j is the three-dimensional world coordinate system position of point j, so that $R(\rho)^{\mathrm{T}} X_j + t$ is the camera coordinate system location of point j. Π is the omnidirectional projection model that computes the image location of the camera coordinate system point. This measurement equation is nonlinear in the estimated parameters, which motivates our use of the iterated extended Kalman filter rather than the standard Kalman filter, which assumes that observations are a linear function of the estimated parameters corrupted by Gaussian noise. We typically assume that the Gaussian errors in the observed feature locations are isotropic with variance (2.0 pixels)^2 in both image x and y directions.

As described, the filter is susceptible to gross errors in the two-dimensional tracking. To improve performance in the face of mis-tracking, we discard the point with highest residual after the measurement step if the residual is over some threshold. 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 mis-tracked, become occluded, or are on independently moving objects in the scene. We typically choose this threshold to be some fraction or small multiple of the expected observation variances, and in our experience choosing a threshold of less than a pixel generally produces the highest accuracy in the estimated motion. However, this requires a highly accurate camera calibration, and we revisit this point in our experimental results.

An initial state estimate distribution must be available before online operation can begin. We initialize both the mean and covariance that specify the distribution using a batch algorithm, which simultaneously estimates the six degree of freedom camera positions corresponding to the first

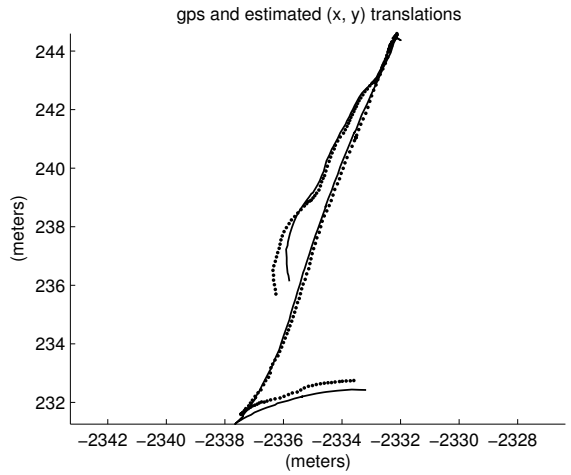

Fig. 8

THE GPS ESTIMATES OF THE (X, Y) TRANSLATION AND THE ESTIMATES FROM ONLINE SHAPE-FROM-MOTION ARE SHOWN AS THE SOLID AND DOTTED LINES, RESPECTIVELY.

several images in the sequence and the three-dimensional positions of the tracked features visible in those image. This estimation is performed by using Levenberg-Marquardt to minimize the re-projection errors for the first several images with respect to the camera positions and point positions, and is described in detail in [7]. We add points that become visible after online operation has begun to the state estimate distribution by adapting the method for incorporating new observations in simultaneous mapping and localization with active range sensors, described by [11], to the case of image data.

B. Results

As mentioned in previous sections, a potential advantage of online shape-from-motion is that it can exploit features tracked over a large portion of the image stream. In the first 300 images of the omnidirectional image stream from Hyperion, 565 points were tracked, with an average of 92.37 points per image and an average of 49.04 images per point.

The ground truth (i.e., GPS) and estimated (x, y) translations that result from applying the online shape-from-motion to the first 300 images are shown together in Figure 8, as the solid and dotted lines, respectively. Because shape-from-motion only recovers shape and motion up to an unknown scale factor, we have applied the scaled rigid transformation to the recovered estimate that best aligns it with the ground truth values. In these estimates, the average and maximum three-dimensional translation errors over the 300 estimated positions are 22.9 and 72.7 cm, respectively; this average error is less than 1% of the approximately 29.2 m traveled during the traverse. The errors, which are largest at the ends of the sequence, are due primarily to the unknown transformation between the camera and mirror. After image 300 this error increases until the filter fails. We are still investigating the details of this behavior.

VI. CONCLUSION

In this paper we have compared two approaches to visual odometry from a omnidirectional image sequence. The robust optical flow method is able to estimate camera intrinsic parameters as well as an estimate of vehicle velocity. The shape-from-motion technique produces higher precision estimation of vehicle motion but it comes at the expense of a larger computation expense. Our current experiments also indicate that it is important to have accurate calibration between the camera and the curved mirror for this technique.

We plan to extend this work to include fisheye lenses, and to incorporate inertial sensor data so as to improve the robustness and reliability of the recovered position.

ACKNOWLEDGMENTS

This work was conducted while the first author was a Visiting Scientist at the Robotics Institute over the period July-October 2003.

REFERENCES

[1] J. S. Chahl and M. V. Srinivasan, "Visual computation of egomotion using an image interpolation technique," *Biological Cybernetics*, 1996.

[2] C. F. Patrick Baker, Abhijit S. Ogale and Y. Aloimonos, "New eyes for robotics," in *Proc. Int. Conf on Intelligent Robots and Systems (IROS)*, 2003.

[3] C. F. Olson, L. H. Matthies, M. Schoppers, and M. W. Maimone, "Robust stereo ego-motion for long distance navigation," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2000)*, vol. 2, Hilton Head, South Carolina, June 2000, pp. 453–458.

[4] M. Ollis, H. Herman, and S. Singh, "Analysis and design of panoramic stereo vision using equi-angular pixel cameras," Carnegie Mellon University, Pittsburgh, Pennsylvania, Tech. Rep. CMU-RI-TR-99-04, January 1999.

[5] P. Corke, "An inertial and visual sensing system for a small autonomous helicopter," *J. Robotic Systems*, vol. 21, no. 2, pp. 43–51, Feb. 2004.

[6] B. D. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision," in *Seventh International Joint Conference on Artificial Intelligence*, vol. 2, Vancouver, Canada, August 1981, pp. 674–679.

[7] D. Strelow, J. Mishler, S. Singh, and H. Herman, "Extending shape-from-motion to noncentral omnidirectional cameras," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2001)*, Wailea, Hawaii, October 2001.

[8] A. Gelb, Ed., *Applied Optimal Estimation*. Cambridge, Massachusetts: MIT Press, 1974.

[9] T. J. Broida, S. Chandrashekhar, and R. Chellappa, "Recursive 3-D motion estimation from a monocular image sequence," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 26, no. 4, pp. 639–656, July 1990.

[10] A. Azarbayejani and A. P. Pentland, "Recursive estimation of motion, structure, and focal length," *IEEE Transations on Pattern Analysis and Machine Intelligence*, vol. 17, no. 6, pp. 562–575, June 1995.

[11] R. Smith, M. Self, and P. Cheeseman, "Estimating uncertain spatial relationships in robotics," in *Autonomous Robot Vehicles*, I. J. Cox and G. T. Wilfong, Eds. New York: Springer-Verlag, 1990, pp. 167–193.