Proceedings of 2004 IEEE/RSJ International Conference on
Intelligent Robots and Systems
September 28 - October 2, 2004, Sendai, Japan

Simultaneous Localization and Mapping using Multiple View Feature Descriptors

Jason Meltzer
Dept. of Computer Science
University of California, Los Angeles
Los Angeles, California, USA
jasonm1@ucla.edu

Rakesh Gupta and Ming-Hsuan Yang
Honda Research Institute
Silicon Valley
Mountain View, California, USA
{rgupta,myang}@honda-ri.com

Stefano Soatto
Dept. of Computer Science
University of California, Los Angeles
Los Angeles, California, USA
soatto@ucla.edu

***Abstract*— We propose a vision-based SLAM algorithm incorporating feature descriptors derived from multiple views of a scene, incorporating illumination and viewpoint variations. These descriptors are extracted from video and then applied to the challenging task of wide baseline matching across significant viewpoint changes.**

The system incorporates a single camera on a mobile robot in an extended Kalman filter framework to develop a 3D map of the environment and determine egomotion. At the same time, the feature descriptors are generated from the video sequence, which can be used to localize the robot when it returns to a mapped location. The kidnapped robot problem is addressed by matching descriptors without any estimate of position, then determining the epipolar geometry with respect to a known position in the map.

I. Introduction

A key component of a mobile robot system is the ability to simultaneously build a map of its environment and localize itself within that environment. In this paper, we describe a vision-based mapping and localization algorithm for mobile robots which uses multiple view feature descriptors as generic landmarks.

In order to determine a 3D model of an environment from images, the locations of sparse 2D points can be integrated over time. A mobile platform with a video camera needs to track such points (typically referred to as *features*) across multiple images. Thus, it must be able to find correspondences among sets of features, leading to the idea of *descriptors,* which ideally provide signatures of distinct locations in space.

Descriptors can be applied to the difficult task of finding correspondences in images taken from significantly different, unknown viewpoints. In practice, this may happen when a robot is placed in a new location ("kidnapped robot problem"), when it is exploring a known environment along a different path, or when estimation of a robot's position is degraded over time. By finding features and their associated descriptors, the correspondence problem can be made simpler by comparing descriptors instead of entire images. Given the correspondences, estimating relative orientation is a matter of computing epipolar geometry.

We address the wide-baseline correspondence problem under the specific scenario of autonomous navigation, where high frame-rate video is available during training ("map building"), but not necessarily during testing (localization with wide baseline), and viewing conditions can change significantly between the two. Such changes affect both the domain of the image (geometric distortion due to changes of the viewpoint) and its range (changes in irradiance). During map building, a video stream is available, making it possible to exploit *small baseline* correspondence by tracking feature locations between closely spaced images. This provides the opportunity to incorporate multiple views of a feature into the descriptor. The primary novelty of such a descriptor is that, rather than discard data after processing each frame, it incorporates information from across multiple adjacent views of a scene to yield a richer representation.

In addition to convenience and availability of the data, there are strong theoretical reasons for using multiple views to derive descriptors. In particular, it is known that generic viewpoint invariants do not exist for single views for either geometry or illumination [2], [1]. However, it can easily be shown that such invariants do exist when combining information from multiple views [14], [3].

In this work, multi-view descriptors are developed using kernel principal component analysis (KPCA) and used to estimate landmark location and robot egomotion using structure-from-motion (SFM) techniques. A map consists of a database of these feature descriptors and their locations in space. This map is incrementally updated over time and is robust to changing environments. The multi-view descriptors are used both to correct drift in the SFM map-building process, and to determine wide-baseline correspondences in a kidnapped-robot scenario.

II. Related Work

Much work in mapping and localization has been performed using range sensing, such as laser scanners or sonar. There has been some recent work with visual sensing by matching features like vertical and horizontal lines, corners in the scene, and active stereo vision to find salient features [29].

Davidson and Murray [5] used active vision for real-time, sequential map-building within a SLAM framework. Assuming that the robot trajectory was given, they controlled the active head's movement and sensing on a short-term tactical basis, making a choice between a selection of currently visible features. Persistent features re-detected

0-7803-8463-6/04/$20.00 ©2004 IEEE

after lengthy neglect could be re-matched, even if the area was passed through along a different trajectory or in a different direction.

The scale invariant feature transform (SIFT) developed by Lowe [12] is invariant to image translation, scaling, rotation, and partially invariant to illumination changes and affine or 3D projection. Se at. al. [24] employed the SIFT scale and orientation constraints for matching stereo images. After matching they used a least-squares procedure to compute the camera egomotion for better localization. Their features had a viewpoint variation limit of 20 degrees.

Wolf et. al. [34] built a vision based localization system by combining techniques from image retrieval with Monte-Carlo localization. The system was able to retrieve similar images even if only a small part of the corresponding scene is seen in the current image. These results were filtered by visibility constraints to globally estimate the position of the robot and to reliably keep track of it and to recover from localization failures.

A stereo vision algorithm for mobile robot mapping and navigation was proposed by Murray et. al. in [17], where a 2D occupancy grid map was built from the stereo data. However, odometry error was not corrected, and hence the map could drift over time.

Little et. al. [11] proposed combining this 2D occupancy map with sparse 3D landmarks for robot localization. They used corners on planar objects as stable landmarks. A trinocular vision system was used to compute neighborhood region planarity. Landmarks were used for matching only in pairs of frames but not kept for matching subsequent images.

Sim and Dudek [26] proposed learning natural visual features for pose estimation. Landmark matching was achieved using principal components analysis, and a tracked landmark is a set of image thumbnails detected in the learning phase, for each grid position in pose space.

Among approaches combining vision and laser sensors, Dellaert et. al. [6] compared brightness values between the images obtained by the robot to those given by a visual map of the ceiling obtained by mosaicing. They used particle filters to represent the multimodal belief of robot location. The Minerva museum tour-guide robot [30] learned its map using this technique in addition to the laser scan occupancy map.

Among approaches using laser range data, Gutmann and Konlolige [7] used global registration and correlation techniques to reconstruct consistent global maps. Recently, Thrun et al. [31] proposed a real-time algorithm combining the EM and incremental algorithms.

III. Scene Feature Representation

A. Overview of the System

Our system consists of the following interacting components: a feature selection and tracking mechanism, which finds and tracks areas of interest across frames of a video input; a feature descriptor database, which builds and stores the multi-view descriptors; a structure from motion system using the extended Kalman filter to determine egomotion and a 3D map of the environment. Figure 1 shows a diagram of how these elements interact.

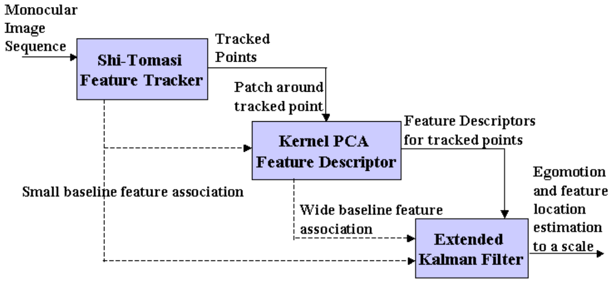
Monocular Image Sequence
Shi-Tomasi Feature Tracker
Tracked Points
Patch around tracked point
Kernel PCA Feature Descriptor
Feature Descriptors for tracked points
Small baseline feature association
Wide baseline feature association
Extended Kalman Filter
Egomotion and feature location estimation to a scale

Fig. 1. Components of the system.

B. Structure from Motion

Our SFM technique is based on Chiuso *et al* [4], which uses a single camera to integrate 3D information causally over time using a robust version of the extended Kalman filter (EKF). Images are captured from a video camera in real-time, and on each frame a number of points of interest are tracked in 2D. The EKF uses these tracks to determine the depth of each point in space, providing a 3D model up to a scale factor.

Localization requires that reliable, persistent features in the environment can be matched even after being lost for long periods of time. This differs from the more common use of visual features in structure from motion, where they are treated as transient entities to be matched over a few frames and then discarded.

Since we use monocular vision with no assumptions on scene geometry, we can only estimate distances up to a scale factor. In the experiments, we do not estimate this scale factor, but the system ensures that it is consistent among structure and motion estimation. If necessary, the scale could be estimated using the odometry data from the mobile robot, by providing the depth of a certain point as prior information to the vision system, or by inserting an object of known size into the scene.

C. Feature Selection and Tracking

During a map building phase, images are collected closely in time, thus the inter-frame motion is small, and appearance changes in features are minimal. Under these circumstances, correspondence becomes easier, making the problem of tracking across frames relatively simple. Many existing feature trackers, such as [10], [25], [33], can produce chains of correspondences by incrementally following small baseline changes between images in the sequence.

We are only concerned with feature *selection* or *tracking* in-so-far as they influence the experimental quality. For our purposes, any consistent feature selector and tracker can be used to estimate the candidate points, or even no feature tracker at all (searching based on odometry, for example). For computational reasons, we used an implementation of the Shi-Tomasi affine multiscale tracker, which proved robust enough for our purposes [25].

D. Multiple View feature Descriptor

The multi-view feature descriptor is generated by kernel principal component analysis (KPCA). A complete discussion of our descriptor can be found in [15], but an overview follows. In contrast to conventional principal component analysis (PCA) which operates in the input image space, KPCA performs the same procedure as PCA in a high dimensional space, F, related to the input by the (nonlinear) map

$$\Phi : \mathbb{R}^N \longrightarrow F, \quad \mathbf{y} \mapsto Y^1$$

If one considers $\mathbf{y} \in \mathbb{R}^N$ to be a (vectorized) image patch, $Y \in F$ is this image patch mapped onto F. The sample covariance for M vectors in F is

$$C' = \frac{1}{M} \sum_{i,j=1}^{M} \Phi(\mathbf{y}_i)\Phi(\mathbf{y}_j)^T$$

assuming $\sum_{k=1}^{M} \Phi(\mathbf{y}_k) = 0$ (see [23] for a method to center $\Phi(\mathbf{y})$). By diagonalizing C', a basis of kernel principal components (KPCs) is found. As demonstrated in [20], by using an appropriate kernel function $k(\mathbf{x}, \mathbf{y}) = \langle \Phi(\mathbf{x}), \Phi(\mathbf{y}) \rangle$, $\mathbf{x}, \mathbf{y} \in \mathbb{R}^N$, one can avoid computing the inner product in the high-dimensional space F. The KPCs are implicitly represented in terms of the inputs (image patches) $\mathbf{y}$, the kernel k, and a set of linear coefficients β, as

$$\Psi = \sum_{i=1}^{M} \beta_i \Phi(\mathbf{y}_i), \Psi \in F.$$

Our method for extracting feature descriptors from image sequences proceeds as follows:

1) **Bootstrap with a small-baseline tracker**: Read a number of frames of the input sequence, track the features using a standard tracking method, and store the image patches of each feature. For an affine-invariant tracker, we use the Shi and Tomasi (ST) [25] algorithm.
2) **Construct kernel basis**: Perform Kernel Principal Component Analysis (KPCA) using the Gaussian kernel separately on each feature's training sequence or reduced training set found in step 3.
3) **Approximate kernel basis**: Form an approximate basis for each feature by finding approximate patches which lead to the least residual estimate of the original basis in high-dimensional space. Create L such patches for each feature. In our algorithm, L is a tuning parameter. Further discussion of "pre-image" approximation in KPCA can be found in [21], [22].

The above algorithm yields a set of descriptors, each corresponding to a particular feature. In order to match a newly observed image to existing descriptors, our algorithm searches the image for patches which have a small residual when projected onto the stored KPCA descriptors. For more details, please see [15].

Unlike [24], [5], and any other system that uses single-view descriptors, our feature representation is viewpoint invariant, so we do not keep track of the viewing angle of the features. Se at. al. [24] store the original view direction of the feature and make a new feature in the database if the view direction varies more than the threshold of 20 degrees. Davidson et. al. [5] expect the feature to be visible only if the angular difference is less than 45 degrees in magnitude.

IV. Robust Extended Kalman Filter

Smith et. al. [27] proposed what is now a widely used approach to Simultaneous Localization And Mapping (SLAM). Their paper describes the use of an extended Kalman filter (EKF) for estimating the posterior distribution over robot pose along with the positions of landmarks. Our use of a robust EKF rather than a particle filter (e.g. [6]) is motivated by the analysis of stability in [4].

We integrate KPCA features with a robust EKF structure from motion system for map building. Our filter is based on a robust version of the work of Chiuso et. al. [4]. Robustness to points that do not move according to a rigid model (such as T-junctions or moving obstacles in the environment) is accomplished by replacing the usual Kalman filter update step with one using an M-estimator for outliers. Outliers are defined by the magnitude of their innovations compared to a threshold defined by the measurement noise (the noise in tracking points through the image sequence).

The scale factor is associated to a reference feature chosen automatically among those visible. When that feature disappears, the reference switches to the best current estimate of another feature. Any error in the localization of that feature results in a global error, which increases every time the reference feature switches, effectively causing a slow drift in the estimates.

A. State Vector and its Covariance

Our state vector consists of the 3D position and orientation of the robot (camera) and 3D positions of the feature locations, both up to a common scale factor. We calculate the uncertainty expected in the measurement in the form of innovation. The innovation of a point is calculated by taking the difference between its measurement and its predicted measurement (reprojection) based on the model. When the innovation exceeds a pre-defined threshold it is classified as an outlier. Because the variance of outliers is increase more rapidly than inliers, their influence on the estimation of motion is mitigated. Outliers are completely removed from the state if their innovations exceed a higher threshold (which is ten times the outlier threshold in our experiments), or if they remain outliers for more than twenty consecutive frames. A test for rejecting outliers based upon such a principle has been proposed previously in [28].

B. Updating and Maintaining the Map

Initial features are selected on the first frame according to the method of Shi and Tomasi [25]; the best features

[1] F is typically referred to as "feature space," but to avoid confusion we will refrain from using that name.

survive for many frames and lead to rich multiple view feature descriptors. When the innovation for a feature is above a particular threshold, the feature is treated as an outlier. The covariances of outliers in the filter are increased in proportion to their innovations, hence their influence on the estimate of structure and egomotion are decreased.

As the field-of-view of the camera shifts in the scene, new features need to be added to the filter and occluded features removed. When a feature is lost, its corresponding rows and columns of the covariance matrix are removed (but the feature remains stored in the descriptor database). Features are added in batches of 30 or more. Because no estimate of the depth of these points can be obtained initially, they must be handled separately to ensure the filter is not corrupted by unstable states. To accomplish this, new features are added in groups to the filter, and each group has its own reference frame. To minimize the effect of adding these points to the filter, they are initialized with large variances. If a group retains fewer than 5 members, it is dropped from the filter. If the primary group being used to compute location estimates disappears, the reference switches to another group leading to some drift.

Even with a known initial position, drift in odometry and SFM causes the estimated localization to deviate from the actual position. When localization is one of the objectives, motion drift must be accounted for by matching features seen earlier during mapping to their current observations. Matching is accomplished with the multi-view descriptors. Drift can be compensated by bundle adjustment techniques [32] to recompute the trajectory when features previously stored during mapping are re-observed.

V. Experiments

We performed two types of tests with the proposed system. The first was to test the efficacy of small and wide baseline correspondence. Wide baseline viewpoint changes test suitability of the system for solving the kidnapped robot problem for initializing localization. In the experiments, two phases were established: *training* and *matching*, which correspond to short and wide baseline correspondence. In the training phase, a video sequence was recorded of a mobile robot moving in a loop around a room. The Shi-Tomasi (ST) [25] tracker was used to obtain an initial set of points, then the procedure of the previous section was used to develop feature descriptors via approximate KPCA and track them via Robust EKF.

In the matching phase, a test image from outside the training sequence was used to find wide-baseline correspondences. First, initial features were selected using the ST or Lukas-Kanade's (LK) [33] selection mechanism. The quality of a candidate match was calculated by finding the projection distance of this patch onto the basis of the descriptor. Finally, candidate matches that fell below a threshold distance were selected, and the best among those was chosen as the matching location on the test image. Results of wide-baseline matching experiments can be found in [15]. One such experiment is included here for completeness, figure 2.

The second test was for performing localization when the robot returned to a previously mapped area by moving around a loop. We corrected for drift using bundle adjustment across several viewpoints keeping the known structure constant. In the first of these experiments (figure 4), the camera was moved around a circle while pointing toward the center of rotation. In the second, a similar path was followed, but significant outliers were introduced into the scene when a person walked across the room about halfway through the loop, causing 40-50 percent of the filtered features to become outliers for 50 frames. This degraded the initial estimation of the path more than in the first experiment, producing a 15 percent error in translational estimation by the time the camera returned to its initial position in the circle. Figure 5 shows the bird's eye view of the uncorrected path produced by the EKF in the second experiment, with translation at every time step and orientation every twenty steps. Corrected locations and orientations are overlayed on this plot. These are computed for a number of frames where the camera has returned to the same location in the circle. Drift in translation estimates present in the raw EKF estimates of egomotion are reduced by this correspondence step to under one percent in each experiment.

While we did not optimize our experiments for speed or programming efficiency, we found the average time for tracking between adjacent frames to be approximately 1/3 seconds for 200 track points and 120 filtered points. The code was executed on a 1.7GHz Pentium IV processor. The video frames and test images were 640x480 8-bit greyscale pixels. To perform wide baseline matching, the algorithm searches over all pairs of features between the set of training features and those on the test image. During the search, matches are declared if the ratio between the second best match score and the best match score is greater than 1.2 (scores range between 0 and 1, with 0 being the best, or smallest projection distance between the test patch and the KPCA descriptor). The process takes about 45 seconds to find matches among pairs of about 100 features each.

VI. Concluding Remarks

We have presented an application of a novel method for extracting feature descriptors from image sequences and matching these to new views of a scene. Rather than derive invariance completely from a model, our system learns the variability in the images directly from data. This technique is applicable in situations where such data is already available, such as robot navigation.

The variations in the appearance of each feature are learned using kernel principal component analysis (KPCA) over the course of image sequences. Our experiments demonstrate robustness to wide appearance variations on non-planar surfaces, including changes in illumination, viewpoint, scale, and geometry of the scene. These techniques have been applied for solving the kidnapped robot problem, where an initial guess of pose is unknown. We have found good results in estimating the robot's position when the robot is within a few feet of the mapped locations.

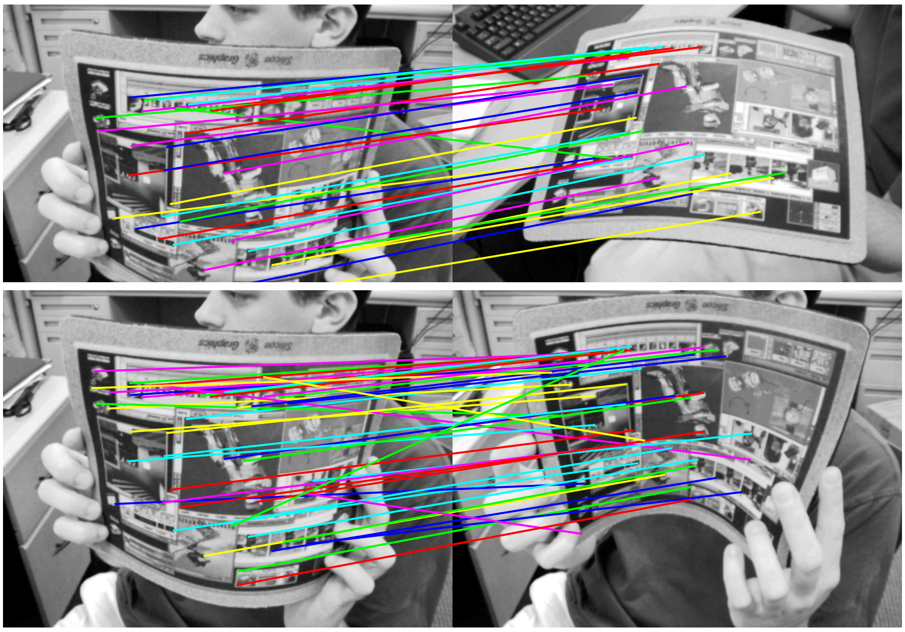

Fig. 2. **Affine tracking + KPCA:** A non-planar surface undergoing warping, viewpoint, and illumination changes. (Top-left) The first image of the training sequence. (Top-right) The test image, outside of the training sequence. 27 feature locations were correctly matched, with 2 false positives. (Bottom-left) Image from the training sequence. (Bottom-right) The warped object. 40 locations correctly matched, 6 false positives. Note that in all cases there are illumination changes across the pad as it changes orientation.

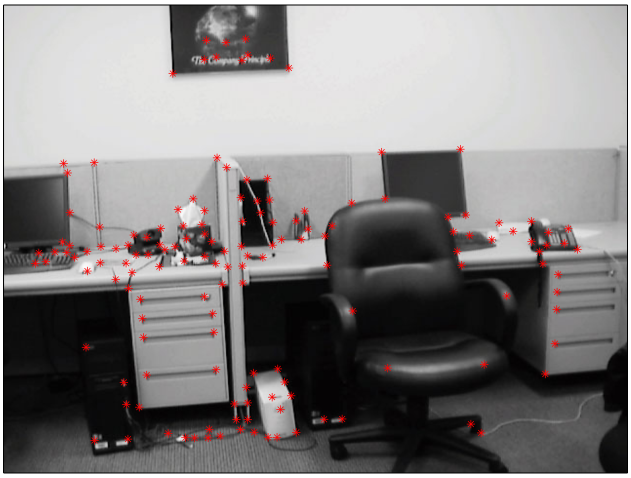

Fig. 3. A sample frame from experimental video with tracked interest points.

We have experimented with closing the loop to recognize a stored feature by combination of prediction using robust extended Kalman filter and KPCA. Once a prior point is recognized we estimate distances using epipolar geometry. Our system is robust to features that violate the model, such as those that are improperly tracked or occluded. In future work, we want to further perform optimization to make system closer to real time performance.

ACKNOWLEDGEMENTS

This work was conducted while Jason Meltzer was an intern at the Honda Research Institute in 2003. Jason Meltzer and Stefano Soatto were supported in part by grants AFOSR F49620-03-1-0095/E-16-V91-G2 and ONR N00014-03-1-0850/N00014-02-1-0720. The authors thank

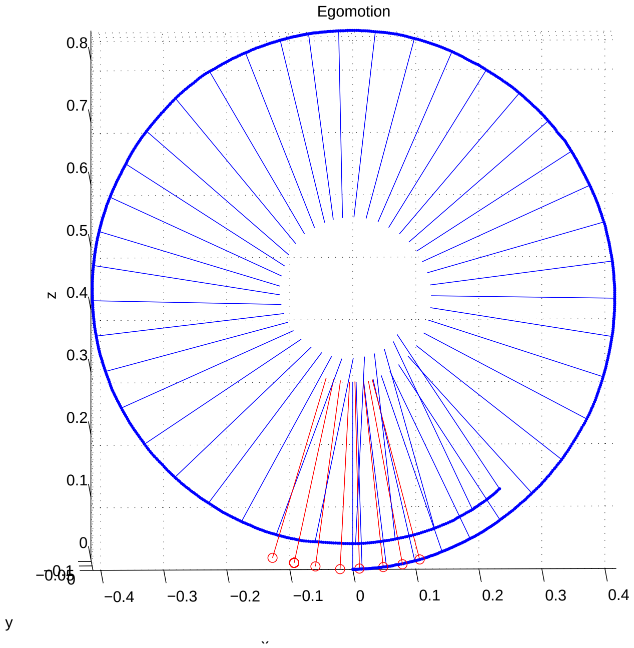

Fig. 4. **Experiment 1:** Uncorrected reconstruction from the EKF with corrected point overlays. The nearly circular path represents the estimated location of the camera in space based on the uncorrected robust EKF, where each time step is indicated by a marker starting at (0,0,0). The thin blue lines show the orientation of the camera at every 20th step. Small red circles and lines indicate the corrected estimates of five locations on the second pass of the camera around the circle. The correspondence and correction step reduced the error from about seven percent to under one percent. The actual path was a circle where the camera was pointing towards its center.

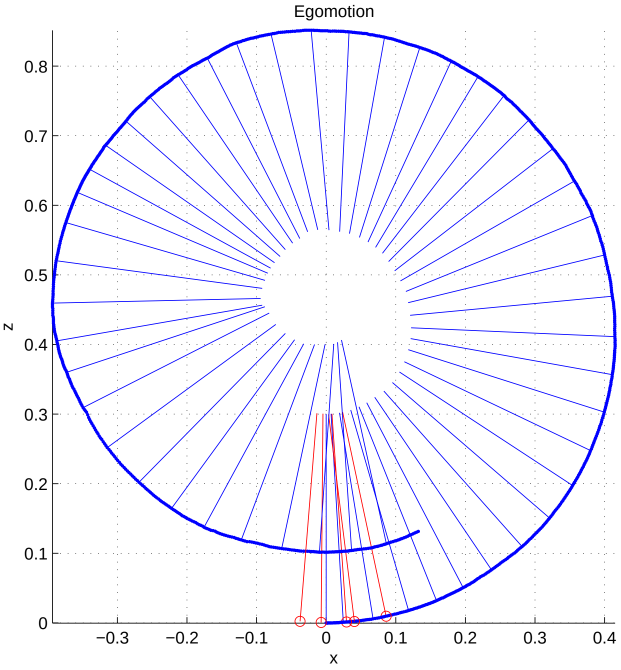

Fig. 5. **Experiment 2:** A circular path similar to that of the previous experiment (figure 4) was followed here, except that a person walked across the field of view, causing a disruption in the estimate of motion due to outliers. The corrected estimates bring the error down from about 15 percent to under 1 percent.

David Ross, David Lowe, Intel OpenCV project, and Paolo Favaro and Hailin Jin from the UCLA Vision Laboratory for providing code and feedback.

References

[1] P. Belhumeur and D. Kriegman. “What is the Set of Images of an Object under all Possible Illumination Conditions?” *Int. J. of Computer Vision*, vol. 28(3), 1998.

[2] J. Burns, R. Weiss and E. Riseman. “The non-existence of general case invariants.” *Geometric Invariance in Machine Vision.* MIT Press, 1992.

[3] H. Chen, P. Belhumeur and D. Jacobs, “In search of illumination invariants.” *IEEE Computer Society Conf. on Computer Vision and Pattern Recognition*, 2000.

[4] A. Chiuso, P. Favaro, H. Jin and S. Soatto. “Structure from Motion Causally Integrated over Time.” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Volume 24, No 4, April 2002.

[5] A. Davidson and D. Murray. “Simultaneous Localization and Map-Building using Active Vision.” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Volume 24, No 7, July 2002.

[6] F. Dellaert, W. Burgard, D. Fox and S. Thrun. “Using the CONDENSATION algorithm for robust, vision-based mobile robot localization.” *IEEE Computer Society Conf. on Computer Vision and Pattern Recognition*, 1999.

[7] J. Gutmann and K. Konolige. “Incremental mapping of large cyclic environments.” In *Proceedings of the IEEE International Symposium on Computational Intelligence in Robotics and Automation (CIRA)*, California, November 1999.

[8] G. Hager, D. Kriegman, E. Yeh and C. Rasmussen. “Image-based prediction of landmark features for mobile robot navigation.” In *IEEE Conf. on Robotics and Automation*, pages 1040–1046, 1997.

[9] D. Hahnel, W. Burgard, B. Wegbreit, and S. Thrun. “Towards lazy data association in SLAM.” In *Proceedings of the 10th International Symposium of Robotics Research (ISRR’03)*, 2003.

[10] C. Harris and M. Stephens. “A combined corner and edge detector.” *Alvey Vision Conference,* 1988.

[11] J. Little, J. Lu, and D. Murray. “Selecting Stable Image Features for Robot Localization Using Stereo.” *Proc. of IEEE/RSJ Int. Conf. on Intelligent Robots and Systems,* pp. 1072-1077, 1998.

[12] D. Lowe. “Object recognition from local scale-invariant features.” *Proc. ICCV,* Corfu, Greece, September 1999.

[13] F. Lu and E. Milios. “Globally Consistent Range Scan Alignment for Environment Mapping.” *Autonomous Robots,* Volume 4, pages 333-349, 1997.

[14] Y. Ma, S. Soatto, J. Kosecka, and S. Sastry. *An invitation to 3D vision, from images to models*. Springer Verlag, 2003.

[15] J. Meltzer, M.-H. Yang, R. Gupta and S. Soatto. “Multiple View Feature Descriptors from Image Sequences via Kernel Principal Component Analysis.” *European Conference on Computer Vision (ECCV),* May 11-14, 2004, Prague, Czech Republic.

[16] K. Mikolajczyk and C. Schmid. “An affine invariant interest point detector.” *Proc. ECCV,* 2002.

[17] D. Murray and C. Jennings. “Stereo vision based mapping and navigation for mobile robots.” In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA’97),* pages 1694-1699, New Mexico, April 1997.

[18] D. Murray and J. Little. “Using real-time stereo vision for mobile robot navigation.” In *Proceedings of the IEEE Workshop on Perception for Mobile Agents,* Santa Barbara, CA, June 1998.

[19] F. Schaffalitzky and A. Zisserman. “Viewpoint invariant texture matching and wide baseline stereo.” *Proc. ICCV,* Jul 2001.

[20] B. Schölkopf, A. Smola. *Learning with Kernels.* Cambridge: The MIT Press, 2002.

[21] B. Schölkopf et al. “Input Space vs. Feature Space in Kernel-Based Methods.” *IEEE Transactions on Neural Networks.* 1999.

[22] B. Schölkopf, P. Knirsch, A. Smola and C. Burges, “Fast Approximation of Support Vector Kernel Expansions, and an Interpretation of Clustering as Approximation in Feature Spaces.” *DAGM Symposium Mustererkennung,* Springer Lecture Notes in Computer Science, 1998.

[23] B. Schölkopf, A. Smola, K. Müller. “Nonlinear component analysis as a kernel eigenvalue problem.” *Neural Computation*, 10, 1299-1319, 1998.

[24] S. Se., D. Lowe, and J. Little. “Vision-based Mobile robot localization and mapping using scale-invariant features.” In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA),* Seoul, Korea, pages 2051–2058, May 2001.

[25] J. Shi and C. Tomasi. “Good Features to Track.” *IEEE CVPR,* 1994.

[26] R. Sim and G. Dudek. “Learning and evaluating visual features for pose estimation.” In *Proceedings of the Seventh International Conference on Computer Vision (ICCV’99),* Kerkyra, Greece, September 1999.

[27] R. Smith, M. Self and P. Cheeseman. “Estimating uncertain spatial relationships in robotics.” *Proceedings of UAI,* pages 435-461, 1986.

[28] S. Soatto and P. Perona. “Three dimensional transparent structure segmentation and multiple 3d motion estimation from monocular perspective image sequences,” In *IEEE Workshop on Motion of Nonrigid and Articulated Objects,* Austin, pages 228–235. IEEE Computer Society, November 1994.

[29] S. Thrun. “Probabilistic Algorithms in Robotics.” *AI Magazine* vol 21 no. 4. 2000.

[30] S. Thrun, M. Bennewitz, W. Burgard, A.B. Cremers, F. Dellaert, D. Fox, D. Hahnel, C. Rosenberg, N. Roy, J. Schulte, and D. Schulz. “Minerva: A second generation museum tour-guide robot.” In *Proceedings of IEEE International Conference on Robotics and Automation (ICRA’99),* Detroit, Michigan, May 1999.

[31] S. Thrun, W. Burgard, and D. Fox. “A real-time algorithm for mobile robot mapping with applications to multi-robot and 3d mapping.” In *IEEE International Conference on Robotics and Automation (ICRA 2000),* San Francisco, CA, April 2000.

[32] B. Triggs, P. McLauchlan, R. Hartley, and A. Fitzgibbon. ”Bundle Adjustment – A Modern Synthesis.” In *Vision Algorithms: Theory and Practice,* Ed. Triggs, Zisserman, and Szeliski. Springer Verlag, pages 298–375, 2000.

[33] C. Tomasi, T. Kanade. “Detection and tracking of point features.” Tech. Rept. CMU-CS-91132. Pittsburgh: Carnegie Mellon U. School of Computer Science, 1991.

[34] J. Wolf, W. Burgard, H. Burkhardt. “Robust Vision-based Localization for Mobile Robots Using an Image Retrieval System Based on Invariant Features,” In *Proc. of the IEEE Int. Conf. on Robotics and Automation (ICRA),* 2002.