# Appearance-Based Minimalistic Metric SLAM

Paul E. Rybski, Stergios I. Roumeliotis, Maria Gini, Nikolaos Papanikolopoulos
Center for Distributed Robotics
Department of Computer Science and Engineering
University of Minnesota, Minneapolis, U.S.A.
{rybski, stergios, gini, npapas}@cs.umn.edu

*Abstract*— **This paper addresses the problem of Simultaneous Localization and Mapping (SLAM) for the case of very small, resource-limited robots which have poor odometry and can typically only carry a single monocular camera. We propose a modification to the standard SLAM algorithm in which the assumption that the robots can obtain metric distance/bearing information to landmarks is relaxed. Instead, the robot registers a distinctive sensor "signature", based on its current location, which is used to match robot positions. In our formulation of this non-linear estimation problem, we infer implicit position measurements from an image recognition algorithm. The Iterated form of the Extended Kalman Filter (IEKF) is employed to process all measurements.**

## I. Introduction

Solving the Simultaneous Localization and Mapping (SLAM) problem for small, resource-limited robots means doing so without the aid of good odometric estimates and accurate metric range sensors. This causes a problem for traditional solutions to the SLAM problem which typically require one or both of the above. The motivating factor for this research is the necessity of doing SLAM on custom miniature robots called Scouts [15] (Fig. 1) that our research group has developed. Scouts, due to their small size, are limited to a monocular camera as their only exteroceptive sensor. Their limited computing capabilities also makes them totally dependent on a wireless proxy-processing scheme in which off-board workstations handle their control and video processing.

We propose a modification to the standard SLAM algorithm in which we relax the assumption that the robots can obtain metric distance information to landmarks. In our approach, we obtain purely qualitative measurements of landmarks where a location "signature" is used to match robot poses. We describe a method by which the Iterated form of the Extended Kalman Filter (EKF) processes all measurements, including both actual odometric and inferred relative positions, and estimate the coordinates of the locations where images were recorded along the
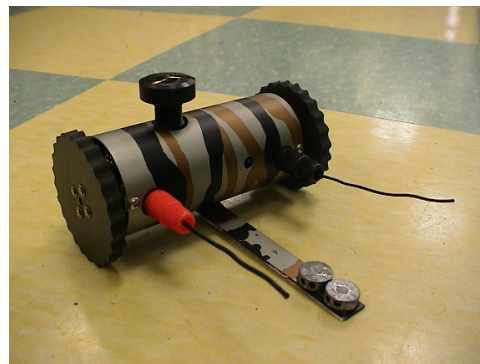
Fig. 1. A Scout robot with an upward-facing Omnitech 190° fisheye lens. The robot is $11\,\mathrm{cm}$ long and $4\,\mathrm{cm}$ in diameter.

trajectory of the robot. In this method, landmarks correspond to images taken at various $(x, y)$ positions of the robot.

## II. Related Work

The Extended Kalman Filter has been used for localizing [9] and performing SLAM [18] on mobile robots for at least a decade. Our approach differs from traditional EKF estimators in that we do not have the ability of resolving specific geometric information about the landmarks we observe in our environment. Instead, the landmark positions are explicitly coupled to the position of the robot.

Bayesian methods have also been used for mobile robot localization (such as Markov Localization) and mapping [19] where the modes of arbitrary robot pose distributions are computed. Statistical methods such as Monte Carlo localization [20] use sampling techniques to more quickly estimate the distribution of possible robot poses. Both of these methods typically use very accurate sensors and/or robots with very accurate odometry that allow them to resolve accurate maps over large distances.

Structure from motion [2] algorithms compute the correspondences between features extracted from multiple images to estimate the geometric shape of landmarks as well as to estimate the robot's pose. While we use a vision system to identify locations by their feature signatures, features are not explicitly tracked. In practice, our vision system could be replaced by any other kind of boolean

sensor modality which can report whether the robot has re-visited a location.

In contrast to explicit metric-based methods, more qualitative methods such as topological maps of nodes have been used as well [17]. Of special note is the representation suggested by Ben Kuipers [8] in his Semantic Spatial Hierarchy (SSH). Locations are explicitly designated by distinctive (but not necessarily unique) sensor signatures. In [22], image "signatures" captured from an omnidirectional camera are used to construct a topological map of an environment by generating histograms of the RGB and HSV (Hue, Saturation, and Value) components.

Finally, physics-based models that involve spring dynamics have been used quite effectively to find minimum energy states in topological map structures [4], [6]. We have had some success with these methods [16] but have found that the parameter choices for the models tend to be very important and that numerically solving for the set of non-linear equations can be unstable.

### III. APPEARANCE-BASED SLAM

Mobile robots like the Scout have wheel encoders that count revolutions. From these counts, odometric measurements of linear and rotational velocity can be calculated. By appropriately integrating this kinetic information, the robot is able to track its pose for a limited period of time. If no absolute positioning measurements are recorded by the robot, the noise in the velocity measurements will eventually cause the computed pose estimates to diverge from their real values. In order to provide periodic corrections, additional information is necessary. In environments where GPS measurements are not available, a robot will have to use information about its surroundings for this purpose. In previous implementations of SLAM algorithms, it is frequently assumed that the robot is able to measure its relative position with respect to features/landmarks [3], [13] or obstacles [19] in the area that it navigates. This implies that the robot carries a distance measuring sensor such as a sonar or a laser scanner. For small scale robots such as the Scout, this is not feasible due to weight, size, and power limitations. An alternative approach is to use a small camera and process relative angular measurements to detected vertical line features, as described in [1]. However the applicability of this algorithm is conditioned on the existence of a sufficient number of identifiable vertical line segments along the trajectory of the robot. Also it is geared toward position tracking while no attempt is made to construct a map populated by these features.

In what follows, we describe and implement a novel methodology that neither relies on any specific type of visual features, nor requires distance measurements. The basic idea behind our approach is to determine a unique visual signature for distinct locations along the robot's path, store this and the estimated pose of the robot at that time instant, and retrieve this information once the robot revisits the same area. Determining whether the robot is at a certain location for a second time is the key element for providing positioning updates. By correlating any two scenes, we infer a relative (landmark to robot, not landmark to landmark) position measurement and use it to update both the current and previous (at locations visited in the past) pose estimates for the robot. This in effect will produce an accurate map of distinct locations within the area that the robot has explored. A Kalman filter estimator is formulated for this purpose.

Before delving into the mathematical formulation of the specific estimator we first discuss in more detail the methodology we employ for assigning and registering visual signatures to certain locations. Although the discussion hereafter is specific to area identification using images from an omnidirectional camera, the remaining of the approach (usage of the identity information by the filter) can be extended to any type of exteroceptive sensor that can be used for identifying an area (e.g. other sensors which measure the magnetic, chemical or audio signature of a location might also be used).

### A. KLT Tracker and Location Identification

The Lucas-Kanade-Tomasi (KLT) feature tracker consists of a registration algorithm that makes it possible to find the best match between two images [11] as well as a feature selection rule which is optimum for the associated tracker under pure translation between subsequent images [21]. We use an implementation of the KLT algorithm* to identify and track features between successive images as a method for determining the match between two images. KLT features are selected from each of the images and are tracked from one image to the next taking into account a small amount of translation. The degree of match is the number of features successfully tracked from one image to the next. A total of 100 features are selected from each image and used for comparison. For the environments used in this paper, 80 or more tracked features is considered a match between locations.

Our image indexing approach is similar in flavor to [10] where a pyramid structure involving several levels of dimensionality reduction are matched from the lowest resolution to the highest. In our case, the KLT features serve as a single level of reduction, but a completely different feature extraction method is employed.

It is important to note that we are not attempting to track the features over multiple frames of video. This technique does not attempt to compute structure from motion on this data primarily because we are ultimately interested in making use of this algorithm on robots that do not have real-time video processing capability.

*Developed by Stan Birchfield [7].

In our mapping formalism, the mobile robot travels around an unknown area and stores images from its camera. KLT is used to compare images recorded at different locations along the trajectory of the robot. When the received image does not match a previously recorded one, we assume that this location is novel and we add the current estimate for the position of the robot to the state vector of landmarks. This constitutes an exploration phase where the robot creates its world model. When the robot encounters an image which matches a previously-seen image, it considers this location to be the same and updates its estimates of (1) the position of all landmarks mapped and (2) its own pose.

We treat the KLT and omnicamera setup as a "virtual sensor" that returns true or false as to whether the robot has returned to a location that it has visited before. This information can be used to infer a relative position measurement $Z = 0_{2 \times 1} + N_z$ between the current position of the robot and that of the same location visited in the past. The accuracy of this measurement $R = E\{N_z N_z^T\}$ can also be inferred by the locus of points (forming an ellipsoid) around a location, with the characteristic that the images recorded at each of them are considered identical by the KLT. This inferred $0_{2 \times 1}$ term reflects the nature of the sensor modality where range or bearing readings are not assumed to be available. Thus, the robot robot can only assign its self-position estimate to the same position as the landmark "signature" when it observes it. Since in practice, it is extremely unlikely that the robot will actually find itself in the exact same location that it was the first time the sensor readings were taken, any uncertainty in the spatial match at time $k$ is captured in the sensor reading covariance matrix $R(k)$.

### B. Kalman Filter Derivation

At this point we describe the process for formulating the Extended Kalman filter required for estimating the current pose of the robot $X_R = [x_R \, y_R \, \phi_R]^T$ and landmarks $X_{L_i} = [x_{L_i} \, y_{L_i}]^T$.

*1) Propagation:* For the vehicle's odometry, we use a generic set of equations. This allows the method to be easily adapted to different types of vehicles. The continuous time equations for the motion expressed in local coordinates (with respect to a frame of reference $\mathcal{R}$ attached to the robot) are:

$$^{\mathcal{R}}\dot{x}_R = V, \quad ^{\mathcal{R}}\dot{y}_R = 0, \quad ^{\mathcal{R}}\dot{\phi}_R = \omega \qquad (1)$$

where $V$ and $\omega$ are the real linear and angular velocity of the robot. The same quantities, as measured by the wheel-encoders' signals in discrete time, are:

$$V_m(k) = V(k) + w_v(k) \qquad (2)$$
$$\omega_m(k) = \omega(k) + w_\omega(k) \qquad (3)$$

where $w_v(k)$, $w_\omega(k)$ are the zero-mean, white Gaussian noise processes that contaminate the velocity measurement signals with known covariance $Q(k) = E\{W_R W_R^T\}$, $W_R = [w_v(k) \, w_\omega(k)]^T$. By integrating the velocity measurements, the state estimate $\hat{X}_R = \left[\hat{x}_R \, \hat{y}_R \, \hat{\phi}_R\right]^T$ is propagated as:

$$\hat{x}_R(k+1) = \hat{x}_R(k) + V_m(k)\delta t \cos \hat{\phi}_R(k) \qquad (4)$$
$$\hat{y}_R(k+1) = \hat{y}_R(k) + V_m(k)\delta t \sin \hat{\phi}_R(k) \qquad (5)$$
$$\hat{\phi}_R(k+1) = \hat{\phi}_R(k) + \omega_m(k)\delta t \qquad (6)$$

Based on Eqs. (1), (2), (3) the linearized discrete-time error-state propagation equation in global coordinates is:

$$\widetilde{X}_R(k+1) = \Phi_R(k+1)\widetilde{X}_R(k) + G_R(k+1)W_R(k) \qquad (7)$$

with

$$\Phi_R = \begin{bmatrix} 1 & 0 & -V_m \delta t \sin \phi_R \\ 0 & 1 & V_m \delta t \cos \phi_R \\ 0 & 0 & 1 \end{bmatrix}, \quad G_R = \begin{bmatrix} \delta t \cos \phi_R & 0 \\ \delta t \sin \phi_R & 0 \\ 0 & \delta t \end{bmatrix} \qquad (8)$$

where $\delta t$ is the time interval between two consecutive odometric measurements.

Since the coordinates of the landmark locations $X_{L_i}$ do not change over time, the real, estimated, and error "motion" equations for a landmark $L_i$ are

$$X_{L_i}(k+1) = X_{L_i}(k) \qquad (9)$$
$$\hat{X}_{L_i}(k+1) = \hat{X}_{L_i}(k) \qquad (10)$$
$$\widetilde{X}_{L_i}(k+1) = I\widetilde{X}_{L_i}(k) + 0W_R(k) \qquad (11)$$

where $I$ is the $2 \times 2$ identity matrix and $0$ is the $2 \times 2$ zero matrix. By augmenting the state vector $X$ with the poses (to be estimated) of all the landmarks $X_{L_i}$ along the pose of the robot $X_R$

$$X = \begin{bmatrix} X_R^T & X_{L_1}^T & \dots & X_{L_N}^T \end{bmatrix}^T \qquad (12)$$

we can derive an expression for the error propagation of this augmented state vector and the covariance matrix associated with it:

$$\tilde{X}(k+1) = \Phi(k)\tilde{X}(k) + G(k)W_R(k) \qquad (13)$$
$$P_{k+1/k} = \Phi(k)P_{k/k}\Phi^T(k) + G(k)Q_R(k)G^T(k) \qquad (14)$$

where

$$\Phi(k) = \begin{bmatrix} \Phi_R(k) & 0 & \dots & 0 \\ 0 & I & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & I \end{bmatrix}, \quad G(k) = \begin{bmatrix} G_R(k) \\ 0 \\ \vdots \\ 0 \end{bmatrix} \qquad (15)$$

*2) Update:* Every time the robot takes an image of its surroundings when it is at $X_R$, it employs the KLT algorithm to determine whether it matches any previously seen locations $X_{L_i}$, or whether it is a new one $X_{L_j}$. In either case the inferred relative position measurement is

$$
\begin{aligned}
Z(k+1) &= 0_{2\times 1} + N_z(k+1) \\
&= {}^{\mathcal{R}}X_{L_i} + N_z
\end{aligned} \tag{16}
$$

From this, one can derive the following expressions for the real and the estimated measurement

$$
\begin{aligned}
Z &= C^T(\phi_R)\left(X_{L_i} - p_R\right) + N_z \tag{17} \\
\hat{Z} &= C^T(\hat{\phi}_R)\left(\hat{X}_{L_i} - \hat{p}_R\right) \tag{18}
\end{aligned}
$$

where $p_R = [x_R\ y_R]^T$, $\hat{p}_R = [\hat{x}_R\ \hat{y}_R]^T$ and

$$
C(\phi_R) = \begin{bmatrix} \cos\phi_R & -\sin\phi_R \\ \sin\phi_R & \cos\phi_R \end{bmatrix} \tag{19}
$$

is the rotation matrix that relates the orientation of the frame of reference $\mathcal{R}$ on the robot with the global coordinate frame. By subtracting the previous two equations the linearized measurement error is computed as

$$
\tilde{Z} = Z - \hat{Z} = H\tilde{X} + N_z \tag{20}
$$

with

$$
\begin{aligned}
\tilde{X} &= \begin{bmatrix} \tilde{X}_R^T & \tilde{X}_{L_1}^T & \dots & \tilde{X}_{L_i}^T & \dots & \tilde{X}_{L_N}^T \end{bmatrix}^T \tag{21} \\
H &= \begin{bmatrix} H_R & 0 & \dots & 0 & H_{L_i} & 0 & \dots & 0 \end{bmatrix} \tag{22} \\
H_r &= \begin{bmatrix} -C^T(\hat{\phi}_r) & -C^T(\hat{\phi}_r)J\left(\hat{X}_{L_i} - \hat{p}_r\right) \end{bmatrix} \tag{23} \\
H_{L_i} &= C^T(\hat{\phi}_r) \tag{24}
\end{aligned}
$$

and $J = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}$. The $H(k+1)$ matrix is used to update the state estimate for the pose of the robot $X_r$ and the positions of the landmarks $X_{L_i}$ every time an image is recorded. The equations for the update cycle of the filter are listed here for completeness:

$$
r(k+1) = Z(k+1) - \hat{Z}(k+1) \tag{25}
$$
$$
S(k+1) = H(k+1)P_{k+1/k}H^T(k+1) + R(k+1) \tag{26}
$$
$$
K(k+1) = P_{k+1/k}H^T(k+1)S^{-1}(k+1) \tag{27}
$$
$$
\hat{X}_{k+1/k+1} = \hat{X}_{k+1/k} + K(k+1)r(k+1) \tag{28}
$$
$$
P_{k+1/k+1} = P_{k+1/k} - K(k+1)S(k+1)K^T(k+1) \tag{29}
$$

Since the accuracy of this update depends on the accuracy of the linearization, we employ the Iterated form of the Extended Kalman filter (IEKF) [5], [12]. First, the IEKF linearizes the measurement equation Eq. (20) around the current estimate $X_{k+1/k}$ of the state and calculates the updated state estimate $X_{k+1/k+1}$ (like the EKF). Then the filter resets $X_{k+1/k}$ to its updated values and the same

process is repeated until it converges. The state covariance $P_{k+1/k}$ is *not* updated until after the state estimate has converged because to do so would artificially and erroneously reduce the uncertainty in the measurements.

## IV. EMPIRICAL VALIDATION

The Scout is a differentially-driven platform with a wheelbase $\alpha$ of $11\,\mathrm{cm}$ and a wheel radius of $2\,\mathrm{cm}$. The $Q_r$ matrix from the state error covariance propagation in Eq. (14) represents the covariance of the robot's linear and translational motions. For a differentially-driven platform, where linear and rotational velocity are a function of the left $v_l$ and right $v_r$ wheelspeeds, i.e. $V_m = (v_l + v_r)/2$, $\omega_m = (v_l - v_r)/\alpha$, this matrix is defined as:

$$
Q_R = \begin{bmatrix} \frac{1}{4}(\sigma_{v_l}^2 + \sigma_{v_r}^2) & \frac{1}{2\alpha}(\sigma_{v_l}^2 - \sigma_{v_r}^2) \\ \frac{1}{2\alpha}(\sigma_{v_l}^2 - \sigma_{v_r}^2) & \frac{1}{\alpha^2}(\sigma_{v_l}^2 + \sigma_{v_r}^2) \end{bmatrix} \tag{30}
$$

where $\sigma_{v_l}$ and $\sigma_{v_r}$ are the standard deviations of the wheelspeed errors.

### A. Simulation Experiment

This method was tested first on a simulated Scout robot. The standard deviation of the estimated wheel encoder error was $1.4\,\mathrm{cm/s}$ and the standard deviation of the sensor error was $1\,\mathrm{cm}$. The true path of the simulated robot is shown in Fig. 2(a) as a square that is traversed twice. Without correcting for odometric error, the path of the simulated robot is shown in Fig. 2(b). Sensor snapshots are taken every $0.5\,\mathrm{m}$ as the robot traverses the path. The first time the robot observes landmark, it adds the estimated positions to its map. When the robot revisits landmarks, all landmark positions are updated by new measurements.

Figs. 2(c) and 2(d) show how the landmark positions are updated as the first landmark is revisited. Figs. 2(e) and 2(f) show how the landmark positions are updated as the last landmark is revisited. Fig. 3 illustrates how the estimated landmark positions are improved by using the Iterated EKF. Fig. 4 shows the sensor residual with a $3\sigma$ upper and lower bounds of uncertainty.

### B. Real-World Experiment

A sequence of images were taken with a Scout equipped with an upward-facing $190°$ vertical/$360°$ horizontal field of view lens along a path that intersected itself five times. An image was taken from the camera roughly every $0.3\,\mathrm{m}$. Ground truth was measured by manually measuring and labeling each location where an image was taken. The KLT algorithm was used to track features between each pair of images in order to find locations where the robot's path crossed itself.

Fig. 5(a) shows the true path of the robot and the estimated path without error correction. Fig. 5(b) shows again the odometric path estimate as well as the estimated positions of the landmarks if there was no correction to
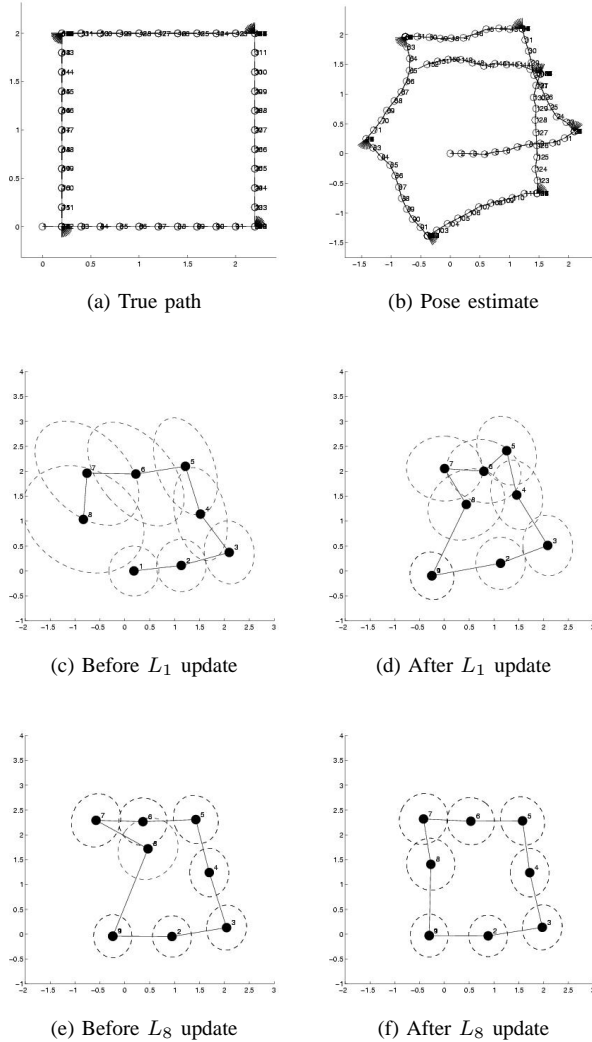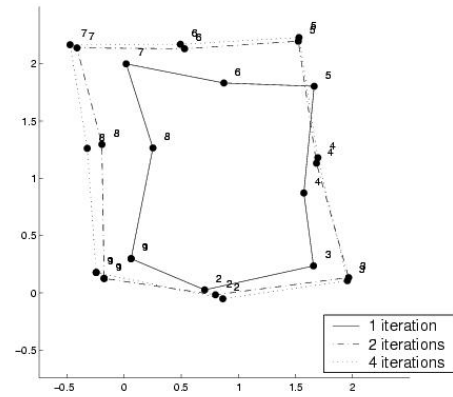
(a) True path      (b) Pose estimate



Fig. 3. Effects of the iterative Kalman Filter on the position estimates. Final landmark positions for 1, 2, and 4 iterations per update step are shown.
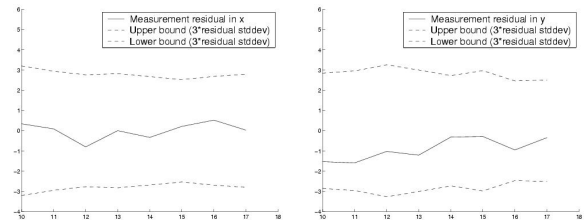


(c) Before $L_1$ update      (d) After $L_1$ update



Fig. 4. Plots showing the sensor residual $r = z - \hat{z}$ and the $3\sigma$ upper and lower bounds of the residual covariance $S$. These residuals are all for landmark positions that have been visited a second time. These residuals were computed with no iterations of the EKF.



(e) Before $L_8$ update      (f) After $L_8$ update

Fig. 2. Illustrations showing various stages of the Kalman update for the simulated runs. The path starts from the lower left, moves counter-clockwise, and is traversed twice. The scale is in meters.

their position. Fig. 5(c) compares the corrected positions of the landmarks with the true path of the robot. Landmarks occurred when the path intersected itself. There is a slight offset of some of the landmark positions, but the relative structure has greatly improved.

## V. Conclusions and Future Work

We introduced a method for performing SLAM with sensor-poor robots in which we formulate the EKF to relax the assumption that our sensors return metric distance information to landmarks. We have proposed the idea of using conventional sensor modalities as a "virtual sensor" which is used to determine whether the robot has returned to a location that it has visited before. In this formalism, landmarks are sensor signatures and indicate locations the robot has visited. The virtual sensor is both the strength

and the weakness of the method as it allows correlations to be found between locations that the robot has visited, but global metric information, such as orientation, can be difficult to capture (Fig. 5 shows a slight global misalignment in rotation). We have shown the effectiveness of this algorithm on simulated and real world data.

In this work, we assumed that sensor readings are unique enough that obtaining the location of where they cross is a simple matter of choosing a threshold of matched features. This technique is only valid for environments in which the individual locations have enough features to be significantly different (e.g. perceptual aliasing effects is ignored). In general, this is not the case and there can be multiple locations that appear to be the same. In future work, we will investigate the use of the Multiple Hypothesis Tracking (MHT) extension to the EKF [14] as a solution to this problem. MHT is capable of representing more general probability distributions and is thus a more appropriate representation of the landmark decision region than a single ellipsoid. Additionally, this method will be evaluated on different sensor modalities besides cameras. Finally, since this is a passive mapping method that is decoupled from exploration, we will explore how to combine it with active sensing techniques.

(a) True path of the robot vs. the erroneous path caused by errors in odometry estimates

(b) The noisy odometric path estimate of the robot and uncorrected landmark estimates. The $1\sigma$ position uncertainty for each landmark are shown with dashed lines.

(c) The true path of the robot and corrected landmark estimates. The $1\sigma$ position uncertainty for each landmark are shown with dashed lines.
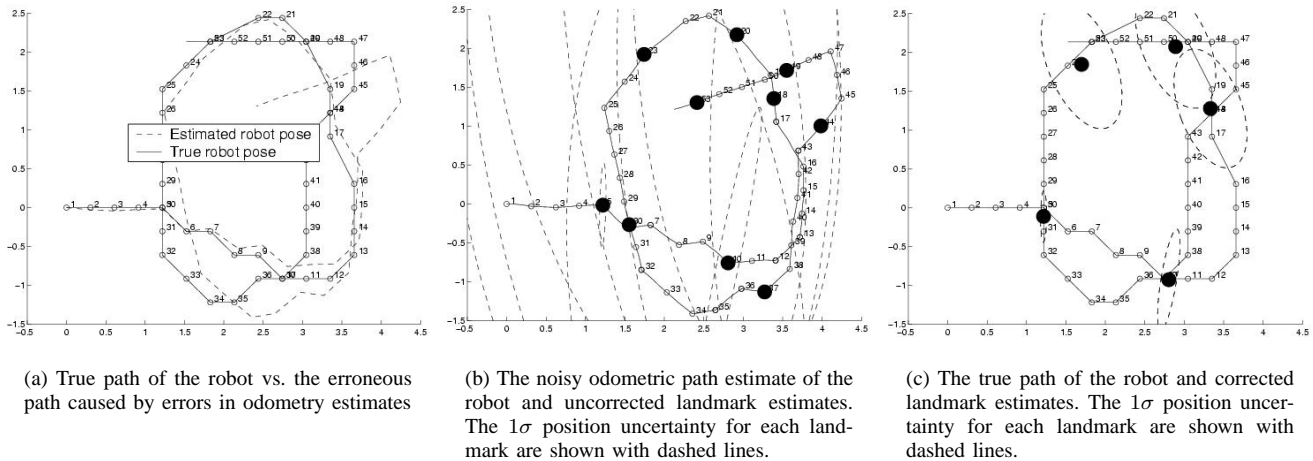
Fig. 5. Real world experiments in an indoor environment (scale is in meters). Landmarks in the true path occur wherever there is an intersection in the path. Positions in the path are labeled chronologically and landmark positions are labeled as large dots.

## VI. REFERENCES

[1] M. Betke and L. Gurvits. Mobile robot localization using landmarks. *IEEE Transactions on Robotics and Automation*, 13(2):251–263, April 1997.

[2] F. Dellaert and A. Stroupe. Linear 2d localization and mapping for single and multiple robots. In *Proc. of the IEEE Int'l Conf. on Robotics and Automation*, May 2002.

[3] M. W. M. G. Dissanayake, P. Newman, S. Clark, H. F. Durrant-Whyte, and M. Csorba. A solution to the simultaneous localization and map building (SLAM) problem. *IEEE Trans. on Robotics and Automation*, 17(3):229–241, June 2001.

[4] T. Duckett, S. Marsland, and J. Shapiro. Learning globally consistent maps by relaxation. In *Proc. of the IEEE Int'l Conf. on Robotics and Automation*, volume 4, pages 3841–3846, 2000.

[5] A. Gelb. *Applied Optimal Estimation*. MIT Press, 1994.

[6] A. Howard, M. Matarić, and G. Sukhatme. Localization for mobile robot teams using maximum likelihood estimation. In *Proc. of the IEEE/RSJ Int'l Conf. on Intelligent Robots and Systems*, EPFL Switzerland, Sept. 2002.

[7] KLT: An implementation of the Kanade-Lucas-Tomasi feature tracker. http://robotics.stanford.edu/~birch/klt/.

[8] B. Kuipers and Y.-T. Byun. A robot exploration and mapping strategy based on a semantic hierarchy of spatial representations. *Journal of Robotics and Autonomous Systems*, 8:47–63, 1991.

[9] J. J. Leonard and H. F. Durrant-Whyte. Mobile robot localization by tracking geometric beacons. *IEEE Trans. on Robotics and Automation*, 7(3):376–382, 1991.

[10] D. Lowe. Object recognition from local scale-invariant features. In *Proceedings of the Seventh IEEE International Conference on Computer Vision*, pages 1150–7, 1999.

[11] B. D. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. *IJCAI*, pages 674–679, 1981.

[12] P. S. Maybeck. *Stochastic Models, Estimation and Control*, volume 141–142 of *Mathematics in Science and Engineering*. Academic Press, 1982.

[13] J. Neira and J. Tardos. Data association in stochastic mapping using the joint compatibility test. *IEEE Trans. on Robotics and Automation*, 17(6):890–897, Dec. 2001.

[14] S. I. Roumeliotis and G. A. Bekey. Bayesian estimation and kalman filtering: A unified framework for mobile robot localization. In *Proc. of the IEEE Int'l Conf. on Robotics and Automation*, pages 2985–2992, San Francisco, CA, April 2000.

[15] P. E. Rybski, S. A. Stoeter, M. Gini, D. F. Hougen, and N. Papanikolopoulos. Performance of a distributed robotic system using shared communications channels. *IEEE Trans. on Robotics and Automation*, 22(5):713–727, Oct. 2002.

[16] P. E. Rybski, F. Zacharias, J.-F. Lett, O. Masoud, M. Gini, and N. Papanikolopoulos. Using visual features to build topological maps of indoor environments. In *Proc. of the IEEE Int'l Conf. on Robotics and Automation*, 2003.

[17] H. Shatkay and L. Kaelbling. Learning topological maps with weak local odometric information. In *Proc. of the 15th Joint Conf. on Artificial Intelligence*, pages 920–927, San Mateo, CA, 1997. Morgan Kaufmann.

[18] R. Smith, M. Self, and P. Cheeseman. Estimating uncertain spatial relationships in robotics. In I. J. Cox and G. T. Wilfong, editors, *Autonomous Robot Vehicles*, pages 167–193. Springer-Verlag, 1990.

[19] S. Thrun, W. Burgard, and D. Fox. A probabilistic approach to concurrent mapping and localization for mobile robots. *Machine Learning*, 31:29–53, 1998.

[20] S. Thrun, D. Fox, W. Burgard, and F. Dellaert. Robust monte carlo localization for mobile robots. *Artificial Intelligence*, 101:99–141, 2000.

[21] C. Tomasi and T. Kanade. Detection and tracking of point features. Technical report, School of Computer Science, Carnegie Mellon University, April 1991.

[22] I. Ulrich and I. Nourbakhsh. Appearance-based place recognition for topological localization. In *Proc. of the IEEE Int'l Conf. on Robotics and Automation*, pages 1023–1029, San Francisco, CA, April 2000.