Location tracker for a mobile robot

Abdul Bais, Robert Sablatnig, Jason Gu and Yahya M. Khawaja

Abstract— **This paper presents landmark based self-localization of a two-wheeled differential drive autonomous mobile robot in a known but highly dynamic environment. The robot is equipped with a pivoted stereo vision system, two digital encoders, a gyro sensor, two 10g accelerometers and a magnetic compass. Global position of the robot is estimated using range measurements of distinct features such as color transitions, corners, junctions and line intersections in the robot environment. However, due to scarcity of distinct features, it is not possible to extract the minimum required features for global position estimation from everywhere in the state space. Therefore, the robot position is tracked between intermittent global localization to have an all time position estimate available to the robot. The robot observation vector is composed of range and bearing measurements of distinct features in the robot environment which is merged with the current position estimate to suppress the accumulating errors. Simulation results illustrate the performance of the location tracker.**

I. Introduction

Position estimation is a basic requirement of autonomous navigation. One of the solutions to this problem is to start at a known location and track the robot position locally using methods such as odometry or inertial navigation [1]. These methods have proven to be efficient and provide good short term position estimates but suffer from unbounded error growth due to integration of minute measurements to obtain the final estimate [2].

Another approach is to estimate the robot position globally using external sensors [2]. The process of global position estimation is simplified by engineering the robot environment with active beacons or other artificial landmarks such as bar code reflectors and visual patterns. Methods that do not require modification of the environment are less accurate and demand significantly more computational power. This leads to techniques where local measurements are fused with measurements from the robot environment [3], [4]. However, the robot must be able to estimate its position from the very beginning or when/if it loses track of its position during navigation.

Extended Kalman filter [5], [6] has been extensively used for information fusion in robot navigation problems. Leonard and Durrant-Whyte [7] have formulated the localization algorithm as a tracking problem. They extract naturally occurring geometric beacons from sonar data as landmarks and match them to a known navigation map to maintain an estimate of robot location. Matthies and Shafer [8] apply an extended Kalman filter for motion estimation using a stereo vision system. The robot position is tracked based on information gathered from features between successive frames as the robot moves from one point to another. The drawback of such an approach is that without referencing features in the global map, the uncertainty increases with the passage of time. A system for RoboCup robots compute depth using calibrated single camera images [9]. Uniquely colored regions in the robot environment are used to help in correspondence analysis. Information from features and odometry is combined using Kalman filter.

Due to its unimodal nature, Kalman filter is not suitable for global localization in environments where features are not distinctive in the global space. To solve this problem some researchers generate and track multiple hypothesis to constitute a framework for global Kalman filter [10], [11]. With the passage of time correct hypothesis collects more evidence and the wrong one disappears.

Other approaches to global self-localization include Markov localization [12] and Monte Carlo localization [13] methods. These methods are capable of recovering a robot from tracking failures and can deal with multi-modal densities. The Markov localization model can represent any probability density function over the robot position and is based on the assumption that the robot position is the only state in the world. Special extensions are required in dynamic environments to filter the damaging effect of sensor data corrupted by external dynamics [14].

A more efficient approach, Monte Carlo localization, represent probability density by a set of samples that are randomly drawn from it. Fichtner and Grossmann [15] report a sensor model and its use in a Monte Carlo frame work for camera based pose estimation. Kraetzschmar and Enderle [16] evaluate the Monte Carlo localization methods in an environment with sporadic features. They use angle and distance information of different features in the environment using single camera images.

For position estimation and interaction with other objects in an intelligent way, autonomous robots need to estimate depth information of interesting objects in its environment. This can be done with a wide range of sensors. The most accurate sensor to compute depth is a laser-range-finder. The drawback of this sensor is that it is too big to be integrated in our robot and also the generation of a complete 3D map of the environment would take too long. Depth estimation using single image is too erroneous and the approach cannot be used

A. Bais started his PhD at Institute of Computer Technology, Vienna University of Technology, Vienna, Austria, in 2004. During his PhD he spent one year at Robotic Research Laboratory of Dalhousie University Canada, where he focused on information fusion for robot navigation (Phone: +43-1-58801-38410, Fax: +43-1-58801-38499, Email: bais@ict.tuwien.ac.at).

R. Sablatnig is with Pattern Recognition and Image Processing Group, Institute of Computer Aided Automation, Vienna University of Technology, Vienna, Austria.

J. Gu is with Robotics Research Laboratory, Department of Electrical and Computer Engineering, Dalhousie University, Halifax, Canada.

Y. M. Khawaja is with the Department of Electrical Engineering, NWFP University of Engineering and Technology, Peshawar, Pakistan.

all the time [17]. Therefore, a stereo vision system is used to obtain 3D information from the robot environment as it can be built small enough to fit the required dimensions.

The robot environment is assumed to be well known but highly dynamic and cannot be modified. Two different types of features are found in the environment: line based and color transitions [18]. Line based features are extracted using gradient based Hough transform which provides the strongest groupings of collinear pixels having roughly the same edge orientation. Corners, junctions and line intersections are determined by semantic interpretation of detected line segments. Color transitions between distinct colored patches are detected using segmentation methods. We have two methods for global position estimation that require as many landmarks at the most [19], [20]. However, instantaneous acquisition of distinct landmarks all the time is not possible as landmarks are few in number and are frequently occluded. In order to have an all time position estimate, this study aims at tracking the robot position. The growing position error is suppressed by acquiring features from the robot environment whenever possible.

We assume that there exist features that can be identified globally. This helps us to avoid local minima traps and saves us from tracking multiple hypothesis or to maintain a multi-modal position belief. Keeping these requirement in view, the extended Kalman filter is considered to be best suited for our application. In addition to the stereo vision system, our robots are equipped with two digital encoders, two 10g acceleration sensors, a gyro sensor and a magnetic compass for determining the absolute orientation of the robot [21].

The balance of the paper is structured as follows: Section II explains the transition model in the framework of an extended Kalman filter, derives control vector and presents its uncertainty analysis. Section III illustrates the state update model, talks about observation prediction and discusses robot observation and its uncertainty. Experimental results are presented in Section IV and finally the paper is concluded in Section V.

II. State transition model

We begin by assuming that we have a function $\mathbf{f}$ that models the transition from robot's state $\mathbf{p}_{k-1}$[1] to $\mathbf{p}_k$ in the presence of control vector $\mathbf{u}_k$ at time k. The control vector is independent of state $\mathbf{p}_{k-1}$ and is supposed to be corrupted by an additive zero mean Gaussian noise $\widetilde{\mathbf{u}}_k$ of covariance $\mathbf{U}_k$. This model is stated as follows:

$$\mathbf{p}_k = \mathbf{f}(\mathbf{p}_{k-1}, \mathbf{u}_k, k) \tag{1}$$

where $\mathbf{u}_k = \widehat{\mathbf{u}}_k + \widetilde{\mathbf{u}}_k$. The quantities $\mathbf{p}_{k-1}$ and $\mathbf{p}_k$ are the desired (unknown) states of the robot. Given robot observations $\mathbf{Z}_{k-1}$, a minimum mean square estimate $\widehat{\mathbf{p}}_{k-1|k-1}$ of the robot state $\mathbf{p}_{k-1}$ at time $k-1$ is $E\{\mathbf{p}_{k-1}|\mathbf{Z}_{k-1}\}$ [6]. The uncertainty of this estimate is denoted by $\mathbf{P}_{k-1|k-1}$ and is evaluated as $E\{\widetilde{\mathbf{p}}_{k-1|k-1}\widetilde{\mathbf{p}}^T_{k-1|k-1}|\mathbf{Z}_{k-1}\}$ where $\widetilde{\mathbf{p}}_{k-1|k-1}$ is the zero mean estimation error.

[1]The notation we use is a slightly changed version of [6]. In our case $\widehat{\mathbf{p}}_{i|j}$ is the minimum mean square estimate of $\mathbf{p}_i$ given observation until time j i.e. $\mathbf{Z}_j$. Similarly, $\mathbf{P}_{i|j}$ is the uncertainty of the estimate $\widehat{\mathbf{p}}_{i|j}$

Using multivariate Taylor series expansion $\mathbf{p}_k$ is linearized around $\widehat{\mathbf{p}}_{k-1|k-1}$ and $\widehat{\mathbf{u}}_k$ to derive $\widehat{\mathbf{p}}_{k|k-1}$, which is an estimate of $\mathbf{p}_k$ using all measurements but the one at time k. This is called *prediction* and can be written as follows:

$$\widehat{\mathbf{p}}_{k|k-1} \approx \mathbf{f}(\widehat{\mathbf{p}}_{k-1|k-1}, \widehat{\mathbf{u}}_k, k) \tag{2}$$

and its error estimate by:

$$\widetilde{\mathbf{p}}_{k|k-1} \approx \mathbf{J}_1\widetilde{\mathbf{p}}_{k-1|k-1} + \mathbf{J}_2\widetilde{\mathbf{u}}_k \tag{3}$$

where $\mathbf{J}_1$ and $\mathbf{J}_2$ are the jacobian of (1) w.r.t $\mathbf{p}_{k-1}$ and $\mathbf{u}_k$ evaluated at $\widehat{\mathbf{p}}_{k-1|k-1}$ and $\widehat{\mathbf{u}}_k$ respectively. Using (3) prediction uncertainty $\mathbf{P}_{k|k-1}$ is given as follows:

$$\mathbf{P}_{k|k-1} = \mathbf{J}_1\mathbf{P}_{k-1|k-1}\mathbf{J}_1^T + \mathbf{J}_2\mathbf{U}_k\mathbf{J}_2^T \tag{4}$$

Our two wheel differential drive robot is assumed to be moving on flat surface with its pose having 3 degrees of freedom. As shown in Fig. 1, the world coordinate system is represented by $\mathrm{X^W}$ and $\mathrm{Y^W}$ axis and the robot coordinate system by $\mathrm{X^{C0}}$ and $\mathrm{Y^{C0}}$ axis. Objects in the world coordinate system have coordinates x, y, and z. The robot motion is always assumed to be a flat surface therefore the robot position is denoted by x, y, and θ. Rotation of the robot coordinate system with respect to the world coordinate system is represented by angle θ. Objects in the robot coordinate system are represented by x^{C0}, y^{C0}, and z^{C0} coordinates. The separation between the two wheels, wheel base, of the robot is *w*. Movement of the robot center is considered as motion of the whole robot. The type of trajectory followed by the robot depends on the velocity of each wheel. If the two values are equal the robot travels along a straight line. A curved path is traversed if the two wheels rotate at different velocities, whereas the robot rotates around its center of mass if the velocities are equal in magnitude but opposite in direction.

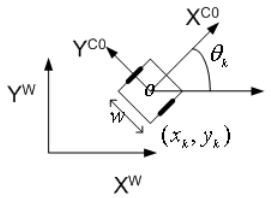


Fig. 1. Representation of robot pose

Fig. 2 shows robot trajectory between time step $k-1$ and k. The distance covered (per unit time) by the left and right wheels of the robot is denoted by v_{lk} and v_{rk} respectively. During this time the robot pose changes from $\mathbf{p}_{k-1}$ to $\mathbf{p}_k$. It is assumed that between the two time intervals the robot moves with a constant velocity, which results in a trajectory of constant radius of curvature c_k. We also assume that at time $k-1$ we have an estimate $\widehat{\mathbf{p}}_{k-1|k-1}$ of the robot pose, its uncertainty $\mathbf{P}_{k-1|k-1}$ and a control vector $\widehat{\mathbf{u}}_k$.

The control vector $\mathbf{u}_k$ is defined as action taken by the robot that causes a state change in the robot frame of reference. It is based on distance covered by the two wheels of the robot and can be derived by referring to Fig. 2 as given by the following transformation:

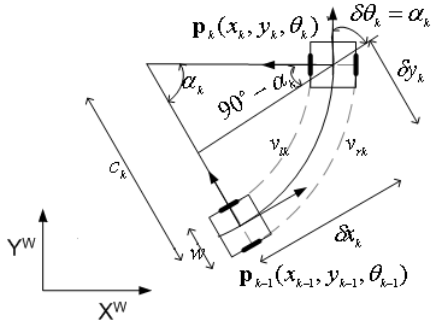


Fig. 2. Geometric construction of the differential drive robot

$$\mathbf{u}_k = \begin{bmatrix} \delta x_k \\ \delta y_k \\ \delta\theta_k \end{bmatrix} = \begin{bmatrix} \frac{w(v_{rk}+v_{lk})}{2(v_{rk}-v_{lk})} sin(\frac{v_{rk}-v_{lk}}{w}) \\ \frac{w(v_{rk}+v_{lk})}{2(v_{rk}-v_{lk})}(1 - cos(\frac{v_{rk}-v_{lk}}{w})) \\ \frac{v_{rk}-v_{lk}}{w} \end{bmatrix} \quad (5)$$

A. Control vector's uncertainty

For uncertainty analysis of the control vector we suppose that the robot's odometry is properly calibrated for systematic errors [22]. For the non-systematic errors we start by making a basic assumption about error in distance traveled by the robot and propagate it to the robot control vector. Suppose the deviation $\widetilde{\mathbf{v}}_k$ of the estimated velocity vector $\widehat{\mathbf{v}}_k$ from its true value $\mathbf{v}_k$ is a random vector of zero mean and covariance $\mathbf{\Sigma}_v$. The two wheels of the robot have same diameter and the distance covered by each wheel is measured by two independent encoders, hence, it is reasonable to assume that error in distance covered by the left wheel is independent of the error in distance covered by the right wheel [1]. The covariance matrix of this error is given as follows:

$$\mathbf{\Sigma}_v = E\{\widetilde{\mathbf{v}}_k \widetilde{\mathbf{v}}_k^T\} = \begin{bmatrix} \sigma_l^2 & 0 \\ 0 & \sigma_r^2 \end{bmatrix} \quad (6)$$

where σ_l^2 and σ_r^2 are the variances of v_{lk} and v_{rk} and are proportional to their absolute values. This error is propagated to $\mathbf{u}_k$ using the transformation given by (5). Using first order approximations of (5) we write the expression for $\mathbf{U}_k$ as follows:

$$\mathbf{U}_k \quad = \quad \mathbf{J}_u \mathbf{\Sigma}_v \mathbf{J}_u^T$$

where $\mathbf{J}_u$ is the jacobian of u_k with respect to $\mathbf{v}_k$ evaluated at $\widehat{\mathbf{v}}_k$ and is given by (7) .

Chong and Kleeman [1] report a detailed analysis of uncertainty in odometry of a differential drive robot. They divide the curve traversed by the robot between two time intervals into infinitely small steps with the assumption that the radius of curvature for all the intermediate steps remains constant. This way the uncertainty in final position of the robot stays the same irrespective of the number of steps the distance is covered in. Our observation is that if the radius of curvature is assumed to be constant then there is no need of this division. The problem that traveling the same distance with different time steps results in different uncertainty is solved by making the uncertainty proportional to the distance covered and if one starts at the low level of wheels counts and distance covered.

B. Special cases and compounding of transformations

Robot trajectory approaches a straight line when the two wheels start turning at the same velocities, whereas it rotates about its center of mass when the two velocities are same in magnitude but opposite in direction. For straight line translation we have $v_{rk} - v_{lk} \rightarrow 0$ which result in $\alpha_k \rightarrow 0$ and $c_k \rightarrow \infty$ and the control vector (5) becomes $\begin{bmatrix} v_k & 0 & 0 \end{bmatrix}^T$. Here v_k is the velocity of the robot center. In case of rotation around robot's center of mass we have $\alpha_k \rightarrow \frac{2v_{rk}}{w}$ and $c_k \rightarrow 0$ when $v_{rk} \rightarrow -v_{lk}$. The control vector in this case becomes $\begin{bmatrix} 0 & 0 & {}^{2v_{rk}}/_{w} \end{bmatrix}^T$.

From motion model of (1) we know that the robot state changes from $\mathbf{p}_{k-1}$ to $\mathbf{p}_k$ under the influence of control vector $\mathbf{u}_k$. A simplified version of this transition is illustrated in Fig. 3, which help us in incorporating the control vector with the current state to arrive at a new state of the robot. Using illustration of Fig. 3 we may write (1) as follows:

$$\mathbf{p}_k = \begin{bmatrix} x_k \\ y_k \\ \theta_k \end{bmatrix} = \begin{bmatrix} x_{k-1} + \delta x_k cos(\theta_{k-1}) - \delta y_k sin(\theta_{k-1}) \\ y_{k-1} + \delta x_k sin(\theta_{k-1}) + \delta y_k cos(\theta_{k-1}) \\ \theta_{k-1} + \alpha_k \end{bmatrix} \quad (8)$$

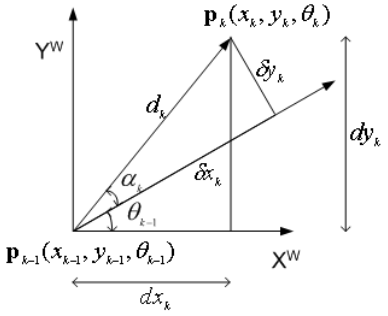


Fig. 3. Compounding of transformation

In Fig. 3 dx and dy represent the x and y components of the total state change in the world coordinate system.

III. Robot observation model

The robot observation model links the current state of the robot with its observation as given by the following transformation:

$$\mathbf{z}_k = \mathbf{h}(\mathbf{p}_k, k) + \mathbf{w}_k \quad (9)$$

Robot observation is assumed to be corrupted by additive zero mean Gaussian noise $\mathbf{w}_k$ of strength $\mathbf{R}$. Expanding (9) around $\widehat{\mathbf{p}}_{k|k-1}$ we get

$$\mathbf{z}_k \approx \mathbf{h}(\widehat{\mathbf{p}}_{k|k-1}) + \mathbf{J}_{zp}(\widehat{\mathbf{p}}_{k|k-1} - \mathbf{p}_k) + \mathbf{w}_k \quad (10)$$

where $\mathbf{z}_{k|k-1} = E\{\mathbf{z}_k|\mathbf{Z}_{k-1}\} = \mathbf{h}(\widehat{\mathbf{p}}_{k|k-1})$ is the predicted observation and $\mathbf{J}_{zp}$ is the jacobian of (9) evaluated at $\widehat{\mathbf{p}}_{k|k-1}$. The deviation of the actual observation from the predicted one is called $innovation$ and is having the following covariance matrix:

$$\mathbf{S} \quad = \quad \mathbf{J}_{zp} \mathbf{P}_{k|k-1} \mathbf{J}_{zp}^T + \mathbf{R} \quad (11)$$

$$\mathbf{J}_u = \begin{bmatrix} \frac{-(v_{rk}^2 - v_{lk}^2)\cos(\frac{v_{rk}-v_{lk}}{w}) + 2wv_{rk}\sin(\frac{v_{rk}-v_{lk}}{w})}{2(v_{rk}-v_{lk})^2} & \frac{(v_{rk}^2 - v_{lk}^2)\cos(\frac{v_{rk}-v_{lk}}{w}) - 2wv_{lk}\sin(\frac{v_{rk}-v_{lk}}{w})}{2(v_{rk}-v_{lk})^2} \\ \frac{-(v_{rk}^2 - v_{lk}^2)\sin(\frac{v_{rk}-v_{lk}}{w}) + 2wv_{rk}(1-\cos(\frac{v_{rk}-v_{lk}}{w}))}{2(v_{rk}-v_{lk})^2} & \frac{(v_{rk}^2 - v_{lk}^2)\sin(\frac{v_{rk}-v_{lk}}{w}) - 2wv_{lk}(1-\cos(\frac{v_{rk}-v_{lk}}{w}))}{2(v_{rk}-v_{lk})^2} \\ \frac{-1}{w} & \frac{1}{w} \end{bmatrix} \quad (7)$$

Innovation is a measure of disagreement between the actual and predicted state of the robot and is used to correct the predicted state estimate as follows:

$$\widehat{\mathbf{p}}_{k|k} = \widehat{\mathbf{p}}_{k|k-1} + \mathbf{K}[\mathbf{z}_k - \mathbf{z}_{k|k-1}] \quad (12)$$

The covariance of this estimate is evaluated as follows:

$$\mathbf{P}_{k|k} = [\mathbf{I} - \mathbf{K}\mathbf{J}_{zp}]\mathbf{P}_{k|k-1}[\mathbf{I} - \mathbf{K}\mathbf{J}_{zp}]^T + \mathbf{K}\mathbf{R}\mathbf{K}^T \quad (13)$$

where $\mathbf{I}$ is a 2×2 identity matrix. The value of $\mathbf{K}$ that minimizes the mean square estimation error is called Kalman gain and is given by the following expression [6]:

$$\mathbf{K} = \mathbf{P}_{k|k-1}\mathbf{J}_{zp}^T\mathbf{S}^{-1} \quad (14)$$

To observe landmark features in its environment, our robot is equipped with a stereo vision system. Assuming identical cameras, parallel image planes and aligned epipolar lines, a point $\mathbf{p}^{\mathrm{C0}} = \begin{bmatrix} x^{\mathrm{C0}} & y^{\mathrm{C0}} & z^{\mathrm{C0}} \end{bmatrix}^T$ in robot coordinate system and its projections $\begin{bmatrix} u_l & v_l \end{bmatrix}^T$ and $\begin{bmatrix} u_r & v_r \end{bmatrix}^T$ in the left and right image can be related under perspective transformation as follows [23]:

$$\mathbf{p}^{\mathrm{C0}} = \begin{bmatrix} x^{\mathrm{C0}} \\ y^{\mathrm{C0}} \\ z^{\mathrm{C0}} \end{bmatrix} = \mathbf{f}\left(\begin{bmatrix} u_l \\ v_l \end{bmatrix}, \begin{bmatrix} u_r \\ v_r \end{bmatrix}\right) = \begin{bmatrix} x_c + \frac{fb}{u_l - u_r} \\ \frac{-b}{2}\frac{u_l + u_r - 2o_u}{u_l - u_r} \\ \frac{-b(v_r - o_v)}{u_l - u_r} \end{bmatrix} \quad (15)$$

where $[o_u \;\; o_v]^T$ is the image center, b is the baseline of the stereo vision system, f focal length of both cameras and x_c is the distance from the center of the robot to the cameras. Image coordinate system is represented by *u* and *v*. The *u* and *v* axes of the image coordinate system are in opposite direction of Y^{C0} and Z^{C0} axes.

Location of landmark $\mathbf{p}_l = [x_l \;\; y_l]^T$ in robot coordinate system is used to construct robot observation vector $\mathbf{z}_k$ as illustrated in Fig. 4 and given by the following expression:

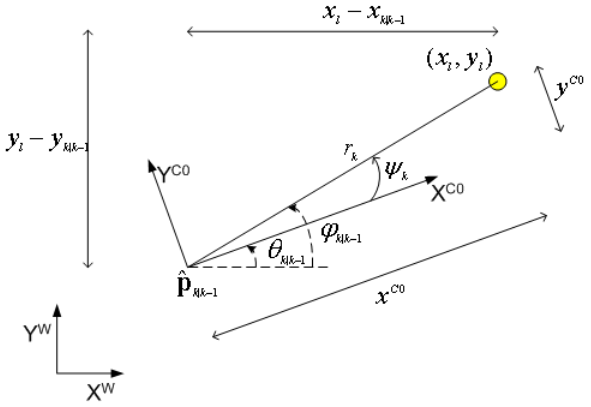


Fig. 4. Illustration of the robot observation: the robot stereo vision system is used to estimate range and bearing with respect to a landmark feature. Four distinct landmark features are shown here

$$\mathbf{z}_k = \begin{bmatrix} r_k \\ \varphi_k \end{bmatrix} = \begin{bmatrix} \sqrt{(x^{\mathrm{C0}})^2 + (y^{\mathrm{C0}})^2} \\ atan2(y^{\mathrm{C0}}, x^{\mathrm{C0}}) \end{bmatrix} \quad (16)$$

where The actual robot observation is compared with what the robot should see from its predicted position. The observation prediction is used in (12) and its derivation is illustrated in Fig. 4 and given by the following expression:

$$\mathbf{z}_{k|k-1} = \begin{bmatrix} \sqrt{(x_l - x_{k|k-1})^2 + (y_l - y_{k|k-1})^2} \\ atan2(y_l - y_{k|k-1}, x_l - x_{k|k-1}) - \theta_{k|k-1} \end{bmatrix} \quad (17)$$

where $\mathbf{p}_l = [x_l \;\; y_l]^T$ represent location of the landmark in world coordinate system, while $x_{k|k-1}$, $y_{k|k-1}$ and $\theta_{k|k-1}$ are components of the robot's predicted pose.

The types of features that are used for self-localization play an important rule in the success or failure of a method. They can broadly be divided into two groups; natural and artificial. Artificial landmarks are specially designed and placed in the environment as an aid in robot navigation. Panzieri et al. [24] reduce accumulating error in the robot position by extracting ceiling lights from vision data as natural landmarks. Similarly, Howard and Kitchen [25] use walls and doorways as landmarks. They maintain a probability distribution across all possible robot positions and track them over time using Baye's rule. Libuda and Kraiss [26] extract natural environment features using a stereo vision system for robot navigation. They use high level interpretation of the scene to extract walls, doors and floor to identify locations.

The drawback of using natural landmarks is that they are difficult to detect with high accuracy. Therefore, the environment is engineered by placing artificial landmarks at predefined positions [27], [28].

For robot observation uncertainty we assume that errors in estimating landmark location in the left and right cameras are independent identically distributed Gaussian random variables having zero mean and covariance matrix given by the following expression:

$$\mathbf{\Sigma}_i = \sigma_{uu}^2 \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad (18)$$

where σ_{uu}^2 is the variance of u_l or u_r. Error in u_l and u_r is propagated to $\mathbf{z}_k$ by the transformation (16) through (15). These systems are nonlinear, however, it is assumed that they can be adequately represented by the first two terms of Taylor series expansion around estimated values of u_l and u_r. Using this assumption we arrive at the following expression of observation covariance:

$$\mathbf{R} = \mathbf{J}_z\mathbf{\Sigma}_i\mathbf{J}_z^T$$

where

$$\mathbf{J}_\mathbf{z} = \frac{-b}{r_k d^2}\begin{bmatrix} fx^{\mathrm{C0}} - (u_r - o_u)y^{\mathrm{C0}} & -fx^{\mathrm{C0}} + (u_l - o_u)y^{\mathrm{C0}} \\ \frac{-x^{\mathrm{C0}}(u_r - o_u) - y^{\mathrm{C0}}f}{r_k} & \frac{x^{\mathrm{C0}}(u_l - o_u) + y^{\mathrm{C0}}f}{r_k} \end{bmatrix} \quad (19)$$

is the jacobian of (16) with respect to $\begin{bmatrix} u_l & u_r \end{bmatrix}^T$.

The error model discussed above adequately captures the uncertainty due to image quantization, however, it fails to handle gross segmentation problems [8]. Moreover, it involves linearization of the perspective transformation which do not hold for distant point correspondences as the disparity decreases and higher order terms will dominate [29].

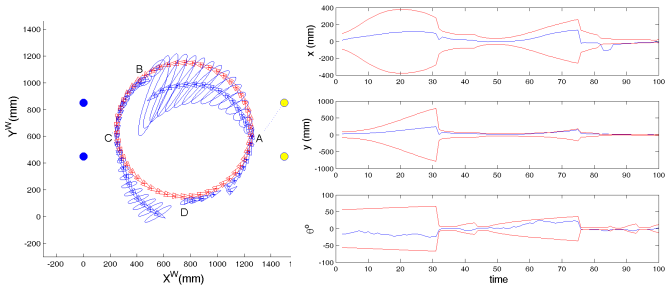
IV. Experimental results

The performance of our algorithm was tested with a simulation as real-world implementations are in progress. The robot wheel base is 75 mm. The stereo vision system is mounted at a height of 70 mm. The separation between the two cameras (stereo baseline) is 30 mm. Image resolution for the experiment was set to be QVGA (320×240 pixels). The experiment was conducted with σ_{uu}^2 value of 0.9. Similarly, σ_l^2 and σ_r^2 values were chosen to be $\left|v_{lk}\right| (0.25)^2$ and $\left|v_{rk}\right| (0.30)^2$ respectively. Image resolution and other simulation parameters were set to satisfy the constraints of the real robots as close as possible.

We group the experiment into three categories according to the trajectory followed by the robot: circular motion, linear motion and rotation about a single point. Thus the curved trajectory and its two special cases are tested. In order to improve the statistics regarding the growth of uncertainty and robot position, different trials are conducted in each category and every trial consists of 100 steps. In each of these trials, the robot either follows a circular path of different diameters, or the starting position of the robot and/or direction of motion is changed. During each trial the robot does not necessarily complete a full 360 rotation.

Fig. 5 shows a single trial of the first category. Here the robot follows a circular path of radius 500 mm. As can be seen from Fig. 5(a) the ideal path that robot is supposed to follow is a perfect circle (the robot starts at 'A' and moves in counterclockwise direction. However, due to imperfections of its sensors the robot deviates from the true path. In this trial the true starting position of the robot is $\begin{bmatrix} 1250 & 650 & 90^\circ \end{bmatrix}$. The robot observes its first landmark feature when it has covered more than a quarter of its intended path marked as 'B'. As can be seen in the figure, uncertainty of the robot position is growing continuously and the pose drifts away from true value. When the robot observes one of the two features on the left side, its uncertainty is reduced and position adjusted. However, as soon as the landmarks are out of sight, drift from the true position starts and the uncertainty increases. Between 'C' and 'D' robot position uncertainty is increasing unbounded. This drift is corrected when the landmarks are visible again from 'D' onward. Fig. 5(b) shows $\pm 3\sigma$ error bound on each component of the robot pose.

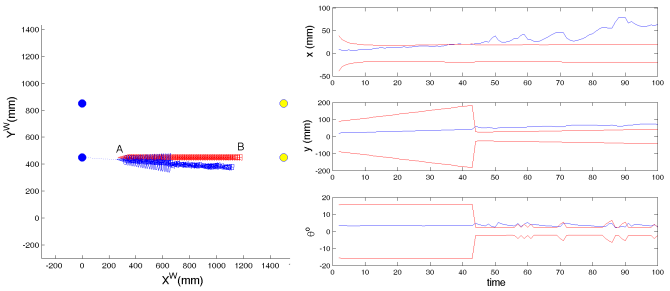
A single trial from the second category is shown in Fig. 6. In this case the robot orientation is fixed at $180^\circ$and it starts at point 'A' on the left side and moves in reverse direction towards the right in small steps of 8.9 mm each. The end point is marked as 'B'. The orientation of the robot is such that the landmarks are always in sight. Fig 6(a) shows the actual and ideal path of the robot. The true starting position of the robot



(a) (b)

Fig. 5. Robot desired trajectory is a circle of radius 500 mm (a) robot starts at a known position with low uncertainty (b) $\pm 3\sigma$ bound on error in x, y and θ

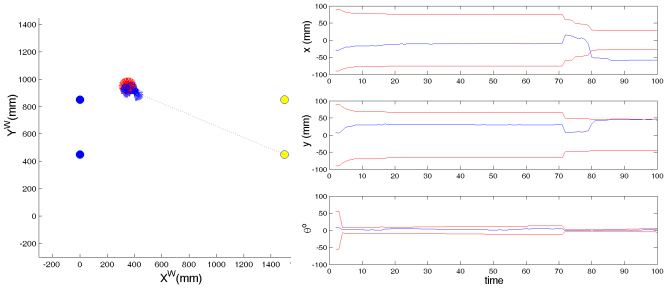
is $\begin{bmatrix} 322 & 450 & 180^\circ \end{bmatrix}$. The error in each component of the robot pose and their corresponding $\pm 3\sigma$ uncertainty bounds are shown in Fig. 6(b).



(a) (b)

Fig. 6. Robot desired path is a straight line (a) robot trajectory (b) position error and the corresponding $\pm 3\sigma$ bound on error in x, y and θ components of the robot pose

Similarly, Fig. 7 shows a single trial from the last category where the robot rotates around its center of mass. Here the robot starts at $\begin{bmatrix} 350 & 950 & -90^\circ \end{bmatrix}$ and completes a 360°rotation in 100 steps in counter clockwise direction. During the entire 100 steps the pose changes of the robot are such that it observes the landmark features only between step 4 and step 8, at step 17, 18 and between 72 and 88. All features observed during the last interval are at a far distance with high uncertainty.



(a) (b)

Fig. 7. The robot is rotating about its center of mass. There is no translation in this case (a) the robot trajectory with an approximately known starting position (b) position error and its covariance ($\pm 3\sigma$ bound)

In the experimental results reported in this paper we initialize the position tracker with a starting position and its corre-

sponding uncertainty. The initial position is obtained by adding random noise to the true position. This can be calculated using landmark based global localization methods [19], [20]. If the initial position is not provided, the robot does not initialize its location tracker and actively searches landmark features until it finds enough that are required for global localization. The ellipses drawn in these figures, illustrate 50% uncertainty for each estimate of the robot position.

Simulation results illustrate that the algorithm can successfully track the robot position despite of the fact that landmarks are sparse. The linearization of the observation model introduce problems for distant features. The robot position error is perfectly bounded when its following a circular path as shown in Fig. 5(b). However, the error is not always bounded in the latter two categories. The reason for this is that in motion along the circle, features are always observed from a short distance whereas in the other cases features are observed from a distance towards the end of the trials. The prominence of this problem comes from the use of a narrow baseline stereo as the construction of the robot does not allow the use of the wide baseline.

V. Conclusion

In this paper we discussed tracking position of an autonomous mobile robot. For this study we assumed that the initial position and its uncertainty is given, however, this can be calculated using global self-localization techniques as reported in [19], [20]. The localization algorithm is not dependent on the presence of artificial landmarks or special structures such as rectangular walls in the robot environment. Furthermore, it is not required that features lie on or close to the ground plane.

A stereo vision system is used as an external sensor that gathers information about features in the robot environment. Features are line based and color transitions [18]. To illustrate the effect of the method and eliminate false correspondences we have used only four landmarks in this study. As a future work we are planning to use all the features found in the robot environment and to analyze the effect of error in feature location in the world coordinate system and in correspondence analysis.

References

[1] K. S. Chong and L. Kleeman, "Accurate odometry and error modelling for a mobile robot," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA '97)*, Albuquerque, New Mexico, April 1997, pp. 2783 – 2788.

[2] J. Borenstein, H. R. Everett, and L. Feng, *Navigating Mobile Robots: Systems and Techniques*. A. K. Peters, Ltd., 1996.

[3] G. Adorni, S. Cagnoni, S. Enderle, and G. K. Kraetzschmar, "Vision-based localization for mobile robots," *Robotics and Autonomous Systems*, vol. 36, pp. 103–119, 2001.

[4] D. Herrero-Pérez, H. Martínez-Barberá, and A. Saffiotti, "Fuzzy self-localization using natural features in the four-legged league," in *RoboCup 2004: Robot Soccer World Cup VIII*, ser. LNCS, D. N. et al., Ed. Springer-Verlag, 2005, pp. 110 – 121.

[5] J. L. Crowley, "Mathematical foundations of navigation and perception for an autonomous mobile robot," in *Workshop on Reasoning with Uncertainty in Robotics*, 1995, pp. 9–51.

[6] P. Newman, *An Introduction to Estimation and its Application to Mobile Robotics*, 2nd ed., Robotics Research Group, Department of Engineering Science, University of Oxford, October 2005, lecture notes.

[7] J. Leonard and H. Durrant-Whyte, "Mobile robot localization by tracking geometric beacons," *IEEE Transactions on Robotics and Automation*, vol. 7, no. 3, pp. 376–382, 1991.

[8] L. Matthies and S. Shafer, "Error modeling in stereo navigation," *IEEE Transaction on Robotics and Automation*, vol. RA-3, no. 3, pp. 239–248, June 1987.

[9] F. Marando, M. Piaggio, and A. Scalzo, "Real time self localization using a single frontal camera," in *Proceedings of the International Symposium on Intelligent Robotic Systems*, Toulouse, France, July 2001.

[10] P. Jensfelt and S. Kristensen, "Active global localization for a mobile robot using multiple hypothesis tracking," *IEEE Transactions on Robotics and Automation*, vol. 17, no. 5, pp. 748–760, October 2001.

[11] K. O. Arras, J. A. Castellanos, M. Schilt, and R. Siegwart, "Feature-based multi-hypothesis localization and tracking using geometric constraints," *Robotics and Autonomous Systems*, vol. 44, pp. 41 – 53, 2003.

[12] D. Fox, W. Burgard, and S. Thrun, "Active markov localization for mobile robots," *Robotics and Autonomous Systems*, vol. 25, no. 3-4, pp. 195–207, November 1998.

[13] S. Thrun, D. Fox, W. Burgard, and F. Dellaert, "Robust monte carlo localization for mobile robots," *Artificial Intelligence*, vol. 128, pp. 99–141, 2001.

[14] D. Fox, W. Burgard, S. Thrun, and A. Cremers, "Position estimation for mobile robots in dynamic environments," in *Fifteenth National Conference on Artificial Intelligence (AAAI-98)*, 1998, pp. 983–988.

[15] M. Fichtner and A. Grossmann, "A visual-sensor model for mobile robot localisation," Artificial Intelligence Institute, Department of Computer Science, Technische Universitaet Dresden, Dresden, Germany, Tech. Rep. WV-03-03/CL-2003-02, 2003.

[16] G. K. Kraetzschmar and S. Enderle, "Self-localization using sporadic features," *Robotics and Autonomous Systems*, vol. 40, pp. 111 – 119, 2002.

[17] S. B. Nickerson, P. Jasiobedzki, D. Wilkes, M. Jenkin, E. Milios, J. Tsotsos, A. Jepson, and O. N. Bains, "The ark project: Autonomous mobile robots for known industrial environments," *Robotics and Autonomous Systems*, vol. 25, no. 1-2, pp. 83–104, October 1998.

[18] A. Bais, R. Sablatnig, and G. Novak, "Line-based landmark recognition for self-localization of soccer robots," in *IEEE International Conference on Emerging Technologies (ICET '05)*, Islamabad, Pakistan, September 2005, pp. 132–137.

[19] A. Bais and R. Sablatnig, "Landmark based global self-localization of mobile soccer robots," in *Computer Vision ACCV 2006: 7th Asian Conference on Computer Vision*, ser. LNCS, P. J. Narayanan, S. K. Nayar, and H.-Y. Shum, Eds., vol. 3852. Hyderabad, India: Springer-Verlag GmbH, January 2006, pp. 842 – 851.

[20] A. Bais, R. Sablatnig, J. Gu, and S. Mahlknecht, "Active single landmark based global localization of autonomous mobile robots," in *Proceedings of 2nd International Conference on Visual Computing (ISVC 2006)*, ser. LNCS, G. B. et al., Ed., no. 4291. Lake Tahoe, Nevada, USA: Springer-Verlag Berlin Heidelberg, November 2006, pp. 202 – 211.

[21] G. Novak and S. Mahlknecht, "TINYPHOON a tiny autonomous mobile robot," in *IEEE International Symposium on Industrial Electronics (ISIE' 05)*, June 2005, pp. 1533–1538.

[22] J. Borenstein and L. Feng, "Measurement and correction of systematic odometry errors in mobile robots," *IEEE Transactions on Robotics and Automation*, vol. 12, no. 6, pp. 869 – 880, December 1996.

[23] E. Trucco and A. Verri, *Introductory TEchniques for 3-D Computer Vision*. Upper Saddle River, NJ, USA: Prentice Hall, 1998.

[24] S. Panzieri, F. Pascucci, R. Setola, and G. Ulivi, "A low cost vision based localization system for mobile robots," in *9th Mediterranean Conference on Control and Automation (MEDSYMP'01)*, Dubrovnik, Croatia, June 2001.

[25] A. Howard and L. Kitchen, "Navigation using natural landmarks," *Robotics and Autonomous Systems*, vol. 26, pp. 99 – 115, 1999.

[26] L. Libuda and K.-F. Kraiss, "Identification of natural landmarks for vision based navigation," in *Conference on Mechatronics and Robotics*, P. P. Drews, Ed., vol. III, Aachen, Germany, September 2004, pp. 877–882.

[27] A. J. Briggs, D. Scharstein, D. Braziunas, C. Dima, and P. Wall, "Mobile robot navigation using self-similar landmarks," in *Proceedings of IEEE International Conference on Robotics and Automation (ICRA '98)*, 2000, pp. 1428–1434.

[28] X. J. Li, A. T. P. So, and S. K. Tso, "Cad-vision-range-based self-localization for mobile robot using one landmark," *Journal of Intelligent and Robotic Systems*, vol. 35, pp. 61–81, 2002.

[29] J. Kriegman, E. Triendl, and T. Binford, "Stereo vision and navigation in buildings for mobile robots," *IEEE Trans. Robotics and Automation*, vol. 5, no. 6, pp. 792–803, December 1989.