14 th International Workshop on
ROBOTICS IN ALPE-ADRIA-DANUBE REGION RAAD'05
BUCHAREST , ROMANIA : MAY 26-28, 2005

International Steering Committee:

G. Belforte Politecnico di Torino, Italy
J. F. Bito Centre of Robotics and Automation, Hungary
Th. Borangiu TU Bucharest , Romania
M. Ceccarelli University of Cassino, Italy
K. Dobrovodsky Academy of Sciences, Slovakia
K. Jezernik University of Maribor, Slovenia
Z. Kolíbal Brno University, Czech Rep.
P. Kopacek TU Vienna , Austria
J. Lenarcic Jozef Stefan Institute, Slovenia
D. Noe University of Ljubljana , Slovenia
A. Rovetta Politecnico di Milano, Italy
I.J. Rudas Budapest Tech , Hungary
G. Schmidt TU München, Germany
B. Heimann TU Hannover, Germany

Local Organising Committee

***Chairman*: Th. Borangiu University Politehnica of Bucharest**
***Members*: N.-A. Ivanescu University Politehnica of Bucharest**
F.D. Anton University Politehnica of Bucharest
S.O. Tunaru University Politehnica of Bucharest
M. Dragoicea University Politehnica of Bucharest
A. Dogar University Politehnica of Bucharest
M. Nitulescu Technical University of Craiova

Workshop Organization

***CIMR* - Centre for Research and Training in Robotics, Industrial Informatics and Materials Engineering**

University Politehnica of Bucharest

***Address* : 313, Spl. Independentei, ED 111 Automatica, sector 6, CP: 77206, Bucharest, Romania**
Tel: +40 21 402 93 68; Fax: +40 21 317 09 12; E-mail: borangiu@cimr.pub.ro

Back To Menu

Proceedings of RAAD'05, 14th International Workshop on Robotics in Alpe-Adria-Danube Region
Bucharest May 26-28, 2005

IMPROVEMENT OF AN EXTENDED KALMAN FILTER BASED MOBILE ROBOT LOCALIZATION

Edouard Ivanjko, Ivan Petrović

University of Zagreb, Faculty of Electrical Engineering and Computing,
Department of Control and Computer Engineering in Automation
Unska 3, HR-10000 Zagreb,
E -mail: edouard.ivanjko@fer.hr; ivan.petrovic@fer.hr

ABSTRACT- A mobile robot must track its pose in the environment in order to perform any useful task. Problem of finding and tracking the mobile robot's pose is called localization, and can be global or local. In this paper we address the local localization or mobile robot pose tracking with prerequisites of known starting pose, mobile robot kinematics and world model. Pose tracking is mostly based on odometry, which has the problem of accumulating errors in an unbounded fashion. To overcome this problem sensor fusion is commonly used. Extended Kalman filter with two approaches has been used in our work for this purpose. First approach uses only sonar's as additional sensors and the second one uses also the built in mobile robot so-called odometric device. Since occupancy grid maps are used, only sonar range measurement uncertainty has to be considered, unlike feature-based maps where an additional uncertainty regarding the feature/range reading assignment must be considered. Thus the numerical complexity is reduced. Proposed localization implementations are experimentally evaluated on a Pioneer 2DX mobile robot in two different set-ups.

Keywords: wheeled mobile robots, localization, sensor fusion, Kalman filter, occupancy grid maps.

INTRODUCTION

Mobile robot localization is one of the very important tasks in navigation of autonomous mobile robots [1]. In an indoor environment with a flat floor plan it becomes a matter of estimating the mobile robot pose, i.e. Cartesian coordinates (x, y) of the robot (mobile robot position) and its orientation Θ. Localization can be local or global. A commonly localization system has both modules implemented. Global localization is usually engaged for self-localization or for lost/kidnapped robot problem solving. Computational complexity of global localization algorithms is much greater then local localization algorithms. In a case of local localization only the offset from the known start pose can be obtained so only a small part of known world model has to be searched/compared to real robot measurements versus global localization which searches the whole known world model. Only so can the unknown mobile robot pose be determined inside of the known world model.

Odometry is one of the most important means of mobile robot local localization. The disadvantage of this method is its unbounded error accumulation due to wheel slippage, floor roughness, etc. To decrease the odometry localization error, odometry is often used with additional sensors like a compass, gyro, laser or sonar range sensors, stereo or mono vision, etc [2]. Odometry and additional sensor information are mostly fused using Kalman filtering techniques [10]. One of them is the Extended Kalman filter (EKF) [4, 9]. The state that has to be estimated using an EKF is the mobile robot pose. A mobile robot kinematics model is firstly used to predict new mobile robot pose, which is then combined with an occupancy grid map to predict sonar range measurements. Comparing predicted sonar range measurements with real sonar range measurements a mobile robot pose correction can be obtained. So the odometry method unbounded errors can be significantly reduced and estimated pose can be used between absolute pose updates based on global localization methods.

The EKF is here used to match recent sensory information against prior knowledge of the environment, i.e. a world model. Due to problems with bad sonar placement on our mobile robot [5], a sonar based local map with high quality can't be determined and so a technique similar as in [9] is adopted that can find direct correspondences between mobile robot sensor measurements and sensor measurements predicted from a world model. Using a threshold comparison between predicted and real sonar range measurements false sonar range measurements can be easily rejected.

Mobile robot pose estimate accuracy depends in this approach on the world model and sensor

measurement quality. Occupancy grid world model quality is limited by used grid cell. Smaller grid cell sizes give a better pose estimate but memory consumption and computational time needed for sonar range measurement prediction increases. To improve the pose estimate without significant memory consumption and computational time increase, an odometric sensor is built in many mobile robots and it is included as an additional sensor into our EKF localization framework [7]. Its advantage is the more frequent access to wheel speed measurements because it's usually implemented in the mobile robot micro controller used as an interface between the mobile robot sensors and mobile robot control computer i.e. low level motor/sensor control and high level navigational tasks.

The paper is organized as follows. The section MOBILE ROBOT MODEL presents used mobile robot kinematic model with brief description of its calibration and section EKF BASED LOCALIZATION describes our firstly implemented EKF localization solution. Then section IMPROVED EKF BASED LOCALIZATION presents the EKF localization, which uses the mobile robot odometric device. Experiment description and experimental results are given in section EXPERIMENTAL RESULTS. The paper ends with CONCLUSION section.

MOBILE ROBOT MODEL

Mobile robot used in our experiments is a three-wheeled robot. Two front wheels are drive wheels with encoder mounted on them and the third wheel is a castor wheel needed for robot stability. Drive wheels can be controlled independently from each other. The encoders can measure the speed or the travelled distance of the wheel. We are using encoders to measure the speed of the wheels. Kinematics of used three wheeled mobile robot are given by the following relations (Figure 1):

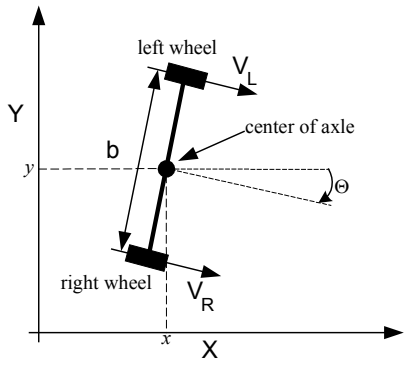

Figure 1. The mobile robot kinematics.

$$x(k+1) = x(k) + D \cdot \cos\Theta(k+1), \quad (1)$$

$$y(k+1) = y(k) + D \cdot \sin\Theta(k+1), \quad (2)$$

$$\Theta(k+1) = \Theta(k) + \Delta\Theta(k), \quad (3)$$

$$D(k) = v_t(k) \cdot T, \quad (4)$$

$$\Delta\Theta(k) = \omega(k) \cdot T, \quad (5)$$

$$v_t(k) = \frac{v_L(k) + v_R(k)}{2} = \frac{\omega_L(k)R + \omega_R(k)R}{2}, \quad (6)$$

$$\omega(k) = \frac{v_R(k) - v_L(k)}{b} = \frac{\omega_R(k)R - \omega_L(k)R}{b}, \quad (7)$$

where are: $x(k)$ and $y(k)$ coordinates of the centre of axle [mm]; $v_t(k)$ robot translation speed [mm/s]; T sampling time [s]; $\Theta(k)$ angle between the vehicle and x-axis [°]; $v_L(k)$ and $v_R(k)$ velocities of the left and right wheel, respectively [mm/s]; $\omega_L(k)$ and $\omega_R(k)$ angular velocities of the left and right wheel, respectively [rad/s]; R radius of the two wheels [mm], and b vehicle axle length [mm]. It is assumed that the wheels have the same radius. Sampling time T was 0.1 [s]. Equations (1) to (7) describe the basic odometry pose tracking model and results obtained by simply propagating these equations through time are marked as uncalibrated odometry (UO). In order to compensate the systematic error regarding the unacquaintance of the exact wheel radius and the unacquaintance of the exact axle length we expand Equations (6) and (7) with three additional parameters [3]:

$$v_t(k) = \frac{k_1 \cdot v_L(k) + k_2 \cdot v_R(k)}{2}, \quad (8)$$

$$\omega(k) = \frac{k_1 \cdot v_R(k) - k_2 \cdot v_L(k)}{k_3 \cdot b}, \quad (9)$$

where parameters k_1 and k_2 compensate the unacquaintance of the exact wheel radius and parameter k_3 the unacquaintance of the exact axle length. Thus the replacement of the above-mentioned equations yields the calibrated odometry (CO), which is also used as the motion model for both EKF implementations.

Calibration parameters values were determined by offline optimisation of two experiment data. In the first experiment robot moved straight ahead, and in the second experiment it turned in place for 180 degrees. Exact final pose, wheel speeds and sampling time were recorded during the experiments. This recorded data combined with the above described mobile robot kinematics model were used to iteratively compute final robot pose in an optimisation function for the MATLAB Optimisation toolbox. Optimisation criterion was to minimise the difference between estimated and measured exact final orientation. A more detailed explanation of used systematic error compensation

and parameter value determination through optimisation can be found in [6].

EKF BASED LOCALIZATION

Basic idea of a general Kalman filter approach is to use a motion model to predict the process state in a time update step and then to use additional sensor (in our case sonar) measurements to correct predicted process state in a measurement update step. Block diagram of our EKF implementation is given in Figure 2. Odometric device measurement can be included into the localization framework or excluded and is so presented with a dashed line. First implementation uses only sonar sensors and the second one uses also the built in odometric device (shown with the dashed line box in Figure 2) as an additional sensor.

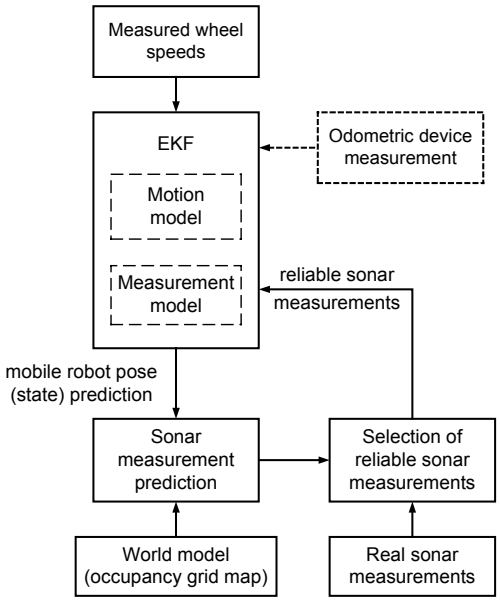

Figure 2. Block diagram of our both EKF localization approaches.

Detailed explanation of firstly implemented EKF localization can be found in [4]. The motion model represents the way in which the current state (in our case the mobile robot pose) is derived from the previous state. State vector is in our case expressed as the mobile robot pose, $\mathbf{x}(k) = [\, x(k),\, y(k),\, \Theta\,(k)\,]^T$, with respect to a global coordinate frame. Each state vector $\mathbf{x}$(k) has a degree of uncertainty that is represented as a 3 by 3 covariance matrix $\mathbf{P}(k)$. Control input, $\mathbf{u}(k)$, represents the movement commands which are acted upon by the mobile robot to move it from time step k to $k+1$. In our implementation control input represents rotation through angle $\Delta\Theta(k)$ followed by a translation through distance $D(k)$:

$$\mathbf{u}(k) = [\, D(k),\, \Delta\Theta(k)\,]^T \tag{10}$$

The state transition function, $\mathbf{f}(\cdot)$, uses the state vector and control input to compute the state vector at the next time step:

$$\mathbf{x}(k+1\,|\,k) = \mathbf{f}(\mathbf{x}(k\,|\,k), \mathbf{u}(k), E\{\mathbf{v}(k)\}), \tag{11}$$

where $\mathbf{v}(k)$ represents unpredictable noise. The noise covariance, $\mathbf{Q}(k)$, was modelled under the assumption that there are two independent sources of error, angular and translational and so the uncertainty in $\Delta\Theta(k)$ and $D(k)$ must be translated into uncertainty in $\mathbf{x}(k+1\,|\,k)$. The expression for $\mathbf{Q}(k)$ is so:

$$\mathbf{Q}(k) = \nabla\mathbf{f}(k)\begin{bmatrix} \Delta\Theta(k)^2\sigma_{\Delta\Theta}^2 & 0 \\ 0 & \sigma_D^2 \end{bmatrix}\nabla\mathbf{f}(k)^T \tag{12}$$

where $\nabla\mathbf{f}(k)$ denotes the Jacobian used to translate the uncertainty in $\Delta\Theta(k)$ and $D(k)$ into uncertainty in $\mathbf{x}(k)$, $\sigma_{\Delta\Theta}^2$ and σ_D^2 are variances of $\Delta\Theta(k)$ and $D(k)$, respectively. The uncertainty in position and orientation $\mathbf{P}(k)$ must also be translated but from time step k to $k+1$:

$$\mathbf{P}(k+1\,|\,k) = \nabla\mathbf{f}'(k)\mathbf{P}(k\,|\,k)\nabla\mathbf{f}'(k)^T + \mathbf{Q}(k), \tag{13}$$

where $\nabla\mathbf{f}'(k)$ denotes the Jacobian used to translate the uncertainty $\mathbf{P}(k)$ from time step k to $k+1$. EKF measurement update step is based on a measurement function that computes the range between the mobile robot and a detected obstacle [9]:

$$h_i(\mathbf{x}(k+1), \mathbf{p}_i) = \sqrt{(x_i - x(k+1))^2 + (y_i - y(k+1))^2}, \tag{14}$$

where $\mathbf{p}_i = (x_i, y_i)$ denotes the point (occupied cell) in the world model detected by the i th sonar. Equations for the EKF measurement update step are then:

$$\mathbf{S}(k) = \nabla\mathbf{h}(k)\cdot\mathbf{P}(k+1)\cdot\nabla\mathbf{h}(k)^T + \mathbf{R}(k), \tag{15}$$

$$\mathbf{K}(k+1) = \mathbf{P}(k+1\,|\,k)\nabla\mathbf{h}(k)^T\mathbf{S}^{-1}(k), \tag{16}$$

$$\mathbf{P}(k+1) = \mathbf{P}(k+1) - \mathbf{K}(k+1)\cdot\mathbf{S}(k)\cdot\mathbf{K}^T(k+1), \tag{17}$$

$$\mathbf{x}(k+1) = \mathbf{x}(k+1) + \mathbf{K}(k+1)\cdot(\mathbf{z}(k+1) - \mathbf{h}(k+1)), \tag{18}$$

where $\mathbf{S}(k)$ presents the innovation covariance matrix, $\nabla\mathbf{h}(k)$ measurement Jacobian, $\mathbf{R}(k)$ measurement noise matrix (diagonal matrix with the $r_i(k)$ noise variance values on the diagonal), $\mathbf{z}(k+1)$ real accepted sonar range measurements,

$\mathbf{h}(k+1)$ predicted accepted sonar range measurements and $\mathbf{K}(k+1)$ Kalman gain. To separate outliers or noisy sonar range measurements, predicted measurements are compared to real measurements and only measurements that differ less than a defined threshold are used for mobile robot pose correction in Equation (18).

IMPROVED EKF BASED LOCALIZATION

As mentioned before most mobile robots are equipped with an odometric device that can measure the mobile robot displacement with a smaller sampling time then the calibrated mobile robot kinematics model in the state transition function (11). The disadvantage of the above-described EKF implementation is that it doesn't use that additional sensor. Also input and sonar noise covariance are constant values. It's well known that poor knowledge off these noise statistics can seriously degrade the Kalman filter performance. To overcome a part of these disadvantages input control noise covariance matrix computation can be expanded into Equation (22) where σ_η^2 presents input noise covariance parameter, which can be adaptively computed during mobile robot motion [7].

To include the odometric device measurement data, the measurement vector $\mathbf{z}(k)$ must be expanded and becomes:

$$\mathbf{z}(k) = \begin{bmatrix} \mathbf{z}_p(k) \\ \mathbf{z}_s(k) \end{bmatrix}, \tag{19}$$

where $\mathbf{z}_s(k)$ represents accepted real sonar measurements and $\mathbf{z}_p(k)$ mobile robot pose updated with the displacement, from previous odometric device pose update through the EKF, measured with the odometric device:

$$\mathbf{z}_p(k) = \begin{bmatrix} x_u + d_1 \\ y_u + d_2 \\ \Theta_u + d_3 \end{bmatrix}. \tag{20}$$

where d_1 represents mobile robot displacement along x-axis, d_2 along y-axis, and d_3 orientation change. Pose $[\, x_u,\, y_u,\, \Theta_u \,]^T$ denotes pose where previous sonar range measurements were available i.e. pose where last odometric device pose correction due EKF was done. Measurement prediction vector $\mathbf{h}(k)$ becomes:

$$\mathbf{h}(k) = \begin{bmatrix} \mathbf{h}_p(k) \\ \mathbf{h}_s(k) \end{bmatrix} = \begin{bmatrix} x(k+1) \\ y(k+1) \\ \Theta(k+1) \\ \mathbf{h}_s(k) \end{bmatrix}, \tag{21}$$

where $\mathbf{h}_s(k)$ represents accepted predicted sonar measurements and $\mathbf{h}_p(k)$ predicted mobile robot pose computed using above described calibrated kinematics model. Measurement noise matrix, $\mathbf{R}(k)$, has also a diagonal form but is expanded to include odometric device measurement noise i.e. it has three additional values (x, y, and orientation pose change measurement noise part).

Odometric device pose is updated every time new sonar range measurements are available and pose correction using Equation (18) is computed. Between these updates it measures the mobile robot displacement to compare it with the predicted one computed using the state transition function (11).

EXPERIMENTAL RESULTS

Described EKF approaches were tested in parallel on a Pioneer 2DX mobile robot from ActivMedia Robotics. Two series of experiments were made to evaluate described localization approaches. The first one was performed in our department corridor, and the second one in a larger room of our department. Main characteristic of the first experiment world, shown in Figure 3, is that it has little features along the axis through the corridor middle and used occupancy grid model is accurate. The mobile robot starts in the right corridor end and travels to its left end. Main characteristic of the second one, shown in Figure 4, is that it has enough features along both axis but used occupancy grid model is less accurate because this room is full of easily movable furniture (like chairs, tables, trash baskets, etc.) which never stays at the same place for a long time. In this experiment set-up the mobile robot starts in the left room and has to reach the right room. These two set-ups enable us to test the local localization algorithm performance in an environment with little features

$$\mathbf{Q}(k) = \sigma_\eta^2 \begin{bmatrix} T + v_t(k)^2 \frac{T^3}{3} \sin^2\Theta(k) & -v_t(k)^2 \frac{T^3}{3} \cos\Theta(k) \sin\Theta(k) & -v_t(k) \frac{T^2}{2} \sin\Theta(k) \\ -v_t(k)^2 \frac{T^3}{3} \cos\Theta(k) \sin\Theta(k) & T + v_t(k)^2 \frac{T^3}{3} \cos^2\Theta(k) & v_t(k) \frac{T^2}{2} \cos\Theta(k) \\ -v_t(k) \frac{T^2}{2} \sin\Theta(k) & v_t(k) \frac{T^2}{2} \cos\Theta(k) & T \end{bmatrix}, \tag{22}$$

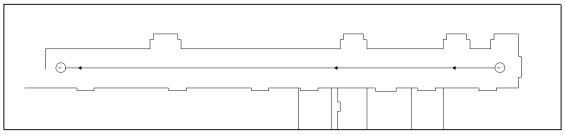

Figure. 3. Model of our department corridor room used for first experiment.

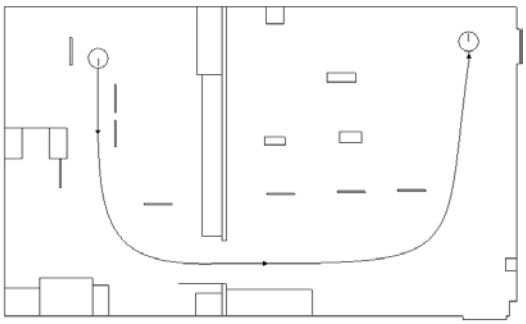

Figure. 4. Model of our department seminar room used for second experiment.

and in a badly or incompletely modelled environment. Both experiment set-ups traversed paths (Figures 3 and 4) are presented by a line with arrows showing mobile robot motion succession. A gradient navigation module [8] is used for mobile robot control, i.e. for path planning and local obstacles avoidance.

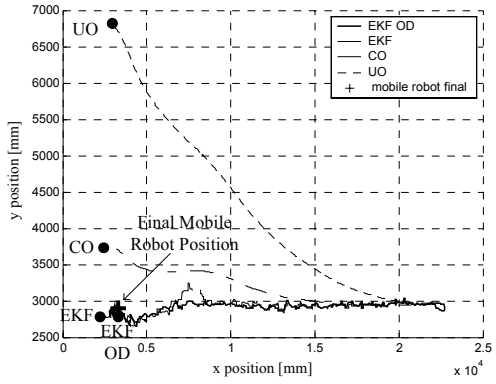

Figure 5. Position estimation performance in the first experiment.

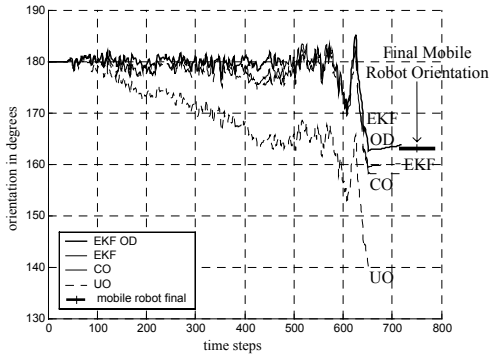

Figure 6. Orientation estimation performance in the first experiment.

New sonar's range measurements $\mathbf{z}(k)$ are available every three time steps on used mobile robot. Raw sonar data $\mathbf{z}(k)$ and predicted measurements $\mathbf{h}(k)$ differ because real mobile robot pose is not exactly known and due to sonar measurement noise, occlusions, specular reflections, outliers, and used occupancy grid inaccuracy. So raw measurements $\mathbf{z}(k)$ were first compared to predicted measurements $\mathbf{h}(k)$ and only those measurements with difference under a certain threshold are accepted. Used threshold is set to 6 cells. That means, in a grid map with cell size of 100 [mm] x 100 [mm], as in our case, measurements with difference less then 600 [mm] are accepted. Obtained results are presented in Figures 5 and 6 for the first experiment set-up and in Figures 7 and 8 for the second experiment set-up. In the first experiment set-up mobile robot orientation was the whole time around 180 degrees so Figure 6 shows the mobile robot orientation estimation during the whole experiment. Second experiment set-up has a greater orientation change (experiment begins with orientation 270, and ends with 90 degrees). To achieve a more clearly result presentation, only orientation estimation results at the experiment end are shown in Figure 8. Exact

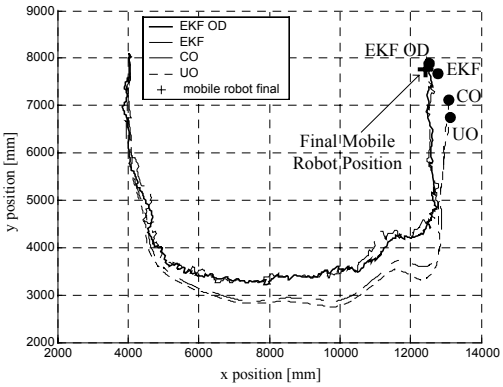

Figure 7. Position estimation performance in the second experiment.

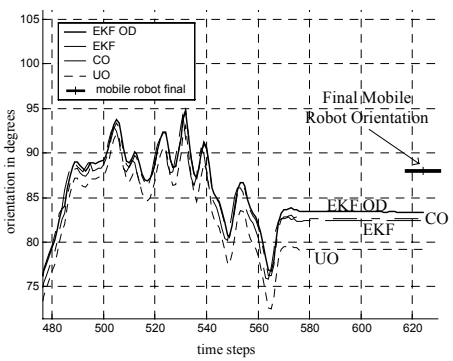

Figure 8. Orientation estimation performance in the second experiment.

	EKF with odometric device	EKF without odometric device	Calibrated odometry (CO)	Uncalibrated odometry (UO)
Position error (%)	0.97	1.52	4.87	9.70
Orientation error (%)	3.46	4.11	5.28	12.73
Average error (%)	2.22	2.82	5.08	11.22

Table 1. Average localization error comparison.

final mobile robot pose was manually measured at experiment end. In here presented figures final mobile robot position part is denoted by a black cross and the orientation part by a short thick line. Estimated final position for each tested technique is denoted by a black dot. EKF implementation that uses the odometric device is labelled as 'EKF OD' and the implementation without it as 'EKF'. Table 1 presents some typical average localization errors from the EKF with odometric device and without it compared to odometry localization without additional sensors. Position error is calculated as:

$$PositionError = \frac{Pos_{act} - Pos_{est}}{Dist} \cdot 100\%, \qquad (23)$$

where Pos_{act} is actual final position, Pos_{est} estimated final position, and $Dist$ total traversed distance. Orientation error is calculated as:

$$OrientationError = \frac{Orient_{act} - Orient_{est}}{Orient_{act}} \cdot 100\%, \qquad (24)$$

where $Orient_{act}$ is actual final orientation and $Orient_{est}$ estimated final orientation. Maximal mobile robot translational speed was set to 300 [mm/s].

CONCLUSION

Two EKF based mobile robot local localization approaches are implemented and experimentally compared in two different experiment set-ups. While the first EKF implementation is a standard one, the second one is its extension based on information inclusion from so-called odometric device. Based on here presented results it can be concluded that both approaches can successfully cope with both environments. Odometric device EKF implementation produces somewhat smoother pose estimation, witch makes it more suitable for navigational tasks. Also pose estimation accuracy is somewhat better. Future work on his topic will include an adaptive framework for parameter σ_η^2, sonar acceptance threshold value, and sonar measurement noise estimation.

ACKNOWLEDGEMENT

This research has been supported by the Ministry of Science and Technology of the Republic of Croatia under grant No. 0036018.

REFERENCES

1. Borenstein J., Everett H. R.,Feng L., Where am I? Sensors and Methods for Mobile Robot Positioning. University of Michigan. 1996.
2. Burgard W., Cremers A. B., Fox D., Hähnel D., Lakemeyer G., Schulz D., Steiner W., Thrun S., Experiences with an interactive museum tour-guide robot, Artificial Intelligence (AI), 1999, 114(1-2), pp. 3-55.
3. Goel P., Roumeliotis S.I., Sukhatme G.S., Robust Localization Using Relative and Absolute Position Estimates, Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 1999.
4. Ivanjko E., Petrović I., Extended Kalman Filter based Mobile Robot Pose Tracking using Occupancy Grid Maps, Proceedings of The 12th IEEE Mediterranean Electro-technical Conference - MELECON 2004, May 12-15, 2004, Dubrovnik, Croatia, pp. 311-314.
5. Ivanjko E., Petrović I., Maček K., Improvements of occupancy grid maps by sonar data corrections, Proceedings of 2003 FIRA Robot Soccer World Congress, Vienna, Austria, 2003.
6. Ivanjko E., Petrović I., Perić N., An approach to odometry calibration of differential drive mobile robots, Proceedings of International Conference on Electrical Drives and Power Electronics EDPE'03, 2003, pp. 519-523.
7. Jetto L., Longhi S., Venturini G., Development and Experimental Validation of an Extended Kalman Filter for the Localization of Mobile Robots, IEEE Transactions on Robotics and Automation, Vol. 15, No. 2, April 1999, pp. 219-229.
8. Konolige K., A Gradient Method for Realtime Robot Control, Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2000), Kagawa University, Takamatsu, Japan, 2000.
9. Lee D., The Map-Building and Exploration Strategies of a Simple Sonar-Equipped Robot, Cambridge University Press, 1996.
10. Welch G., Bishop G., An introduction to the Kalman filter, University of North Carolina at Chapel Hill, Technical Report TR 95-041, 2000.