Mathematical and Computational Applications, Vol. 18, No. 3, pp. 361-372, 2013

PERFORMANCE IMPROVEMENT IN VSLAM USING STABILIZED FEATURE POINTS

Caner Şahin and Mustafa Ünel

Faculty of Engineering and Natural Sciences, Sabanci University, 34956, Orhanlı-Tuzla, Istanbul
canersahin@sabanciuniv.edu, munel@sabanciuniv.edu

Abstract- Simultaneous localization and mapping (SLAM) is the main prerequisite for the autonomy of a mobile robot. In this paper, we present a novel method that enhances the consistency of the map using stabilized corner features. The proposed method integrates template matching based video stabilization and Harris corner detector. Extracting Harris corner features from stabilized video consistently increases the accuracy of the localization. Data coming from a video camera and odometry are fused in an Extended Kalman Filter (EKF) to determine the pose of the robot and build the map of the environment. Simulation results validate the performance improvement obtained by the proposed technique.

Key Words- vSLAM, Video stabilization, Feature extraction, Extended Kalman Filter.

1. INTRODUCTION

SLAM has been one of the key research areas for autonomous mobile robots. It is the process of building the map of an unknown environment and determining the location of the robot using this map concurrently. The inception of the SLAM problem occured at the 1986 IEEE Robotics and Automation Conference as reported in [1]. After this time, probabilistic methods were incorporated into the robotics research where they were primarily being used in the guidance, navigation and control of robots. SLAM has been an active research area after the influential work of Smith *et al*. [2] about stochastic mapping. Ayache and Faugeras introduced the combination of visual work and navigation using stereo pairs [3]. Experimental work of Moutarlier *et al*. [4] attracted the interest of the researchers to the SLAM problem. Lowe et al. used the Scale Invariant Feature Transform (SIFT) algorithm in vSLAM for a mobile robot [5]. Davison et al. developed an EKF based monocular vSLAM in [6].

In this paper, we propose a performance improvement technique that extracts stabilized Harris corner features using template matching based stabilized video sequences. When a non-holonomic wheeled mobile robot (WMR) navigates in an unknown environment, some undesired phenomena such as vibrations on the mobile robot and the speed bump constructions in the environment might occur. With the proposed technique, these problems are eliminated, and as a result stabilized feature extraction is achieved. Stabilized keypoint extraction ensures both consistency in map building and localization of the mobile robot.

The rest of the paper is organized as follows: In section 2, the sensor fusion algorithm is introduced. In section 3, mathematical model of a non-holonomic mobile robot is described. EKF algorithm is summarized in section 4. Stabilized feature point extraction is detailed in section 5. Simulation results are presented in section 6, and finally, the paper is concluded in section 7.

2. SENSOR FUSION ARCHITECTURE

The sensor fusion architecture developed in this work is shown in Figure 1 and composed of several modules. Data generated by both the camera and the odometry are used in feature extraction (FE) and dead reckoning (DR) blocks, respectively. The output of FE is the observation, and the output of DR is the robot state prediction. In measurement prediction block, predicted states obtained from the robot model are used and the sensor measurement model is utilized to predict the measurements. In matching module, measurement predictions are subtracted from observations to calculate the innovation and innovation covariance. The output of the matching block is transferred to EKF update block to estimate the non-holonomic WMR states and build the map.

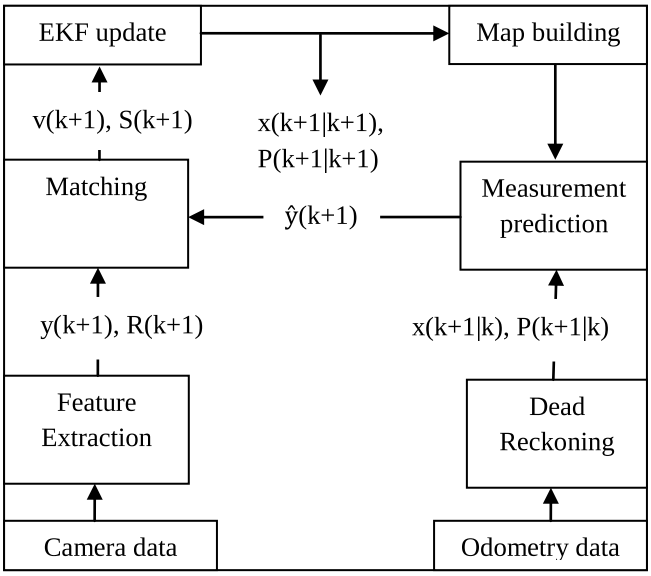


Figure 1. Sensor fusion architecture

3. MATHEMATICAL MODEL OF THE MOBILE ROBOT

The non-holonomic WMR shown in Figure 2 includes two driving wheels and a back caster that are non deforming. The robot moves on the horizontal plane and the contact of the wheels with the ground is assumed to satisfy rolling without any skidding or slipping.

3.1. Kinematic Model

In the kinematic modeling of the non-holonomic WMR, orientation must be considered since it affects the robot movement along x and y directions based on the kinematic constraints of the system.

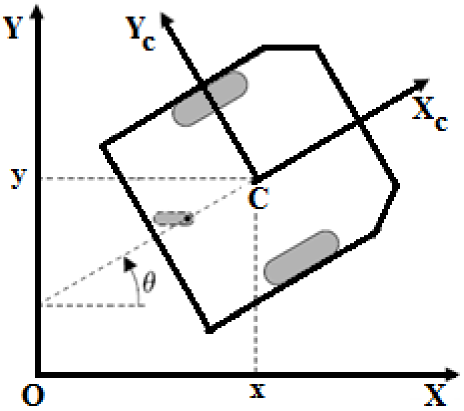


Figure 2. Non-holonomic wheeled mobile robot

The kinematic model of the NWMR is described by the following equations [7]:

$$\begin{aligned} \dot{x} &= v cos\theta \\ \dot{y} &= v sin\theta \\ \dot{\theta} &= w \end{aligned} \tag{1}$$

or, can be written in a more compact form as

$$\dot{\boldsymbol{x}} = f(\boldsymbol{x}, \boldsymbol{u}) \tag{2}$$

where $\boldsymbol{x} = [x, y, \theta]^T$ is the pose (position and orientation) of the centre of mass of the mobile robot C, with respect to world coordinate frame O, $\boldsymbol{u} = [v, w]^T$ is the control input vector, where v is the linear velocity and w is the angular velocity of the mobile robot, respectively. Using Euler's forward difference approximation for the derivative, the discrete form of the mobile robot kinematic model can be written as:

$$\begin{aligned} x_{k+1} &= x_k + Tv cos\theta_k \\ y_{k+1} &= y_k + Tv sin\theta_k \\ \theta_{k+1} &= \theta_k + wT \end{aligned} \tag{3}$$

or, in a more compact form as

$$\boldsymbol{x_{k+1}} = f(\boldsymbol{x_k}, \boldsymbol{u_k}) \tag{4}$$

$$f(\boldsymbol{x_k}, \boldsymbol{u_k}) = \begin{bmatrix} f_x \\ f_y \\ f_\theta \end{bmatrix} = \begin{bmatrix} x_k + Tv cos\theta_k \\ y_k + Tv sin\theta_k \\ \theta_k + wT \end{bmatrix} \tag{5}$$

where k is the discrete time index, T is the sampling period and $f(\boldsymbol{x_k}, \boldsymbol{u_k})$ is a nonlinear mapping [8]. In order to implement EKF, this nonlinear system must be linearized. In [9], it is shown that applying the Taylor series approximation to the right-hand side of Eq. 2 and ignoring the higher order terms yields the following linear state-space model of the mobile robot:

$$\boldsymbol{x}(k+1) = A(k)\boldsymbol{x}(k) + B(k)\boldsymbol{u}(k) \tag{6}$$

The state $A(k)$ and input $B(k)$ matrices are defined as follows:

$$A(k) = \begin{bmatrix} \frac{\partial f_x}{\partial x_k} & \frac{\partial f_x}{\partial y_k} & \frac{\partial f_x}{\partial \theta_k} \\ \frac{\partial f_y}{\partial x_k} & \frac{\partial f_y}{\partial y_k} & \frac{\partial f_y}{\partial \theta_k} \\ \frac{\partial f_\theta}{\partial x_k} & \frac{\partial f_\theta}{\partial y_k} & \frac{\partial f_\theta}{\partial \theta_k} \end{bmatrix} = \begin{bmatrix} 1 & 0 & -Tv(k)sin\theta_k \\ 0 & 1 & Tv(k)cos\theta_k \\ 0 & 0 & T \end{bmatrix} \tag{7}$$

$$B(k) = \begin{bmatrix} \frac{\partial f_x}{\partial u_k} & \frac{\partial f_x}{\partial w_k} \\ \frac{\partial f_y}{\partial u_k} & \frac{\partial f_y}{\partial w_k} \\ \frac{\partial f_\theta}{\partial u_k} & \frac{\partial f_\theta}{\partial w_k} \end{bmatrix} = \begin{bmatrix} Tcos\theta_k & 0 \\ Tsin\theta_k & 0 \\ 0 & T \end{bmatrix} \tag{8}$$

3.2. Camera Sensor Model

Ideal pin hole camera model is used as a measurement model. Acquired measurements from the camera generate the measurement vector $\boldsymbol{y}$,

$$\boldsymbol{y} = [y_{1k,}\, y_{2k,} \dots, y_{pk}]^T \tag{9}$$

where p is the number of the features observed at a particular time index k. At the same time, all the observed image features build up the map of the environment. At any time k, for one observed image feature camera model implies:

$$\begin{bmatrix} m_{ix} \\ m_{iy} \end{bmatrix} = \begin{bmatrix} O_x + f_c \frac{s_{ix}^C}{s_{iz}^C} \\ O_y + f_c \frac{s_{iy}^C}{s_{iz}^C} \end{bmatrix} \; for\ i = 1,2,\dots,p \tag{10}$$

where f_c is the focal length of the camera, $(O_x,\ O_y)$ is the principal point of the image plane in pixels, $s^C = [s_{ix}^C, s_{iy}^C, s_{iz}^C]^T$ is the 3D location of the extracted feature with respect to the camera frame. 3D location of the i^{th} feature with respect to the world coordinate frame is given as [10]:

$$q_i = [X_i, Y_i, Z_i]^T = r + R_C^W s_i^C \tag{11}$$

where q_i is the 3D location of the image feature in world frame, R_C^W is the rotation matrix that defines the orientation of the camera frame with respect to the world frame, r is the 3D translation vector from world frame to camera frame. A rotation matrix can be parameterized by three independent variables such as Euler angles. Due to the planar robot motion assumption, the orientation matrix will be just in terms of the yaw angle [13]:

$$R_C^W = \begin{bmatrix} cos\theta & -sin\theta & 0 \\ sin\theta & cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{12}$$

In Eq. 12, θ (heading angle) is taken from the estimated states of the EKF that will be summarized in the next section. By rearranging Eq. 11, one can calculate the s_i^C as:

$$s_i^C = R_W^C(q_i - r) \tag{13}$$

where R_W^C is simply the transpose of the rotation matrix R_C^W. Plugging Eq. 13 into the measurement model yields the extracted feature location in image plane:

$$\begin{bmatrix} m_{ix} \\ m_{iy} \end{bmatrix} = \begin{bmatrix} O_x + f_c \frac{cos\,\theta(X_i - r_x) + sin\,\theta(Y_i - r_y)}{Z_i - r_z} \\ O_y + f_c \frac{-sin\,\theta(X_i - r_x) + cos\,\theta(Y_i - r_y)}{Z_i - r_z} \end{bmatrix} \tag{14}$$

The measurement Jacobian H_k is calculated by taking the derivative of the right hand side of the Eq. 14 with respect to the states of the mobile robot $\boldsymbol{x_k}$. Thus,

$$H_k = \begin{bmatrix} \frac{\partial m_{ix}}{\partial r_x} & \frac{\partial m_{ix}}{\partial r_y} & \frac{\partial m_{ix}}{\partial \theta} & \frac{\partial m_{ix}}{\partial X_i} & \frac{\partial m_{ix}}{\partial Y_i} \\ \frac{\partial m_{iy}}{\partial r_x} & \frac{\partial m_{iy}}{\partial r_y} & \frac{\partial m_{iy}}{\partial \theta} & \frac{\partial m_{iy}}{\partial X_i} & \frac{\partial m_{iy}}{\partial Y_i} \end{bmatrix}$$

$$H_k = \left(\frac{f_c}{Z_i - r_z}\right) \begin{bmatrix} -cos\theta & -sin\theta & -sin\theta(X_i - r_x) + cos\theta(Y_i - r_y) & cos\theta & sin\theta \\ sin\theta & -cos\theta & -cos\theta(X_i - r_x) - sin\theta(Y_i - r_y) & -sin\theta & cos\theta \end{bmatrix} \tag{15}$$

Observation and measurement prediction data are fused in EKF to calculate the innovation and innovation covariance.

4. EXTENDED KALMAN FILTER (EKF)

The mobile robot navigates in an unknown environment, without any a priori knowledge about the map, takes measurements to extract feature points and consequently localizes itself. External (camera) and internal (odometry) sensory data will be fused in EKF. The robot pose $(\boldsymbol{x})$ and the locations of the extracted feature points $(\boldsymbol{X_F})$ with respect to the world frame can be stacked in a new state vector as:

$$\boldsymbol{X} = \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{X_F} \end{bmatrix}$$

where $\boldsymbol{x} = [x, y, \theta]^T$ defines position and orientation of the robot, and is governed by the following nonlinear model:

$$\boldsymbol{x_{k+1}} = f(\boldsymbol{x_k}, \boldsymbol{u_{k+1}}, w_k)$$

$$\boldsymbol{y_{k+1}} = h(\boldsymbol{X_{k+1}}, v_k) \tag{16}$$

where w_k and v_k are the process and the measurement noise, which are modeled as zero-mean, independent Gaussian distributions with covariance matrices Q_k and R_k, respectively.

The second element of $\boldsymbol{X}$ is defined as

$$\boldsymbol{X_F} = \begin{bmatrix} X_{fi} \\ Y_{fi} \end{bmatrix} \quad \text{for } i = 1,2,\dots,n \tag{17}$$

where $\boldsymbol{X_F} = [X_{fi}, Y_{fi}]^T$ are the locations of the extracted features with respect to the world frame and added to the map at time k. Since the positions of the extracted features are not changed, they remain at the same locations during the navigation; i.e.

$$\boldsymbol{X_{F,k+1}} = \begin{bmatrix} X_{fi} \\ Y_{fi} \end{bmatrix}_{k+1} = \boldsymbol{X_{F,k}} \tag{18}$$

Linearization of Eqs. 16 and 18 with respect to $\boldsymbol{X}$ imply new Jacobians for the process model [7]:

$$\bar{A} = \begin{bmatrix} A & O_1 \\ O_1^T & I \end{bmatrix}, \quad \bar{B} = \begin{bmatrix} B \\ O_2 \end{bmatrix}$$

$$\bar{H} = \begin{bmatrix} \dfrac{\partial m_{ix}}{\partial\left(r_x, r_y, \theta, \{X_{fi}, Y_{fi}\}_{i=1,\dots n}\right)} \\ \dfrac{\partial m_{iy}}{\partial\left(r_x, r_y, \theta, \{X_{fi}, Y_{fi}\}_{i=1,\dots n}\right)} \end{bmatrix} \tag{19}$$

where $A \in R^{3x3}$, $O_1 \in R^{3x2n}$ (zero matrix), $I \in R^{2nx2n}$ (identity matrix), $B \in R^{3x2}$ and $O_2 \in R^{2nx2}$ (zero matrix) with n being the number of features extracted at time k. With this framework, the following algorithm summarizes the recursions involved in computing the EKF [11]:

$$\boldsymbol{X_{k+1|\,k}} = f(\boldsymbol{X_k}, \boldsymbol{u_{k+1}}) \tag{20}$$

$$P_{k+1|\,k} = \bar{A}_{k+1,k} P_k \bar{A}^T_{k+1,k} + \bar{Q}_k \tag{21}$$

$$K_{k+1} = P_{k+1|\,k} \bar{H}^T_{k+1} [\bar{H}_{k+1} P_{k+1|\,k} \bar{H}^T_{k+1} + R_k]^{-1} \tag{22}$$

$$\boldsymbol{X_{k+1}} = \boldsymbol{X_{k+1|\,k}} + K_{k+1}(\boldsymbol{y_{k+1}} - h(\boldsymbol{X_{k+1|\,k}})) \tag{23}$$

$$P_{k+1} = (I - K_{k+1} \bar{H}_{k+1}) P_{k+1|\,k} \tag{24}$$

where $\bar{Q}_k$ is the covariance matrix of the combined state $\boldsymbol{X}$. To initialize the filter, $\boldsymbol{X_0}$ and P_0 are set to some arbitrary random values.

5. STABILIZED FEATURE POINT EXTRACTION

Extracting feature points accurately increases the performance of vSLAM algorithm since they are used in EKF measurement update. It provides improvement in both map building and localization of the mobile robot. In this section, stabilization of video sequences and Harris corner detection algorithm are detailed.

Video stabilization is one of the most crucial video processes that reduces the blurring level of image sequences and unwanted camera motions. Extracting point features from stabilized video frames improves the consistency of the static landmarks and provides robust matching between corresponding points. Proposed video stabilization method in this work is based on a template matching that uses the sum of absolute differences (SAD) algorithm:

$$SAD = \sum_{(i,j)\epsilon W} |I_1(i,j) - I_2(x+i, y+j)| \tag{25}$$

where I_1 and I_2 are two consecutive image frames. $I_1(i,j)$ and $I_2(x+i,y+j)$ defines the pixel intensity values. In I_1, a window W, e.g. size of $(15 \text{ x } 15)$, is generated around an interest point. Meanwhile, each pixel in the second video frame is scanned by shifting this window along horizontal (x) and vertical (y) directions. Note that the intensity values in the second window is subtracted from those values in the first window. The absolute values of all these pixel intensities in W are summed. If there is a correct match, the SAD function gives a near 0 value. Thus, a similar window is created in the second video frame [14]. Scan process can be applied both over the entire image or just using a region of interest. In each subsequent video frame, SAD algorithm determines the camera motion relative to the previous frame. It uses this information to remove unwanted translational camera motions and generate a stabilized video.

Feature extraction from consecutive images is one of the essential steps of vision based simultaneously localization and mapping applications. In this work, extracted image features are corners that are obtained via Harris corner detector. Due to space limitations, details about Harris corners are omitted here, and the interested reader may refer to [12].

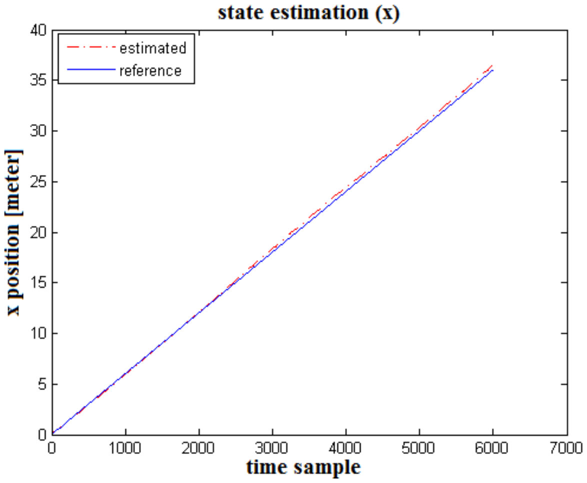
6. SIMULATION RESULTS

In this section, the performance of the proposed technique is verified with simulation results. Ramp and circular inputs are used to generate the odometry data. Odometry and camera outputs are fused in EKF to estimate states of the mobile robot. Extended Kalman filter both estimates the mobile robot states and generates the map of the unknown environment. Inputs for the system are summarized in Table 1.

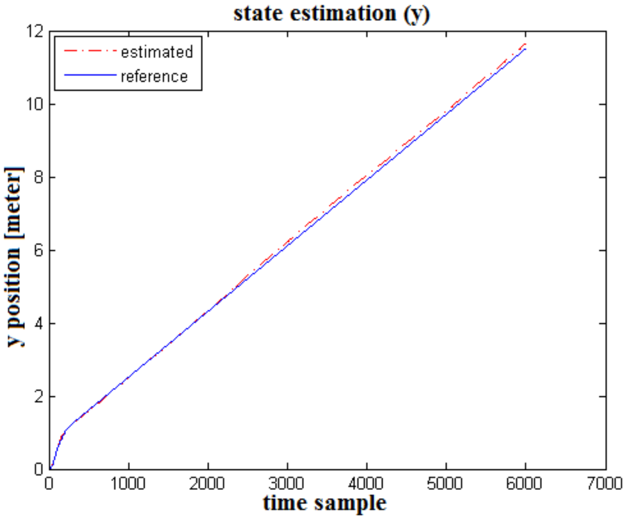
Table 1. System inputs

Type of Input	Input
Ramp trajectory	$v_r = 0.3$ [m/s] $w_r = 0$ [rad/s] $\theta_r = v_r$ [rad] $x_r = v_r t$ [m] $y_r = 0.09t + 0.7$ [m]
Circular trajectory	$v_r = 0.3$ [m/s] $w_r = 0.6$ [rad/s] $\theta_r = w_r t$ [rad] $x_r = x_0 + 5\, sin\, \theta_r$ [m] $y_r = y_0 - 5\, cos\, \theta_r$ [m] $x_0 = 2\ (center\ coordinate\ of\ the\ circle)$ [m] $y_0 = 2\ (center\ coordinate\ of\ the\ circle)$ [m]

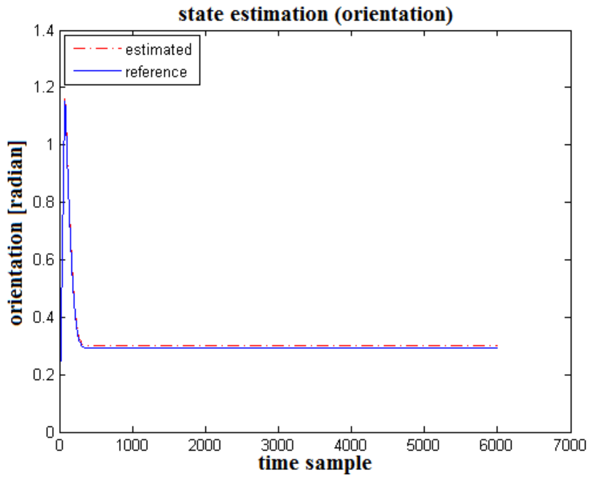
In this table x_r, y_r, θ_r indicate the reference pose of the mobile robot and v_r, w_r denote reference linear and angular velocities of the mobile robot, respectively. Simulation results for the ramp trajectory is depicted for 120 seconds, and 1/50 is chosen for sampling time both for EKF and the camera. In Figure 3 (a), (b) and (c) robot pose estimation is shown. According to the leftmost and center graphs, x and y positions of the mobile robot increase as time increases. Given the control input that is shown in Table 1 for ramp input, x position coordinate of the mobile robot increases more rapidly than the y coordinate position. Initial robot pose as well as the initial camera frame are used as the reference coordinate system and all estimates are represented with respect to this frame. On the rightmost graph in Figure 3, θ, heading angle estimation is shown. When mobile robot starts to navigate in the environment, it has a rotation at the beginning of the movement for trajectory tracking that is related to the ramp control input. As shown in Figure 3, the errors between reference and estimated pose states are less than 1%.



(a)



(b)

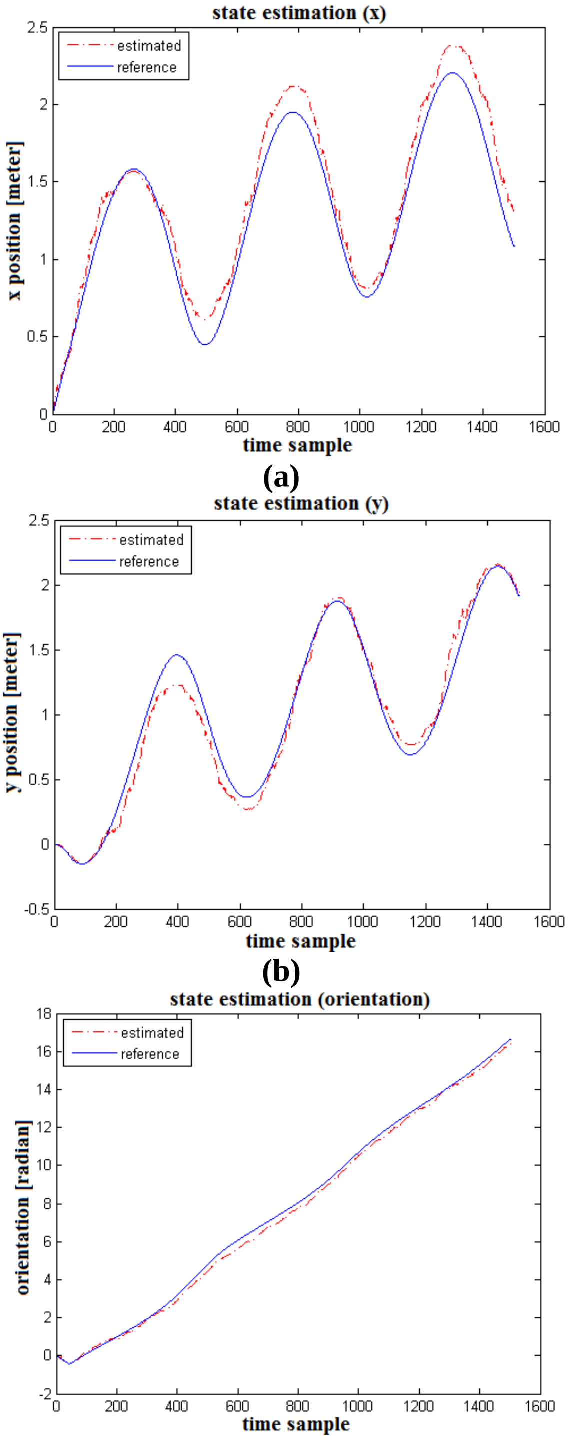


(c)

Figure 3. x, y and θ state (pose) estimations by EKF for ramp input

In Figure 4 (a), (b) and (c) pose estimation of the NWMR is shown for the circular trajectory. The simulation for circular trajectory is performed for 30 seconds and 1/50 sampling time is chosen again for both EKF and the camera as in the ramp input. In the first and second graphs of the figure, x and y position estimates are depicted. Given constant linear and angular velocity inputs, 0.3 [m/s] and 0.6 [rad/s] respectively, mobile robot navigates in circular trajectory in the environment. In the leftmost graph, at 800 and 1300 time samples, there occurs some differences between reference and estimated states. The reason why these differences occur is the rapid increase in heading angle and hence decrease in the overlap area in consecutive image frames. Reduction in the overlapped area between consecutive frames gives rise decrease in stable feature point extraction and consequently higher noise in map building.

In our vSLAM algorithm the accuracy of the mobile robot localization is highly dependent on the map building. Errors in these regions are approximately 8%. However, between 800 and 1300 time samples, the reference and the estimated states are very close to each other, i.e. the error rate is below 1%. This promising result is obtained thanks to the stabilized extracted feature points and validates the performance of our proposed algorithm. On the rightmost graph in Figure 4, it is seen that heading angle increases continuously with time.



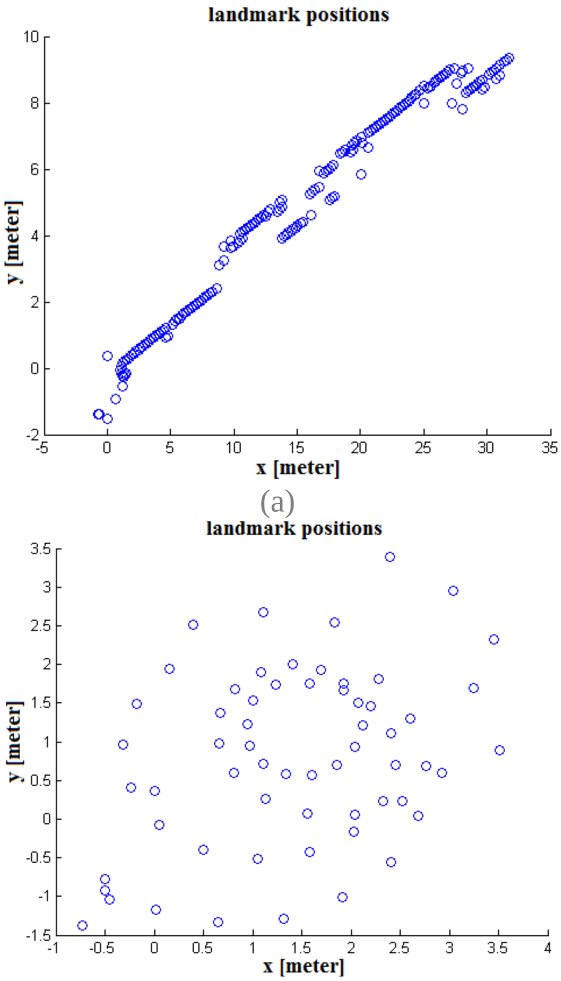
(a)

(b)

(c)

Figure 4. x, y and θ state (pose) estimations by EKF for circular input

The most prominent result of the proposed technique is the accuracy improvement of visual simultaneous localization and map building algorithm using stabilized feature point extraction. Subsequent video frames are stabilized and Harris corner features are extracted from stabilized video sequences. In Figure 5 (a) and (b), landmark positions for ramp and circular inputs are shown. While mobile robot is travelling in the unknown environment with given control inputs, naturally located planar landmarks are extracted and used for measurement update in EKF. In vSLAM algorithms, generating consistent map is one of the most crucial processes to obtain accurate navigation results. Acquiring these naturally located features in a consistent way by neglecting unwanted camera motion and jitter, our technique builds a consistent map and improves the localization correctness as shown in Figures 3 and 4.



(a)

(b)

Figure 5. Landmark positions: (a) ramp trajectory, (b) circular trajectory

7. CONCLUSION

In this paper, we proposed a performance improvement technique for vSLAM problem of mobile robots. We incorporated video stabilization into vSLAM for feature extraction, correspondingly map building and localization. In vSLAM, the performance of the algorithm depends on both the accuracy of the map and localization of the robot. In this work, it is shown that consistent feature extraction technique both improves the accuracy of map building and localization of the mobile robot by neglecting unwanted sensor motion and the noises that are caused by the external factors. Simulation results verified that EKF state estimation performance is improved thanks to the utilization of stabilized landmarks in measurement update.

As a future work, we plan to suggest a new approach to vSLAM problem that will use vision only and eliminate some of the assumptions made in this work such as 2D visual features and planar motion. Instead of using odometry data in EKF, we will estimate robot ego-motion utilizing extracted 3D visual features.

8. REFERENCES

1. H. D. Whyte and T. Bailey, Simultaneous Localization and Mapping: Part I, *IEEE Robotics & Automation Magazine* **13(2)**, 99-100, 2006.

2. R. Smith, M. Self, and P. Cheeseman, A Stochastic Map for Uncertain Spatial Relationships, *Proceedings of the 4th International Symposium on Robotics Research*, 467-474, 1988.

3. N. Ayache and O. Faugeras, Building, Registrating, and Fusing Noisy Visual Maps, *Int. Jornal of Robotics Research* **7(6)**, 45–65, 1988.

4. P. Moutarlier and R. Chatila, Stochastic Multisensory Data Fusion for Mobile Robot Location and Environment Modeling, *Proceedings of the 5th International Symposium on Robotics Research*, 207-216, Tokyo, Japan, 1989.

5. S. Se, D. Lowe and J. Little, Mobile Robot Localization and Mapping with Uncertainity using Scale-Invariant Visual Landmarks, *The International Journal of Robotic Research* **21(8)**, 735-758, 2002.

6. A. J. Davison, I. D. Reid, N. D. Molton and O. Stasse, MonoSLAM: Real-Time Single Camera SLAM, *IEEE Trans. on Pattern Analysis and Machine Intelligence* **29(6)**, 1052-1067, 2007.

7. J. Guivant and E. Nebot, Optimization of the Simultaneous Localization and Map Building Algorithm for Real Time Implementation, *IEEE Trans. on Robotics and Automation* **17(3)**, 242-257, 2001.

8. F. Kühne, J. M. G. da Silva Jr. and W. F. Lages, Model Predictive Control of a Mobile Robot using Input-Output Linearization, *IEEE Proceedings of Mechatronics and Robotics*, ISBN 0-7803-9044-X, 2005.

9. W. F. Lages and J. A. V. Alves, Real-Time Control of a Mobile Robot using Linearized Model Predictive Control, *Proc. of 4th IFAC Symposium on Mechatronics Systems* **4(1)**, 968-973, 2006.

10. Y. T. Wang, Y. C. Feng and D. Y. Hung, Detection and Tracking of Moving Objects in SLAM using Vision Sensors, *IEEE Trans. Instrumentation and Measurement Technology Conference*, Binjiang, China, 1-5, 2011.

11. S. Haykin, *Kalman Filtering and Neural Networks*, John Wiley and Sons Inc., Hamilton, Canada, 2001.

12. Y. Ma, S. Soatto, J. Kosecka and S. S. Sastry, *An Invitation to 3-D Vision*, Springer, 2003.

13. J. J. Craig, *Introduction to Robotics : Mechanics and Control*, Addison-Wesley, 1989.

14. R. A. Hamzah, R. A. Rahim and Z. M. Noh, Sum of Absolute Differences Algorithm in Stereo Correspondence Problem for Stereo Matching in Computer Vision Application, *IEEE Comp. Science and Information Technology* **1**, 652-657, 2010.