

Localization of Leader-Follower Robot Using Extended Kalman Filter

Siti Nurmaini¹, Sahat Pangidoan²

Intelligent System Research Group, Universitas Sriwijaya, Indonesia
¹sitinurmaini@gmail.com, ²sahatsilalahi141@gmail.com

ABSTRACT

Non-holonomic leader-follower robot must be capable to find its own position in order to be able to navigate autonomously in the environment this problem is known as localization. A common way to estimate the robot pose by using odometer. However, odometry measurement may cause inaccurate result due to the wheel slippage or other small noise sources. In this research, the Extended Kalman Filter (EKF) is proposed to minimize the error or the inaccuracy caused by the odometry measurement. The EKF algorithm works by fusing odometry and landmark information to produce a better estimation. A better estimation acknowledged whenever the estimated position lies close to the actual path, which represents a system without noise. Another experiment is conducted to observe the influence of numbers of landmark to the estimated position. The results show that the EKF technique is effective to estimate the leader pose and orientation pose with small error and the follower has the ability traverse close to leader based-on the actual path.

Keywords: Estimator, Extended Kalman Filter, Localization, Odometry, Leader-Follower Robot

1. INTRODUCTION

In order to navigate safely and reliably, an autonomous mobile robot must be capable of finding out its location relative to the environment independently [1][2][3]. Localization is one important component in robot navigation system in terms of position and orientation (x, y, θ) . The goal of the localization is to keep track of the position while the robot is navigating through the environment [3][4]. According to the methods in determining the location, a robot has access to two kinds of information, relative or absolute [5][6]. By relative, also known as dead-reckoning, the robot collecting and integrating its information from different sensors where the integration is started from the initial position and continuously updated through times. Absolute method is different from the other one, because the robot does not need to derive some integrated sequence of measurement to gain information, but the robot itself does a direct measurement to supply information [3][7].

However, the fact that mobile robots have some unbounded growth of time integration errors along their duties to traverse their environment is undeniable [8][9]. The wheels attached to mobile robot are susceptible to slippage. It is known that slippage can disturb the sensor reading. Mathematical modeling, including sensor modeling may cause some inaccuracies in a system. It is obvious those errors

will continue to grow if they are left untreated [1][7][8]. A mobile robot with untreated accumulating errors will behave unpredictable from what it is expected to behave. Hence, the mobile robot control must be overcome the limitation, in order to reach the target with small error.

In this paper, the Extended Kalman Filter (EKF) is proposed to reduce the accumulation error in the actuator. The information gathered by relative and absolute manners will be fused to suppress the error, resulting in leader-follower mobile robot could localize itself [9]. EKF is selected due to it produces an optimal algorithm based-on recursive filter. It is an estimator widely used in a nonlinear system. EKF will be used to fuse the measurement information from odometer and landmark in order mobile robot could localize itself accurately [9] [10]. The result of this research will be presented as a performance graph consists of x , y , θ coordinates and the error graph. The rest of the paper is organized as follows. Section II will cover method used in this research. Section III will provide the experimental results and analysis, and finally, the conclusion will be presented in section IV.

2. LEADER-FOLLOWER LOCALIZATION

The EKF has a cycle that consists of prediction phase and correction phase [11] [12]. As a cycle, prediction phase is where the mobile robot predicts its next location with a prior knowledge, whereas the correction phase will correct the corrupted predicted phase. However, to enable the mobile robot to move from one pose to another pose, it is necessary to define a motion model before.

2.1 LEADER-FOLLOWER KINEMATIC SYSTEM

The kinematic system of the leader follower robot is generated with the parameters that will be measured are the relative distance between the leader and the follower robot [2] [10]. The modelling of leader-follower system has been derived directly by the kinematic analysis of relative robot follower along the x and y coordinates associated with the robot leader. The leader L has configuration vector $[x_L, y_L, \theta_L]^T$ while the follower F has a vector $[x_F, y_F, \theta_F]^T$. The control inputs of the leader and the follower are the linear and angular velocities $[v_L, \omega_L]^T$ and $[v_F, \omega_F]^T$, respectively [10].

The relative distance between leader and follower must be determine, thus they can be move in the same trajectory. To illustrate the relative position between the robots in Cartesian coordinates Figure 1(b) is utilized, it must to projected the relative distance along the x and y directions. In x - y Cartesian coordinates, the distance between the robot leader and the follower robot is l . By using the properties of trigonometric functions i.e., $a.b=|a| \cdot |b| \cos\theta$, the rotation matrix Equation for robot follower is obtained shown in Equation (1) as follow [11],

$$\begin{bmatrix} x_F \\ y_F \\ \theta_F \end{bmatrix} = \begin{bmatrix} \cos\theta_F & \sin\theta_F & 0 \\ -\sin\theta_F & \cos\theta_F & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (1)$$

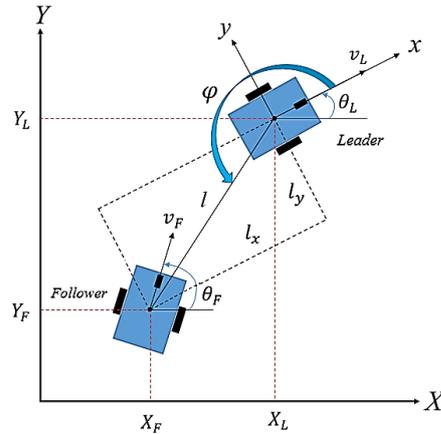


FIGURE 1. Leader-follower kinematic system

Based on Figure 1, and assuming the relative distance Equation can be derived using the matrix rotation in Equation (3) the relative robot leader's distance to the follower robot is defined in Equation (5). Where the relative position the follower robot along the x direction is l_x and along the y direction is l_y with relative orientation θ .

$$\begin{bmatrix} l_x \\ l_y \\ e_\theta \end{bmatrix} = \begin{bmatrix} -\cos\theta_L & -\sin\theta_L & 0 \\ \sin\theta_L & -\cos\theta_L & 0 \\ 0 & 0 & -1 \end{bmatrix} \begin{bmatrix} X_L - X_F \\ Y_L - Y_F \\ \theta_L - \theta_F \end{bmatrix} \quad (2)$$

If the position of the leader robot (X_L, Y_L) is determined and (l_x, l_y) are known and fixed to achieve and maintain the desired formation, parameter (l_x, l_y) must be controlled, then the position with respect to the robot leader can be determined. By controlling $l_x \rightarrow l_x^d$ where l_x^d is the desired relative position along the x direction and $l_y \rightarrow l_y^d$, where l_y^d is the desired relative position along the y direction. In the normal conditions, the relative distance between the leader's robot and the follower robot is l^d , it needs to be simultaneously projected and to control the movement of the follower robot against the leader robot by using Equations (3) to (6) as follow [11],

$$l_x^d = l^d \cos\varphi^d \text{ or,} \quad (3)$$

$$\dot{l}_x^d = \dot{l}^d \cos\varphi^d - l^d \dot{\varphi}^d \sin\varphi^d \quad (4)$$

$$l_y^d = l^d \sin\varphi^d \text{ or,} \quad (5)$$

$$\dot{l}_y^d = \dot{l}^d \sin\varphi^d - l^d \dot{\varphi}^d \cos\varphi^d \quad (6)$$

The desired relative distance l^d between the robot leader and the follower robot is required to be constant or $l^d = l_0$, whereas the relative angle φ^d is varied with time. Therefore, the Equation (4) and (6) becomes Equation (7) and (8),

$$\dot{l}_x^d = -l_0 \dot{\varphi}^d \sin \varphi^d \quad (7)$$

$$\dot{l}_y^d = -l_0 \dot{\varphi}^d \cos \varphi^d \quad (8)$$

From Equation (2) the model of \dot{l}_x as follow,

$$\begin{aligned} \dot{l}_x &= -(\dot{X}_L - \dot{X}_F) \cos \theta_L + (X_L - X_F) \dot{\theta}_L \sin \theta_L - (\dot{Y}_L - \dot{Y}_F) \sin \theta_L + (Y_L - Y_F) \dot{\theta}_L \cos \theta_L \\ &= l_y \dot{\theta}_L + \dot{X}_F \cos \theta_L + \dot{Y}_F \sin \theta_L - \dot{X}_L \cos \theta_L - \dot{Y}_L \sin \theta_L \\ &= l_y \dot{\theta}_L + \dot{X}_F \cos \theta_L + \dot{Y}_F \sin \theta_L - v_L \end{aligned} \quad (9)$$

where v_L represents the linear velocity of the leader robot. The new state variable is defined to represent the orientation angle difference between the robot leader and the follower robot as,

$$e_\theta = \theta_F - \theta_L \quad \text{or} \quad \theta_L = \theta_F - e_\theta$$

$$\dot{e}_\theta = \dot{\theta}_F - \dot{\theta}_L = \omega_F - \omega_L \quad (10)$$

If Equation (10) is substituted to Equation (9) become Equation (11) as follow,

$$\begin{aligned} \dot{l}_x &= l_y \dot{\theta}_L + \dot{X}_F \cos(\theta_F - e_\theta) + \dot{Y}_F \sin(\theta_F - e_\theta) - v_L \\ &= l_y \dot{\theta}_L - v_L + \cos e_\theta (\dot{X}_F \cos \theta_F + \dot{Y}_F \sin \theta_F) - \sin e_\theta (\dot{Y}_F \cos \theta_F - \dot{X}_F \sin \theta_F) \end{aligned} \quad (11)$$

Due to the holonomic constraint of mobile robot in Equation (12), it transforms Equation (11) become Equation (13),

$$\dot{Y}_F \cos \theta_F - \dot{X}_F \sin \theta_F = 0 \quad (12)$$

$$\dot{l}_x = l_y \omega_L + v_F \cos e_\theta - v_L \quad (13)$$

where $\omega_L = \dot{\theta}_L$ represents the angular velocity of the leader's robot, and v_F represents the linear velocity of follower's robot. In the same way from Equation (8) the model of \dot{l}_y will be obtained as follow,

$$\dot{l}_y = l_x \omega_L + v_F \sin e_\theta \quad (14)$$

From the overall Equation of the leader-follower kinematic model can be summarize in Equation (15) as follow,

$$\begin{bmatrix} \dot{l}_x \\ \dot{l}_y \\ \dot{e}_\theta \end{bmatrix} = \begin{bmatrix} \cos e_\theta & 0 & -1 & l_y \\ \sin e_\theta & 0 & 0 & l_x \\ 0 & 1 & 0 & -1 \end{bmatrix} \quad (15)$$

$$u = [v_F, \omega_F, v_L, \omega_L]^T$$

where ω_F is the angular velocity of the follower robot, v_F is the linear velocity of the follower robot, ω_L is angular velocity the leader robot and v_L is the linear velocity the leader robot. By using the leader-follower approach, ω_L and v_L is a time function that varies from input control ω and v .

2.2. MOTION MODEL

Motion model of mobile robot is used to find the current position of the mobile robot. There are two widely used motion model which is velocity and odometry. Odometry model as the motion model in the prediction phase of the EKF algorithm is utilized in this paper. The motion model of a mobile consists of three actions as shown in Figure 1: first rotation, then translation, and the second rotation [12] [13] [14].

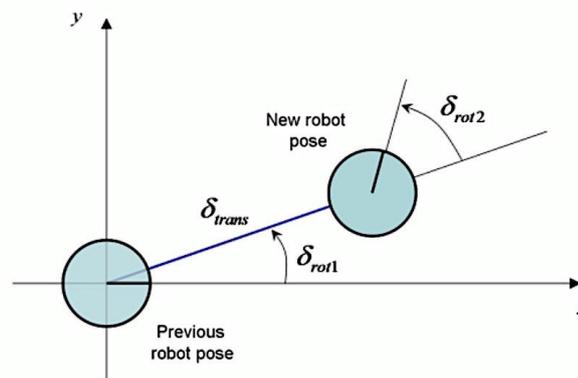


FIGURE 2. Basic action of mobile robot motion

The motion model uses the *relative* information of the internal odometer. In the time interval (t_{t-1}, t) , mobile robot will move from initial pose x_{t-1} to target pose x_t and the report back the related odometry measurement information as $\hat{x}_{t-1} = (\hat{x} \hat{y} \hat{\theta})^T$ to $\hat{x}'_t = (\hat{x}' \hat{y}' \hat{\theta}')^T$. The related information will be used to generate odometer motion model as follows [14] [15]:

$$\begin{pmatrix} x' \\ y' \\ \theta' \end{pmatrix} = \begin{pmatrix} x \\ y \\ \theta \end{pmatrix} + \begin{pmatrix} \delta_{trans} \cos(\theta + \delta_{rot1}) \\ \delta_{trans} \sin(\theta + \delta_{rot1}) \\ \theta + \delta_{rot} \end{pmatrix} \quad (16)$$

where, translation δ_{trans} and rotation δ_{rot} is define as following:

$$\delta_{trans} = \frac{D_r + D_l}{2} \text{ and } \delta_{rot} = \frac{D_r - D_l}{2b}$$

2.3. LANDMARK

Landmark is a feature of the environment that can be detected by the mobile robot sensor. Landmark could be an artificial where a landmark is purposely set in order to give the benefit to the mobile robot. While a natural landmark is part of the environment itself that cannot be manipulated like doors or windows. Based on the characteristic, landmark can be divided into active and passive. An active landmark actively sent the location information directly to the mobile robot, while the passive landmark cannot send the location information. Thus, the mobile robot has to actively look for these landmarks is acquired the position measurements [6]. The landmark location can be defined as follows;

$$l_k = \begin{bmatrix} l_{[x],k} \\ l_{[y],k} \\ l_{[\theta],k} \end{bmatrix} \text{ and } z_k = \begin{bmatrix} z_{[x],k} \\ z_{[y],k} \\ z_{[\theta],k} \end{bmatrix} \quad (17)$$

where $l_{[x],k}$ and $l_{[y],k}$ are the coordinates and $l_{[\theta],k}$ is the orientation of the landmark in the global coordinate system. The measurement z_k is assumed to be the location of a landmark from the viewpoint of the robot.

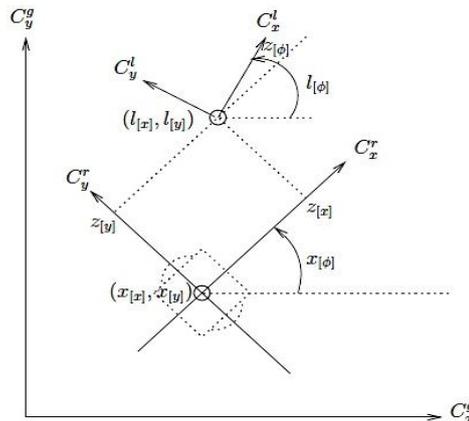


FIGURE 3. Relation between the robot at location x , a landmark l , and a measurement z

Given the location of the robot x_k and the location of the landmark l_k in global coordinates (see. Figure 3). The measurement model is defined $h(\cdot, \cdot)$. That is the measurement function that relates the robot's location with the landmark location.

$$z_k = h(x_k, l_k) = \begin{bmatrix} h_x(x_k, l_k) \\ h_y(x_k, l_k) \\ h_\theta(x_k, l_k) \end{bmatrix} \quad (18)$$

2.4. EXTENDED KALMAN FILTER

The EKF is an estimator algorithm used when a system is governed by a nonlinear function. The EKF processes all available measurements to estimate the state and use knowledge of the system and sensor dynamics, the system and measurement noises, and any available data about the initial values of the state [15]. In a non-linear system, the vector $x \in R^n$ is changed to the form used as follows $x_k = f(x_{k-1}) + w_{k-1}$, a measurement vector $z \in R^n$ $z_k = h(x_k) + v_k$ and a perturbation component Δx_{k-1} , $x_k = x_{k-1}^{nom} + \Delta x_{k-1}$. The EKF estimation algorithm for estimating the pose and orientation robot is expressed in Figure 4. The process is divided into 2, prediction and correction phase. Such flowchart is utilized for leader robot and the follower only follow the leader path.

The prediction Equation is defined as follows,

$$\begin{aligned} P_k^- &= E[e_k^-(e_k^-)^T] \\ &= E[(A_k \Delta x_{k-1} + w_{k-1})(A_k \Delta x_{k-1} + w_{k-1})^T] \\ &= A_k E[\Delta x_{k-1} (\Delta x_{k-1})^T] A_k^T + E[w_{k-1} w_{k-1}^T] \\ P_k^- &= A_k P_{k-1} A_k^T + Q_{k-1} \end{aligned} \quad (19)$$

and the correction Equation is defined as follows:

$$\begin{aligned} P_k^+ &= P_k^- - K_k^T H_k^T P_k^- - K_k H_k P_k^- + K_k D K_k^T \\ P_k^+ &= P_k^- - H_k^T P_k^- D^{-1} H_k^T P_k^- - H_k^T P_k^- D^{-1} H_k P_k^- + H_k^T P_k^- D^{-1} D H_k^T P_k^- D^{-1} \\ P_k^+ &= P_k^- - H_k^T P_k^- D^{-1} H_k P_k^- \\ P_k^+ &= P_k^- - H_k^T P_k^- (H_k P_k^- H_k^T + R_k)^{-1} H_k P_k^- \\ P_k^+ &= (1 - K_k H_k) P_k^- \end{aligned} \quad (20)$$

The Kalman gain is defined,

$$K_k = P_k^- H_k^T (H_k P_k^- H_k^T + R_k)^{-1} \quad (21)$$

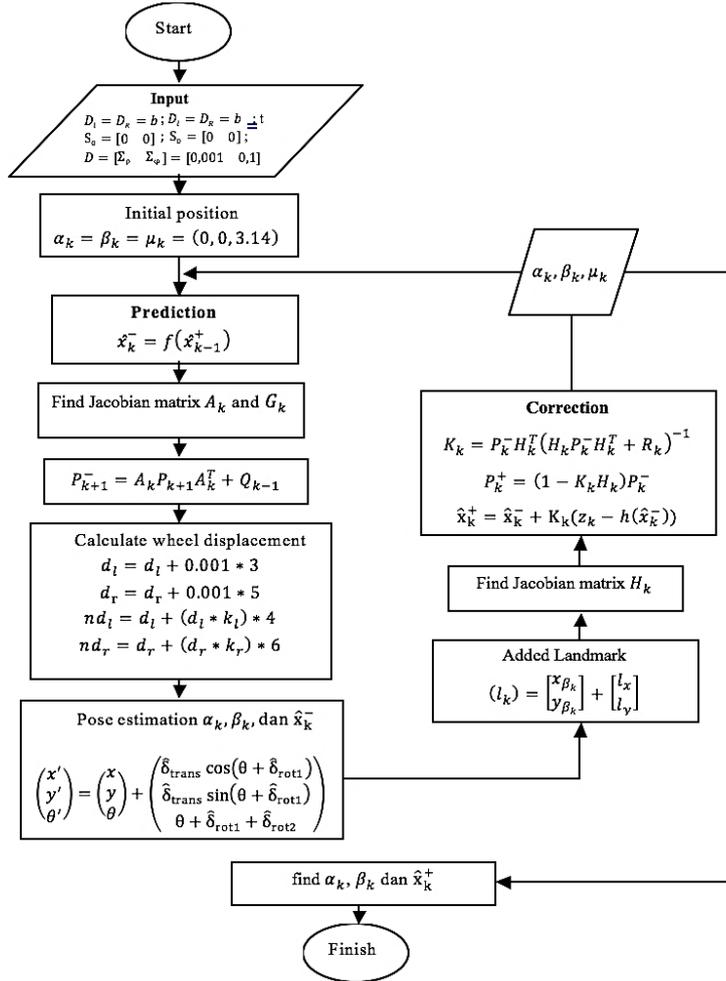


FIGURE 4. Robot localization with EKF process

The input parameters are $\mu_{k-1} = [x_{k-1} \ y_{k-1} \ \theta_{k-1}]$, $[K_l, K_r]^T$, P_{k-1}^+ and $Z = [\rho \ \phi]$, where μ_{k-1} and P_{k-1}^+ is respectively the estimated position and the position covarian matrix at time $t - 1$. $[K_l, K_r]^T$ is encoder error, and Z is the actual relative measurement. $u_k = [D_l \ D_r]^T$ is the control input for left-wheel and right-wheel. The prediction phase consists of prior estimation $\hat{\mu}_k$ with covarian matrix P_k^- [9]. The EKF localization uses the motion model from Equation (19) with the control input from Equation (20) and (21). A_k and G_k respectively are the jacobian of the estimated pose μ_k at previous pose μ_{k-1} and control input u_k . Q_k is Covariance matrix of the noise in control. Correction phase consist of posterior estimation μ_k with covarian matrix P_k^+ , and the kalman gain K . The estimated relative measurement model is \hat{Z} , l_x and l_y respectively is the landmark pose relative to the mobile robot. The jacobian matrix of the predicted measurement model with respect to the robot location is H_r , $\hat{\mu}_k = [\hat{x}_k \ \hat{y}_k \ \hat{\theta}_k]$. The Covariance matrix of the measurement of noise is R_t .

3. RESULTS AND ANALYSIS

The EKF algorithm will basically prevent the estimated position from drifting. In order to see the effectiveness of the EKF, a simulation made to show how the mobile robot move if it depends only to odometer. Meanwhile another simulation was made with two kinds of motion models, which is the actual and the estimated model. The actual motion model is the motion reference which is represents the real movement of the mobile robot in a noisy environment, where the estimated motion model is the noise-filtered pose of the mobile robot motion. Finally, the mobile robot is programmed to move in a random path in effort to always move close to the actual path.

The experiments are conducted to show how the EKF algorithm fuses the odometer information and landmark information to improve the information about location with minimized error. Figure 5 (a) and 5(c) shows the pose and orientation estimation of the leader follower robot with a fixed value control input motion with the initial leader pose is set to $[x, y, \theta]^T = [0, 0, 3.14]$. The leader robot only used pure odometry estimation produce accumulating errors. The leader pose will drift from the actual path as shown in Figure 5 (a). As shown in Figure 5 (b), the green line is represented the actual path, and the red line is represented the leader path with EKF. The EKF will keep the pose of the mobile robot close to the actual path, The Orientation in the graphs shows the evolution of the mobile robot angles between $-\pi$ and π .

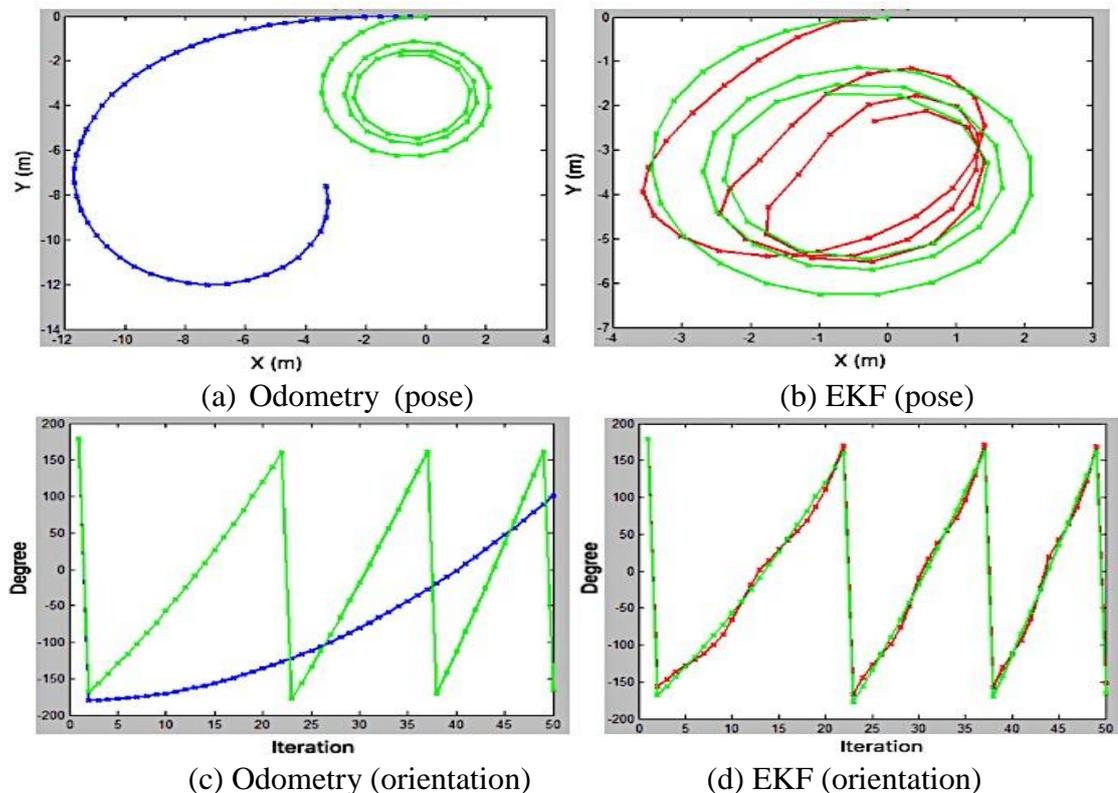


FIGURE 5. Odometry and EKF estimation (blue-line odometry estimation and green-line actual path, red-line EKF estimation)

In Figure 5 (c) the leader orientation far from actual path if only use the odometry estimation, but when it uses EKF to estimation the pose, the leader can keep the distance. It produces good estimation to the actual pose.

The performance of both odometry estimation and EKF estimation can be shown in Figure 6 based on the distance error. The error of the odometer (EAO) is a difference of actual estimation and odometer, whereas the EKF distance error (EAE) is the difference of the actual estimation and EKF estimation as shown in Equation (22) and (23). From the EAO which is represented by blue line shows the odometry estimation errors that greatly increasing demonstrating a drifting position. However, the EKF estimation errors which is represented by red line shows small changing compared to the EAO, which indicating a success implementation of the EKF algorithm.

$$EAO = \sqrt{(x_{\beta_k} - x_{\alpha_k})^2 + (y_{\beta_k} - y_{\alpha_k})^2} \quad (22)$$

$$EAE = \sqrt{(x_{\beta_k} - \hat{x}_k^+)^2 + (y_{\beta_k} - \hat{y}_k^+)^2} \quad (23)$$

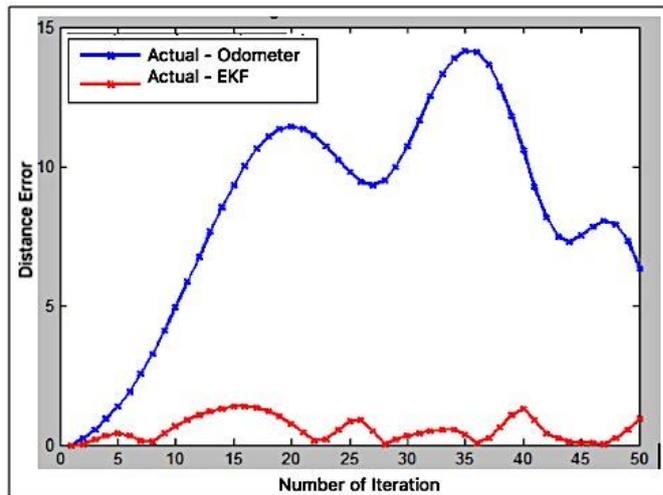
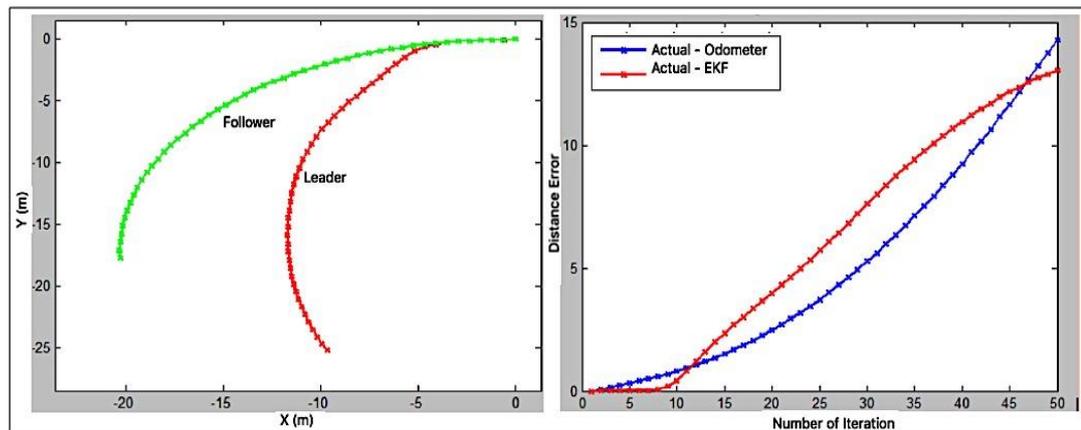


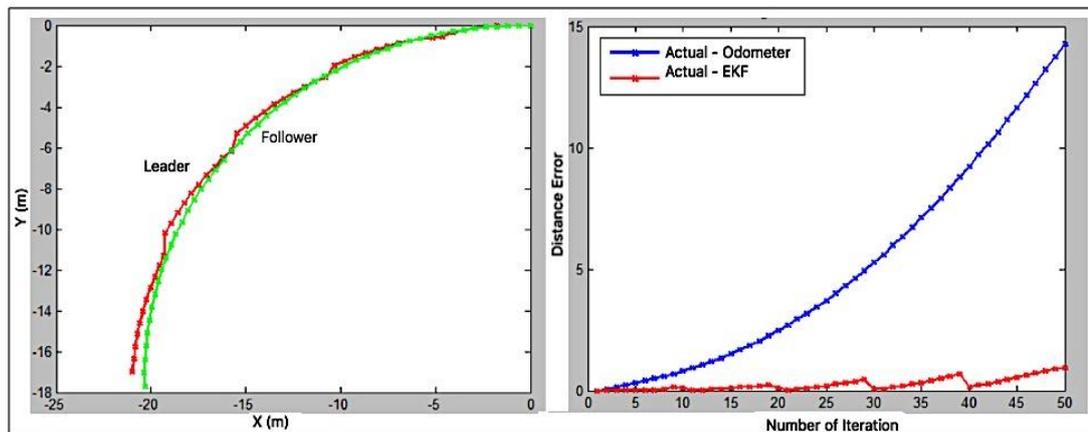
FIGURE 6. Distance error

In this paper the landmarks are utilized to improve the motion of mobile robot to achieve the target with minimized error. The landmark is an important feature for a mobile robot localization to recognize the environment. Moreover, the landmark itself can affect the correction capabilities of the mobile robot. Figure 8 shows the robot localization estimation without landmark and Figure 7 represents the EKF estimation with landmark activation every 10 iteration. However, the EKF algorithm will incorporate the actual-noisy path with the landmark to provide a system with corrected poses. It can be seen that the EKF estimation correct the pose every time it meets the landmark. The distance error of the mobile robot is increase, due to the EKF estimation does not have any information to fuse with estimator to improve the location itself.



(a) Trajectory (b) Distance error of leader robot

FIGURE 7. Mobile robot trajectory with landmark



(a) Trajectory (b) Distance error of leader

FIGURE 8. Mobile robot trajectory without landmarks

The distance error to ensure the location estimation using landmark is successful. The following experiment results in Figure 9 shows a different trajectories and distances error according to the different numbers of landmark with $N = 2, 5, 7, 10$. Assuming the mobile robot moves with a random control input, which is also represents the movement with increasing uncertainties, the performances of EKF estimation and Odometer estimation after 10 times repetition can be shown in Table 1. EOAO and EPAO respectively are position error of EKF estimation and pure odometry estimation, while EOAE and EOAO represent Orientation error of EKF estimation and pure odometry estimation. According to the experimental results, a system with more landmark shows small error on the average. The estimation with 10 landmarks detected about 2.1 % error compared with just 2 landmark which yield 42.88 % error in pose estimation. Error orientation by using 2 landmarks with odometry about 25.52 %, but if EKF is used the error decrease to 12.06 %.

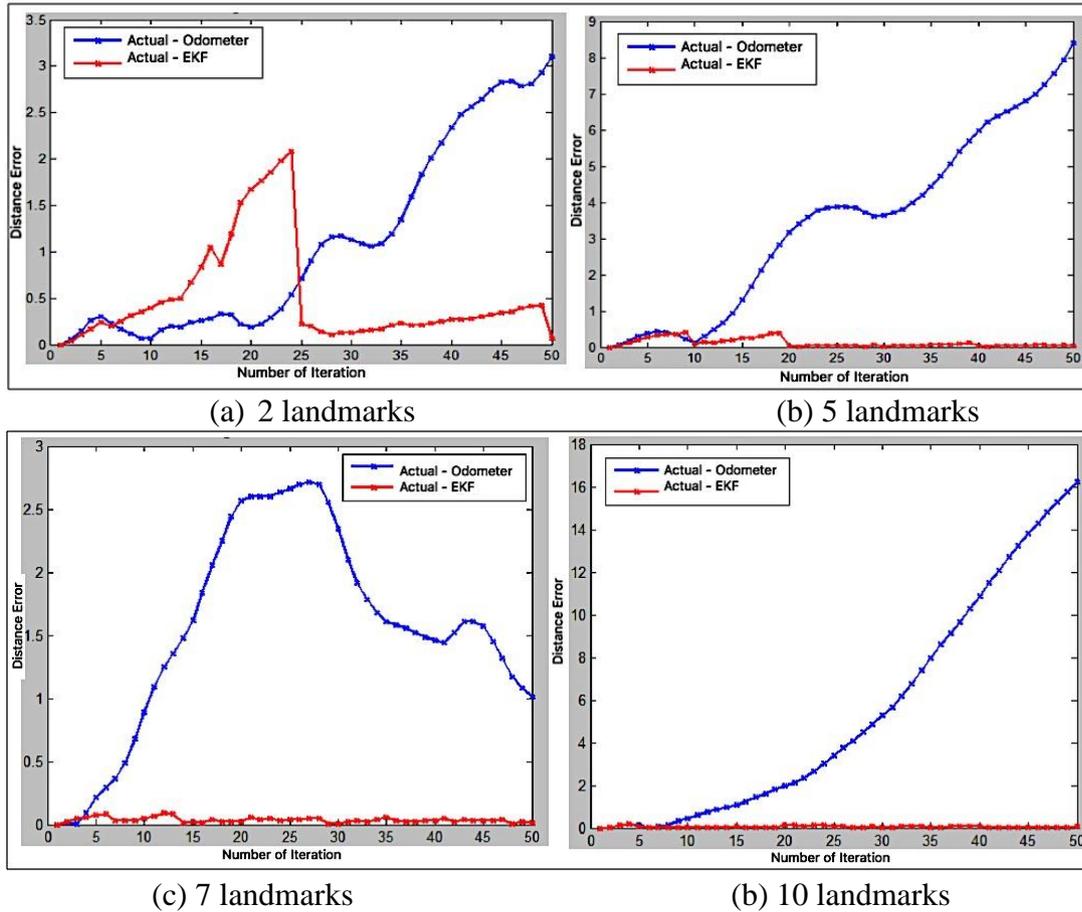


FIGURE 9. Distance error of a leader robot with 2, 5, 7, and 10 landmarks

TABLE 1
 Error in pose and orientation estimation

Numbers of Landmark (N)	EPAE	EPAO	EOAE	EOAO
2	0.4288	3.9401	0.2552	2.1700
5	0.1034	6.1096	0.2502	2.8136
7	0.0629	5.0902	0.2244	3.4198
10	0.0294	3.7726	0.1206	2.287

4. CONCLUSION

This study shows the effectiveness of implementation of EKF algorithm to improve the localization of the leader-follower robot using the information acquired from odometry measurements and landmarks. As shown in the experiment result before, an EKF algorithm implemented in a motion model of a leader-follower robot will make the mobile robot traverse close to actual path. However, there is a problem which cannot solve by the robot itself. Due to the localization, cannot accomplished if the landmark is not detected. But if the landmark is added to the environment the error is reduced until 40% for pose estimation and 12 % for orientation estimation. In the future, the EKF technique will be combine with other

localization approach to enhance the performance and the validation of hybrid algorithm is conducted with or without landmark information.

REFERENCES

- [1]. S. Nurmaini, and T. Bambang. "Intelligent Robotics Navigation System: Problems, Methods, and Algorithm." *International Journal of Electrical and Computer Engineering* 7, no. 6, 3711-3726, 2017.
- [2]. G. Dongbing, and Z. Wang. "Leader–follower flocking: algorithms and experiments." *IEEE Transactions on Control Systems Technology* 17, no. 5, 1211-1219, 2009.
- [3]. T. Bambang, S. Nurmaini, and G. Ogi. "Fuzzy Logic-Ant Colony Optimization for Explorer-Follower Robot with Global Optimal Path Planning." *Computer Engineering and Applications Journal* 7, no. 1, 2018.
- [4]. K. Fantian, C. Youping, X. Jingming, Z. Gang, and Z. Zude, Z. Mobile robot localization based on extended Kalman filter. In Proceedings of the 2006 6th World Congress on Intelligent Control and Automation, Dalian, China, 21–23; pp. 9242–9246, June 2006.
- [5]. M. Grewal and A. Andrews. (1993). "*Kalman filtering: theory and practice*". Prentice-Hall, Inc., Englewood Cliffs, New Jersey, June 2006
- [6]. R. Negenborn. "*Robot Localization and Kalman Filters on finding your position in a noisy world*". University of Utrecht, 2003.
- [7]. M. Ribeiro, Isabel. "*Kalman and Extended Kalman Filters: Concept, Derivation and Properties*". University of Lisbon, Portugal, 2004.
- [8]. G. Tzafestas, Spyros. 2014. "*Introduction to Mobile Robot Control*". Elsevier: London. Welch, G. and Bishop, G. "An Introduction to the Kalman Filter", 2006.
- [9]. S. Janabi, Farrokh, and M. Mohammed. "A Kalman-filter-based method for pose estimation in visual servoing." *IEEE transactions on Robotics*, 26, no. 5:939-947, 2010.
- [10]. F. Gita, and S. Nurmaini. "Formation control of leader-follower robot using interval type-2 fuzzy logic controller." In *Electrical Engineering and Computer Science (ICECOS), 2017 International Conference on*, pp. 44-49, 2017.
- [11]. P. Zhaoxia, G. Wen, A. Rahmani, and Y. Yu. "Leader–follower formation control of nonholonomic mobile robots based on a bioinspired neurodynamic based approach." *Robotics and autonomous system*, vol. 61, no. 9: 988-996, 2013.
- [12]. S.Y. Chen. "Kalman filter for robot vision: a survey." *IEEE Transactions on Industrial Electronics*, 59, no. 11:4409-4420, 2012.
- [13]. G. Huang, P, I. Anastasios, Mourikis, and I. Stergios and Roumeliotis. "Analysis and improvement of the consistency of extended Kalman filter based SLAM." In *2008 IEEE International Conference on Robotics and Automation*, pp. 473-479, 2008.
- [14]. A. Mary, and G. Hancke. "Pose estimation of a mobile robot based on fusion of IMU data and vision data using an extended Kalman filter. *Sensors*, Vol. 17, no. 10:2164, 2017.

- [15]. I. Edouard, and I. Petrovic. "Extended Kalman filter based mobile robot pose tracking using occupancy grid maps." In *Electrotechnical Conference, MELECON 2004. Proceedings of the 12th IEEE Mediterranean*, vol. 1, pp. 311-314, 2004.