Comparative Study on Object Tracking Algorithms for mobile robot Navigation in GPS-denied Environment

H.S Hewawasam*,M. Yousef Ibrahim†, Gayan Kahandawa‡ and T. A. Choudhury§

School of Engineering and Information Technology
Federation University Australia, Churchill
Vic. 3842, Australia
Email: *h.hewawasam@federation.edu.au, †yousef.ibrahim@federation.edu.au, ‡g.appuhamillage@federation.edu.au,
§t.choudhury@federation.edu.au

***Abstract*—This paper presents a comparative study conducted on the performance of the commonly used object tracking and location prediction algorithms for mobile robot navigation in a dynamically cluttered and GPS-denied mining environment. The study was done to test the different algorithms for the same set criteria (such as accuracy and computational time) under the same conditions.**

The identified commonly used algorithms for object tracking and location prediction of moving objects used in this investigation are Kalman filter (KF), extended Kalman filter (EKF) and particle filter (PF). The study results of those algorithms are analyzed and discussed in this paper. A trade-off was apparent. However, in overall performance KF has shown its competitiveness.

The result from the study has found that the KF based algorithm provides better performance in terms of accuracy in tracking dynamic objects under commonly used benchmarks. This finding can be used in development of an efficient robot navigation algorithm.

***Index Terms*—GPS-denied environment, object tracking, mobile robot navigation, Kalman filter, Particle filter, autoregressive model, point based tracking, physical based tracking, statistical based tracking**

I. Introduction

Navigation of Mobile robots and autonomous vehicles in underground mining can be challenging and hazardous task. Global Positioning System (GPS) signal can be either weak or non-existent in underground mines. Nevertheless, for safe and efficient navigation of mobile robots in those GPS-denied and dynamically cluttered environments would require not only moving objects tracking but also reliable locations prediction of those objects. Objects tracking in addition to accurate and reliable location prediction of moving objects are fundamental requirements for the safe and efficient navigation of mobile robots in underground mining environment.

Considerable amount of research is being done in the field of Multiple/Single Object Tracking (M/SOT) or Multiple/single Target Tracking (M/STT). This field has become a major research area due to its increasing commercial applications such as navigating robots in mines, surveillance systems, security systems, etc. [1]. Predicting the locations of moving objects can be achieved by using object tracking and prediction algorithms [1]–[7]. This paper test, compares and analyses the various commonly used techniques to achieve those tasks. The comparative study was done under common bench marking object motion conditions to evaluate the suitability of selected algorithms for the robot navigation problem.

The main steps of object tracking can be identified as object detection, object classification and object tracking [8]. There are mainly two strategies used for object detection. They are detection-free tracking (DFT) and detection-based tracking (DBT) [9], Fig.1. In DFT objects are tracked manually. In DBT some specific algorithms such as frame difference, background subtraction and optical flow are used [1]. When multiple object tracking is considered various sensors are used such as laser range finder (LRF) sensors [3], [10], [11], sonar sensors [10], and cameras [1], [12], where computer vision plays an important role [7], [8], [13], [14].

Once the objects are detected, object differentiation methods are used to identify objects separately. Object classification methods have been developed based on objects appearance and its motion. In the final step object tracking and prediction of the states of the object is done. Object tracking and prediction algorithms can be mainly divided into three groups [15] as shown in Fig. 1 :

1) Physical-based tracking and prediction methods
2) statistical-based methods
3) cooperative-based.

The physical-based object tracking algorithms can be further divided into three main groups [9]:

- Point based

978-1-5386-6376-9/19/$31.00 ©2019 IEEE

- Kernel based [16]
- Silhouette based algorithms [17]

Physical-based prediction methods are commonly used to estimate the positions of moving objects and track them in a dynamic environment. Reasons for using physical-based prediction methods will be discussed in section II.

Moving objects can be divided into two groups as slow maneuvering objects and fast maneuvering moving objects. Slow maneuvering objects have constant velocity or small acceleration. Fast maneuvering objects rapidly change their velocity [18]. Generally, dynamic objects can be taken as slow maneuvering objects if a sufficiently high sampling rate is maintained [19]. Those objects can be tracked by point based algorithms.

When slow maneuvering objects are tracked by point based algorithms, motion models for these objects must be derived [15], [19] A few motion models have been developed for slow maneuvering objects. Constant velocity models and constant acceleration models are widely used to model the motion of slow maneuvering objects.

This paper analyses the physical-based object tracking algorithms used for online object tacking of slow maneuvering objects for mobile robot navigation. The paper is arranged in the following order: Section II describes physical and statistical based object tracking. Section III discusses the development of the bench-marking model. Section IV describes the point based object tracking algorithms. Section V provides a comparative study about KF, EKF and PF algorithms for slow maneuvering objects tracking using the results generated. Section VI presents discussion of the obtained results and conclusions.

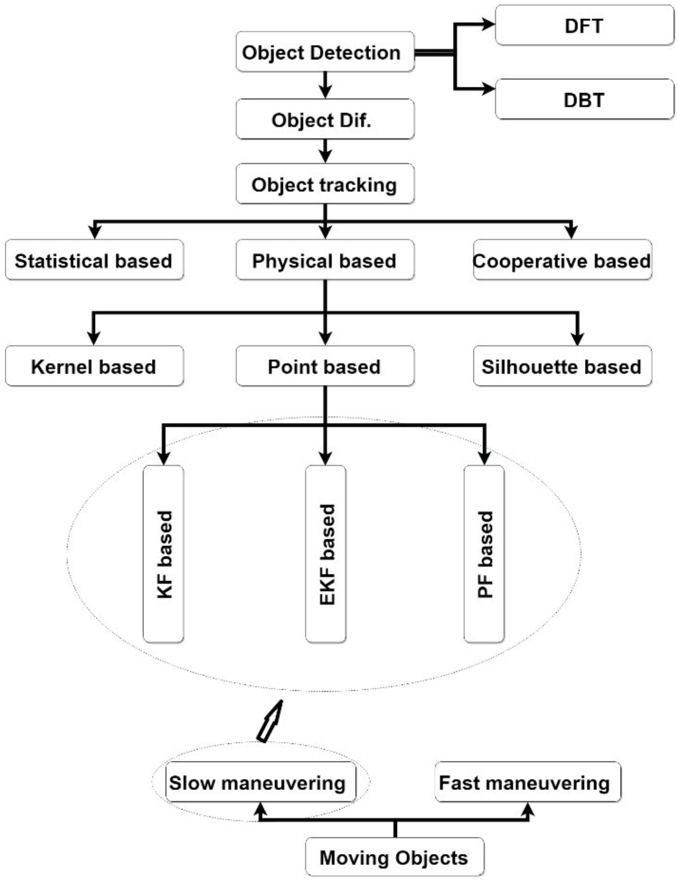

Fig. 1. Object tracking techniques.

II. Physical and statistical based object tracking

Object tracking algorithms can be mainly divided into three groups based on their motion prediction approach as mentioned in section I. Cooperative-based tracking methods are developed by combining the physical and statistical based methods. Cooperative-based prediction methods require the knowledge about the behaviors of moving objects which has to be collected and processed in advance [15]. Because of this limitation cooperative based methods are not suitable for most of the online tracking applications in unknown environment. A through comparative study on physical-based and statistical-based algorithms has been provided in this section.

A. Physical-based tracking

Physical-based location prediction techniques generally assume that a motion model can be developed to represent the movements of a dynamic object. This can be achieved if the required parameters (such as measurements, system matrix and sampling time) are known. Those techniques usually employ an algorithm to estimate the current location, velocity, acceleration, and orientation of moving objects. Coefficients of the motion model are learned by an online learning process. The motion model is then used to compute the next states of the moving objects [15].

Physical-based tracking and prediction can be done by commonly using either Kalman Filter (KF), Extended Kalman Filter (EKF) or Partial Filter (PF).

B. Statistical-based tracking

In this method, moving objects tend to move in accordance with certain trajectories that depend on the nature of the object and the architecture of the environment. These prediction methods use a two stage process: learning stage and prediction stage. Despite the fact that this method can be used for long term prediction; the learning stage requirement makes it unsuitable for online tracking [20]. Grid-based techniques [21], [22] and Cluster-based techniques [23] are some of the common algorithms used in this method.

C. Comparison of statistical-based tracking and physical-based tracking

As shown in table 1 it can be said that physical based tracking is good for online short-term prediction of slow manoeuvring objects in an unknown dynamic environment. Statistical based tracking methods are good for offline, short term or long-term prediction of known moving objects in known dynamic environment. Therefore, it is essential to use a physical based tracking method if the application needs online tracking. Point based algorithms can be identified as a subgroup of physical based tracking.

TABLE I
ADVANTAGES AND DISADVANTAGES OF PHYSICAL-BASED AND STATISTICAL-BASED TRACKING

	Advantages	Disadvantages
Physical based	Online prediction method	Need a proper motion model
	Can start perdition from the 1st time step(in most cases)	Most cases take constant acceleration or velocity models which will reduce the accuracy of estimation
	Low computation cost	Inaccurate in long term prediction
	Can predict location, velocity, acceleration (some cases), orientation, angular velocity, angular acceleration (some cases)	Sudden changes in objects motion may create issues in prediction
	Good for short term prediction	Noisy measurements due to noisy sensors
Statistical based	No need of motion model	Need offline training
	There are no acceleration or velocity constrains	Motion patterns have to be derived
	Has less issues with noise	Motion rules have to be made
	Can predict long term states of objects	

III. Bench-marking Model

As stated in Section I if a good sampling rate is maintained, a moving object can be considered as a slow maneuvering object. In this section a slow maneuvering object is modeled with constant acceleration. The same model was used to examine and compare the tracking performance for KF, EKF and PF based algorithms. The motion model is expressed below, Equations 1-8

$$s = ut + 1/2at^2 \tag{1}$$
$$v = u + at \tag{2}$$

Where:

$s \overset{\text{def}}{=}$ distance travelled in a sample
$u \overset{\text{def}}{=}$ velocity in n^{th} sample
$a \overset{\text{def}}{=}$ acceleration
$v \overset{\text{def}}{=}$ velocity in n^{th} sample

Applying 1 and 2 to x direction

$$x(t+1) = x(t) + dx(t)/dt + \frac{1}{2}u(t)t^2 \tag{3}$$

$$dx(t+1)/dt = dx(t)/dt + u(t)t \tag{4}$$

Applying 1 and 2 to y direction:

$$y(t+1) = y(t) + dy(t)/dt + \frac{1}{2}u(t)t^2 \tag{5}$$

$$dy(t+1)/dt = dy(t)/dt + u(t)t \tag{6}$$

$$\begin{bmatrix} x(t+1) \\ y(t+1) \\ dx(t+1)/dt \\ dy(t+1)/dt \end{bmatrix} = \begin{bmatrix} 1 & 0 & t & 0 \\ 0 & 1 & 0 & t \\ 0 & 0 & 1 & t \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x(t) \\ y(t) \\ dx(t)/dt \\ dy(t)/dt \end{bmatrix} + \begin{bmatrix} 1/2t^2 \\ 1/2t^2 \\ t \\ t \end{bmatrix} u(t) + w(t) \tag{7}$$

$x(t) \overset{\text{def}}{=}$ position in x direction
$y(t) \overset{\text{def}}{=}$ position in y direction
$w(t) \overset{\text{def}}{=}$ process noise

Location parameters x and y can be measured in 2D space . Therefore, output equation can be expressed as follows.

$$\begin{bmatrix} x(t) \\ y(t) \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} x(t) \\ y(t) \\ dx(t) \\ dy(t) \end{bmatrix} + v(t) \tag{8}$$

Where:
v(t)$\overset{\text{def}}{=}$ measurement noise

IV. Point based object tracking algorithms

In point-based object tracking, feature points of moving objects are used to represent and track the moving objects [24]. Kalman filter (KF), extended Kalman filter (EKF) and partial filter (PF) are some of the powerful approaches used in point based multiple object tracking algorithms.

A. Kalman Filter (KF)-Based Object Tracking Algorithms

Kalman-based algorithms work in two process states: the prediction state and the correction/update state. In the prediction state, Kalman filter yields the estimates of current state variables, along with their own uncertainties. In the update state, pre- estimates are updated by a weighted average [25]. These algorithms are recursive. To track a moving object the algorithms use the current measurement and estimation in the previous sample. Kalman filter does not make any of the assumption that the errors are Gaussian. However, the filter does produce the exact conditional probability estimate in the special case that if all the errors are Gaussian distributed [25]. If a system can be described by linear system equations the standard KF based algorithms can be used to estimate states of the system.

1) Example algorithms using KF techniques: Two examples of algorithms using KF technique are presented and analysed in this paper.

a) KF-First example: Elbagir et. al. [19] have demonstrated the use of KF to predict future position and orientations of freely moving objects. However, the proposed algorithm and framework has limited constraints when it is applied for robot motion planning in dynamic environments. The main advantage of the proposed frame work is its ability to start prediction of the states from the first time step without the need of a prior history [19].

b) KF-Second example: In this example G'omez-Ortega et. al. have presented predictive navigation system for mobile robots in [26]. The system deals with unexpected moving obstacles which future positions over a prediction horizon are estimated with a Kalman filter approach. Genetic algorithms have been used for real time optimization problem involved in the model based predictive control problem [26].

2) Analysing the algorithms : The performance of the KF-based algorithms was analysed using the bench-marking motion model described in Section III. Both algorithms used a common filter for object tracking. This filter was redeveloped using Matlab. The object model developed in section III was tracked in one direction using the developed filter (used in the above two algorithms). Observed graphs are shown in Fig. 2

The object was tracked for 10s with a sampling period of 0.1s. The graph in Fig .2. shows the actual position of the moving object, measured/ observed position from sensor and the estimated position using KF with respect to time. When the measured and estimated positions are compared it is clear that KF technique has improved tracking accuracy.

From analysis it can be justified that the KF based object tracking algorithms,

- use a developed motion model for dynamic objects
- can predict states of the moving object one-time step ahead
- assume that previous and current positions are available from sensory devices
- are capable of tracking slow manuring objects

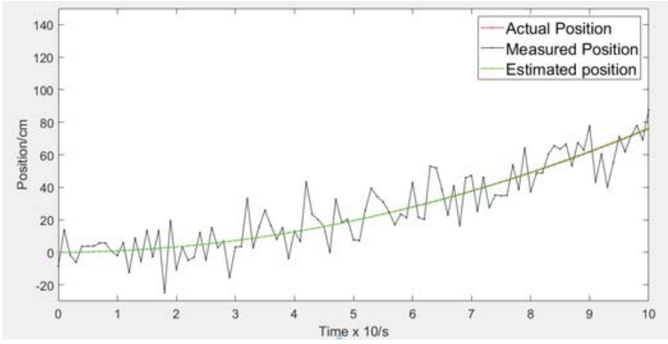

Fig. 2. Estimation results of KF-based algorithm.

B. Extended Kalman Filter (KF)-Based Object Tracking Algorithms

When the system is nonlinear, KF based algorithms do not converge [27]. Therefore, KF cannot be used for tracking non-linear systems. The EKF based algorithms linearizes the state space model using first order approximation. The mean and covariance of states are propagated using Jacobian matrices [28]. Ones the function is linearized using Jacobian matrix, these algorithms use the filter operating point to estimate the states. This process is performed at each time step in EKF based algorithms.

In order to use EKF based algorithms for a non-linear system, transition and observation models can be nonlinear but they should be differentiable. At the same time process noise and measurement noise should be zero mean Gaussian distributions.

1) Example algorithms using EKF techniques: The following two examples reported below have attempted to use EKF.

a) EKF-First example: Madhavan et. al. in [10] reported work done using an EKF-based Algorithm for a moving object prediction framework for off-road autonomous navigation. This method can predict moving objects' future position for path planning of unmanned ground vehicle navigation in dynamic environments. The short-term predictions are based on an EKF working symbiotically with a probabilistic object classification scheme. The proposed framework was shown to deliver reliable position estimates for a wheeled vehicle.

b) EKF-Second example: Rebai et. al. [11] have proposed a method to detect and track moving objects. The method was based on EKF algorithm for tracking mobile objects. In this precented approach authors have used a laser range finder to detect moving objects. Once the objects are detected and classified the EKF based algorithm deals with dynamic object tracking. This is achieved by a set of Extended Kalman Filters

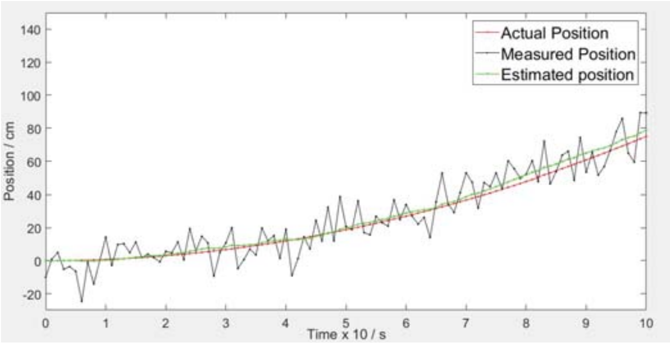

Fig. 3. Estimation results of EKF-based algorithm.

(EKF) where an EKF is initialized for each mobile object and evaluated for the prediction and update phases. Authors have modelled the moving object nonlinearly, hence they had to use an EKF-based algorithm. The used motion model is expressed bellow, Equations 9-11.

$$x_{k+1} = x_k + v_k.\Delta t.\cos(\theta_k) \tag{9}$$

$$y_{k+1} = y_k + v_k.\Delta t.\sin(\theta_k) \tag{10}$$

$$\theta_{k+1} = \theta_k + \Delta\theta \tag{11}$$

Where the state vector $X = [x, y, \theta]'$. Also x, y are the coordinates of the object centre and θ is the direction of the velocity vector. $'v'$ is the translational velocity and $\Delta\theta$ is the orientation change.

2) Analysing the algorithms : These two algorithms have used an EKF based technique to predict and track moving objects. The filters used in the above two algorithms were redeveloped using Matlab. The same bench marking model in section III was tracked by the developed filter with the same conditions previously used. Observed results are as in shown in Fig. 3.

The graph in Fig .3. shows the actual position of the moving object, measured/ observed position from sensor and the estimated position using PF with respect to time. It was observed that the sensors' measured/observed positions were noisy and had considerably higher error. However, it was also seen from the obtained results that the estimated positions are much more accurate than the measured position, Fig. 3. That was the effect of applying the filter.

From the analysis it can be justified that the EKF based object tracking algorithms,

- can be used to track slow maneuvering objects
- use an object model
- can predict one step ahead
- are capable of tracking non-linear systems (Developed filter proves the linearization)

C. Particle Filter-Based Object Tracking Algorithms

Particle filter based algorithms are sequential Monte Carlo methods based on point mass (or "particle") representations of probability densities [29], [30]. PF-based methods eliminate the main constrains of KF methods. Therefore, PF-based methods can be applied to any state-space model (non-linear and non-Gaussian). To model dynamic systems, the focus will be on the discrete-time formulation of the problem. At least two models are required for particle filter based algorithms to analyze and make inference about a dynamic system [30]. Firstly, a model describing the evolution of the state with time (the system model) and, secondly, a model relating the noisy measurements to the state (the measurement model). It has been assumed that these models are available in a probabilistic form.

Firstly, a model describing the evolution of the state with time (the system model) and, secondly, a model relating the noisy measurements to the state (the measurement model). It has been assumed that these models are available in a probabilistic form. A posterior probability density function (pdf) of the state is constructed in Bayesian approach. All available information, including the measurements are used to develop the pdf. This pdf embodies all available statistical information. Therefore, it represents a broader solution to the estimation problem. In principle, an optimal estimate of the state may be obtained from the pdf. For a smoother navigation, an estimate is required every time that a measurement is received. In this case, a recursive filter is used as a convenient solution. The prediction stage uses the system model to predict the state pdf for the next iteration. The update operation uses the latest measurement to modify the prediction pdf. This is achieved using Bayes theorem. This is the mechanism for updating knowledge about the target state in the light of extra information from new data [30].

1) Example algorithms using PF techniques: Two examples of algorithms using PF technique are presented and analysed in this paper.

a) Partical filter based object tracking using sensors information: The presented method in [29] is based on Particle Filters and Sample-based Joint Probabilistic Data Association (SJPDAF). This method can be used to deal with the non-linear and non-Gaussian models. It uses a probability density description of the model instead of a linear and Gaussian model. It is mentioned that particle filters are used to estimate the states of non-linear and non-Gaussian systems. In order to estimate states, particle filters are used to approximate a posterior density function as set of weighted samples, "particles".

b) PF-Second example: Almeida et. al have presented a method for tracking multiple moving objects, in dynamic environments, using particles filters and SJPDAFs in [31]. However, the research problem which has been addressed

TABLE II
PERFORMANCE OF PF WITH DIFFERENT NUMBER OF PARTICLES

Number of Particles	Maximum Error/m	Variation/m2	Computation Time/s
100	5.169	10.08	0.2864
1000	3.7494	1.8363	4.4930
5000	2.6349	1.2633	123.3613

was the development of sensor-based method to track moving objects around the robot. This framework is capable of modelling the dynamic environment and predicting the stats of moving objects. This allows the robot to detect and track several moving objects using a range finder sensor. Particle filters and Sample-based Joint Probabilistic Data Association Filters are the methods used in this approach.

2) Analysing the algorithms : In order to analyse the performance of PF, a PF algorithm was developed using Matlab. An object with the same motion model (described in Section III) was tracked using PF model. . Then results were plotted as shown in Fig. 4 (PF with 100 particles). Performance of the PF was further analysed by changing the number of particles used in the filter results are shown in table 2. For the comparative study PF with 100 partials has been used in order to keep the computation time in-rage with other two methods.

Fig .4 shows the actual position of the moving object, measured/ observed position from sensor and the estimated position using PF with respect to time. It was noticed that measured/observed positions were noisy and had considerably higher error. However, this noise was suppressed resulting in a more accurate estimation following the application of the filter.

It was also observed when higher number of particles were used the estimation error was further reduced, but the computation time was considerably increased, Table 2.

From the analysis it can be justified that the PF based object tracking algorithms,

- can be use to track slow maneuvering objects
- can be used to track non-Gaussian models
- use an object motion model
- become more accurate when filter uses more particles
- can track non-linear systems

D. Comparison of Algorithms According to the State Estimator

A general comparison on the algorithms based on error distribution, computation cost and system model is provided in Table 3. KF-based algorithms can be used to track liner Gaussian systems. EKF-based state estimators are capable of tracking non-linear but Gaussian systems. PF-base algorithms can track non-linear and non-Gaussian systems.

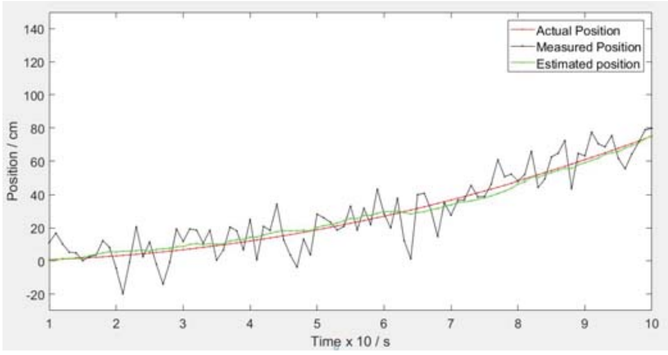

Fig. 4. Estimation results of PF-based algorithm.

TABLE III
COMPARISON OF ALGORITHMS

State Estimator	System model	Error Distribution	Computation Cost
KF	Linear	Gaussian	Low
EKF	Non-Linear (Locally Linear)	Gaussian	Low
PF	Non-Linear/Linear	Non- Gaussian /Gaussian	High

V. RESULTS AND DISCUSSION

A. Testing Results

Three selected algorithms (KF, EKF and PF based) were used to track the slow maneuvering object, modelled in section III. The same type of Gaussian distributed noise was used for measurement and process noise in all three cases. A same sampling rate was maintained in all three algorithms.

In all three case the actual, measured and estimated positions were plotted in the same graph as shown in Fig. 2, Fig. 3 and Fig. 4. Different between actual position and estimated position which is known as estimation error was then calculated for each estimation. The estimation errors for all the three algorithms were plotted in the same graph as shown in Fig. 5. Maximum estimation error, variance of estimation errors and computation time for each case were calculated and displayed in the bar chart shown in Fig. 6. The developed graphs and bar chart were used to analyse the performances of each algorithms in terms of accuracy. A brief analysis on computation time also been completed.

Figure 2, 3, and 4 show actual, measured and estimated positions of a single moving object which was tracked using KF, EKF and PF based algorithms respectively. In order to compare the accuracy and keep the computation time of the PF in the same range as KF and EKF, the number of particles in PF algorithm was adjusted to 100. The results show that tracking have been significantly improved by using these algorithms. As shown in Fig. 2, 3 and 4 measured positions

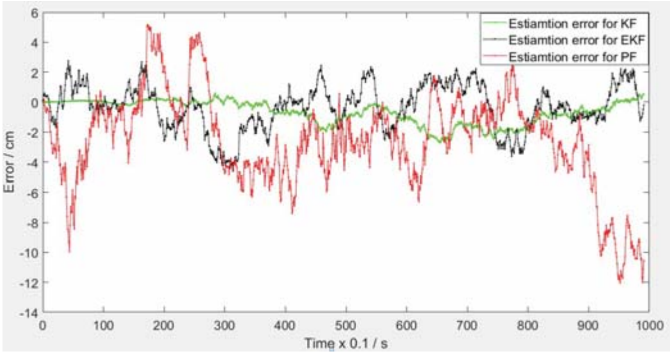

Fig. 5. Estimation error comparison between KF, EKF and PF based algorithms.

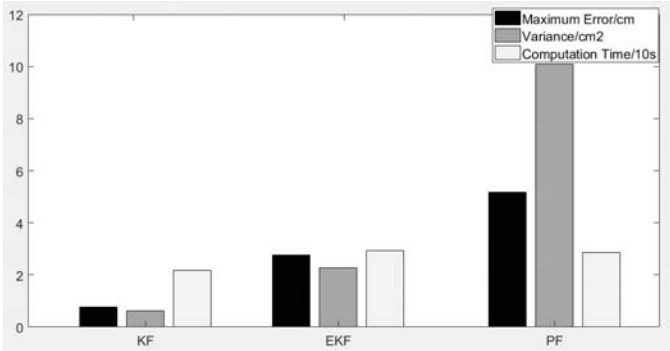

Fig. 6. Estimation error comparison between KF, EKF and PF based algorithms.

(black lines) are much deviated from actual positions (red lines). After applying the tracking algorithms, the accuracy of the estimated positions (green lines) were improved. Fig. 5 shows the estimation error of used algorithms during the entire process. The results in Fig. 5 show that when a slow maneuvering object is tracked by KF based algorithm the estimation error and variation are smaller compared to other two methods. PF based algorithm shows higher variations in estimation error.

Maximum estimation error was found with PF-based algorithm compared with the other two algorithms. The error was around 5.1cm (for a=1.5cm/s2, and sampling period is 0.1s than maximum error was 5.1cm). KF based algorithm has the lowest maximum estimation error as shown in Fig. 6. Maximum estimation error for KF based algorithm was 0.7cm. PF-based algorithm has the highest estimation error variance and the lowest is the KF based algorithm as shown in Fig. 6. Accuracy of the particle filter can be improved by increasing the number of particles as shown in Table 2. However, the computation time is increased compared with the other two methods. As shown in Fig. 6 computation times for KF, EKF and PF were 0.21834s, 0.2938s and 0.2864s respectively. KF based algorithm was slightly faster than other tested algorithms.

When the overall estimation error is considered, during the entire estimation process, KF based algorithm had the lowest error, while PF based algorithm had the highest as shown in Fig. 6.

VI. Conclusion

This paper presented a comparative study on dynamic objects tracking for mobile robots path planning in dynamically cluttered GPS-denied environment. KF, EKF and PF based algorithms have been used to track slow maneuvering objects in a dynamic environment. The moving objects were modelled as constant acceleration objects by maintaining a sufficient sampling rate.

The conducted comparative simulation based analysis shows, KF-based algorithms are more accurate than EKF and PF algorithms. EKF based algorithms are more suitable for tracking non-linear but Gaussian models because EKF linearizes the state space model. If the system model is non-linear and non-Gaussian, then PF based algorithms would be a better option due to its point mass representations of probability densities. However, the accuracy of PF based algorithm was low when tracking constant acceleration model under the used bench-marking motion model. This can be improved by increasing the number of particles but, this will lead to an increased computation cost. It can be concluded that KF based algorithms perform better in tracking slow maneuvering objects, for mobile robot navigation in dynamically cluttered GPS-denied environments.

References

[1] H. Naeem, J. Ahmad, and M. Tayyab, "Real-time object detection and tracking," in *INMIC*, 2013, Conference Proceedings, pp. 148–153.

[2] W. Yi, M. R. Morelande, L. Kong, and J. Yang, "An efficient multi-frame track-before-detect algorithm for multi-target tracking," *IEEE Journal of Selected Topics in Signal Processing*, vol. 7, no. 3, pp. 421–434, 2013.

[3] A. H. A. Rahman, H. Zamzuri, S. A. Mazlan, M. A. A. Rahman, and M. A. Zakaria, "Tracking uncertain moving objects using dynamic track management in multiple hypothesis tracking," in *2014 International Conference on Connected Vehicles and Expo (ICCVE)*, 2014, Conference Proceedings, pp. 345–350.

[4] T. Chen, R. Wang, B. Dai, D. Liu, and J. Song, "Likelihood-field-model-based dynamic vehicle detection and tracking for self-driving," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 11, pp. 3142–3158, 2016.

[5] J. Dey and N. Praveen, "Moving object detection using genetic algorithm for traffic surveillance," in *2016 International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT)*, 2016, Conference Proceedings, pp. 2289–2293.

[6] D. Crouse, "Basic tracking using nonlinear continuous-time dynamic models [tutorial]," *IEEE Aerospace and Electronic Systems Magazine*, vol. 30, no. 2, pp. 4–41, 2015.

[7] H. Kim and K. Sim, "A particular object tracking in an environment of multiple moving objects," in *ICCAS 2010*, 2010, Conference Proceedings, pp. 1053–1056.

[8] S. R. Balaji and S. Karthikeyan, "A survey on moving object tracking using image processing," in *2017 11th International Conference on*

Intelligent Systems and Control (ISCO), 2017, Conference Proceedings, pp. 469–474.

[9] W. Luo, J. Xing, A. Milan, X. Zhang, W. Liu, X. Zhao, and T.-K. Kim, *Multiple Object Tracking: A Literature Review*, 2017.

[10] R. Madhavan and C. I. Schlenoff, "Moving object prediction for off-road autonomous navigation," in *AeroSense 2003*, vol. 5083. SPIE, 2003, Conference Proceedings, p. 12.

[11] R. Karima, A. Benabderrahmane, O. Azouaoui, and N. Ouadah, *Moving obstacles detection and tracking with laser range finder*, 2009.

[12] L. O. Russo, G. A. Farulla, M. Indaco, S. Rosa, D. Rolfo, and B. Bona, "Blurring prediction in monocular slam," in *2013 8th IEEE Design and Test Symposium*, 2013, Conference Proceedings, pp. 1–6.

[13] J. S. Kulchandani and K. J. Dangarwala, "Moving object detection: Review of recent research trends," in *2015 International Conference on Pervasive Computing (ICPC)*, 2015, Conference Proceedings, pp. 1–5.

[14] S. V. Amarasinghe, H. S. Hewawasam, W. B. D. K. Fernando, J. V. Wijayakulasooriya, G. M. R. I. Godaliyadda, and M. P. B. Ekanayake, "Vision based obstacle detection and map generation for reconnaissance," in *2014 9th International Conference on Industrial and Information Systems (ICIIS)*, 2014, Conference Proceedings, pp. 1–6.

[15] c. Jayawardena, "A technical review of motion prediction methods for indoor robot navigation," 2015, Book.

[16] Q. Miao and Y. Gu, "Kernel-based online object tracking via gaussian mixture model learning," in *2016 Sixth International Conference on Instrumentation Measurement, Computer, Communication and Control (IMCCC)*, 2016, Conference Proceedings, pp. 522–525.

[17] W. Liang, T. Tieniu, N. Huazhong, and H. Weiming, "Silhouette analysis-based gait recognition for human identification," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 12, pp. 1505–1518, 2003.

[18] M. Keshmiri and M. Keshmiri, "Performance comparison of various navigation guidance methods in interception of a moving object by a serial manipulator considering its kinematic and dynamic limits," in *2010 15th International Conference on Methods and Models in Automation and Robotics*, 2010, Conference Proceedings, pp. 212–217.

[19] A. Elnagar, "Prediction of moving objects in dynamic environments using kalman filters," in *Proceedings 2001 IEEE International Symposium on Computational Intelligence in Robotics and Automation (Cat. No.01EX515)*, 2001, Conference Proceedings, pp. 414–419.

[20] D. Vasquez and T. Fraichard, "Motion prediction for moving objects: a statistical approach," in *IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA '04. 2004*, vol. 4, 2004, Conference Proceedings, pp. 3931–3936.

[21] A. F. Foka and P. E. Trahanias, "Probabilistic autonomous robot navigation in dynamic environments with human motion prediction," *International Journal of Social Robotics*, vol. 2, no. 1, pp. 79–94, 2010. [Online]. Available: https://doi.org/10.1007/s12369-009-0037-z

[22] R. Nicholas, N. Paul, and S. Siddhartha, *Modeling and Prediction of Pedestrian Behavior Based on the Sub-Goal Concept*. MITP, 2013, p. 1. [Online]. Available: http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=6578005

[23] C. Hermes, C. Wohler, K. Schenk, and F. Kummert, "Long-term vehicle motion prediction," in *2009 IEEE Intelligent Vehicles Symposium*, 2009, Conference Proceedings, pp. 652–657.

[24] U. K. J. H. S. Parekh, D. G. Thakore, "A survey on object detection and tracking methods," *International Journal of Innovative Research in Computer and Communication Engineering*, vol. 2, 2014.

[25] M. Marron, J. C. Garcia, M. A. Sotelo, M. Cabello, D. Pizarro, F. Huerta, and J. Cerro, "Comparing a kalman filter and a particle filter in a multiple objects tracking application," in *2007 IEEE International Symposium on Intelligent Signal Processing*, 2007, Conference Proceedings, pp. 1–6.

[26] J. Gómez-Ortega, D. R. Ramírez, D. Limón-Marruedo, and E. F. Camacho, "Genetic algorithms based predictive control for mobile robot navigation in changing environments," in *2001 European Control Conference (ECC)*, 2001, Conference Proceedings, pp. 3179–3184.

[27] Q. Li, R. Li, K. Ji, and W. Dai, "Kalman filter and its application," in *2015 8th International Conference on Intelligent Networks and Intelligent Systems (ICINIS)*, 2015, Conference Proceedings, pp. 74–77.

[28] D. H. Santosh and P. G. K. Mohan, "Multiple objects tracking using extended kalman filter, gmm and mean shift algorithm - a comparative study," in *2014 IEEE International Conference on Advanced Communications, Control and Computing Technologies*, 2014, Conference Proceedings, pp. 1484–1488.

[29] J. Almeida, A. Almeida, and R. Araujo, "Tracking multiple moving objects for mobile robotics navigation," in *2005 IEEE Conference on Emerging Technologies and Factory Automation*, vol. 1, 2005, Conference Proceedings, pp. 8–210.

[30] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-gaussian bayesian tracking," *IEEE Transactions on Signal Processing*, vol. 50, no. 2, pp. 174–188, 2002.

[31] J. Almeida and R. Araujo, "Tracking multiple moving objects in a dynamic environment for autonomous navigation," in *2008 10th IEEE International Workshop on Advanced Motion Control*, 2008, Conference Proceedings, pp. 21–26.