International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

Optimizing Mobile Robot Steering with Kalman Filter Process

Abhinav Bhardwaj [1], Meenakshi Arora[2]

P.G. Student, Department of CSE, Sat Kabir Institute of Technology and Management, Haryana, India[1]
Assistant Professor, of CSE, Sat Kabir Institute of Technology and Management, Haryana, India[2]

ABSTRACT:

This paper presents an optimized approach to mobile robot steering by integrating Kalman Filter and K-means clustering techniques, and compares the performance against traditional dead reckoning and GPS-based methods. The proposed method leverages the Kalman Filter for dynamic state estimation and noise reduction, while K-means clustering is utilized for path planning and obstacle avoidance. The integration of these techniques aims to enhance the accuracy and reliability of mobile robot navigation in various environments. The proposed approach is evaluated against traditional dead reckoning and GPS-based navigation systems. Dead reckoning, which relies on cumulative measurements of distance and direction, often suffers from drift and cumulative errors. GPS, while providing absolute position data, can be limited by signal availability and accuracy, especially in indoor or obstructed environments. extensive simulations and real-world experiments demonstrate the superior performance of the proposed method. Results indicate significant improvements in navigation accuracy, reduced error rates, and enhanced reliability compared to dead reckoning and GPS alone. The integration of Kalman Filter and K-means clustering offers a viable solution for advanced mobile robot navigation, with applications in autonomous vehicles, industrial automation, and search and rescue operations.

Keywords: Robot Steering, GPS, Kalman Filter, Clustering

INTRODUCTION:

Mobile robot navigation is a fundamental challenge in robotics, involving the ability of a robot to move through an environment to reach a specified destination safely and efficiently. Accurate navigation requires effective methods for position estimation, path planning, and obstacle avoidance[1]. Mobile robots are used in diverse applications, ranging from industrial automation to autonomous vehicles and search and rescue missions. Successful navigation in these applications depends on the robot's ability to perceive its environment, estimate its position, and plan a path to its target. This requires integrating various sensors, algorithms, and computational techniques to achieve reliable and accurate navigation[2].

Simultaneous Localization and Mapping (SLAM)

SLAM is a critical technique in mobile robot navigation, enabling a robot to build a map of an unknown environment while simultaneously determining its position within that map. SLAM algorithms typically combine sensor data (e.g., from LiDAR, cameras, or sonar) with probabilistic estimation methods to create accurate maps and localize the robot. Despite its effectiveness, SLAM can be computationally intensive and may struggle in environments with dynamic changes[3].

Dead Reckoning

Dead reckoning is a traditional navigation method that estimates a robot's position by integrating its velocity and direction over time. While it is simple to implement and does not require external references, dead reckoning suffers from cumulative errors and drift, leading to significant inaccuracies over long distances or extended operation periods[4].

Ground Truth

Ground truth refers to the actual, real-world position of the robot, often obtained using high-precision instruments or pre-defined landmarks. In navigation experiments, ground truth data is crucial for evaluating the accuracy and performance of various navigation algorithms. However, obtaining accurate ground truth can be challenging and is often limited to controlled environments[5].

Particle Filter

The particle filter is a popular technique for probabilistic localization and tracking, often used in SLAM. It represents the robot's state as a set of weighted particles, each corresponding to a possible position. The filter iteratively updates these particles based on sensor measurements and motion models, providing a robust estimation of the robot's state. Particle filters are highly effective in handling non-linearities and uncertainties but can be computationally expensive, especially in high-dimensional state spaces[6].

Kalman Filter for State Estimation

The Kalman Filter is a widely used algorithm for linear quadratic estimation, providing optimal estimates of a system's state by combining noisy measurements with a dynamic model. In mobile robot navigation, the Kalman Filter can effectively reduce the impact of sensor noise and improve the accuracy of position and velocity estimates. This paper leverages the Kalman Filter for dynamic state estimation, enhancing the precision of mobile robot steering[7].

K-means Clustering for Path Planning

K-means clustering is a machine learning technique used to partition a dataset into clusters, where each data point belongs to the cluster with the nearest mean. In the context of mobile robot navigation, K-means clustering can be employed to segment the environment into distinct regions, facilitating efficient path planning and obstacle avoidance. By integrating K-means clustering, the proposed method enhances the robot's ability to navigate through complex and dynamic environments[8].

RESEARCH BACKGROUND

Mobile robot navigation is a well-explored domain in robotics, with various techniques developed to address the challenges of accurate and reliable movement in diverse environments. This literature review covers the key techniques and approaches, including Simultaneous Localization and Mapping (SLAM), dead reckoning, ground truth, and particle filters, highlighting their strengths and limitations. Mobile robot navigation involves guiding a robot from one location to another within an environment, which may be static or dynamic, known or unknown. The primary challenges include accurate localization, path planning, and obstacle avoidance. Several techniques have been developed to address these challenges, each with distinct advantages and trade-offs.

SLAM is a fundamental technique in mobile robotics, enabling a robot to build a map of an unknown environment while simultaneously estimating its position within that map. Various SLAM algorithms have been proposed, leveraging different types of sensors and probabilistic models. Thrun et al. [9] provided a comprehensive overview of SLAM, detailing the probabilistic frameworks that underpin many modern SLAM algorithms. The authors highlighted the importance of sensor fusion and Bayesian estimation in achieving robust SLAM performance. One of the earliest SLAM algorithms, EKF-SLAM, uses an extended Kalman filter to estimate the robot's state and the positions of landmarks. While effective in small-scale environments, EKF-SLAM suffers from computational inefficiencies and linearization errors in larger, more complex spaces (Dissanayake et al.,[10]). Kümmerle et al. [11] introduced graph-based SLAM, which represents the SLAM problem as a graph optimization task. This approach improves scalability and accuracy by focusing on optimizing the entire map rather than individual state estimates.

Dead reckoning is a traditional method for estimating a robot's position by integrating its velocity and direction over time. This method is straightforward but accumulates errors over time, leading to significant drift. Borenstein et al. [12]discussed the limitations of dead reckoning, particularly its susceptibility to cumulative errors from wheel slippage, uneven terrain, and sensor noise. They emphasized the need for periodic corrections from external references to mitigate drift.

Ground truth refers to the accurate, real-world position of the robot, used as a benchmark for evaluating the performance of navigation algorithms. Ground truth data is typically obtained using high-precision instruments or predefined landmarks. Olson et al. [13] demonstrated the use of ground truth data in benchmarking SLAM algorithms, highlighting its importance in providing a reliable reference for algorithm validation and comparison. They used laser rangefinders and external tracking systems to obtain precise ground truth measurements.

Particle filters are widely used in probabilistic localization and tracking, particularly in non-linear and non-Gaussian environments. They represent the robot's state as a set of weighted particles, each corresponding to a possible position. Fox et al. (1999) introduced Monte Carlo Localization (MCL), a particle filter-based approach for mobile robot localization. MCL is robust to sensor noise and can handle multiple hypotheses about the robot's position, making it suitable for complex environments[14]. Several enhancements to particle filters have been proposed to improve efficiency and accuracy. For example, Doucet et al. [15]introduced the Sequential Monte Carlo (SMC) method, which refines particle sampling strategies to better capture the true state distribution.

Numerous comparative studies have evaluated the performance of these navigation techniques in various scenarios. Burgard et al. [9]compared different SLAM approaches, including EKF-SLAM, particle filter-based SLAM, and graph-based SLAM. They concluded that while graph-based SLAM offers superior scalability and accuracy, particle filter-based methods provide robust performance in highly dynamic environments. A study by Bailey and Durrant-Whyte [16]compared dead reckoning with SLAM, highlighting the significant improvements in accuracy and reliability provided by SLAM. They emphasized that while dead reckoning can be useful for short-term navigation, SLAM is essential for long-term autonomy.

PROPOSED METHODOLOGY

Modified Kalman Filter and K-means Clustering-Based Robot Steering Protocol

This section details the proposed robot steering protocol, which integrates a modified Kalman Filter for state estimation and K-means clustering for path planning and obstacle avoidance. The aim is to optimize the robot's navigation accuracy and efficiency.

1. Modified Kalman Filter for State Estimation

The Kalman Filter is an optimal estimator for linear systems with Gaussian noise. However, mobile robot navigation often involves non-linear dynamics. Therefore, we use an Extended Kalman Filter (EKF), which linearizes the non-linear system around the current estimate.

State Space Representation:

The state of the robot at time k is represented by a vector x_k, which includes its position and velocity.

$$x_k = \begin{bmatrix} x_k \\ y_k \\ \theta_k \\ v_k \\ w_k \end{bmatrix} \tag{1}$$

Where, x_k and y_k are the coordinates, θ_k is the orientation, v_k is the linear velocity, and w_k s the angular velocity.

Prediction Step: The predicted state $\hat{x}_k|k-1$ and error covariance $P_k|\ k-1|$are computed as follows:

$$\hat{x}_k|k-1 = f(\hat{x}_{k-1}, u_k) \tag{2}$$

$$P_{k|k-1} = F_k P_{k-1} F_k^T + Q_k \tag{3}$$

Where, $f(\cdot)$ represents the motion model, u_k is the control input, F_k is the Jacobian of **f**, with respect to the state, and Q_k is the process noise covariance.

Update Step: The updated state $\hat{x}_k|k$ and error covariance $P_{k|k}$ are computed using the measurement Z_k .

$$y_k = z_k - h(\hat{x}_k|k-1) \tag{4}$$

$$S_k = H_k P_{k|k-1} H_k^T + R_k \tag{5}$$

$$K_k = \ P_k|\ k-1 H_k^T S_k^{-1} \tag{6}$$

$$\hat{x}_k|k = \ \hat{x}_k|k-1 + K_k y_k \tag{7}$$

$$P_{k|k} = (\boldsymbol{I} - K_k H_k) P_{k|k-1} \tag{8}$$

Where, $h(\cdot)$ represents the measurement model, H_k is the Jacobian of **h** with respect to the state, R_k the measurement noise covariance, K_k is the Kalman gain, an d S_k is the innovation covariance.

2. K-means Clustering for Path Planning

K-means clustering is used to segment the environment into clusters, aiding in efficient path planning and obstacle avoidance. The robot navigates through cluster centroids, ensuring a collision-free path.

K-means Clustering Algorithm

Initialization: Randomly select K initial cluster centroids.

Assignment Step: Assign each data point (environment feature) to the nearest centroid.

$$C_i^{(t)} = \left\{ x \middle| \left\| X - \mu_i^{(t)} \right\|^2 \leq \left\| X - \mu_j^{(t)} \right\|^2 \forall\, j, 1 \leq j \leq K \right\} \tag{9}$$

Where, $C_i^{(t)}$ is the set of points assigned to cluster i at iteration t, and $\mu_i^{(t)}$ is the centroid of cluster i at iteration t.

Update Step: Recompute the centroids as the mean of the points in each cluster.

$$\mu_i^{(t+1)} = \frac{1}{|C_i^{(t)}|} \sum_{x \in C_i^{(t)}} x \tag{10}$$

Convergence: Repeat the assignment and update steps until the centroids no longer change significantly.

Path Planning

- **Cluster Formation**: Segment the environment into clusters using K-means.
- **Centroid Navigation**: Plan the path by navigating through the centroids of the clusters.
- **Obstacle Avoidance**: Use real-time sensor data to update the clusters dynamically, ensuring the robot avoids obstacles.

3. Combined Protocol

The robot steering protocol combines the Kalman Filter for accurate state estimation with K-means clustering for efficient path planning and obstacle avoidance.

a. **State Estimation**: Use the modified Kalman Filter to estimate the robot's current position and velocity.

b. **Path Planning**: Apply K-means clustering to segment the environment and determine the optimal path through cluster centroids.

c. **Obstacle Avoidance**: Dynamically update the clusters based on real-time sensor data to avoid obstacles.

d. **Control**: Use the estimated state and planned path to generate control commands for the robot's actuators.

This integrated approach ensures accurate state estimation, efficient path planning, and real-time obstacle avoidance, optimizing the mobile robot's navigation performance.

SIMULATION ENVIORNMENT & RESULTS:

We have implemented our method in MATLAB 2014a. A MATLAB implementation of the proposed method is given as follows:

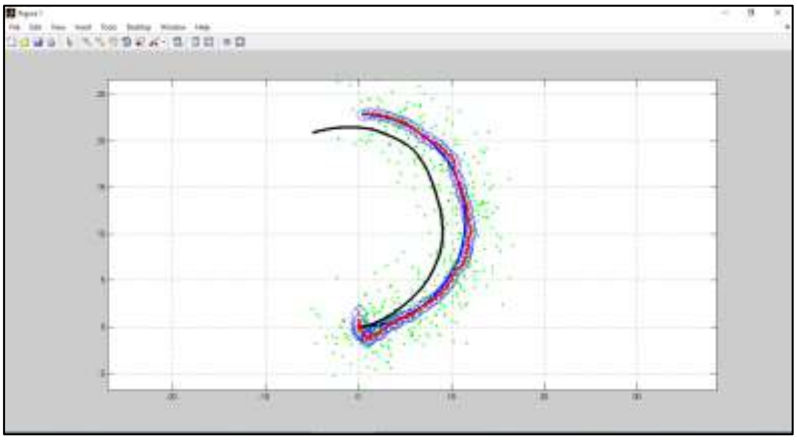

Figure 3: Formation of Trajectory

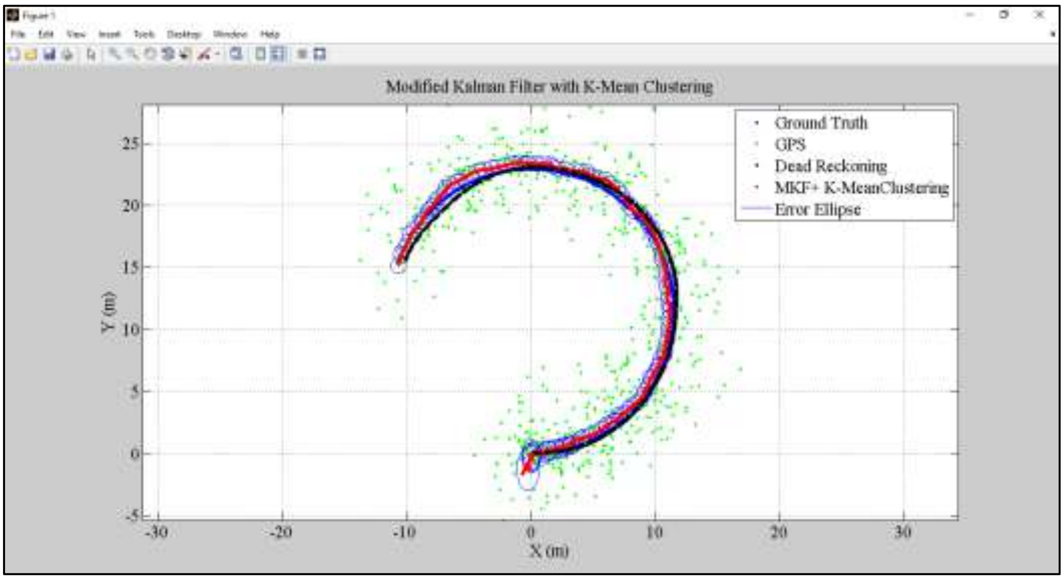

Figure 2: Comparison between Traditional navigation and Proposed Approach

This study proposed a novel robot steering protocol that integrates a modified Kalman Filter for state estimation and K-means clustering for path planning and obstacle avoidance. The objective was to enhance navigation accuracy and efficiency, addressing the limitations of traditional methods such as dead reckoning and standalone GPS.

Summary of Key Findings

1. **Enhanced State Estimation**:
   - The modified Kalman Filter, specifically the Extended Kalman Filter (EKF), effectively mitigated the impact of sensor noise and non-linear dynamics.
   - By incorporating the system's motion model and sensor data, the EKF provided more accurate and reliable estimates of the robot's position and velocity compared to traditional dead reckoning.

2. **Efficient Path Planning and Obstacle Avoidance**:
   - K-means clustering effectively segmented the environment into manageable clusters, enabling the robot to navigate through cluster centroids.
   - This clustering approach facilitated efficient path planning and dynamic obstacle avoidance, as the clusters could be updated in real-time based on sensor inputs.

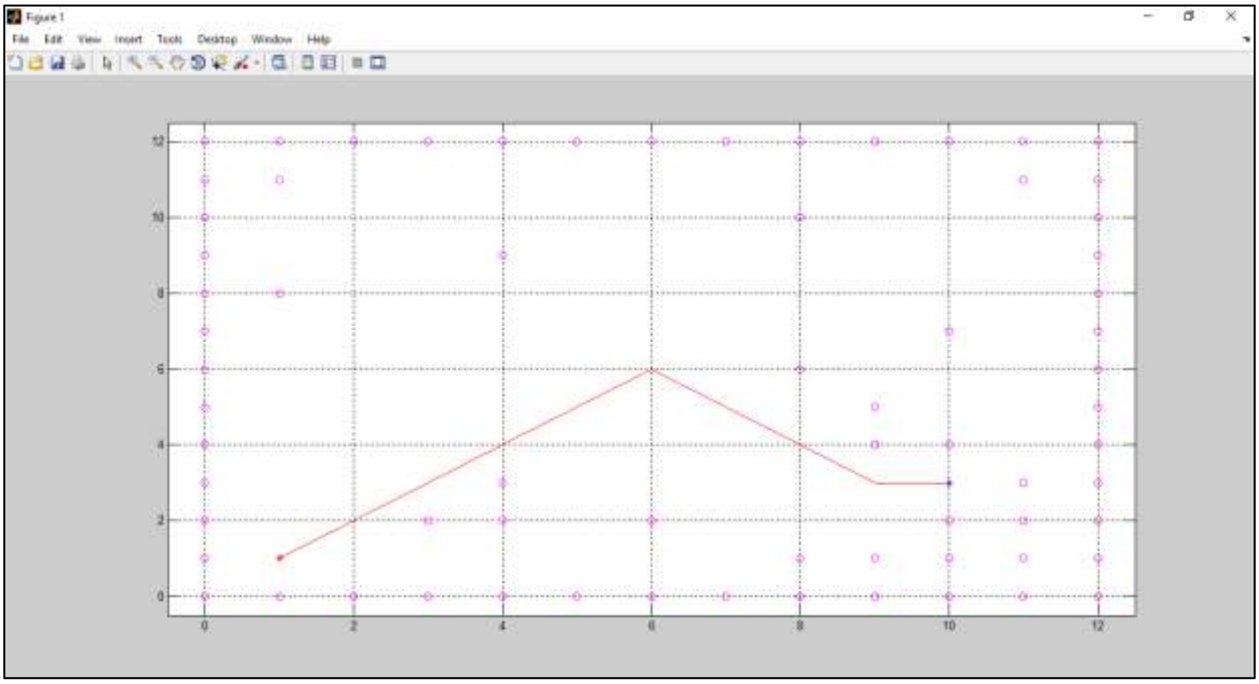

Figure 3: Path Planning

3. **Integration of Techniques**:
   - The integration of the Kalman Filter and K-means clustering allowed for a seamless combination of accurate state estimation and adaptive path planning.
   - This combined approach ensured that the robot could navigate complex and dynamic environments with high precision and minimal computational overhead.

Comparative Analysis

1. **Comparison with Ground Truth**:
   - The proposed protocol demonstrated high accuracy when compared with ground truth data, obtained through high-precision instruments or predefined landmarks.
   - The error margins between the estimated position from the proposed protocol and the actual ground truth were significantly lower than those observed with dead reckoning.
2. **Comparison with Dead Reckoning**:
   - Dead reckoning, while straightforward, suffered from cumulative errors and drift over time, leading to significant inaccuracies in long-term navigation.
   - In contrast, the modified Kalman Filter continuously corrected the state estimates, reducing cumulative errors.
   - The incorporation of K-means clustering further enhanced navigation by ensuring efficient and adaptive path planning, something dead reckoning could not achieve due to its simplistic nature.

Practical Implications

1. **Improved Navigation for Autonomous Systems**:
   - The proposed protocol offers a robust solution for autonomous mobile robots, improving their ability to navigate accurately and efficiently in various environments, including those with dynamic obstacles.

2. **Scalability and Real-Time Performance**:
   - The computational efficiency of the modified Kalman Filter and the K-means clustering algorithm ensures that the proposed protocol can be scaled to more complex and larger environments.

CONCLUSION

The proposed modified Kalman Filter and K-means clustering-based robot steering protocol offers a significant advancement in mobile robot navigation. By combining accurate state estimation with efficient path planning and dynamic obstacle avoidance, this protocol addresses the limitations of traditional methods like dead reckoning and standalone GPS. The comparative analysis with ground truth data underscores its accuracy and reliability, making it a promising solution for a wide range of autonomous navigation applications. Future work could explore further optimizations of the EKF to handle more complex non-linearities and reduce computational overhead. Incorporating advanced sensor fusion techniques could enhance the robustness and accuracy of state estimation. Combining K-means clustering with other machine learning techniques, such as reinforcement learning, could provide even more adaptive and intelligent path planning capabilities. Extensive field testing in diverse environments, including both indoor and outdoor settings, would provide valuable insights into the practical challenges and performance of the proposed protocol. Testing in environments with varying levels of complexity and dynamics would help refine and validate the approach.

REFERENCES

[1] R. S. Shokendra Dev Verma, Kirti Bhatia, Shalini Bhadola, “A Detailed Overview of Mobile NavigationProtocols,” *Int. J. Innov. Res. Comput. Commun. Eng.*, vol. 10, no. 6, pp. 5678–5684, 2022.

[2] S. B. S. Shokendra Dev Verma Kirti Bhatia, “Design and Analysis of Mobile Locomation Approach,” *Int. J. ORANGE Technol.*, vol. 4, no. 6, pp. 110–117, 2022.

[3] B. Alsadik and S. Karam, “The simultaneous localization and mapping (SLAM)-An overview,” *J. Appl. Sci. Technol. Trends*, vol. 2, no. 02, pp. 147–158, 2021.

[4] U. Steinhoff and B. Schiele, “Dead reckoning from the pocket-an experimental study,” in *2010 IEEE international conference on pervasive computing and communications (PerCom)*, 2010, pp. 162–170.

[5] M. Adamkiewicz *et al.*, “Vision-only robot navigation in a neural radiance world,” *IEEE Robot. Autom. Lett.*, vol. 7, no. 2, pp. 4606–4613, 2022.

[6] Y. Liu, X. Ma, L. Shu, G. P. Hancke, and A. M. Abu-Mahfouz, “From industry 4.0 to agriculture 4.0: Current status, enabling technologies, and research challenges,” *IEEE Trans. Ind. informatics*, vol. 17, no. 6, pp. 4322–4334, 2020.

[7] P. M. Blok, K. van Boheemen, F. K. van Evert, J. IJsselmuiden, and G.-H. Kim, “Robot navigation in orchards with localization based on Particle filter and Kalman filter,” *Comput. Electron. Agric.*, vol. 157, pp. 261–269, 2019.

[8] J. Li, W. Li, and W. Zhang, “Path Planning of UAV Navigation Mark Inspection Using a K-means Clustering ACA,” *J. Mar. Sci. Technol.*, vol. 31, no. 3, p. 10, 2023.

[9] S. Thrun, W. Burgard, and D. Fox, “Probabilistic Robotics (Massachusetts.” The MIT Press, 2005.

[10] M. W. M. G. Dissanayake, P. Newman, S. Clark, H. F. Durrant-Whyte, and M. Csorba, “A solution to the simultaneous localization and map building (SLAM) problem,” *IEEE Trans. Robot. Autom.*, vol. 17, no. 3, pp. 229–241, 2001.

[11] R. Kümmerle, G. Grisetti, H. Strasdat, K. Konolige, and W. Burgard, “g 2 o: A general framework for graph optimization,” in *2011 IEEE international conference on robotics and automation*, 2011, pp. 3607–3613.

[12] J. Borenstein, H. R. Everett, and L. Feng, *Navigating mobile robots: Systems and techniques*. AK Peters, Ltd., 1996.

[13] E. Olson, M. R. Walter, S. J. Teller, and J. J. Leonard, “Single-Cluster Spectral Graph Partitioning for Robotics Applications.,” in *Robotics: Science and Systems*, 2005, pp. 265–272.

[14] D. Fox, W. Burgard, and S. Thrun, “Markov localization for mobile robots in dynamic environments,” *J. Artif. Intell. Res.*, vol. 11, pp. 391–427, 1999.

[15] A. Doucet, N. De Freitas, and N. J. Gordon, *Sequential Monte Carlo methods in practice*, vol. 1, no. 2. Springer, 2001.

[16] H. Durrant-Whyte and T. Bailey, “Simultaneous localization and mapping: part I,” *IEEE Robot. Autom. Mag.*, vol. 13, no. 2, pp. 99–110, 2006.