Journal of Mechatronics, Electrical Power, and Vehicular Technology 16 (2025) 69-83

Journal of Mechatronics, Electrical Power, and Vehicular Technology

e-ISSN: 2088-6985
p-ISSN: 2087-3379

mev.brin.go.id

Review of Kalman filter variants for SLAM in mobile robotics with linearization and covariance initialization

Muhammad Haniff Gusrial [a], Nur Aqilah Othman [a, *], Hamzah Ahmad [a], Mohd Hasnun Arif Hassan [b]

[a] *Faculty of Electrical and Electronics Engineering Technology*
Universiti Malaysia Pahang Al-Sultan Abdullah, 26600 Pekan, Pahang, Malaysia

[b] *Faculty of Mechanical and Automotive Engineering Technology*
Universiti Malaysia Pahang Al-Sultan Abdullah, 26600 Pekan, Pahang, Malaysia

Abstract

Simultaneous localization and mapping (SLAM) has become a foundational concept in robotics navigation which enabling autonomous systems to build maps of unknown environments while estimating their own position. This article aims to provide a comprehensive review of the SLAM concept in the context of mobile robotics navigation by focusing on theoretical principles, estimation problems, algorithms involved, and related applications. The existing literature is systematically analyzed and classified based on three main perspectives of navigation, which are localization, mapping, and path planning. Particular attention is given to Kalman filters and their variants in SLAM-based systems, along with crucial consideration of the linearization and covariance initialization. This article identifies the strengths and limitations of current SLAM approaches. Therefore, this article concludes by outlining research gaps and recommending directions for future exploration, with the intention of serving as a foundation for continued innovation in SLAM-based robotic navigation systems.

Keywords: simultaneous localization and mapping; mobile robot navigation; Kalman filter initialization; extended Kalman filter; unscented Kalman filter; covariance matrix; nonlinear system linearization.

I. Introduction

Robotics is a widely known field nowadays due to its demand not only in manufacturing work, but also in several branches such as medical [1], welding industries [2], education [3], space exploration, search and rescue operations, and many more [4]. The term 'robotic' can be defined as one of the branches related to computing and engineering that involves the concept, design, manufacture, and operation of a robot to assist humans [5]. Mobile robot, which is a member of the wheeled robot family, have become one of the research areas that emerged in the robotic field, which has been rapidly evolving with the recent demands. Mobile robot is widely used as it is able to perform many possibilities. Commonly, mobile robots act as service robots to help humans in their daily tasks. Currently, three main focuses in mobile robot study are ground robot, aerial robot (air), and underwater robot, with a variety of physical designs, sensors, and algorithms [6].

Mobile robots need to model an environment in order to perform their mission, especially the

* Corresponding Author. nuraqilah@umpsa.edu.my (N. A. Othman)

https://doi.org/10.55981/j.mev.2025.1096

Received 4 December 2024; 1st revision 15 June 2025; 2nd revision 19 June 2025; 3rd revision 23 June 2025; accepted 25 June 2025; available online 30 July 2025; published 31 July 2025

How to Cite: M. H. Gusrial *et al.*, "Review of Kalman filter variants for SLAM in mobile robotics with linearization and covariance initialization," *Journal of Mechatronics, Electrical Power, and Vehicular Technology*, vol. 16, no. 1, pp. 69-83, July, 2025.

autonomous navigation using sensors [4]. These applications are often related to the main navigation categories such as localization, mapping, and path planning. The applications are using the most commonly used method by researchers, which is simultaneous localization and mapping, or SLAM, in their working principle. SLAM is a well-known method for its various functionality and applicability that was first developed by Michael Csorba in 1997 [7]. Due to the existence of errors during navigation procedures, an algorithm is needed in order to filter out the non-related data or so-called errors for producing an accurate result. Common filters in SLAM-based approaches are the Kalman filter with its extensions. The filter itself has the ability to initialize and perform linearization (with its extensions) in order for an accurate and reliable result to be presented.

The remaining sections of this article are structured as follows: Section II presents navigation strategies used in mobile robotics. Section III provides an overview of simultaneous localization and mapping (SLAM). Section IV discusses the application of Kalman filter-based approaches in SLAM. Section V focuses on linearization techniques. Section VI addresses initialization via initial covariance in SLAM. Finally, the conclusion is presented in Section VII.

II. Navigation Strategies for Mobile Robots

Navigation is one of the famous applications that involves the use of a mobile robot. Navigation can be classified into three different categories, which are based on sub-problems related to navigation, such as localization, mapping, and path planning [8]. The overall concept of navigation is as illustrated in Figure 1 and discussed throughout this article. Localization, which is one of the navigation branches, is a process for a robot to estimate its location by considering all objects within its environment [9][10]. Mapping is the capability of a robot to represent the encountered features, such as landmarks, obstacles, and other relevant features that may exist in an environment by creating a map [9][11]. A robot that needs to find the best path or a collision-free and safe path for a specific outcome can be defined under path planning, or sometimes called trajectory planning [12][13]. The process of mapping and self-localization (localization) of the robot can be done simultaneously, which is simultaneous localization and mapping or SLAM using the help of sensors [14]. SLAM was developed by a well-known researcher, Michael Csorba, in 1997, where he concluded that the errors from the vehicle and the estimated map have a correlation between them, and it has become the fundamental theory for the implementation of this method [7].

A. Localization

Navigation from a localization point of view is successfully performed by following the four phases, which are perception, localization, cognition, and motion control. In the perception phase, data from the sensor's interpretations will be extracted by the robot, and the robot's current location in the environment will be estimated using the sensor's data in the localization phase. With the estimated position, the robot can plan the right steps to reach its goal in the cognition phase. The motor outputs are then modified to follow the desired trajectory in the motion control phase.

Problems related to localization can be simplified into three sections, which are position tracking, global localization, and the kidnapped robot. Position tracking is where a robot knows its initial position, but due to the position uncertainties, the robot cannot localize itself. As for global localization, the robot has no idea of its initial position. Similar to that, the kidnapped robot problem is where the robot is lost in an unknown location and can only be solved if the robot is autonomously set [10].

Some of the approaches that have been researched to solve the localization problems are the probabilistic approach, the radio-frequency identification (RFID) approach, and several evolutionary approaches. The most common probabilistic approaches are Markov and Kalman filter localization, where the theorem of total probability (in the prediction phase) and Bayes' rule (in the update phase) is used [15]. In a simple word, Markov localized a robot in an unknown environment by updating its possible position probability and continuously estimating, correcting, as well as updating the current position by incorporating sensor information to predict the new position by applying Bayes' rule. Similar to Markov, the Kalman filter applies the prediction and update phase, but the initial position is initially known, which makes the Kalman filter very efficient in solving the position tracking problem, but not suitable for solving global localization and the kidnapped robot problem. Position tracking problem can also be solved using an RFID approach. RFID uses a set of tags that are arranged in an environment to provide the location information that will be extracted by the robot when it passes through to help in determining the current position [10][16].

B. Mapping

Mapping in navigation perspective can be classified into three different techniques, such as occupancy grid

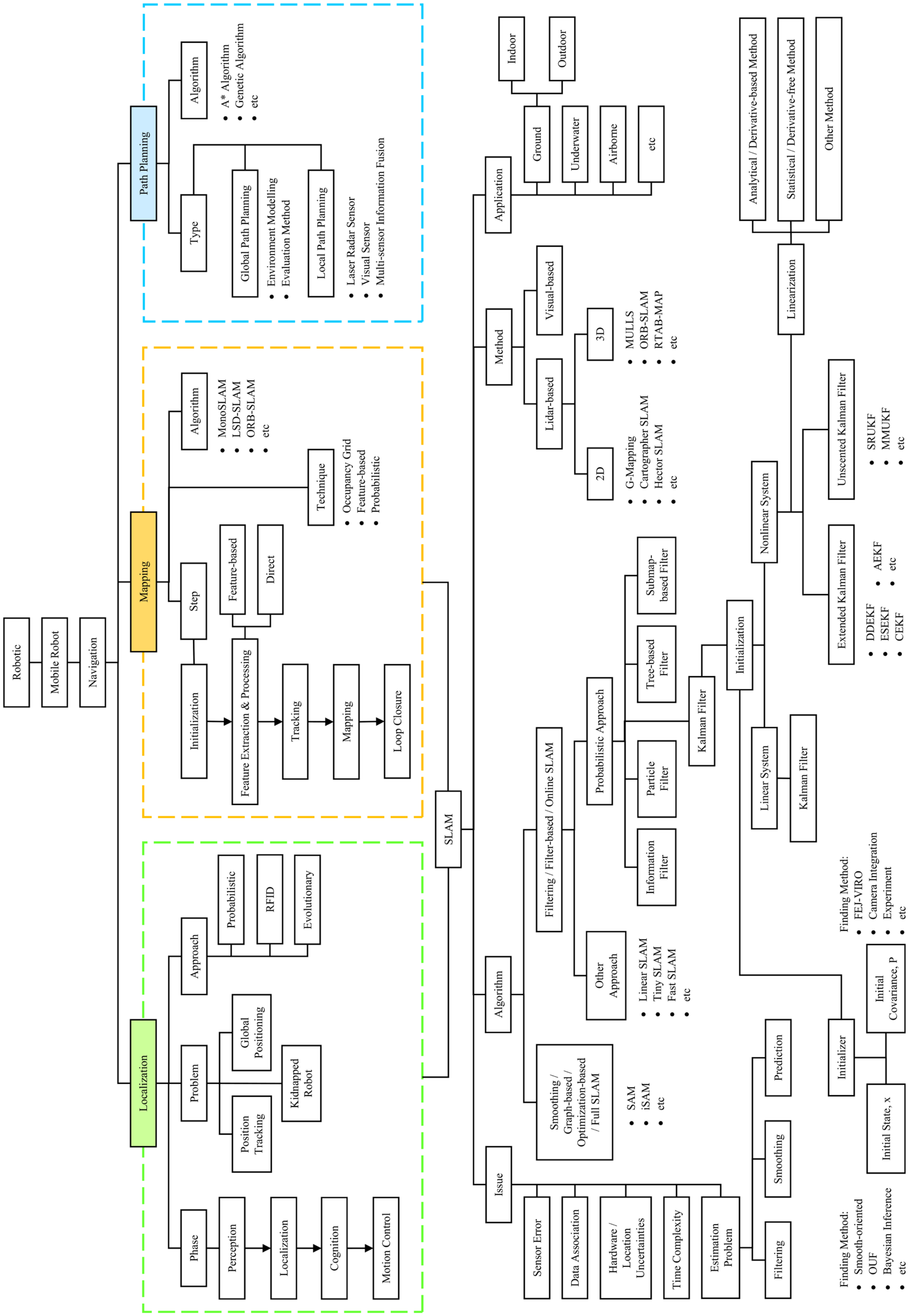

Figure 1. Summary mind map.

mapping, feature-based mapping, and probabilistic mapping, with several famous visual algorithms that have been presented by previous researchers. Examples of algorithms are monocular SLAM (MonoSLAM), parallel tracking and mapping (PTAM), dense tracking and mapping (DTAM), large-scale direct monocular SLAM (LSD-SLAM), oriented features from accelerated segment test (FAST) and rotated BRIEF-SLAM (ORB-SLAM), convolutional neural network SLAM (CNN-SLAM), etc. Like many mapping implementations by previous researchers, it always comes with the implementation of localization through SLAM due to its nature to create a map for the purpose of locating the robot or any obstacle based on the detected location through localizing in an environment.

In most of the mapping techniques, there are five fundamental steps that are involved, which are initialization, feature extraction and processing, tracking, mapping, and loop closure. Starting with the initialization step, the data is acquired from the sensors, and it is crucial for the sensors to be calibrated to ensure distortion-free data. After initialization, the distinct features such as corners, edges, or high-contrast points are identified for feature matching and reconstruction using a feature-based method, and in the direct method, the raw data is used directly for estimation and mapping purposes. In the tracking step, the extracted data is used for estimating the location and orientation of the robot in each frame by comparing the features within the current and previous frames. After the comparison is completed, the selected information will be presented in a map. Finally, a loop closure is needed to correct the errors during the previous steps by continuously analyzing and comparing the current data with the previous data. Any revisiting actions will be updated to help in correcting any drifts to ensure an accurate map is produced [11].

C. Path planning

Path planning is important for obtaining the optimal path, especially when a robot operates in a free space environment where numerous possible paths can be taken [17]. Path planning in navigation perspective acts as the bridge between the perception and motion control phases, [18] where it is generally divided into two different types of path planning according to the environment information. There are global path planning and local path planning for known and unknown environments, respectively [19]. A known environment is when the obstacle in an environment is fixed.

Global path planning, also called offline or static path planning where the robot is aware of the environment and obstacles around it and can reach the desired destination with prior information due to the static environment [19]. It requires the robot to build a global map and understand it using a ‘search and seek’ algorithm in order for an optimal path to be obtained [18]. The method related to global path planning is the environment modelling and evaluation method of path quality [20]. Environment modelling is where the perceived environment is organized and utilized to help the robot operate efficiently. Several approaches in environment modelling are the grid method (GM), topological method (TM), geometric characteristic method (GCM), and mixed representation (MR) method. The evaluation method of path quality is a method that evaluates the path planning by seven different evaluations explained by a previous researcher [18].

Local path planning, also called online or dynamic path planning, on the other hand, is where the robot is partially known or completely unknown to the environment. This type of path planning is able to perform real-time monitoring with good flexibility, but the path planned is not guaranteed to be optimal. Several methods that incorporate the use of sensors in local path planning, such as laser radar sensor (LRS), visual sensor (VS), and multi-sensor information fusion (MIF), act as the environment detection device [18].

The algorithms involved in path planning are divided into several categories due to some researcher having their own classifications depending on their own interpretations based on the environment dynamics, principles, and mechanisms. If the dynamic of an environment becomes its benchmark, then the algorithm will be classified into three groups, which are classic, heuristic, and evolutionary approaches [19]. If separated by principle and mechanisms, it can be classified into four categories, such as reactive computing, soft computing, C-space search, and optimal control, which can then be further classified into more sub-categories in detail [17]. Some of the algorithms that are commonly used for path planning are A* algorithm and genetic algorithm (GA) [18][19][20].

D. Application in navigation

There are many robots is being used by researchers to performed navigation such as Jetank AI mobile robot with mounted odometry and camera [21], Kian-I mobile robot with mounted ultrasonic sensors and infrared sensors [22], Jackal mobile robot with mounted inertial measurement unit (IMU) sensor [23], ambulance robot (Ambubot) with mounted IMU

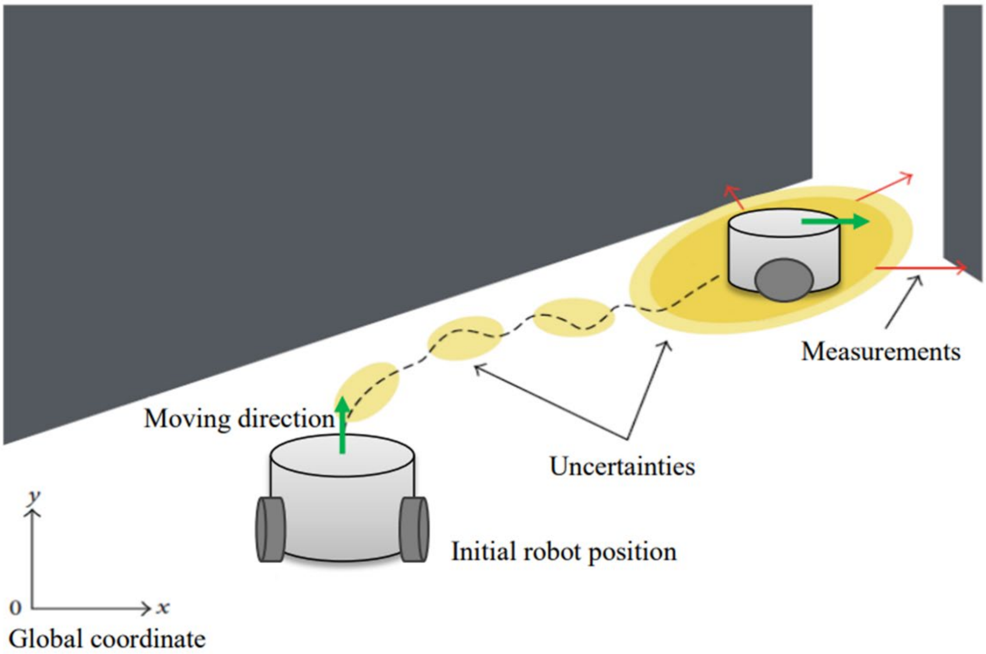

Figure 2. SLAM concept in general.

sensor and outdoor global positioning system (GPS) [24] as well as custom mobile robot with multiple sensors integration [25][26].

III. Simultaneous Localization and Mapping

Simultaneous localization and mapping (SLAM) is a process in which a mobile robot explores an unknown environment while simultaneously constructing a map and estimating its own position within that map [27][28]. This is done without any prior knowledge of the robot's location or the structure of the environment. SLAM is formerly known as concurrent mapping and localization (CML) [29], which refers to the ability of a robot to construct a map in both artificial and real environments while continuously estimating its position as it observes its surroundings [30]. This process relies on onboard proprioceptive sensors, such as odometers, and exteroceptive sensors such as laser scanners. By using the data provided by these sensors and applying suitable algorithms to convert the information from the robot's reference frame, the robot can estimate its current position along with the positions of detected features or landmarks. However, the SLAM process is susceptible to various sources of error or uncertainties, including sensor noise, modelling inaccuracies, system limitations, and algorithmic weaknesses as portrayed in Figure 2.

A. SLAM system architecture

SLAM in navigation systems involves four stages, which are sensors integration, front-end or treatment stage, back-end or processing stage with feedback that will act as a closed loop between these two stages (front-end and back-end), and the final stage produces the result as shown in Figure 3. The first stage is where sensors receive sensory raw data collected from the environment. The commonly used sensors are odometry, IMU, light detection and ranging (LiDAR), and laser sensor. Generally, odometry provides the estimation of the motion, IMU presents the orientation and acceleration data, and LiDAR, as well as a laser sensor, generate a precise distance measurement to the surrounding obstacles [31][32].

In the front-end stage, where the data from sensors is being given specific treatment as required by the objectives of the research, which commonly involves

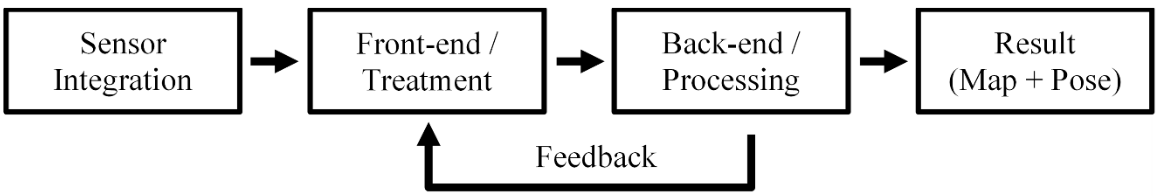

Figure 3. SLAM system architecture.

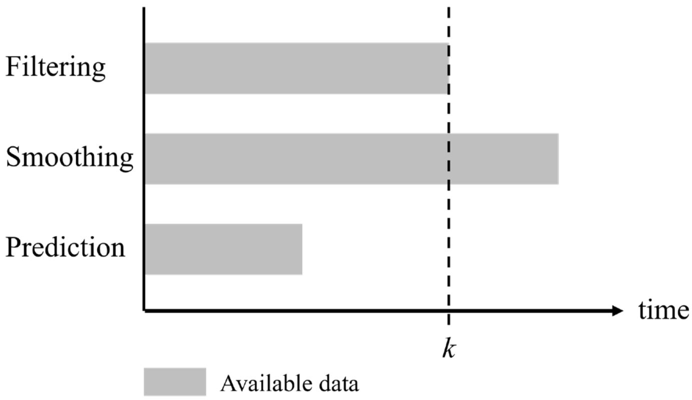

Figure 4. Estimation problem.

feature extraction, data association, and outlier rejection to filter out the unnecessary information. Next, the process is proceeded with the back-end stage where all the data and treatment are being evaluated and refined, while the feedback mechanisms will provide the system's improvement to ensure the result is in an optimal state. Finally, the reliable map and pose in trajectory form can be estimated accurately after all of the stages are complete [33][34].

B. Estimation problem in SLAM

In order to further understand SLAM, it is better to explore the estimation problem. Estimation refers to the process of the system incorporating the initial parameters known as initializers and control inputs to begin the prediction phase. Extracting the desired information from the observation by systematically utilizing the disturbance, uncertainties, errors, and prior knowledge regarding the system, as well as the predicted data, is used to continue the estimation process in the update phase. The result produced is used again by the prediction phase, and the process will be repeated according to the desired needs [7]. There are mainly three types of estimation problems, which are filtering, smoothing, and prediction. Filtering is when an estimation is done using the last measured data (available data at time k). Smoothing is where the estimation is done within the collected data range (range of data at time k) [35]. Prediction is where the estimation is done beyond the available data, where future estimation is needed (beyond the range of data at time k) [36]. These estimations at time k are illustrated in Figure 4.

A successful navigation would require a mobile robot to understand its environment and have a reliable location tracking within the environment [33]. Robots, including mobile robots, tend to have the most common problem when undergoing navigation, which is with the sensors. Sensors help to gain information in creating a model of the environment that the mobile robot encounters. During exploration, sensors tend to produce errors, which cause an emergence of uncertainties due to sensors' weaknesses, such as low resolution, reliability, and others [4]. This inaccurate sensory information phenomenon can sometimes occur due to the raw data information collected by the sensor is not used directly, but it is processed by a mathematical model that translates it into some meaningful information [33].

C. SLAM algorithm

The SLAM algorithm can be divided into graph-based SLAM and filter-based SLAM [37], as shown in Table 1. Graph-based SLAM or optimization-based uses a graph in representing the robot's pose at each time occurrence as nodes, with the edges representing the spatial constraint between each pose [38]. This algorithm is also called as smoothing approach or full SLAM because its style of estimation uses using full set of measurements, which causes the computational cost to be relatively high [33][39]. Some examples of this type are square root smoothing and mapping (SAM), incremental smoothing and mapping (iSAM), and many more [26].

Table 1.
SLAM algorithm comparison.

Criteria	Graph-based SLAM	Filter-based SLAM
Estimation method	Optimization based	Recursive based (filtering)
Measurement usage	Full set of measurement	Current and previous measurement
Computational cost	High	Moderate to low
Initialization sensitivity	Low	High
Common uses	Offline mapping	Real-time navigation
Example	SAM, iSAM, etc	Kalman filter, IF, PF, etc

On the other hand, filter-based SLAM, online SLAM, or sometimes called as filtering approach, uses the obtained robot's current pose data with the map to estimate the new pose and position at the next time occurrence [26]. This type is divided into two categories, which are probabilistic approaches and other approaches. Probabilistic approaches are usually based on the Bayesian rule. The most common examples of this approach are the Kalman filter, information filter (IF), particle filter (PF), submap-based filter, and tree-based filter [33]. Other approaches are FastSLAM, tinySLAM, linear SLAM, and many more [33][37].

The SLAM algorithm can be used in various sensor-based methods, such as lidar-based SLAM and camera-based or visual SLAM. Lidar-based SLAM incorporated the use of LiDAR sensors in the system, while camera-based SLAM used cameras as its data collection tools. LiDAR-based SLAM is capable of handling two-dimensional (2D) and three-dimensional (3D) systems. Some examples of 2D systems are Grisetti's mapping (GMapping) [40], Cartographer SLAM [41], and Hector SLAM [37], while 3D system is multi-metric linear least square (MULLS) [37], LiDAR odometry and mapping (LOAM) [38], lightweight and ground-optimized LOAM (LeGO-LOAM) [34], and LiDAR inertial odometry via smoothing and mapping (LIO-SAM) [32]. GMapping and Cartographer SLAM are very well known in the 2D SLAM navigation system, especially in mobile robots, as it is successfully become examples for loop closing localization with mapping techniques [42] and generated occupancy grid-based map as a result representation method [41], respectively. For visual SLAM, the well-known examples are ORB-SLAM [30], real-time appearance-based mapping (RTAB-Map), Low Dimensional SLAM (L-SLAM), LSD-SLAM, and many more [37].

Table 2.
SLAM application.

Environment	Application	Example of SLAM method
Indoor (ground)	Multi-robot, trajectory estimation	GMapping, Hector SLAM, RTAB-Map, ORB-SLAM
Outdoor (ground)	Autonomous robot	EKF SLAM, GMapping
Underwater	Underwater mapping	3D SLAM
Airborne	Aircraft navigation	Radio SLAM
Simulated	Robot flocks, path planner, delivery system, cooperative control, etc	Any SLAM method

D. SLAM application

Many applications related to SLAM has been done by researchers in various environments, including ground, which are indoor [28][37][43], outdoor [26][44], underwater [45][46], and airborne [47]. There are also many researchers implemented SLAM in simulation for different purposes, such as robot flocks [48], path planning [40], delivery system [49] cooperative control of mobile robots [50], etc. A robot operating system (ROS) -based platform mobile robot, such as Turtlebot3 burger, [51] is also implementing SLAM in order to perform the navigation procedures such as environment mapping, covariance determination, and many more, as shown in Table 2.

IV. Kalman Filter-Based SLAM Approaches

A. Kalman filter for linear systems

The most common filtering algorithm in the robotics field, especially in SLAM implementation, would be the Kalman filter due to its simplicity in implementation [33]. Kalman filter or also known as the prediction-update algorithm, uses the dynamic state data to predict the next state data and eventually produces updated data through a numerical integration technique, which is Euler's method or Runge-Kutta [52]. This algorithm is also a recursive algorithm that estimates the state variable by integrating all available measurements used in the prediction and estimation of robot tracking and positioning (also called localization), where the robot is in an unfamiliar map with uncertainties, and where a mathematical model of the system is used in the state estimation [53]. Optimally, this algorithm works in a linear environment where the state equation of the system can predict the state value to update the new state in the tracking procedure by applying the observation model [54].

Kalman filter is formulated based on a linear state space system in the form of state vector, x, which is an equation that defines position and velocity expression, where many derivations have been done using kinematic equations and observation model, z. All variables and notations have been listed as in Table 3. These expressions were rewritten in terms of the state vector x_k and observation model z_{k-1}. Noted that the noise vector, w, and measurement noise vector, v is neglected in the following steps (during applying the Kalman filter) because its value is assumed to be random errors with zero mean. The prediction-update task can be performed at each time step, which is denoted as k. It is also important to note that the prediction process will always begin by applying initializers into the algorithm, which are the initial state estimate, $\hat{x}_0$, and initial state error covariance matrix or also called the initial covariance matrix, P_0 [54][55].

The prediction process begins with predicting the state vector, $\hat{x}_{k|k-1}$ and state error covariance matrix, $P_{k|k-1}$ where the previous estimated state vector, $\hat{x}_{k-1}$ or $\hat{x}_0$ (for the first time step), input vector, u, dynamic system matrices (state transition matrix, F and control matrix, G), previous estimated state error covariance matrix, P_{k-1} or P_0 (for the first time step), and process noise covariance matrix, Q, are also included. The Kalman gain matrix, K_k, is calculated once the predicted value has been obtained, where the output-defined matrix or observation matrix, H, for describing the observation and measurement noise covariance, R, is included [54].

The updated state vector, $\hat{x}_{k|k}$, and updated state error covariance, $P_{k|k}$ with the incorporation of the identity matrix, I [54][56]. The predicted output, $H_k\hat{x}_{k|k-1}$ is sometime denoted as $\hat{y}_{k|k-1}$. The subscript $k|k-1$ is a notation for a state at discrete time k using its previous state at discrete time $k-1$ while the subscript $k|k$ is a notation for a state at discrete time k (first k refers to update step) using its previous state at discrete time k (second k refers to prediction step).

Table 3.
Variables and notations included in the Kalman filter implementation.

Variable/ notation	Description	Variable/ notation	Description
k	Time step (at time k)	G	Control matrix
k-1	Time step (at time k-1)	H	Observation/ output-defined matrix
x	State vector	I	Identity matrix
$\hat{x}_0$	Initial state vector	K_k	Kalman gain matrix
z / y	Observation/ measurement model	P	State covariance matrix
u	Input vector	P_0	Initial state covariance matrix
w	Process noise vector	Q	Process noise covariance matrix
v	Measurement noise vector	R	Measurement noise covariance matrix
F	Transition matrix	$\nabla f/\nabla h$	Jacobian matrix

B. Extended Kalman filter for nonlinear systems

In a nonlinear environment, a normal Kalman filter would not be the best choice to rely on due to errors that emerged because of its poor tracking effect [27], absence of a loop closure (ability to recognize previously visited locations for state estimation), and data association problem [33]. One of the solutions for solving those problems would be to introduce extended Kalman filter (EKF) which is one of the extensions of Kalman filter that has been the choices for many applications due to its low computational cost [57]. This filter is introduced to solve a nonlinear system problem such as robot tracking and positioning [54] as well as its ability to build map with the accuracy of above 90 % [57]. EKF works by linearizing the nonlinear system into a linear system (linearization) by applying a first-order Taylor series expansion [27][58]. Linearization and sampling approximation are involved in this algorithm can fuse data from sensors to estimate the robot pose in three dimensions (3D) [59].

EKF includes robot pose (position and orientation) with the landmark location to estimate the state vector [58]. The pose and landmark mean value are stored in the form of a column matrix $(n \times 1)$ and the covariance value in the form of a square matrix $(n \times n)$. Robot pose is represented as (x, y, θ) where x and y are the position in 2D axes with θ is the orientation angle, while landmark, on the other hand, is represented as (r, θ) [4]. EKF consists of a similar mathematical structure to the normal Kalman filter, but with a slight difference. Due to the nature of EKF in handling nonlinear systems, the state vector and observation model will be based on a nonlinear model state equation. Same as Kalman filter, process noise vector, w, and measurement noise vector, v, are also neglected in the following steps, and it is denoted as zero [54].

In implementing EKF, the nonlinear equation is linearized using Taylor series expansion by incorporating Jacobian matrices denoted as ∇f. Due to the linearization, the nonlinear functions f and h will be directly used for the predicted state vector and observation model. In the prediction of the state vector, $\hat{x}_{k|k-1}$, and state error covariance matrix, $P_{k|k-1}$, the elements included are similar to the Kalman filter equations, the main difference is in the existence of Jacobian matrices ∇f. After this step, the EKF algorithm is same as the normal Kalman filter, which is continued with the Kalman gain, and updated equations with the existence of Jacobian matrices, ∇h [54]. The Kalman filters and EKF estimation process are summarized as shown in Figure 5.

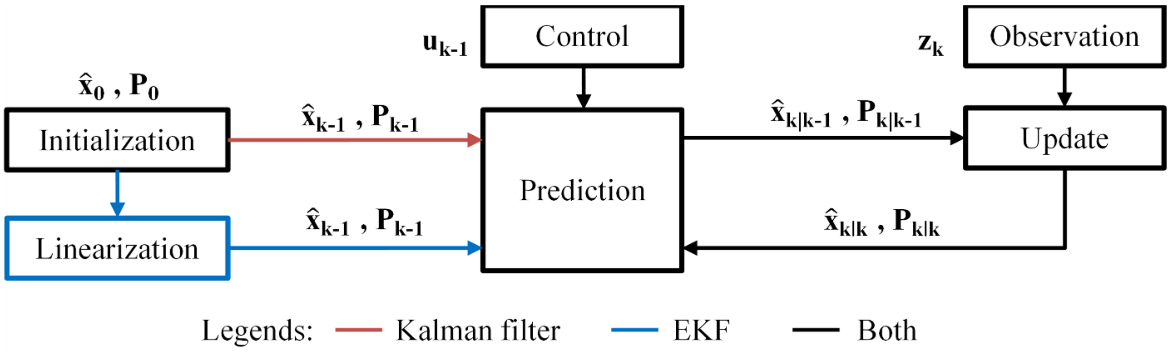

Figure 5. Estimation process in Kalman filter and EKF.

The linearized EKF system can sometimes present a poor nonlinear function due to some error that may have been produced during the linearization because of EKF is only suitable for solving Gaussian problems [26][60]. Those errors can sometimes be very obvious due to the extremely nonlinear system, which can lead to filter deviation [61]. The fact that EKF's known for its simple implementation operation with white Gaussian noise, limited in large-scale navigation and other factors, many researchers have proposed several methods that are developed from the principle of EKF to overcome the weaknesses drawn from this algorithm [62]. For example, a decorrelated distributed extended Kalman filter (DDEKF) is proposed to ensure the robot's orientation can be revised during estimation to provide a consistent and accurate system by incorporating a magnetic compass sensor as an input for the measurement vector [26].

Other researchers also suggested using error-state extended Kalman filter (ES EKF) as it is known for its ability to perform real-time localization in the global navigation satellite system (GNSS) denied environment, such as an indoor environment [63], better than normal EKF [23]. Compressed extended Kalman filter (CEKF) is also being introduced as the improved version of EKF, which divides the state vector into passive and active parts, where in the local area, only the active part will be updated, but once the robot moves, both parts (passive and active) will be updated [62]. There is also an algorithm that specifically targets improving the system's accuracy based on the odometric parameters known as the augmented extended Kalman filter (AEKF) [62].

C. Unscented Kalman filter for nonlinear systems

Other than EKF, the unscented Kalman filter (UKF) is also commonly used in nonlinear systems. UKF focused on using sampling and weight variables to perform the prediction and estimation in SLAM [27]. UKF has the ability to minimize the linearization error by applying the unscented transform (UT) to the approximation due to its consideration of high-order terms and avoiding the derivation of Jacobian matrices [62][64]. When a nonlinear map is changed, UKF basically helps to determine which transformation allows approximation of the covariance and mean of random vectors of length n by computing σ-points or also known as $2n+1$ points [61][64].

Some research clearly shows that UKF is precise when encountering Gaussian noises during approximations up to third order and precise up to second order in terms of covariance estimation, while EKF is only able to precisely accomplish those estimates in the first order only [61][64]. It is also clearly shown that UKF is more accurate than EKF in robot tracking applications [27] and robot position reconstruction, due to its approximation properties are far superior to EKF [64].

Similar to EKF, UKF also tends to draw some disadvantages in its implementation. Accuracy in UKF tends to reduce when encountering large localization coverage and high velocity navigation [61]. Some researchers even concluded that UKF is slower than EKF in a nonlinear system [33]. In order to achieve perfect estimation, researchers have fused several methods with UKF to obtain an ideal result. Square root UKF (SRUKF) has been introduced to improve numerical stability, Masreliez-Martin UKF (MMUKF) is introduced to improve low tracking accuracy [27], and many more. Even though EKF seems not to be perfectly accurate compared to other algorithms, including UKF, it is still commonly used for SLAM-related navigation applications such as positioning [65][66], map building [4][57] and many more.

V. Linearization Techniques in SLAM

EKF and UKF are widely known over other filters for their ability to perform nonlinear state estimation. Basically, both filters applied a specific linearization method to linearize a nonlinear system. As for EKF, analytical linearization or derivative-based method is used, while statistical linearization or derivative-free method is used in UKF [67][68]. There are also some other methods that have been proposed by previous researchers.

Analytical or functional linearization specifically for EKF is based on the first order of the Taylor series expansion to linearize the nonlinear motion and observation model [69]. Previous researchers have developed an evolutionary method, which is the mean EKF (MEKF) that is based on the normal EKF to lessen the linearization error by replacing new formula that included real state vector, x_k approach in the Jacobian matrix of the observation function, that will eventually increase the accuracy of the linear approximation [70]. All steps are continued as normal EKF.

Statistical linearization is known as an alternative to the analytical or functional linearization, as it is a Jacobian-free method with no derivatives required. This method is also being used due to the Jacobian may not exist in all nonlinear applications, which can be difficult to determine analytically, and deriving it numerically may cause problems [71]. This method also has been used in cubature Kalman filter (CKF) [72], quadrature Kalman filter (QKF) [73], and UKF itself [74].

Both methods have their pros and cons in certain situations. Previous researchers have compared both methods with polynomial and trigonometric functions in order for the commonly encountered nonlinearities in estimation problems to be identified in terms of true mean and variance. The tests are run assuming the random signal to be Gaussian with known mean and standard deviation. In summary, the result of the test is not significantly winning on one side only, which can be seen as shown in Table 4 with the Monte Carlo method as its reference [75].

Other linearization methods that have been proposed by other researchers are point-to-point linearization, which focuses on linearizing a nonlinear trajectory for representing the displacement terms of an object position in different frames of an image, where a fuzzy logic algorithm is used [76]. There is also a combination of EKF with discrete wavelet transform (DWT), which is called hybrid linearization. This method has been used in the medical field specifically to examine the condition of the human heart, where DWT helps in dividing and analyzing the electrocardiogram (ECG) continuous-time signal into different frequencies in order to denoise the signal [77].

VI. Initialization via Initial Covariance in SLAM

In order to implement any Kalman filter technique, initialization is an important step that needs to be looked into. A variable called covariance needs to be determined due to its existence as one of the initializers for implementing the algorithm. Covariance can be defined mathematically as the relationship between two (2) random variables (x and y) that are dependent on one another. Covariance may exist in the range from $-\infty$ to $+\infty$ where it includes the mean of variable x ($\bar{x}$) and mean of variable y ($\bar{y}$) [78]. Those values will be in the form of a matrix to illustrate the covariance matrix. The covariance value will be located at the diagonal of the matrix, with its off-diagonal denoted as zero due to diagonalization [79][80].

In EKF implementation, the initial covariance, P_0, is an important value that needs to be determined to preserve the algorithm's efficiency and avoid major reduction in the performance when inaccurate statistical analysis is implemented. Other than that, as a recursive algorithm, it is always assumed the initial variable to be known a priori, whereas in a real scenario, this does not usually happen, which can contribute to undiscovered bias due to the initialization error through the recursion. If it happened, it may produce a transient period, as shown in Figure 6, in which the algorithm produces unreliable estimates until sufficient measurements accumulate. During this transient phase, the Kalman gain fails to provide an optimal balance between model predictions and measurement updates, thus compromising both the optimality and unbiasedness of the Kalman filter. Such transients are

Table 4.
Analytical and statistical linearization method comparison [75].

Nonlinear function		Mean cases	Result
Polynomial	$y = x^k$	Zero mean	Statistical linearization provides good estimation for mean and variance estimate error
Trigonometric	$y = sin(x)$	Zero mean	Statistical linearization provides good accuracy for variance estimate error
	$y = sin(z)$	Non-zero mean	Statistical linearization provides good accuracy for mean estimate error while analytical linearization provides good accuracy for variance estimate error

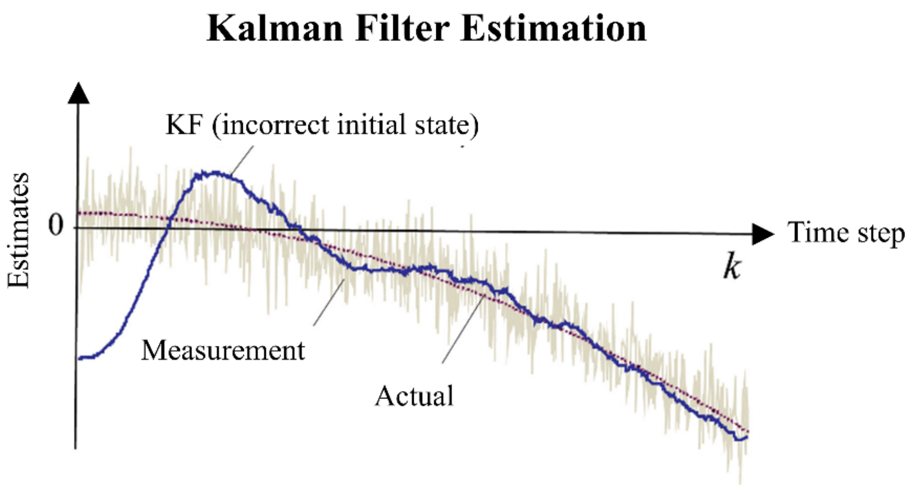

Figure 6. Transient period (shown by incorrect initial state) along the time step, k [55].

particularly problematic in applications requiring rapid convergence or where measurements are sampled at large intervals, as initial errors, although eventually diminishing if the filter is stable, may lead to prolonged periods of unsatisfactory performance [55]. This problem can be solved by controlling the initial covariance, P_0, which can be set according to the confidence level of the mean value used [81].

Some researchers had already proposed several approaches for determining the initial covariance. Some examples that have been done previously are smoother-oriented initialization (step-forward strategy) [82], linear optimal unbiased filter (OUF) [83], and Bayesian inference technique, where those have their drawback in terms of delay estimation and are limited to a linear system only [55]. Other than that, the first-estimate Jacobian visual-inertial-ranging odometry (FEJ-VIRO) method has been proposed to minimize the localization drift problem by incorporating ultra-wideband (UWB) ranging measurement into the visual-inertial odometry (VIO) framework, by incorporating the initialization process that will estimate the covariance matrix [84]. The integration of a camera can also be part of the covariance approximation by validating the matching accuracy between experiment and simulation camera poses using an image processing algorithm to compare variances and scores of the image produced [85].

Several methods are used in determining the covariance for a system, nowadays not limited to robotic fields. Covariance can be determined for solving the near real-time (NRT) modelling problem of global ionospheric total electron (TEC), which involves the changes of spherical harmonic (SH) between epochs in the astronomy field [86]. Large sparse covariance matrix can be determined by using ℓ_1 penalized covariance estimator with the help of the majorization-minimization (MM) algorithm [87].

In the localization or mapping of robotics-focused research, some researchers used pre-formulated data such as EuRoC public dataset [88], identity matrix [54], or even a random value [89] rather than determining their initial covariance value for the purpose of simplicity. This is not very reliable due to not every system is suitable for using the covariance from the same dataset. This will eventually drive the result to become inaccurate. Therefore, more initialization approaches, especially in experimental approaches are need to be explored in the future, especially for a nonlinear system.

VII. Conclusion

SLAM remains a key component in mobile robotics as it is a very interesting field to be explored. This article reviewed the three main perspectives of SLAM navigation, focusing on mobile robotics through theoretical foundation, estimation problem, algorithms, and applications, along with the relevance of Kalman filters with their nonlinear extensions, linearization, and covariance initialization procedure. Although Kalman filter-based methods are widely used but each of their variants differs due to their individual simplicity, compatibility, and computational efficiency, which is challenging when applied in highly dynamic or uncertain environments. Important implementation factors such as the accuracy of the linearization process and the initialization of the covariance matrix significantly affect the performance and reliability of SLAM. The current literature lacks comprehensive comparisons between various types of SLAM estimation methods with their extensions, especially in filter-based SLAM. This highlights the need for more flexible and adaptive filtering approaches since it has become one of the common SLAM-based approaches explored nowadays. Future research should consider combining classical filtering approaches with data-driven methods, artificial intelligence (AI)-based, machine learning-based, or deep learning-based, and not to forget the impact of linearization and initialization in SLAM-based systems. Hence, improving robustness under various sensing conditions and extending SLAM applications into various engineering domains, such as autonomous vehicle navigation, inspection robotics, smart manufacturing systems, and allowing various fields to be explored, such as technical education, economic development, and human resource matters. Therefore, this article is intended as a stepping stone to guide further research and development in SLAM-based applications with practical and relevant implementations in the future.

Acknowledgements

The author would like to acknowledge Universiti Malaysia Pahang Al-Sultan Abdullah and Malaysian Ministry of Higher Education for the continuous supports under Supervisory Incentive Scheme, FRGS/1/2022/TK07/UMP/02/20 and RDU220113 grant.

Declarations

Author contribution

All authors contributed equally as the main contributor of this paper. All authors read and approved the final paper.

Funding statement

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

Competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Additional information

Reprints and permission: information is available at https://mev.brin.go.id/.

Publisher's Note: National Research and Innovation Agency (BRIN) remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

References

[1] C. Liu, Y. Liu, R. Xie, Z. Li, S. Bai, and Y. Zhao, "The evolution of robotics: research and application progress of dental implant robotic systems," *Int. J. Oral Sci.*, vol. 16, no. 1, p. 28, Apr. 2024.

[2] S. I. Wahidi, S. Oterkus, and E. Oterkus, "Robotic welding techniques in marine structures and production processes: A systematic literature review," *Mar. Struct.*, vol. 95, no. February, p. 103608, May 2024.

[3] F. Ouyang and W. Xu, "The effects of educational robotics in STEM education: a multilevel meta-analysis," *Int. J. STEM Educ.*, vol. 11, no. 1, p. 7, Feb. 2024.

[4] O. Ozisik and S. Yavuz, "Simultaneous localization and mapping with limited sensing using Extended Kalman Filter and Hough transform," *Teh. Vjesn. - Tech. Gaz.*, vol. 23, no. 6, pp. 1731–1738, Dec. 2016.

[5] Chenguang, "Short Note on Robotics," *J. Biomed. Syst. Emerg. Technol.*, vol. 8, no. 2, 2021.

[6] R. G. Prasuna and S. R. Potturu, "Deep reinforcement learning in mobile robotics – a concise review," *Multimed. Tools Appl.*, vol. 83, pp. 70815–70836, Feb. 2024.

[7] 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, Jun. 2001.

[8] I. Lluvia, E. Lazkano, and A. Ansuategi, "Active Mapping and Robot Exploration: A Survey," *Sensors*, vol. 21, no. 7, p. 2445, Apr. 2021.

[9] M.-F. R. Lee and S. H. Yusuf, "Mobile robot navigation using deep reinforcement learning," *Processes*, vol. 10, no. 12, p. 2748, Dec. 2022.

[10] P. K. Panigrahi and S. K. Bisoy, "Localization strategies for autonomous mobile robots: A review," *J. King Saud Univ. - Comput. Inf. Sci.*, vol. 34, no. 8, pp. 6019–6039, Sep. 2022.

[11] B. Singh, P. Kumar, and N. Kaur, "Advancements and challenges in visual-only simultaneous localization and mapping (V-SLAM): A systematic review," *2nd Int. Conf. Sustain. Comput. Smart Syst. ICSCSS 2024 - Proc.*, pp. 116–123, 2024.

[12] A. Loganathan and N. S. Ahmad, "A systematic review on recent advances in autonomous mobile robot navigation," *Eng. Sci. Technol. an Int. J.*, vol. 40, p. 101343, Apr. 2023.

[13] S. M. Nasti and M. A. Chishti, "A review of AI-Enhanced navigation strategies for mobile robots in dynamic environments," in *2024 ASU International Conference in Emerging Technologies for Sustainability and Intelligent Systems (ICETSIS)*, IEEE, Jan. 2024, pp. 1239–1244.

[14] A. M. Barros, M. Michel, Y. Moline, G. Corre, and F. Carrel, "A comprehensive survey of visual SLAM algorithms," *Robotics*, vol. 11, no. 1, p. 24, Feb. 2022.

[15] J. Uhlmann and S. J. Julier, "Gaussianity and the Kalman filter: A simple yet complicated relationship," *J. Cienc. e Ing.*, vol. 14, no. 1, pp. 21–26, Jun. 2022.

[16] M. Schumann and J. Haase, "Development of an RFID-based indoor localization system to support firefighters in burning buildings," in *IECON Proceedings (Industrial Electronics Conference)*, IEEE, Nov. 2024, pp. 1–6.
[17] J. R. Sánchez-Ibáñez, C. J. Pérez-del-Pulgar, and A. García-Cerezo, "Path planning for autonomous mobile robots: A review," *Sensors*, vol. 21, no. 23, p. 7898, Nov. 2021.
[18] L. Liu, X. Wang, X. Yang, H. Liu, J. Li, and P. Wang, "Path planning techniques for mobile robots: Review and prospect," *Expert Syst. Appl.*, vol. 227, p. 120254, Oct. 2023.
[19] P. K. Mohanty, A. K. Singh, A. Kumar, M. K. Mahto, and S. Kundu, "Path planning techniques for mobile robots: A review," in *Proceedings of the 13th International Conference on Soft Computing and Pattern Recognition (SoCPaR 2021)*, Springer, Cham, 2022, pp. 657–667.
[20] H. Y. Zhang, W. M. Lin, and A. X. Chen, "Path planning for the mobile robot: A review," *Symmetry (Basel)*., vol. 10, no. 10, p. 450, 2018.
[21] S. Sellak, Y. Alj, and Y. Salih-Alj, "Monocular visual odometry in mobile robot navigation," in *ISIVC 2024 - Proceedings: 12th IEEE International Symposium on Signal, Image, Video, and Communications*, IEEE, May 2024, pp. 1–6.
[22] A. Abbasi, S. MahmoudZadeh, A. Yazdani, and A. J. Moshayedi, "Feasibility assessment of Kian-I mobile robot for autonomous navigation," *Neural Comput. Appl*., vol. 34, no. 2, pp. 1199–1218, Jan. 2022.
[23] I. Brigadnov, A. Lutonin, and K. Bogdanova, "Error state extended Kalman filter localization for underground mining environments," *Symmetry (Basel)*., vol. 15, no. 2, p. 344, Jan. 2023.
[24] H. Samani and R. Zhu, "Robotic Automated External Defibrillator Ambulance for Emergency Medical Service in Smart Cities," *IEEE Access*, vol. 4, pp. 268–283, 2016.
[25] D. Nemec, V. Šimák, A. Janota, M. Hruboš, and E. Bubeníková, "Precise localization of the mobile wheeled robot using sensor fusion of odometry, visual artificial landmarks and inertial sensors," *Rob. Auton. Syst*., vol. 112, pp. 168–177, Feb. 2019.
[26] F. Pei, M. Zhu, and X. Wu, "A decorrelated distributed EKF-SLAM system for the autonomous navigation of mobile robots," *J. Intell. Robot. Syst*., vol. 98, pp. 819–829, Jun. 2020.
[27] M. Tang, Z. Chen, and F. Yin, "Robot tracking in SLAM with Masreliez-Martin unscented Kalman filter," *Int. J. Control. Autom. Syst*., vol. 18, no. 9, pp. 2315–2325, 2020.
[28] C. Nica, M. Oprea, and A.-C. Stan, "On the development of a mobile TurtleBot3 burger multi-robot system for manufacturing environment monitorization," in *Proceedings of Emerging Trends and Technologies on Intelligent Systems (ETTIS 2021)*, Springer Singapore, 2022, pp. 323–337.
[29] M. Kanderske and T. Thielmann, "Simultaneous localization and mapping and the situativeness of a new generation of geomedia technologies," *Commun. Public*, vol. 4, no. 2, pp. 118–132, Jun. 2019.
[30] C. Ruan, Q. Zang, K. Zhang, and K. Huang, "DN-SLAM: a visual SLAM with ORB features and NeRF mapping in dynamic environments," *IEEE Sens. J*., vol. 24, no. 4, pp. 5279–5287, Feb. 2024.
[31] L. Luo, F. Peng, and L. Dong, "Improved multi-sensor fusion dynamic odometry based on Neural networks," *Sensors*, vol. 24, no. 19, p. 6193, Sep. 2024.
[32] Y. Cai, Y. Ou, and T. Qin, "Improving SLAM techniques with integrated multi-sensor fusion for 3D reconstruction," *Sensors*, vol. 24, no. 7, p. 2033, Mar. 2024.
[33] H. Taheri and Z. C. Xia, "SLAM; definition and evolution," *Eng. Appl. Artif. Intell*., vol. 97, p. 104032, Jan. 2021.
[34] X. Dang, Z. Rong, and X. Liang, "Sensor fusion-based approach to eliminating moving objects for SLAM in dynamic environments," *Sensors*, vol. 21, no. 1, p. 230, Jan. 2021.
[35] J. M. Pak and C. K. Ahn, "State estimation algorithms for localization: A survey," *Int. J. Control. Autom. Syst*., vol. 21, no. 9, pp. 2771–2781, 2023.
[36] H. H. Afshari, S. A. Gadsden, and S. Habibi, "Gaussian filters for parameter and state estimation: A general review of theory and recent trends," *Signal Processing*, vol. 135, pp. 218–238, Jun. 2017.
[37] C. Nehate, R. Shinde, S. Naik, M. Aradwad, A. Bhurke, and F. Kazi, "Implementation and evaluation of SLAM systems for a mobile robot," in *2021 IEEE 18th India Council International Conference (INDICON)*, IEEE, Dec. 2021, pp. 1–6.
[38] Y. Liu, W. Zhang, F. Li, Z. Zuo, and Q. Huang, "Real-time lidar odometry and mapping with loop closure," *Sensors*, vol. 22, no. 12, p. 4373, Jun. 2022.
[39] L. Buonocore, S. R. Barros dos Santos, A. A. Neto, and C. L. Nascimento, "FastSLAM filter implementation for indoor autonomous robot," in *2016 IEEE Intelligent Vehicles Symposium (IV)*, IEEE, Jun. 2016, pp. 484–489.
[40] P. M. de Assis Brasil, F. U. Pereira, M. A. de Souza Leite Cuadros, A. R. Cukla, and D. F. Tello Gamarra, "A Study on global path planners algorithms for the simulated TurtleBot 3 robot in ROS," in *2020 Latin American Robotics Symposium (LARS), 2020 Brazilian Symposium on Robotics (SBR) and 2020 Workshop on Robotics in Education (WRE)*, IEEE, Nov. 2020, pp. 1–6.
[41] A. Dwijotomo, M. A. Abdul Rahman, M. H. Mohammed Ariff, H. Zamzuri, and W. M. H. Wan Azree, "Cartographer SLAM method for optimization with an adaptive multi-distance scan scheduler," *Appl. Sci*., vol. 10, no. 1, p. 347, Jan. 2020.
[42] R. P. Guan, B. Ristic, L. Wang, and J. L. Palmer, "KLD sampling with Gmapping proposal for Monte Carlo localization of mobile robots," *Inf. Fusion*, vol. 49, pp. 79–88, Sep. 2019.
[43] T. J. Lee, C. H. Kim, and D. I. D. Cho, "A monocular vision sensor-based efficient SLAM method for indoor service robots," *IEEE Trans. Ind. Electron*., vol. 66, no. 1, pp. 318–328, 2019.
[44] B. L. E. A. Balasuriya et al., "Outdoor robot navigation using Gmapping based SLAM algorithm," in *2016*

Moratuwa Engineering Research Conference (MERCon), IEEE, Apr. 2016, pp. 403–408.

[45] A. Palomer, P. Ridao, and D. Ribas, "Multibeam 3D underwater SLAM with probabilistic registration," *Sensors*, vol. 16, no. 4, p. 560, Apr. 2016.

[46] P. Du et al., "View-based underwater SLAM using a stereo camera," in *OCEANS 2017 - Aberdeen*, IEEE, Jun. 2017, pp. 1–6.

[47] Z. M. Kassas et al., "Aircraft navigation in GNSS-denied environments via radio SLAM with terrestrial signals of opportunity," *IEEE Trans. Intell. Transp. Syst*., vol. 25, no. 10, pp. 14164–14182, Oct. 2024.

[48] J. Fernandes, K. Li, J. Mirabile, and G. Vesonder, "Application of robot operating system in robot flocks," in *2019 IEEE 10th Annual Ubiquitous Computing, Electronics and Mobile Communication Conference (UEMCON)*, IEEE, 2019, pp. 0359–0363.

[49] R. Safin, R. Lavrenov, K.-H. Hsia, E. Maslak, N. Schiefermeier-Mach, and E. Magid, "Modelling a TurtleBot3 based delivery system for a smart hospital in gazebo," in *2021 International Siberian Conference on Control and Communications (SIBCON)*, IEEE, May 2021, pp. 1–6.

[50] S. Zhang and H. Zhao, "Research and implementation of cooperative control for ROS mobile robot," *Proc. Int. Conf. Artif. Life Robot*., vol. 29, pp. 43–48, Feb. 2024.

[51] M. H. Gusrial, M. L. H. Abdullah, N. A. Othman, and H. Ahmad, "Enhancing navigation accuracy of Turtlebot3 burger mobile robot through initial covariance matrix determination," in *Lecture Notes in Electrical Engineering*, vol. 1213, 2024, pp. 525–542.

[52] G. Zeng, S. Chen, H. Wu, and M. Yang, "Continuous-discrete extended Kalman filtering based on the neural ordinary differential equations method," *Eng. Appl. Artif. Intell*., vol. 133, p. 108611, Jul. 2024.

[53] Y. Bai, B. Yan, C. Zhou, T. Su, and X. Jin, "State of art on state estimation: Kalman filter driven by machine learning," *Annu. Rev. Control*, vol. 56, p. 100909, 2023.

[54] M. B. Rhudy, R. A. Salguero, and K. Holappa, "A Kalman filtering tutorial for undergraduate students," *Int. J. Comput. Sci. Eng. Surv*., vol. 08, no. 01, pp. 01–18, Feb. 2017.

[55] S. Zhao and B. Huang, "Trial-and-error or avoiding a guess? Initialization of the Kalman filter," *Automatica*, vol. 121, p. 109184, Nov. 2020.

[56] N. A. Robins, "A comparison into current SLAM implementations," Thesis, Toronto Metropolitan University, 2024.

[57] M. S. A. Khan, D. Hussian, Y. Ali, F. U. Rehman, A. Bin Aqeel, and U. S. Khan, "Multi-sensor SLAM for efficient navigation of a mobile robot," in *2021 4th International Conference on Computing & Information Sciences (ICCIS)*, IEEE, Nov. 2021, pp. 1–5.

[58] I. Ullah, X. Su, X. Zhang, and D. Choi, "Simultaneous localization and mapping based on Kalman filter and extended Kalman filter," *Wirel. Commun. Mob. Comput*., vol. 2020, pp. 1–12, Jun. 2020.

[59] T. Moore and D. Stouch, "A generalized extended Kalman filter implementation for the robot operating system," in *Advances in Intelligent Systems and Computing*, vol. 302, Springer, 2016, pp. 335–348.

[60] A. Eman and H. Ramdane, "Mobile robot localization using extended Kalman filter," in *2020 3rd International Conference on Computer Applications & Information Security (ICCAIS)*, IEEE, Mar. 2020, pp. 1–5.

[61] I. Ullah, X. Su, J. Zhu, X. Zhang, D. Choi, and Z. Hou, "Evaluation of localization by extended Kalman filter, unscented Kalman filter, and particle filter-based techniques," *Wirel. Commun. Mob. Comput*., vol. 2020, pp. 1–15, Oct. 2020.

[62] N. Ayadi, N. Derbel, N. Morette, C. Novales, and G. Poisson, "Simulation and experimental evaluation of the EKF simultaneous localization and mapping algorithm on the wifibot mobile robot," *J. Artif. Intell. Soft Comput. Res*., vol. 8, no. 2, pp. 91–101, Apr. 2018.

[63] H. Yao *et al*., "A Benchmark of Absolute and Relative Positioning Solutions in GNSS Denied Environments," *IEEE Internet Things J*., vol. 11, no. 3, pp. 4243–4273, Feb. 2024.

[64] L. D'Alfonso, W. Lucia, P. Muraca, and P. Pugliese, "Mobile robot localization via EKF and UKF: A comparison based on real data," *Robotics and Autonomous Syst*., vol. 74, pp. 122–127, Dec. 2015.

[65] T. Kim and T.-H. Park, "Extended Kalman filter (EKF) design for vehicle position tracking using reliability function of radar and LiDAR," *Sensors*, vol. 20, no. 15, p. 4126, Jul. 2020.

[66] C.-Y. Yang, H. Samani, Z. Tang, and C. Li, "Implementation of extended kalman filter for localization of ambulance robot," *Int. J. Intell Robot. Appl*., vol.8, pp. 960-973, Jun. 2024.

[67] G. G. Ravali and K. N. Raju, "Performance analysis of CKF algorithm for battery SoC estimation with its aging effect," *Int. J. Appl. Power Eng*., vol. 13, no. 4, pp. 886-899, Dec. 2024.

[68] M. Kiani and R. Ahmadvand, "The strong tracking innovation filter," *IEEE Trans. Aerosp. Electron. Syst*., vol. 58, no. 4, pp. 3261–3270, Aug. 2022.

[69] M. Mohammad, A. Naiari, and M. Hosseynzadeh, "Passive tracking of a target based on Supervisory adaptive EKF and CKF," in *2020 IEEE 17th International Conference on Smart Communities: Improving Quality of Life Using ICT, IoT and AI (HONET)*, IEEE, Dec. 2020, pp. 1–5.

[70] F. F. Yadkuri and M. J. Khosrowjerdi, "Methods for improving the linearization problem of extended Kalman filter," *J. Intell. Robot. Syst*., vol. 78, pp. 485–497, Jun. 2015.

[71] M. Moradi and R. Havangi, "Nonlinear filter-based estimation of wheel-rail contact forces and related considerations using inertial measurement unit," *J. Electr. Comput. Eng. Innov*., vol. 13, no. 2, pp. 353–364, 2025.

[72] Y. Wang, Z. Yang, Y. Wang, Z. Li, V. Dinavahi, and J. Liang, "Resilient dynamic state estimation for power system using Cauchy-Kernel-based maximum correntropy cubature Kalman filter," *IEEE Trans. Instrum. Meas*., vol. 72, pp. 1–11, 2023.

[73] B. Liu and Z. Wu, "Maximum correntropy quadrature Kalman filter based interacting multiple model approach for maneuvering target tracking," *Signal, Image and Video Process.*, vol. 19, no. 76, pp. 1-10, Jan. 2025.
[74] Z. Song, Y. Zhang, Y. Yu, and C. Tang, "Cooperative positioning algorithm based on manifold gradient filtering in UAV-WSN," *IEEE Sens. J.*, vol. 24, no. 8, pp. 12676–12688, Apr. 2024.
[75] M. Rhudy, Y. Gu, and M. R. Napolitano, "An analytical approach for comparing linearization methods in EKF and UKF," *Int. J. Adv. Robot. Syst.*, vol. 10, no. 4, p. 208, Apr. 2013.
[76] S. M. M. Yousefi, S. S. Mohseni, H. Dehbovid, and R. Ghaderi, "A practical approach to tracking estimation using object trajectory linearization," *Int. J. Comput. Intell. Syst.*, vol. 17, no. 175, pp. 1-14, Jul. 2024.
[77] P. S. Lakshmi and V. L. Raju, "ECG de-noising using hybrid linearization method," *TELKOMNIKA Indones. J. Electr. Eng.*, vol. 15, no. 3, p. 504, Sep. 2015.
[78] M. Franzese and A. Iuliano, "Correlation analysis," in *Encyclopedia of Bioinformatics and Computational Biology*, vol. 1, Elsevier, pp. 706–721, 2019.
[79] H. Ahmad, N. A. Othman, M. M Saari, M. S Ramli, M. M Mazlan, and T. Namerikawa, "A hypothesis of state covariance decorrelation effects to partial observability SLAM," *Indones. J. Electr. Eng. Comput. Sci.*, vol. 14, no. 2, pp. 588–596, May 2019.
[80] M. M. Mazlan, N. A. Othman, and H. Ahmad, "Improving covariance matrix diagonalization in SLAM of mobile robot," in *Lecture Notes in Electrical Engineering*, vol. 770, Springer Singapore, pp. 995–1007, 2022.
[81] A. Rauh, S. Wirtensohn, P. Hoher, J. Reuter, and L. Jaulin, "Reliability assessment of an unscented Kalman filter by using ellipsoidal enclosure techniques," *Mathematics*, vol. 10, no. 16, p. 3011, Aug. 2022.
[82] M. Berardi and J. K. Galimberti, "Smoothing-based initialization for learning-to-forecast algorithms," *Macroecon. Dyn.*, vol. 23, no. 3, pp. 1008–1023, Apr. 2019.
[83] S. Zhao, B. Huang, and F. Liu, "Linear optimal unbiased filter for time-variant systems without apriori information on initial conditions," *IEEE Trans. Automat. Contr.*, vol. 62, no. 2, pp. 882–887, 2017.
[84] S. Jia, Y. Jiao, Z. Zhang, R. Xiong, and Y. Wang, "FEJ-VIRO: A consistent first-estimate Jacobian visual-inertial-ranging odometry," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE, Oct. 2022, pp. 1336–1343.
[85] M. Rupp, B. Blagojevic, C. Knoll, M. P. Zapf, Z. Weimin, and O. Sawodny, "Approximation of covariance matrices based on matching accuracy," in *2020 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)*, IEEE, Jul. 2020, pp. 691–696.
[86] P. Chen, H. Liu, M. Schmidt, Y. Yao, and W. Yao, "Near real-time global ionospheric modeling based on an adaptive Kalman filter state error covariance matrix determination method," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–12, 2022.
[87] Q. Wei and Z. Zhao, "Large covariance matrix estimation with oracle statistical rate," in *2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Rhodes Island, Greece: IEEE, Jun. 2023, pp. 1–5.
[88] P. Shi, Z. Zhu, S. Sun, Z. Rong, X. Zhao, and M. Tan, "Covariance estimation for pose graph optimization in visual-inertial navigation systems," *IEEE Trans. Intell. Veh.*, vol. 8, no. 6, pp. 3657–3667, Jun. 2023.
[89] S. Zhang, J. Shan, and Y. Liu, "Variational Bayesian estimator for mobile robot localization with unknown noise covariance," *IEEE/ASME Trans. Mechatronics*, vol. 27, no. 4, pp. 2185–2193, Aug. 2022.