



sensors



Review

Advances in Visual Simultaneous Localisation and Mapping Techniques for Autonomous Vehicles: A Review

Jibril Abdullahi Bala, Steve Adetunji Adeshina and Abiodun Musa Aibinu



<https://doi.org/10.3390/s22228943>

Review

Advances in Visual Simultaneous Localisation and Mapping Techniques for Autonomous Vehicles: A Review

Jibril Abdullahi Bala ^{1,*}, Steve Adetunji Adeshina ² and Abiodun Musa Aibinu ¹¹ Department of Mechatronics Engineering, Federal University of Technology, Minna 920211, Nigeria² Department of Computer Engineering, Nile University of Nigeria, Abuja 900001, Nigeria

* Correspondence: jibril.bala@futminna.edu.ng

Abstract: The recent advancements in Information and Communication Technology (ICT) as well as increasing demand for vehicular safety has led to significant progressions in Autonomous Vehicle (AV) technology. Perception and Localisation are major operations that determine the success of AV development and usage. Therefore, significant research has been carried out to provide AVs with the capabilities to not only sense and understand their surroundings efficiently, but also provide detailed information of the environment in the form of 3D maps. Visual Simultaneous Localisation and Mapping (V-SLAM) has been utilised to enable a vehicle understand its surroundings, map the environment, and identify its position within the area. This paper presents a detailed review of V-SLAM techniques implemented for AV perception and localisation. An overview of SLAM techniques is presented. In addition, an in-depth review is conducted to highlight various V-SLAM schemes, their strengths, and limitations. Challenges associated with V-SLAM deployment and future research directions are also provided in this paper.

Keywords: autonomous vehicles; computer vision; localisation; mapping; visual SLAM



Citation: Bala, J.A.; Adeshina, S.A.; Aibinu, A.M. Advances in Visual Simultaneous Localisation and Mapping Techniques for Autonomous Vehicles: A Review. *Sensors* **2022**, *22*, 8943. <https://doi.org/10.3390/s22228943>

Academic Editor: Ikhlas Abdel-Qader

Received: 19 October 2022
Accepted: 16 November 2022
Published: 18 November 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Autonomous Vehicles (AVs) have received widespread attention globally due to their low deployment costs, low on-board sensor sizes, and improved technology [1–4]. AVs have the ability to carry out various driving tasks with minimal or no human supervision. This unique ability of AVs significantly reduces driver workload while increasing driver comfort and efficiency. Additionally, these systems improve vehicle safety by reducing human errors, such as drowsiness, intoxication, fatigue, and loss of concentration, which in turn lead to vehicle accidents [2,5–7]. The end-goal of AV development is to create a driverless system that can perform diverse driving tasks [8]. The successful operation of AVs depends on five operational modules which are perception, localisation, decision making, planning, and control [8,9]. While decision making, planning, and control majorly involve problem solving, the vehicle can only perform these tasks based on the input from the perception and localisation modules. Perception, planning, and control are significant challenges in AV security and safety [10]. Understanding the surroundings is critical in a vehicle's operation [11], and sensing the environment is achieved using a variety of perception technologies such as cameras, lidar, radar, Global Position System (GPS) devices, and ultrasonic sensors [12]. Localisation involves a vehicle estimating its position and orientation within a perceived environment. Robust localisation is another key challenge in AV deployment [13]. Several studies have implemented GPS and Global Navigation Satellite System (GNSS)-based localisation schemes. However, these techniques have proven insufficient in terms of precision, and as such, recent research efforts have focused on the use of lidar and cameras for localisation [13]. A popular technique which utilises cameras and lidar sensors for perception and localisation is Simultaneous Localisation and Mapping (SLAM).

SLAM is a method in which an autonomous navigation system obtains 2D or 3D geometric information about its surroundings, which is usually unknown, estimates its pose within that environment, and generates a map of the area [14,15]. SLAM-based systems have been used in a wide range of applications such as drones, mobile robots, virtual reality, and augmented reality [14,16]. A variety of sensors have been used in implementing SLAM such as lidar, GPS, and Inertial Measurement Unit (IMU) sensors [14,17]. However, SLAM based on cameras has been widely studied due to the low cost, simpler configuration, low energy consumption, and versatility of cameras [14,16]. This technique, known as Visual SLAM (V-SLAM), uses cameras to estimate an autonomous agent's position and orientation within an environment, as well as map that environment.

V-SLAM is a technique under Visual Odometry (VO) which uses cameras to calculate a body's change in position over time [18]. V-SLAM has been implemented with different camera types and configurations. Monocular, stereo, and RGB-D cameras have also been successfully used in developing SLAM-based systems [19]. Additionally, these techniques utilise a variety of feature extraction techniques [20] and some researchers have combined traditional V-SLAM schemes with machine and deep learning [21]. Furthermore, the maps generated by SLAM algorithms provide a visual representation of the environment in 2D forms, as in the case of binary occupancy grid maps [22] or 3D forms such as point cloud maps [23].

In this paper, a detailed review of V-SLAM techniques implemented in AVs and mobile robots is presented. The study takes an in-depth look at the techniques, strengths, and limitations of the methods implemented in these systems. The effectiveness and accuracy of current V-SLAM techniques are also highlighted in this study. The main objectives of this paper are as follows:

- Develop a hierarchy of existing V-SLAM methods with a focus on their respective implementation techniques and perceived advantages over their counterparts;
- Discuss key characteristics of V-SLAM techniques in literature and highlight their advantages and limitations;
- Perform a comparative analysis of recent V-SLAM technologies and identify their strengths and shortcomings;
- Identify open issues and propose future research directions in V-SLAM schemes for AVs.

The rest of this paper is organised as follows. Section 2 provides a background on SLAM with focus on the different types of SLAM algorithms and their implementations. An in-depth look at V-SLAM schemes, their types, architectures, and applications are presented in Section 3. Proposed future directions in the area of V-SLAM research are presented in Section 4 while the conclusion is provided in Section 5.

2. Simultaneous Localisation and Mapping (SLAM)

Simultaneous Localisation and Mapping (SLAM) is a major problem in the field of robotics and autonomous navigation systems. The solution to the SLAM problem enables a mobile robot or AV to create a map of its environment while simultaneously keeping track of its pose within the same environment [24,25]. In the last three decades, significant achievements have been recorded in the field of SLAM [16]. SLAM has been implemented in ground [26], aerial [27], and underwater [28] systems.

The SLAM problem has been solved using a variety of approaches. Filter-based algorithms such as Extended Kalman Filter (EKF) and Particle Filter (PF) have been used in SLAM systems. Estimation theoretic approaches such as Bayesian estimation have also been implemented to solve the SLAM problem [24]. For instance, in [29], an optimized self-localization technique for SLAM was presented. This method was developed for dynamic scenes using hypothesis density filters. The novel approach, called GEM-SLAM, achieved major improvements over three benchmark algorithms namely, the Rao Blackwellized Probability Hypothesis Density (RB-PHD) filter, the Single Cluster Probability Hypothesis Density (SC-PHD) filter, and the Factored Solution to SLAM (FastSLAM) algorithm.

2.1. SLAM Techniques

SLAM techniques can be categorised into two groups: the Filter-based SLAM and the Graph-based SLAM [29]. The graph-based techniques depend on non-linear optimization techniques and thus, the method used must ensure convergence to global minima. Additionally, pose graph optimisation is susceptible to accumulated errors. Loop closure is used to detect a previously visited location and recalibrates the system's trajectory [29]. Other graph-based techniques include Occupancy Grid-based SLAM [30], Graph SLAM [30], and Incremental Smoothing and Mapping (iSAM) [31,32].

Because of the limitations of graph-based approaches, such as convergence to local minima and cumulative errors, filter-based techniques have been investigated as possible alternatives to graph-based techniques. The EKF is one of the earliest methods for solving SLAM and was proposed by [33]. A clutter resistant SLAM algorithm for autonomous guided vehicles in dynamic industrial environments was presented in [24]. This clutter-resistant technique uses point features generated from reflectors and line features to improve SLAM robustness. The technique proved to be accurate and efficient by keeping the localisation error within 19 mm and 31 mm for the X-axis and Y-axis, respectively.

The Unscented Kalman Filter (UKF) is related to the class of Linear Regression Kalman Filters. This technique is widely used in SLAM systems to linearize random variable non-linear functions using linear regression [34]. A localisation technique for Wireless Sensor Networks (WSNs) based on UKF and PF algorithms was presented in [34]. The authors modified a Kalman Filter based on UKF and PF localisation. This proposed technique showed a variety of applications such as target tracking and robot localisation based on the simulation results.

The PF is also another filter-based method used in SLAM. The PF samples multiple possibilities (particles) of the observer and filters them based on the likelihood of them having the correct pose [35,36]. A fast algorithm of SLAM for mobile robots based on ball PF was presented in [26]. This technique was a modification of the box PF and the firefly algorithm is used to maintain the diversity of ball particles. This process increased the consistency of the pose estimation and the results demonstrated the superiority in performance of the technique. In addition, ref. [37] presented an embedded PF SLAM implementation using a cost effective platform. In this study, a Genetic Algorithm (GA)-based approach is implemented for calibration and to prevent overfitting. This process provided more generalizable results, robustness, and improved performance. Furthermore, a PF SLAM using differential drive mobile robot was presented in [38]. The study implemented a PF-based SLAM technique on a mobile robot called e-puck. This study observed that the number of particles used had an effect of the performance of the algorithm. Also, ref. [39] developed a stratified PF SLAM based on monocular cameras. This study used a sample weighting algorithm to stratify the particles and was implemented for an Unmanned Aerial Vehicle (UAV). The results were compared to different sample weighting approaches and exhibited improved robustness and accuracy.

In addition, the Rao-Blackwellized Particle Filter (RBPF) is used for robust estimation in SLAM [29]. For instance, ref. [40] presented a study which improved grid-based SLAM with RBPF by adaptive proposals and selective resampling. This technique computed an accurate proposal distribution which considers both the movement of the robot and its most recent observation. The results exhibited the advantage of this technique over other previous approaches. Another popular filter-based technique used in SLAM is the Factored Solution to Simultaneous Localization and Mapping (FastSLAM) which is based on the RBPF [41,42]. The FastSLAM algorithm was later improved by Montemerlo [43,44]. A hierarchy of the types of SLAM techniques is presented in Figure 1.

2.2. Sensors Used in SLAM

Autonomous navigation systems use a variety of sensors to perceive the environment and localise themselves. These sensors include lidar, sonar, a Radio Frequency Identification (RFID), Global Positioning System (GPS), Inertial Measurement Units (IMU), and

vision sensors [17]. For instance, ref. [45] evaluated the exact flow of particles used for state estimations in unmanned aerial system navigation. The technique utilised sensor information obtained from IMU and GPS sensors and implemented a novel Bayesian filtering SLAM technique for aerial environments. The results proved the effectiveness of this technique over existing methods in terms of speed of convergence and accuracy. Furthermore, ref. [46] presented a PF-based landmark mapping for SLAM of mobile robots. This technique was based on a Radio Frequency Identification (RFID) system and used two separate filters to estimate position and orientation of the robot and integrated circuit tags. The experimental results validated the technique and showed its computational efficiency.

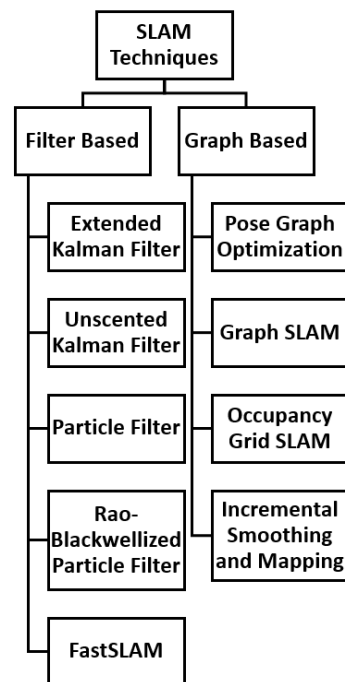


Figure 1. Hierarchy of SLAM Techniques.

SLAM based on lidar sensors is a popular research area since the sensor data is not affected by illumination changes unlike cameras [47]. This technique is considered to be an accurate and effective method for autonomous agents to create a map and localise themselves [25]. A conic feature-based SLAM technique was developed for open environments in [25]. The technique utilised 2D lidar and a defined conic feature-based parametric approach. The experimental results proved the accuracy of the algorithm in open environments. In addition, ref. [48] benchmarked PF algorithms for efficient Velodyne-based vehicle localisation. Due to the large amount of information obtained by the 3D Velodyne lidar sensor, an analysis was performed to ascertain the how many points were required to achieve optimal efficiency and positioning accuracy. The study conclude that an initial density of two particles per square metre were required to achieve total convergence.

Additionally, Global Navigation Satellite System (GNSS) has been successfully used to implement localisation techniques in SLAM. GNSS provides multiple positioning solutions such as precise point positioning, single point position, and Real-time Kinematic (RTK). GNSS is low cost and works well in an open sky environment [47]. A continuous positioning algorithm based on RTK and Visual Inertial SLAM (VI-SLAM) was presented in [49]. In this study, VI-SLAM was used to complement the limitations of RTK such as blockage of satellite signals by trees and buildings. The results showed the effectiveness of the technique to provide continuous positioning solutions. Furthermore, sonar sensors have been successfully implemented in SLAM systems. The sensors' ability to perform effectively in underwater water scenarios has made it a popular choice. A PF SLAM with 3D evidence

grids in underwater environments was presented in [50]. This study utilised the RBPF in a flooded subterranean environment using sonar data, and provided successful results.

Laser scanners have also been implemented to solve SLAM problems due to their ability to provide bearing and range information [51]. A PF-based outdoor robot localization technique using natural features extracted from a laser scanner was presented in [51]. This method applies batch processing which extracts features from full laser scans. The PF is applied to reduce the estimation error associated with EKFs and the technique was verified in real world experiments. Additionally, ref. [52] presented a distributed SLAM technique using improved PF for mobile robot localisation. This method used data obtained from laser sensors and dead reckoning. The system exhibited better tolerance and robustness over Distributed PF SLAM. Table 1 shows the various sensors used in SLAM and highlights their merits and demerits. The sensors outlined have been implemented in various SLAM techniques with varying configurations. Thus, they can be used for both feature-based and graph-based SLAM implementations.

Table 1. Overview of SLAM Sensors.

Sensor	Advantages	Limitations
Lidar	<ul style="list-style-type: none"> • Can provide 3D and 2D information [25]. • Can be used as a standalone SLAM technique [25]. 	<ul style="list-style-type: none"> • High Cost [49]. • Noisy Data [53].
Radar	<ul style="list-style-type: none"> • Provides accurate range information [54]. 	<ul style="list-style-type: none"> • High Cost [49].
Sonar	<ul style="list-style-type: none"> • Can be used effectively in underwater scenarios [50]. 	<ul style="list-style-type: none"> • High memory requirements [50]. • High Cost [49].
Camera	<ul style="list-style-type: none"> • Low cost, versatile, and simple configuration [17]. • Can extract semantic information [55]. 	<ul style="list-style-type: none"> • Monocular cameras provide no scale information [16]. • Performance varies in scenes prone to illumination changes [56].
Inertial Measurement Unit	<ul style="list-style-type: none"> • Can measure angular velocity and acceleration [16]. • Serves as complementary sensor to cameras [49]. • Good relocalisation capabilities [16]. 	<ul style="list-style-type: none"> • Difficult to implement as a stand-alone SLAM sensor. • Performance unsatisfactory in dynamic environments since it cannot extract semantic information [16].
Wheel Encoders	<ul style="list-style-type: none"> • Not susceptible to drift with temperature [57]. • Requires no initialisation [57]. 	<ul style="list-style-type: none"> • Only suitable for 2D motion [57].
Laser	<ul style="list-style-type: none"> • Can provide accurate range information [24]. 	<ul style="list-style-type: none"> • Susceptible to errors in reflective environments [24]. • High cost [49].
GPS	<ul style="list-style-type: none"> • - Provides accurate location information [49]. 	<ul style="list-style-type: none"> • Susceptible to failure in environments where coverage is blocked by trees or buildings [49]
GNSS	<ul style="list-style-type: none"> • Provides multiple positioning solutions [49]. 	<ul style="list-style-type: none"> • Fails in environments where coverage is blocked by trees or buildings [49].

3. Visual Simultaneous Localisation and Mapping (V-SLAM)

3.1. Background of V-SLAM

V-SLAM is a technique in which an autonomous navigation system uses a vision sensor to construct and update a map of an unknown environment while at the same time keeping track of its position and orientation within that environment [58,59]. In comparison to other sensor data such as lidar, camera data can provide rich and abundant information, which enhances high level operations [60]. The camera route is represented as a set of relative poses in a world reference frame. The environment is represented by landmarks which are objects or keypoint features in each frame. The landmarks remain still in static environments, while the landmarks change their position in the case of dynamic environments [60].

The V-SLAM process can be broken down into the following steps [58]:

- Acquire, read, and pre-process the data from the camera and other devices;
- Estimate motion and local map of the scene from adjacent camera frames;
- Optimise and adjust camera poses;
- Detect loops to eliminate errors and complete the map.

V-SLAM primarily uses cameras to solve the SLAM problem due to their low cost, versatility, efficiency, and simple configuration [14]. The cameras can be classified into three major types [49], namely: monocular, stereo, and RGB-D cameras. In the monocular camera setup, only one camera is used for visual perception. Significant research has been conducted in monocular V-SLAM with excellent results. MonoSLAM is considered to be the first pure visual monocular SLAM technique. It was presented in 2003 and was capable of pose estimation and feature measurement on a desktop PC [61]. In 2009, PTAM was presented, which comprised of two threads; one for feature point tracking and the other for mapping [62].

ORB-SLAM [63] is a popular and open source monocular SLAM technique which breaks the SLAM process into three threads: tracking, mapping, and loop detection. Numerous monocular SLAM techniques have been built on the ORB-SLAM framework with impressive results. A SLAM map restoration algorithm based on submaps and an undirected connected graph was presented in [14]. This technique is a monocular SLAM method and uses submap connections to reinitialise and rebuild parts of a map in the event of a system trace failure. The method was simulated on a UAV dataset and the results showed that the integrity of the map was preserved in the case of tracking failures. Additionally, semi-direct monocular SLAM with three levels of parallel optimizations was developed in [64]. In this study, a novel SLAM framework is developed. The first half of the technique utilises direct method for camera pose tracking while the second half of the technique uses a feature-based method for refinement of key frames, loop closures, and mapping. This method showed an increase in accuracy and robustness in motion estimation.

In the monocular camera setup, the scale of the system and environment is ambiguous, and the field of view is limited. It is difficult to determine the depth of the scene or estimate distance from the monocular camera setup [49]. Due to these limitations, studies have been conducted to minimise the effect of inability of monocular cameras to estimate scale. For instance, a simple but effective scale estimation technique for monocular visual odometry in road driving scenarios was presented in [65]. The study utilised 3D ground points to estimate the scale of the camera in monocular V-SLAM. The results showed that the method was able to achieve an average translation error of 1.19% on the KITTI dataset, thus, outperforming conventional monocular visual SLAM techniques.

Stereo camera setup utilises two or more cameras to perceive the environment. In this configuration, the cameras can be placed on both the left and right hand sides of the field of vision. With this setup, it is easier to estimate the scale of the environment. Additionally, this configuration allows the use of smartphones in V-SLAM processes since a significant number of these devices come with multiple cameras [49]. ORB-SLAM2 [19] which is an extension of ORB-SLAM suitable for stereo and RGB-D cameras caters for the limitations associated with monocular cameras. In [66], a persistent map saving technique was developed for visual localisation for AVs. This method was an extension of ORB-SLAM2 and utilised stereo vision to develop a map saving feature for the technique. Experimental results show that the method was capable of keeping the relative translation error of the localisation under 1%. Similarly, a robust stereo visual SLAM technique for dynamic environments with moving object was developed in [67]. This study is based on the ORB-SLAM2 technique as well, and uses a fundamental matrix to identify dynamic feature points and then uses on the static points to estimate the pose. This in turn improved the localisation accuracy and robustness of the system. The authors in [68] developed an indirect visual simultaneous localization and mapping scheme based on linear models. The model was based on the stereo camera configuration and utilised key-frame insertion and map management to minimise computational redundancy and landmark unreliability.

The technique showed improved performance when compared to other techniques such as ORB-SLAM2 and COP-SLAM.

Despite the usefulness of stereo vision in depth estimation, in some devices such as smartphones, the cameras are closely placed to each other making accurate depth estimation beyond 2 metres challenging. RGB-D cameras solve this challenge by using Time of Flight technology to estimate the depth of the environment [49]. KinectFusion proposed the use of Kinect cameras for 3D reconstruction, laying the groundwork for research in RGB-D-based SLAM systems although this method was computationally intensive and slow [58]. ORB-SLAM2 [19] has proven to be an effective and popular choice for RGB-D-based V-SLAM techniques. A pixel-wise motion segmentation for SLAM in dynamic environments was developed by [69]. The study developed a novel pixel-wise segmentation technique for dynamic elements in a quest to improve accuracy and robustness of RGB-D SLAM. The experimental results showed the novel technique outperformed other state of the art methods for dynamic object removal in SLAM schemes. Additionally, ref. [17] developed a real-time cloud visual simultaneous localization and mapping technique for indoor service robots. The study presents the development of a real-time network-based SLAM technique which aims at minimising computing costs. The results showed the efficiency of the method and its ability to bear the network delay in a Local Area Network (LAN).

3.2. V-SLAM Categories

V-SLAM can be categorised into three major classifications, namely: Feature-based methods (Indirect method), Direct methods, and Multi-Sensor methods [14]. These methods are presented in Figure 2.

In the feature based or indirect methods, unique features of the camera frame are extracted and matched with the features of the next frame for tracking. ORB (Oriented FAST and rotated BRIEF) feature extraction [19,63] is a popular choice in V-SLAM. The choice of this technique is based on the fact that ORB features are fast to compute and are invariant to viewpoint. Several V-SLAM studies utilise ORB feature extraction as their method of choice. In [60], a robust visual localization technique in dynamic environments based on sparse motion removal was developed. The authors developed a V-SLAM technique for dynamic environments using Sparse Motion Removal. This method uses ORB feature extraction and evaluates the similarities between two consecutive frames and the difference between the current frame and the reference frame. This process identifies dynamic regions in the SLAM process and the results showed an improvement in the localisation accuracy and robustness in dynamic environments. Similarly, ref. [70] developed a novel integrated framework for V-SLAM in an attempt to bridge the gap between visual servoing and V-SLAM. This study, in a quest to enhance robustness and efficiency, developed a novel integrated framework which uses V-SLAM to stabilise servo tasks. The technique also removes feature points of moving objects to enhance operability in dynamic environments. Furthermore, ref. [71] developed MGC-VSLAM which is a meshing-based and geometric constraint V-SLAM technique for dynamic indoor environments. The study improves the accuracy and robustness of the ORB-SLAM2 algorithm by utilising a novel meshing-based uniform distribution approach. Additionally, a modified geometric constraint method is used to filter out the dynamic features. The results showed the technique improved the positioning accuracy of ORB-SLAM2 in highly dynamic surroundings.

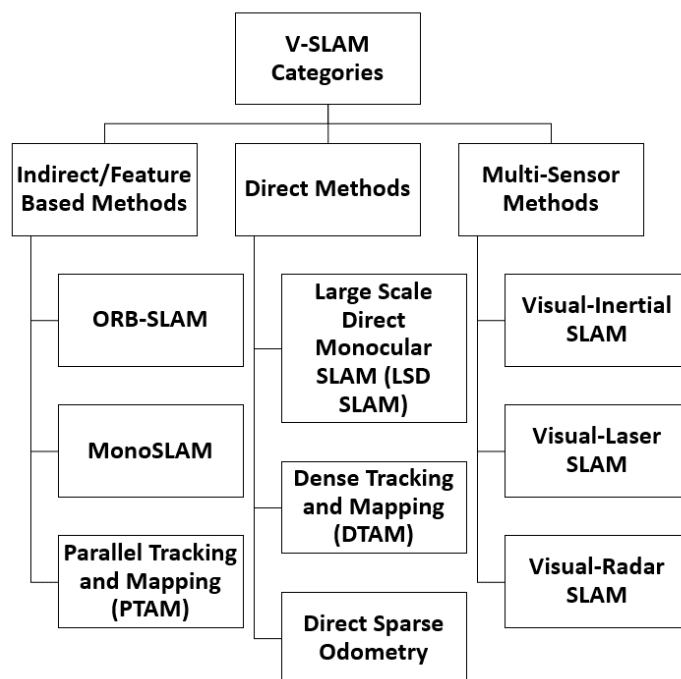


Figure 2. V-SLAM Categories.

ORB feature matching can be adversely affected by illumination changes and weak texture, leading some researchers to explore other feature extraction techniques such as lines and planes [14,72]. For instance, a SLAM system based on RGBD image and point-line feature was developed in [72]. This technique uses RGB-D images and point-line features to improve the accuracy and reliability of SLAM estimations. The results showed an improvement in accuracy of pose estimation and map reconstruction. Additionally, ref. [73] developed DT-SLAM which is a dynamic thresholding-based corner point extraction technique in SLAM systems. The study presents a dynamic self-adaptive threshold technique for corner point detection in a quest to improve SLAM localisation performance. This technique improved upon the ORB corner point extraction in FAST which in turn improved the localisation performance. Furthermore, ref. [74] developed PL-GM which is an RGB-D SLAM with a novel 2D and 3D geometric constraint model of point and line features. The author presented a novel geometric constraint model to utilise both 2D and 3D information within points and lines. The results showed a comparable performance to contemporary SLAM techniques based on point and line features. A line flow-based SLAM technique was presented in [75]. This study uses a line flow to encode line segment observations along the temporal dimension. This technique showed higher efficiency and localisation accuracy. The experimental results showed good localisation and mapping abilities in challenging scenarios.

The direct methods cater for large time consumption associated with feature extraction and matching. Additionally, since the extracted features only represent a small part of the overall image, the direct methods use global pixel information. These techniques work on the assumption that the image intensity should be consistent in corresponding spatial points of neighbouring frames [14]. For instance, ref. [76] presented a semi-direct visual odometry which involves extracting the FAST feature points in the camera frame and subsequently evaluating the pose of the camera transformation according to the information around the feature points. Similarly, ref. [77] presented Large-Scale Direct (LSD) SLAM method. This technique computes the depth of semi-dense points with sudden gradient changes such as corners and edges in place of feature extraction. The technique can be run in real-time on a CPU and also has the ability to deal with weaker textures and larger scales. Other

direct V-SLAM methods include Direct Tracking and Mapping (DTAM) and Direct Sparse Odometry [78].

Due to the ambiguity associated with monocular V-SLAM, the technique can be combined with other devices such as GNSS and IMU sensors. This is the case in the multi sensor approach [14,49]. These hybrid methods solve the scale ambiguity problem, albeit with additional configuration, computation, and cost requirements [49]. Visual Inertial SLAM (VI-SLAM) is a technique where the camera is fused with an IMU sensor. This method is useful in real-time applications due to its unique properties such as bias correction, automatic estimator initialisation, online extrinsic calibration, and loop detection [49]. A Visual-Inertial RGB-D SLAM with encoders for a differential wheeled robot deep learning-based v-slam techniques was presented in [79]. The study combined encoders with RGB-D cameras to develop a SLAM technique with improved accuracy and a lower Root Mean Square Error (RMSE). Additionally, a stereo visual inertial mapping algorithm for autonomous mobile robot was presented in [80]. Here, the authors presented a stereo visual inertial mapping system in order to enhance the accuracy and consistency of the generated map. The experimental results proved the robustness and effectiveness of the algorithm. Furthermore, ref. [27] developed VPS-SLAM which is a visual planar semantic SLAM technique for aerial robots. The authors presented robust and lightweight visual semantic SLAM technique for aerial robots. This method combines visual inertial odometry techniques and YOLO v2 to extract semantic information and maps an indoor environment, the effectiveness of which was validated with several experiments. A summary of additional works reviewed under the different types of V-SLAM is presented in Table 2.

Table 2. Related Works in V-SLAM Research

Ref.	Work	Observations
[57]	Uses measurements of RGB-D camera and encoder to produce robot poses and octo-map, relies on CPU not GPU, works in both static and dynamic indoor environments.	Not designed for outdoor environments, problem of wheel slipping causes inaccuracy, inability to track dynamic objects.
[81]	Technique proved agreement between system pose estimates and ground truth.	Performance on dynamic outdoor environment could not be determined, and depth camera in Kinect blinded by sunlight during daytime.
[49]	Uses Visual-Inertial SLAM to complement limitations of RTK such as blockage of satellite signals due to buildings and trees. This was achieved with a common smartphone instead of extra specialised devices.	Further work required to evaluate the model more accurately, not designed for dynamic outdoor environments. RTK systems are reliant on further infrastructure which comes with additional costs according to [66].
[82]	System uses pre-existing map and compares obtained images to evaluate user's position within the map.	Use of pre-existing map not suitable for dynamic outdoor environments.
[83]	The study presents the development of a new processing chain based on V-SAM for UAVs.	Data processing performance in real time is low, and the technique focuses on aerial motion and thus, no information was provided for ground movement.
[84]	In this study, the authors developed a semantic depth filter for RGB-D SLAM operations making it more accurate in dynamic environments.	Simulated using TUM dataset and thus performance on dynamic outdoor environment could not be determined.
[85]	The study presents novel panoramic Visual Inertial SLAM which utilises a wheel encoder to achieve improved robustness and localisation accuracy.	Simulated using University of Michigan North Campus Long-Term Vision and LiDAR Dataset (NCLT) dataset and thus performance on dynamic outdoor environment could not be determined.

Table 2. Cont.

Ref.	Work	Observations
[86]	The study presents a SLAM algorithm coupled with wheel encoder measures to enhance localisation. A low cost map was generated to enhance speed and memory efficiency.	Despite its ability to handle tracking in dynamic scenarios, the system only considered indoor settings and no information on outdoor performance was provided.
[87]	The study uses wheel odometer measurements and monocular camera to develop a Visual Inertial Odometry model coupled with non-holonomic constraints.	Simulated using KITTI and KAIST Complex Urban datasets and thus, real time performance on dynamic outdoor environment could not be accurately determined.
[88]	This article presents Integrated Visual Odometry with a Stereo Camera (IVO-S), a unique low-cost underwater visual navigation approach. Unlike pure visual odometry, the suggested approach combines data from inertial sensors and a sonar to function in context-sparse situations.	The suggested approach performs effectively in underwater sparse-feature settings with high precision, but existing visual slams or odometries, like as ORB-SLAM2 and OKVIS, do not. However, the technique does not include loop closure detection and map reconstruction operations.
[89]	This research presents a real-time and resilient point-line based monocular visual inertial SLAM (VINS) system for smart city mobility robots heading towards 6G. EDLines with adaptive gamma correction are used to extract a higher proportion of long line features among all extracted line features faster.	The experimental findings reveal that the VINS system outperforms other sophisticated systems in terms of localization accuracy, and robustness in challenging situations. However, the performance in outdoor scenes could not be accurately determined since the model was not deployed in outdoor settings.

3.3. Deep Learning Applications in V-SLAM

In recent years, V-SLAM has been combined with deep learning techniques. These deep learning models are utilised in various operations in the SLAM pipeline such as depth prediction [90], object detection [91], and semantic segmentation [92]. Due to the significant improvements in accuracy and performance achieved by deep learning techniques, these methods have become a de facto solution for most computer vision tasks such as object detection and classification [28].

In [16], a real-time visual-inertial localization technique using semantic segmentation towards dynamic environments was presented. The study utilises CNN-based semantic segmentation and multi-view geometric constraints to identify and avoid using dynamic object feature points. The results showed the technique had higher localisation accuracy and robustness when compared to other state of the art SLAM techniques. In addition, ref. [93] presented RDMO-SLAM which is a real-time V-SLAM for dynamic environments using semantic label prediction with optical flow. This technique, which is an extension of [92], is a novel semantic visual SLAM technique based on Mask R-CNN and PWC-Net and was developed to improve the tracking and real time performance of the SLAM algorithm. The results showed an improvement in real-time performance while retaining robust tracking capabilities.

Furthermore, ref. [58] developed a mobile robot visual slam system with enhanced semantics segmentation. The technique was able to improve tracking and speed using camera and encoder data. This was achieved using enhanced semantic segmentation. In [94], the authors present a Visual SLAM method based on semantic segmentation and geometric constraints. This method was suitable for dynamic indoor environments. The test results on the Oxford and TUM datasets demonstrate that the modified approach improves the consistency of feature points extracted by 56.3%. Visual SLAM placement error is decreased by 68.8% on average, and the produced semantic map contains rich semantic information with less redundancy. However, the accuracy and computation time could be improved upon. A summary of additional works reviewed in the area of deep learning-based V-SLAM techniques is presented in Table 3.

Table 3. Deep Learning-Based V-SLAM Techniques

Ref.	Work	Observations
[20]	This study presents a SLAM technique that uses objects and walls as elements of the environment model. The objects are identified using YOLO v3 technique. The system exhibited better performance than the RGB-D SLAM and a comparable performance to ORB SLAM.	The technique was tested in a static indoor environment and thus, the performance on a dynamic outdoor environment could not be determined.
[95]	In this study, a semantic filter-based faster R-CNN is utilised to solve fundamental matrix calculations in ORB SLAM. This method reduced the trajectory error, number of low quality feature correspondences, and position error.	Simulated using KITTI and ETH datasets and thus performance on dynamic outdoor environment could not be determined.
[55]	Developed a novel RGB-D SLAM method combined with deep learning in order to decrease impact of moving objects in the estimation of camera pose. This was achieved using semantic segmentation and multi-view geometry.	Real time performance needs to be improved, not designed for outdoor environments.
[96]	Technique combined ORB-SLAM2 and PSPNet-based semantic segmentation to identify and eliminate dynamics points. This reduced the trajectory and pose errors.	Simulated using the TUM RGB-D dataset and thus performance on dynamic outdoor environments could not be determined.
[97]	Here, a Decoder-Encoder Model (DEM) was developed which uses CNNs to improve depth estimation performance. Additionally, a loss function was developed to enhance the training of the DEM.	Simulated using indoor NYU-Depth-v2 and outdoor KITTI datasets and thus performance on dynamic outdoor environment could not be determined.
[98]	In this study, YOLO v3 was used to provide semantic information in order to distinguish edge features and reduce the effect of unstable features. This process improved the positioning accuracy of the system.	Simulated using public TUM RGB-D dataset and thus performance on dynamic outdoor environment could not be determined.
[16]	The study utilises CNN-based semantic segmentation and multi-view geometric constraints to identify and avoid using dynamic object feature points.	Simulated using ADVIO dataset and thus performance on dynamic outdoor environment could not be determined.
[99]	The study presents a dynamic point detection and rejection algorithm centred on neural network-based semantic segmentation. This eliminates dynamic object interference during pose estimation.	The technique was simulated on the EuRoC dataset and collected underground images tunnel. However, the real-time performance could not be evaluated on dynamic ground outdoor environments.
[100]	Presented in this study was a novel Visual Place Recognition technique capable of operating under changing viewpoint and appearance conditions. The system avoids the use of CNN which has high computational requirements.	The system was simulated on various public VPR datasets but focused mainly on static environments.
[101]	A deep learning-based real-time visual SLAM technique is proposed in this work. A parallel semantic thread is created using the lightweight object detection network YOLOv5s to obtain semantic information in the scene more quickly.	The experimental findings suggest that the system improves in terms of accuracy as well as real-time performance. However, for practicality, the map generating process and computation speed need to be improved.

4. Challenges and Open Issues in V-SLAM

Based on the reviewed works, a number of challenges associated with V-SLAM deployment were identified. These salient limitations can serve as open issues for researchers and provide a focus for future research directions. These challenges are presented as follows:

- (a) **Reliability in Outdoor Environments:** There is room for improvement in the reliability of V-SLAM implementations, especially in outdoor environments. The ineffectiveness of lidar and radar sensors in extreme weather conditions coupled with their high cost makes them [13,102,103] unsuitable for outdoor conditions. Additionally, despite the high precision and strong anti-interference ability of laser scans, they provide no semantic information about the environment [58]. The use of V-SLAM techniques cater for these limitations, however, these methods are susceptible to unpredictable and uncontrollable environmental conditions such as illumination changes [104].

- (b) **Operability in Dynamic Scenes:** The traditional SLAM and V-SLAM techniques assume a static environment, which is not always the case. V-SLAM techniques based on static scenes fail when deployed to dynamic environments [57]. The dynamic environment comprises of moving objects which needs to be taken into account in localisation and mapping operations. In the case of ORB-SLAM, for instance, it is not possible to determine if the extracted feature points are from static or dynamic objects [58]. Although significant research has been carried out in the area of object detection [91] and semantic segmentation [55], V-SLAM implementations in highly dynamic sceneries such as road networks and highways have not been exhaustively explored. Autonomous systems still need to fully comprehend dynamic scenarios and cope with dynamic objects [55,99].
- (c) **Robustness in Challenging Scenes:** V-SLAM techniques need to be robust enough to handle various scenarios. V-SLAM systems have the tendency to fail in situations involving fast motion [57,99]. Additionally, conventional V-SLAM systems rely on stable visual landmarks, which makes implementation difficult [104]. Therefore, achieving robust performance in challenging sceneries is paramount to the success of V-SLAM techniques [105].
- (d) **Real-time Deployment:** Deployment onto embedded hardware is another open issue for V-SLAM implementations [106]. Existing techniques have high computational requirements and slow real-time performance, thus, resulting in high deployment costs. With a high demand of unmanned systems for deployment in sectors such as Agriculture, Oil and Gas, and Military, the need arises for V-SLAM methods can be deployment onto microcontrollers and microcomputer systems.
- (e) **Control Scheme for Navigation:** Majority of the reviewed works lack an effective control technique for navigation based on the V-SLAM output. Considering path planning and control are major modules in AV deployment [8,9], there is a need for an effective control mechanism to navigate the AV in relation to the perceived environment. This will significantly contribute to the advancement towards fully autonomous vehicles.

5. Conclusions

A detailed review of V-SLAM approaches used for AV perception and localisation was presented in this paper. In addition, an in-depth study of state-of-the-art methodologies in V-SLAM was undertaken, as well as a thorough examination of these methods.

An overview on SLAM methods was provided. The classic SLAM methodology is divided into two types: filter-based and graph-based approaches. Filter-based approaches are preferable for solving the SLAM problem because they are not prone to accumulation errors and do not suffer from convergence to local minima. Furthermore, the various types of sensors utilized in the traditional SLAM technique were addressed, with an emphasis on their benefits and drawbacks. The various camera configurations (monocular, stereo, and RGB-D) and V-SLAM algorithms were also investigated. ORB-SLAM has been recognized as a popular alternative among researchers, particularly as a foundation for developing modified V-SLAM approaches.

Furthermore, a thorough examination was carried out to highlight various V-SLAM systems, their strengths and weaknesses. It was observed that Deep learning models have been used to include tasks such as object identification, semantic segmentation, and depth prediction into SLAM as a result of developments in deep learning. This paper also discussed the challenges of V-SLAM implementation and future research directions. The primary research gaps highlighted were dependability in outdoor environments, operability in dynamic scenarios, robustness in demanding circumstances, real-time deployment, and navigation control methods.

Author Contributions: Conceptualization, J.A.B. and S.A.A.; Methodology, J.A.B. and S.A.A.; Software, J.A.B.; Validation, A.M.A. and S.A.A.; Formal analysis, J.A.B., S.A.A. and A.M.A.; Investigation, J.A.B. and S.A.A.; Resources, S.A.A. and A.M.A.; Data Curation, J.A.B.; Writing—original draft preparation, J.A.B.; Writing—review and editing, J.A.B., A.M.A. and S.A.A.; Supervision, A.M.A. and S.A.A.; Project administration, S.A.A.; Funding Acquisition, A.M.A. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Tertiary Education Trust Fund (TETFUND), Nigeria under the project titled ‘Novel Road Accident Monitoring and Prevention System for Nigerian Roads and Highways’ (TETF/ES/DR&D-CE/NRF2020/SETI/95/VOL1) under the 2020 National Research Fund (NRF) grant cycle.

Acknowledgments: The authors wish to acknowledge Tertiary Education Trust Fund (TETFUND), Nigeria for funding this research under the project titled ‘Novel Road Accident Monitoring and Prevention System for Nigerian Roads and Highways’ (TETF/ES/DR&D-CE/NRF2020/SETI/95/VOL1) under the 2020 National Research Fund (NRF) grant cycle.

Conflicts of Interest: The authors declare no conflict of interest.

References

- Du, H.; Leng, S.; Wu, F.; Chen, X.; Mao, S. A new vehicular fog computing architecture for cooperative sensing of autonomous driving. *IEEE Access* **2020**, *8*, 10997–11006. [\[CrossRef\]](#)
- Jeong, Y.; Kim, S.; Yi, K. Surround vehicle motion prediction using LSTM-RNN for motion planning of autonomous vehicles at multi-lane turn intersections. *IEEE Open J. Intell. Transp. Syst.* **2020**, *1*, 2–14. [\[CrossRef\]](#)
- Mellucci, C.; Menon, P.P.; Edwards, C.; Challenor, P.G. Environmental feature exploration with a single autonomous vehicle. *IEEE Trans. Control. Syst. Technol.* **2019**, *28*, 1349–1362. [\[CrossRef\]](#)
- Tang, L.; Yan, F.; Zou, B.; Wang, K.; Lv, C. An improved kinematic model predictive control for high-speed path tracking of autonomous vehicles. *IEEE Access* **2020**, *8*, 51400–51413. [\[CrossRef\]](#)
- Dixit, S.; Montanaro, U.; Dianati, M.; Oxtoby, D.; Mizutani, T.; Mouzakitis, A.; Fallah, S. Trajectory planning for autonomous high-speed overtaking in structured environments using robust MPC. *IEEE Trans. Intell. Transp. Syst.* **2019**, *21*, 2310–2323. [\[CrossRef\]](#)
- Gu, X.; Han, Y.; Yu, J. A novel lane-changing decision model for autonomous vehicles based on deep autoencoder network and XGBoost. *IEEE Access* **2020**, *8*, 9846–9863. [\[CrossRef\]](#)
- Yao, Q.; Tian, Y.; Wang, Q.; Wang, S. Control strategies on path tracking for autonomous vehicle: State of the art and future challenges. *IEEE Access* **2020**, *8*, 161211–161222. [\[CrossRef\]](#)
- Chae, H.; Yi, K. Virtual target-based overtaking decision, motion planning, and control of autonomous vehicles. *IEEE Access* **2020**, *8*, 51363–51376. [\[CrossRef\]](#)
- Liao, J.; Liu, T.; Tang, X.; Mu, X.; Huang, B.; Cao, D. Decision-making strategy on highway for autonomous vehicles using deep reinforcement learning. *IEEE Access* **2020**, *8*, 177804–177814. [\[CrossRef\]](#)
- Benterki, A.; Boukhniifer, M.; Judalet, V.; Maaoui, C. Artificial intelligence for vehicle behavior anticipation: Hybrid approach based on maneuver classification and trajectory prediction. *IEEE Access* **2020**, *8*, 56992–57002. [\[CrossRef\]](#)
- Yan, F.; Wang, K.; Zou, B.; Tang, L.; Li, W.; Lv, C. Lidar-based multi-task road perception network for autonomous vehicles. *IEEE Access* **2020**, *8*, 86753–86764. [\[CrossRef\]](#)
- Guo, H.; Li, W.; Nejad, M.; Shen, C.C. Proof-of-event recording system for autonomous vehicles: A blockchain-based solution. *IEEE Access* **2020**, *8*, 182776–182786. [\[CrossRef\]](#)
- Ort, T.; Gilitschenski, I.; Rus, D. Autonomous navigation in inclement weather based on a localizing ground penetrating radar. *IEEE Robot. Autom. Lett.* **2020**, *5*, 3267–3274. [\[CrossRef\]](#)
- Zhan, Z.; Jian, W.; Li, Y.; Yue, Y. A slam map restoration algorithm based on submaps and an undirected connected graph. *IEEE Access* **2021**, *9*, 12657–12674. [\[CrossRef\]](#)
- Saeedi, S.; Bodin, B.; Wagstaff, H.; Nisbet, A.; Nardi, L.; Mawer, J.; Melot, N.; Palomar, O.; Vespa, E.; Spink, T.; et al. Navigating the landscape for real-time localization and mapping for robotics and virtual and augmented reality. *Proc. IEEE* **2018**, *106*, 2020–2039. [\[CrossRef\]](#)
- Zhao, X.; Wang, C.; Ang, M.H. Real-time visual-inertial localization using semantic segmentation towards dynamic environments. *IEEE Access* **2020**, *8*, 155047–155059. [\[CrossRef\]](#)
- Zheng, Y.; Chen, S.; Cheng, H. Real-time cloud visual simultaneous localization and mapping for indoor service robots. *IEEE Access* **2020**, *8*, 16816–16829. [\[CrossRef\]](#)
- Yusefi, A.; Durdu, A.; Aslan, M.F.; Sungur, C. Lstm and filter based comparison analysis for indoor global localization in uavs. *IEEE Access* **2021**, *9*, 10054–10069. [\[CrossRef\]](#)
- Mur-Artal, R.; Tardós, J.D. Orb-slam2: An open-source slam system for monocular, stereo, and rgb-d cameras. *IEEE Trans. Robot.* **2017**, *33*, 1255–1262. [\[CrossRef\]](#)

20. Rusli, I.; Trilaksono, B.R.; Adiprawita, W. RoomSLAM: Simultaneous localization and mapping with objects and indoor layout structure. *IEEE Access* **2020**, *8*, 196992–197004. [[CrossRef](#)]
21. Steenbeek, A. CNN Based Dense Monocular Visual SLAM for Indoor Mapping And Autonomous Exploration. Master's Thesis, University of Twente, Enschede, The Netherlands, 2020.
22. Ozaki, M.; Kakimuma, K.; Hashimoto, M.; Takahashi, K. Laser-based pedestrian tracking in outdoor environments by multiple mobile robots. *Sensors* **2012**, *12*, 14489–14507. [[CrossRef](#)] [[PubMed](#)]
23. Himmelsbach, M.; Mueller, A.; Lüttel, T.; Wünsche, H.J. LIDAR-based 3D object perception. In Proceedings of the 1st International Workshop on Cognition for Technical Systems, München, Germany, 6–7 October 2008; Volume 1.
24. Wang, W.; Wu, Y.; Jiang, Z.; Qi, J. A clutter-resistant SLAM algorithm for autonomous guided vehicles in dynamic industrial environment. *IEEE Access* **2020**, *8*, 109770–109782. [[CrossRef](#)]
25. Zhao, J.; Huang, S.; Zhao, L.; Chen, Y.; Luo, X. Conic feature based simultaneous localization and mapping in open environment via 2D lidar. *IEEE Access* **2019**, *7*, 173703–173718. [[CrossRef](#)]
26. Luo, J.; Qin, S. A fast algorithm of simultaneous localization and mapping for mobile robot based on ball particle filter. *IEEE Access* **2018**, *6*, 20412–20429. [[CrossRef](#)]
27. Bavle, H.; De La Puente, P.; How, J.P.; Campoy, P. VPS-SLAM: Visual planar semantic SLAM for aerial robotic systems. *IEEE Access* **2020**, *8*, 60704–60718. [[CrossRef](#)]
28. Teixeira, B.; Silva, H.; Matos, A.; Silva, E. Deep learning for underwater visual odometry estimation. *IEEE Access* **2020**, *8*, 44687–44701. [[CrossRef](#)]
29. Evers, C.; Naylor, P.A. Optimized self-localization for SLAM in dynamic scenes using probability hypothesis density filters. *IEEE Trans. Signal Process.* **2017**, *66*, 863–878. [[CrossRef](#)]
30. Thrun, S. Probabilistic robotics. *Commun. ACM* **2002**, *45*, 52–57. [[CrossRef](#)]
31. Kaess, M.; Ranganathan, A.; Dellaert, F. iSAM: Incremental smoothing and mapping. *IEEE Trans. Robot.* **2008**, *24*, 1365–1378. [[CrossRef](#)]
32. Kaess, M.; Johannsson, H.; Roberts, R.; Ila, V.; Leonard, J.J.; Dellaert, F. iSAM2: Incremental smoothing and mapping using the Bayes tree. *Int. J. Robot. Res.* **2012**, *31*, 216–235. [[CrossRef](#)]
33. Smith, R.C.; Cheeseman, P. On the representation and estimation of spatial uncertainty. *Int. J. Robot. Res.* **1986**, *5*, 56–68. [[CrossRef](#)]
34. Ullah, I.; Shen, Y.; Su, X.; Esposito, C.; Choi, C. A localization based on unscented Kalman filter and particle filter localization algorithms. *IEEE Access* **2019**, *8*, 2233–2246. [[CrossRef](#)]
35. Thrun, S. Particle Filters in Robotics. In *UAI*; Citeseer: Princeton, NJ, USA, 2002; Volume 2, pp. 511–518.
36. Thrun, S.; Fox, D.; Burgard, W.; Dellaert, F. Robust Monte Carlo localization for mobile robots. *Artif. Intell.* **2001**, *128*, 99–141. [[CrossRef](#)]
37. Llofriu, M.; Andrade, F.; Benavides, F.; Weitzenfeld, A.; Tejera, G. An embedded particle filter SLAM implementation using an affordable platform. In Proceedings of the 2013 16th International Conference on Advanced Robotics (ICAR), Montevideo, Uruguay, 25–29 November 2013; pp. 1–6.
38. Yatim, N.M.; Buniyamin, N. Particle filter in simultaneous localization and mapping (Slam) using differential drive mobile robot. *J. Teknol.* **2015**, *77*, 91–97. [[CrossRef](#)]
39. Slowak, P.; Kaniowski, P. Stratified particle filter monocular SLAM. *Remote Sens.* **2021**, *13*, 3233. [[CrossRef](#)]
40. Grisetti, G.; Stachniss, C.; Burgard, W. Improving grid-based slam with rao-blackwellized particle filters by adaptive proposals and selective resampling. In Proceedings of the 2005 IEEE International Conference on Robotics and Automation, Barcelona, Spain, 18–22 April 2005; pp. 2432–2437.
41. Murphy, K.P. Bayesian map learning in dynamic environments. *Adv. Neural Inf. Process. Syst.* **1999**, *12*, 1015–1021.
42. Murphy, K.; Russell, S. Rao-Blackwellised particle filtering for dynamic Bayesian networks. In *Sequential Monte Carlo Methods in Practice*; Springer: Berlin/Heidelberg, Germany, 2001; pp. 499–515.
43. Montemerlo, M.; Thrun, S.; Koller, D.; Wegbreit, B.; et al. FastSLAM 2.0: An improved particle filtering algorithm for simultaneous localization and mapping that provably converges. In Proceedings of the IJCAI, Acapulco, Mexico, 9–15 August 2003; Volume 3, pp. 1151–1156.
44. Montemerlo, M.; Thrun, S.; Koller, D.; Wegbreit, B. FastSLAM: A factored solution to the simultaneous localization and mapping problem. *Aaai/iaai* **2002**, 593598, 593–598.
45. Duymaz, E.; Oğuz, A.E.; Temeltas, H. Exact flow of particles using for state estimations in unmanned aerial systems navigation. *PLoS ONE* **2020**, *15*, e0231412. [[CrossRef](#)]
46. Wang, J.; Takahashi, Y. Particle filter based landmark mapping for SLAM of mobile robot based on RFID system. In Proceedings of the 2016 IEEE International Conference on Advanced Intelligent Mechatronics (AIM), Banff, AB, Canada, 12–15 July 2016; pp. 870–875.
47. Javanmardi, E.; Javanmardi, M.; Gu, Y.; Kamijo, S. Factors to evaluate capability of map for vehicle localization. *IEEE Access* **2018**, *6*, 49850–49867. [[CrossRef](#)]
48. Blanco-Claraco, J.L.; Mañas-Alvarez, F.; Torres-Moreno, J.L.; Rodriguez, F.; Gimenez-Fernandez, A. Benchmarking particle filter algorithms for efficient velodyne-based vehicle localization. *Sensors* **2019**, *19*, 3155. [[CrossRef](#)]
49. Niu, Z.; Zhao, X.; Sun, J.; Tao, L.; Zhu, B. A continuous positioning algorithm based on RTK and VI-SLAM with smartphones. *IEEE Access* **2020**, *8*, 185638–185650. [[CrossRef](#)]

50. Fairfield, N.; Kantor, G.; Wettergreen, D. Towards particle filter SLAM with three dimensional evidence grids in a flooded subterranean environment. In Proceedings of the 2006 IEEE International Conference on Robotics and Automation, ICRA, Orlando, FL, USA, 15–19 May 2006; pp. 3575–3580.
51. Adams, M.; Zhang, S.; Xie, L. Particle filter based outdoor robot localization using natural features extracted from laser scanners. In Proceedings of the IEEE International Conference on Robotics and Automation, ICRA'04, New Orleans, LA, USA, 26 April–1 May 2004; Volume 2, pp. 1493–1498.
52. Pei, F.; Wu, M.; Zhang, S. Distributed SLAM using improved particle filter for mobile robot localization. *Sci. World J.* **2014**, *2014*, 239531. [[CrossRef](#)] [[PubMed](#)]
53. Xie, D.; Xu, Y.; Wang, R. Obstacle detection and tracking method for autonomous vehicle based on three-dimensional LiDAR. *Int. J. Adv. Robot. Syst.* **2019**, *16*, 1729881419831587. [[CrossRef](#)]
54. Lu, H.; Yang, S.; Zhao, M.; Cheng, S. Multi-robot indoor environment map building based on multi-stage optimization method. *Complex Syst. Model. Simul.* **2021**, *1*, 145–161. [[CrossRef](#)]
55. Ai, Y.; Rui, T.; Lu, M.; Fu, L.; Liu, S.; Wang, S. DDL-SLAM: A robust RGB-D SLAM in dynamic environments combined with deep learning. *IEEE Access* **2020**, *8*, 162335–162342. [[CrossRef](#)]
56. Zhang, S.; Zheng, L.; Tao, W. Survey and evaluation of RGB-D SLAM. *IEEE Access* **2021**, *9*, 21367–21387. [[CrossRef](#)]
57. Yang, D.; Bi, S.; Wang, W.; Yuan, C.; Wang, W.; Qi, X.; Cai, Y. DRE-SLAM: Dynamic RGB-D encoder SLAM for a differential-drive robot. *Remote Sens.* **2019**, *11*, 380. [[CrossRef](#)]
58. Li, F.; Chen, W.; Xu, W.; Huang, L.; Li, D.; Cai, S.; Yang, M.; Xiong, X.; Liu, Y.; Li, W. A mobile robot visual SLAM system with enhanced semantics segmentation. *IEEE Access* **2020**, *8*, 25442–25458. [[CrossRef](#)]
59. Zeng, F.; Wang, C.; Ge, S.S. A survey on visual navigation for artificial agents with deep reinforcement learning. *IEEE Access* **2020**, *8*, 135426–135442. [[CrossRef](#)]
60. Cheng, J.; Wang, C.; Meng, M.Q.H. Robust visual localization in dynamic environments based on sparse motion removal. *IEEE Trans. Autom. Sci. Eng.* **2019**, *17*, 658–669. [[CrossRef](#)]
61. Davison, A.J. Real-time simultaneous localisation and mapping with a single camera. In Proceedings of the Computer Vision, IEEE International Conference on. IEEE Computer Society, Nice, France, 13–16 October 2003; Volume 3, p. 1403–1403.
62. Klein, G.; Murray, D. Parallel tracking and mapping for small AR workspaces. In Proceedings of the 2007 6th IEEE and ACM International Symposium on Mixed and Augmented Reality, Nara, Japan, 13–16 November 2007; pp. 225–234.
63. Mur-Artal, R.; Montiel, J.M.M.; Tardos, J.D. ORB-SLAM: A versatile and accurate monocular SLAM system. *IEEE Trans. Robot.* **2015**, *31*, 1147–1163. [[CrossRef](#)]
64. Lu, S.; Zhi, Y.; Zhang, S.; He, R.; Bao, Z. Semi-Direct Monocular SLAM With Three Levels of Parallel Optimizations. *IEEE Access* **2021**, *9*, 86801–86810. [[CrossRef](#)]
65. Fan, M.; Kim, S.W.; Kim, S.T.; Sun, J.Y.; Ko, S.J. Simple But Effective Scale Estimation for Monocular Visual Odometry in Road Driving Scenarios. *IEEE Access* **2020**, *8*, 175891–175903. [[CrossRef](#)]
66. Nobis, F.; Papanikolaou, O.; Betz, J.; Lienkamp, M. Persistent map saving for visual localization for autonomous vehicles: An orb-slam 2 extension. In Proceedings of the 2020 Fifteenth International Conference on Ecological Vehicles and Renewable Energies (EVER), Monte-Carlo, Monaco, 10–12 September 2020; pp. 1–9.
67. Li, G.; Liao, X.; Huang, H.; Song, S.; Liu, B.; Zeng, Y. Robust stereo visual slam for dynamic environments with moving object. *IEEE Access* **2021**, *9*, 32310–32320. [[CrossRef](#)]
68. Chien, C.H.; Hsu, C.C.J.; Wang, W.Y.; Chiang, H.H. Indirect visual simultaneous localization and mapping based on linear models. *IEEE Sens. J.* **2019**, *20*, 2738–2747. [[CrossRef](#)]
69. Hempel, T.; Al-Hamadi, A. Pixel-wise motion segmentation for SLAM in dynamic environments. *IEEE Access* **2020**, *8*, 164521–164528. [[CrossRef](#)]
70. Li, C.; Zhang, X.; Gao, H.; Wang, R.; Fang, Y. Bridging the gap between visual servoing and visual SLAM: A novel integrated interactive framework. *IEEE Trans. Autom. Sci. Eng.* **2021**, *19*, 2245–2255. [[CrossRef](#)]
71. Yang, S.; Fan, G.; Bai, L.; Li, R.; Li, D. MGC-VSLAM: A meshing-based and geometric constraint VSLAM for dynamic indoor environments. *IEEE Access* **2020**, *8*, 81007–81021. [[CrossRef](#)]
72. Li, D.; Liu, S.; Xiang, W.; Tan, Q.; Yuan, K.; Zhang, Z.; Hu, Y. A SLAM System Based on RGBD Image and Point-Line Feature. *IEEE Access* **2021**, *9*, 9012–9025. [[CrossRef](#)]
73. Wu, R.; Pike, M.; Lee, B.G. DT-SLAM: Dynamic Thresholding Based Corner Point Extraction in SLAM System. *IEEE Access* **2021**, *9*, 91723–91729. [[CrossRef](#)]
74. Zhang, C. PL-GM: RGB-D SLAM with a novel 2D and 3D geometric constraint model of point and line features. *IEEE Access* **2021**, *9*, 9958–9971. [[CrossRef](#)]
75. Wang, Q.; Yan, Z.; Wang, J.; Xue, F.; Ma, W.; Zha, H. Line Flow Based Simultaneous Localization and Mapping. *IEEE Trans. Robot.* **2021**, *37*, 1416–1432. [[CrossRef](#)]
76. Forster, C.; Pizzoli, M.; Scaramuzza, D. SVO: Fast semi-direct monocular visual odometry. In Proceedings of the 2014 IEEE International Conference on Robotics and Automation (ICRA), Hong Kong, China, 31 May–7 June 2014; pp. 15–22.
77. Engel, J.; Schöps, T.; Cremers, D. LSD-SLAM: Large-scale direct monocular SLAM. In Proceedings of the European Conference on Computer Vision, Zurich, Switzerland, 6–12 September 2014; pp. 834–849.

78. Cremers, D. Direct methods for 3d reconstruction and visual slam. In Proceedings of the 2017 Fifteenth IAPR International Conference on Machine Vision Applications (MVA), Nagoya, Japan, 8–12 May 2017; pp. 34–38.
79. Zhu, Z.; Kaizu, Y.; Furuhashi, K.; Imou, K. Visual-Inertial RGB-D SLAM With Encoders for a Differential Wheeled Robot. *IEEE Sens. J.* **2021**, *22*, 5360–5371. [[CrossRef](#)]
80. Zhang, M.; Han, S.; Wang, S.; Liu, X.; Hu, M.; Zhao, J. Stereo visual inertial mapping algorithm for autonomous mobile robot. In Proceedings of the 2020 3rd International Conference on Intelligent Robotic and Control Engineering (IRCE), Oxford, UK, 10–12 August 2020; pp. 97–104.
81. Wu, R. *A Low-cost SLAM Fusion Algorithm for Robot Localization*; University of Alberta: Edmonton, Canada, 2019.
82. Chen, C.H.; Wang, C.C.; Lin, S.F. A Navigation Aid for Blind People Based on Visual Simultaneous Localization and Mapping. In Proceedings of the 2020 IEEE International Conference on Consumer Electronics-Taiwan (ICCE-Taiwan), Taoyuan, Taiwan, 28–30 September 2020; pp. 1–2.
83. Haavardsholm, T.V.; Skauli, T.; Stahl, A. Multimodal Multispectral Imaging System for Small UAVs. *IEEE Robot. Autom. Lett.* **2020**, *5*, 1039–1046. [[CrossRef](#)]
84. Cui, L.; Ma, C. SDF-SLAM: Semantic depth filter SLAM for dynamic environments. *IEEE Access* **2020**, *8*, 95301–95311. [[CrossRef](#)]
85. Jiang, F.; Chen, J.; Ji, S. Panoramic visual-inertial SLAM tightly coupled with a wheel encoder. *ISPRS J. Photogramm. Remote Sens.* **2021**, *182*, 96–111. [[CrossRef](#)]
86. Canovas, B.; Nègre, A.; Rombaut, M. Onboard dynamic RGB-D simultaneous localization and mapping for mobile robot navigation. *ETRI J.* **2021**, *43*, 617–629. [[CrossRef](#)]
87. Huang, C. Wheel Odometry Aided Visual-Inertial Odometry in Winter Urban Environments. Master's Thesis, Schulich School of Engineering, Calgary, AB, Canada, 2021.
88. Xu, Z.; Haroutunian, M.; Murphy, A.J.; Neasham, J.; Norman, R. An Integrated Visual Odometry System With Stereo Camera for Unmanned Underwater Vehicles. *IEEE Access* **2022**, *10*, 71329–71343. [[CrossRef](#)]
89. Kuang, Z.; Wei, W.; Yan, Y.; Li, J.; Lu, G.; Peng, Y.; Li, J.; Shang, W. A Real-time and Robust Monocular Visual Inertial SLAM System Based on Point and Line Features for Mobile Robots of Smart Cities Towards 6G. *IEEE Open J. Commun. Soc.* **2022**, *3*, 1950–1962. [[CrossRef](#)]
90. Lee, S.J.; Choi, H.; Hwang, S.S. Real-time depth estimation using recurrent CNN with sparse depth cues for SLAM system. *Int. J. Control. Autom. Syst.* **2020**, *18*, 206–216. [[CrossRef](#)]
91. Soares, J.C.V.; Gattass, M.; Meggiolaro, M.A. Crowd-SLAM: Visual SLAM Towards Crowded Environments using Object Detection. *J. Intell. Robot. Syst.* **2021**, *102*, 1–16. [[CrossRef](#)]
92. Liu, Y.; Miura, J. RDS-SLAM: Real-time dynamic SLAM using semantic segmentation methods. *IEEE Access* **2021**, *9*, 23772–23785. [[CrossRef](#)]
93. Liu, Y.; Miura, J. RDMO-SLAM: Real-time visual SLAM for dynamic environments using semantic label prediction with optical flow. *IEEE Access* **2021**, *9*, 106981–106997. [[CrossRef](#)]
94. Yang, S.; Zhao, C.; Wu, Z.; Wang, Y.; Wang, G.; Li, D. Visual SLAM Based on Semantic Segmentation and Geometric Constraints for Dynamic Indoor Environments. *IEEE Access* **2022**, *10*, 69636–69649. [[CrossRef](#)]
95. Shao, C.; Zhang, C.; Fang, Z.; Yang, G. A deep learning-based semantic filter for ransac-based fundamental matrix calculation and the ORB-slam system. *IEEE Access* **2019**, *8*, 3212–3223. [[CrossRef](#)]
96. Han, S.; Xi, Z. Dynamic scene semantics SLAM based on semantic segmentation. *IEEE Access* **2020**, *8*, 43563–43570. [[CrossRef](#)]
97. Tu, X.; Xu, C.; Liu, S.; Xie, G.; Huang, J.; Li, R.; Yuan, J. Learning depth for scene reconstruction using an encoder-decoder model. *IEEE Access* **2020**, *8*, 89300–89317. [[CrossRef](#)]
98. Liu, W.; Mo, Y.; Jiao, J.; Deng, Z. EF-Razor: An Effective Edge-Feature Processing Method in Visual SLAM. *IEEE Access* **2020**, *8*, 140798–140805. [[CrossRef](#)]
99. Li, D.; Yang, W.; Shi, X.; Guo, D.; Long, Q.; Qiao, F.; Wei, Q. A visual-inertial localization method for unmanned aerial vehicle in underground tunnel dynamic environments. *IEEE Access* **2020**, *8*, 76809–76822. [[CrossRef](#)]
100. Tomitá, M.A.; Zaffar, M.; Milford, M.J.; McDonald-Maier, K.D.; Ehsan, S. Convsequential-slam: A sequence-based, training-less visual place recognition technique for changing environments. *IEEE Access* **2021**, *9*, 118673–118683. [[CrossRef](#)]
101. Su, P.; Luo, S.; Huang, X. Real-Time Dynamic SLAM Algorithm Based on Deep Learning. *IEEE Access* **2022**, *10*, 87754–87766. [[CrossRef](#)]
102. Yue, J.; Wen, W.; Han, J.; Hsu, L.T. LiDAR data enrichment using deep learning based on high-resolution image: An approach to achieve high-performance LiDAR SLAM using Low-cost LiDAR. *arXiv* **2020**, arXiv:2008.03694.
103. Hu, T.; Sun, X.; Su, Y.; Guan, H.; Sun, Q.; Kelly, M.; Guo, Q. Development and performance evaluation of a very low-cost UAV-LiDAR system for forestry applications. *Remote Sens.* **2020**, *13*, 77. [[CrossRef](#)]
104. Cheng, J.; Zhang, H.; Meng, M.Q.H. Improving visual localization accuracy in dynamic environments based on dynamic region removal. *IEEE Trans. Autom. Sci. Eng.* **2020**, *17*, 1585–1596. [[CrossRef](#)]
105. Alkendi, Y.; Seneviratne, L.; Zweiri, Y. State of the art in vision-based localization techniques for autonomous navigation systems. *IEEE Access* **2021**, *9*, 76847–76874. [[CrossRef](#)]
106. Yu, J.; Gao, F.; Cao, J.; Yu, C.; Zhang, Z.; Huang, Z.; Wang, Y.; Yang, H. CNN-based Monocular Decentralized SLAM on embedded FPGA. In Proceedings of the 2020 IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW), New Orleans, LA, USA, 18–22 May 2020; pp. 66–73.